UNIVERSIDAD POLITÉCNICA DE VALENCIA

Nuevos algoritmos para el problema de secuenciación en máquinas paralelas no relacionadas y generalizaciones

Luis Fanjul Peyró

TESIS DOCTORAL

Dirigida por:
Rubén Ruiz García

Valencia, 2010

A Celia, por soportarme.

A Nacho y María por ser mis hijos.

Resumen

Para iniciar esta Tesis Doctoral se buscó un problema de producción sencillo pero de amplia aplicación práctica que permitiera adaptarlo para llegar a problemas más generales y de más amplia aplicación. Por este motivo, nos centramos en las máquinas paralelas, y dentro de ellas, en las no relacionadas dado que son una generalización de los casos de máquinas idénticas y de las uniformemente relacionadas. Escogimos el objetivo de minimizar el tiempo máximo de finalización o $C_{\text{máx}}$, uno de los más comunes de la literatura. Este problema tiene la facultad de que, a pesar de su carácter teórico, tiene una amplia aplicación práctica, como el caso de secuenciar las tareas de los hornos de cocción cerámicos. Por otra parte se quería ampliar el problema para el caso en que no se usaran todas las máquinas o no se hicieran todos los trabajos necesariamente.

Las metas perseguidas son el presentar unos algoritmos sencillos y potentes para la resolución del problema $R//C_{\text{máx}}$, capaces de constituirse en el estado del arte. Dado que los modernos ordenadores montan casi en su totalidad varios núcleos en su CPU y los algoritmos se van adaptando a este hecho, también se ha buscado realizar una adaptación de los algoritmos para su uso en paralelo. Finalmente, se pone como meta el encontrar métodos eficaces y sencillos para la resolución de problemas de este tipo en donde no se emplearan todas las máquinas o no se realizaran todos los trabajos.

En la presente Tesis Doctoral se realizó un amplio estudio de la literatura existente respecto al problema de máquinas paralelas no relacionadas y se extrajo el estado del arte, así como un estudio del posible tipo de instancias a emplear, dado que no existía una grupo de instancias tipo para este problema.

Se presentan cuatro algoritmos iniciales sencillos que mejoran los resultados del estado del arte en algunos casos y dan mejores resultados de media en el conjunto total de instancias tratadas. Dichos algoritmos se basan en métodos iterativos en los que se realiza una búsqueda local de inserción seguida de

una búsqueda local de intercambio hasta óptimo local de ambas, seguidas de diversos métodos de modificación parcial de la solución para volver de nuevo con esta solución modificada a las búsquedas locales.
Se introducen métodos que buscan disminuir el grado de aleatoriedad de los primeros algoritmos, donde se llegan a desarrollar tres nuevos algoritmos que mejoran los anteriores y que destacan con mejores resultados que el estado del arte en prácticamente todos los casos. Un nuevo algoritmo híbrido donde se aúnan todas las características de los métodos desarrollados hasta ese momento nos lleva a mejorar significativamente los resultados obtenidos.
No obstante los buenos resultados, se proponen nuevos métodos basados en una disminución del número de variables a tener en cuenta en la resolución del problema que acaban derivando en cinco nuevos algoritmos que nos llevan a mejorar aún más los valores ya obtenidos por los anteriores métodos propuestos. Es de destacar que los algoritmos propuestos no solo obtienen buenos resultados, sino que van mejorando estos resultados a medida que se les da más tiempo de ejecución.
Se realizan las modificaciones pertinentes a los mejores algoritmos desarrollados en aras de paralelizar parte de las tareas que realizan y esto nos permite compararlos con el estado del arte. Los resultados muestran como el mejor algoritmo desarrollado es capaz de superar al método representativo del estado del arte también en el ámbito paralelo.
Finalmente, se presenta el problema de máquinas opcionales y selección de trabajos, realizando primeramente su formulación matemática para posteriormente mostrar los métodos más favorables para la solución de este tipo de problemas, basados, en el caso de la selección de máquinas, en una elaboración de un ranking de máquinas, seguidos de una selección iterativa de ellas y una resolución final por distintos métodos.
Por último hacemos una reflexión sobre todo lo estudiado y una discusión sobre las posibles líneas de investigación que deja abiertas la presente Tesis Doctoral.

Resum

Per iniciar aquesta tesi doctoral es va buscar un problema de producció senzill però de gran aplicació pràctica que permetés adaptar per arribar a problemes més generals i de més àmplia aplicació. Per aquest motiu, ens centrem en les màquines paral.leles, i dins d'elles, en les no relacionades donat que són una generalització dels casos de màquines idèntiques i de les uniformement relacionades. Vam escollir l'objectiu de minimitzar el temps màxim de finalització o $C_{\max}$, un dels més comuns de la literatura. Aquest problema té la facultat que, tot i el seu caràcter teòric, té una àmplia aplicació pràctica, com per al cas de seqüenciar les tasques dels forns de cocció ceràmics. D'altra banda es volia ampliar el problema per al cas en que no fan falta totes les màquines o no es fessin tots els treballs necessàriament.

Les metes perseguides són el presentar uns algoritmes senzills i potents per a la resolució del problema $R//C_{\max}$, capaços de constituir-se en l'estat de l'art. Atès que els moderns ordinadors munten quasi totalment diversos nuclis a la seva CPU i els algorismes es van adaptant a aquest fet, també s'ha buscat realitzar una adaptació dels algorismes per al seu ús en paral.lel. Finalment, es posa com a meta el trobar mètodes eficaços i senzills per a la resolució de problemes d'aquest tipus on no s'utilitzaran totes les màquines o no es realitzaran tots els treballs.

En la present tesi doctoral es va realitzar un ampli estudi de la literatura existent respecte al problema de màquines paral.leles no relacionades i es va extraure l'estat de l'art, així com un estudi del possible tipus d'instàncies a utilitzar, ja que no existia una grup d'instàncies tipus per aquest problema.

Es presenten quatre algoritmes inicials senzills que milloren els resultats de l'estat de l'art en alguns casos i donen millors resultats de mitjana en el conjunt total de instàncies tractades. Aquests algorismes es basen en mètodes iteratius en què es realitza una cerca local d'inserció seguida d'una cerca local d'intercanvi fins òptim local de les dues, seguides de diversos mètodes de modificació

parcial de la solució per tornar de nou amb aquesta solució modificació a les recerques locals.
S'introdueixen mètodes que busquen disminuir el grau d'aleatorietat dels primers algoritmes, on s'arriba a desenvolupar tres nous algoritmes que milloren els anteriors i que destaquen amb millors resultats que l'estat de l'art en pràcticament tots els casos. Un nou algorisme híbrid a on es reuneixen totes les característiques dels mètodes desenvolupats fins aquell moment ens porta a millorar significativament els resultats obtinguts.
No obstant els bons resultats, es proposen nous mètodes basats en una disminució del nombre de variables a tenir en compte en la resolució del problema que acaben derivant en cinc nous algoritmes que ens porten a millorar encara més els valors obtinguts pels anteriors mètodes proposats. És de destacar que els algorismes proposats no només obtenen bons resultats, sinó que van millorant aquests resultats a mesura que se'ls dóna més temps d'execució.
Es realitzen les modificacions pertinents als millors algorismes desenvolupats en nom de parallelitzar part de les tasques que realitzen i això ens permet comparar-los amb l'estat de l'art. Els resultats mostren com el millor algorisme desenvolupat és capaç de superar el mètode representatiu de l'estat de l'art també en l'ambient paral.lel.
Finalment, es presenta el problema de màquines opcionals i selecció de treballs, realitzant primerament la seva formulació matemàtica per a posteriorment mostrar els mètodes més favorables per a la solució d'aquest tipus de problemes, basats, en el cas de la selecció de màquines, en un elaboració de un rànquing de màquines, seguits d'una selecció iterativa d'elles i una resolució final per diferents mètodes.
Per últim fem una reflexió sobre tot el estudiat i una discussió sobre les possibles línies d'investigació que deixa obertes aquesta tesi doctoral.

Abstract

At the onset of this thesis we aimed for a simple problem but with wide practical applications. The initial objective was to reach more general problems and wider applications after adaptations. For this reason, we focus on parallel machines scheduling problems, and within them, in unrelated parallel machine problems as they are a generalization of the identical and uniform parallel related machines cases. We chose to minimize the maximum completion time or $C_{\max}$, one of the most commonly studied objectives in the literature. This problem, despite its theoretical nature, has many practical applications, as in the case of sequencing the work of ceramic kilns. Furthermore, the problem has been expanded in thesis for some not studied extensions like the case in which not all machines have to be used or in the case where not all jobs must be processed.

The goals of this thesis are to present simple and powerful algorithms for solving the $R//C_{\max}$ problem, capable of constituting the new state-of-the-art in the literature. Modern computers are built with multi-core CPUs and modern algorithms are starting to take this into account. Therefore, we also set a goal to make parallel computing adaptations of our presented algorithms. Finally, we set the goal of finding effective and simple methods for solving problems where not all machines have to be used and problems where not all jobs must be processed.

In the present thesis we have conducted and extensive study of existing literature for the unrelated parallel machines scheduling problem and extracted the state-of-the-art. We also study the possible types of instances to be used, since the literature was lacking a common benchmark.

We present four simple initial algorithms that improve the performance of existing state-of-the-art methods in some cases and better results, on average, in the entire set of instances treated. These algorithms are based on iterative methods. An iterative application to local optimality of an insertion local

search followed by a interchange local search has proven beneficial. This, together with various methods of partial modification of the solution are the main core of the proposed algorithms.

We introduce procedures to reduce the degree of randomness of the initially proposed algorithms, where we develop three new methods that improve the previous algorithms and give better results than the state-of-the-art in virtually all cases. A new hybrid algorithm which combines all the features of the methods developed so far leads us to a significant improvement in the benchmark results.

Despite the excellent performance, we further propose new methods based on a reduction in the number of variables to consider when solving the mathematical assignment problem. This results in five new algorithms that further improve the solutions obtained by all previously proposed methods. It is worth noting that the proposed algorithms not only achieve good results, but they do so with an inherent simplicity.

Some changes are made to the best algorithms in order to parallelize them. This allows us to compare with the state-of-the-art in parallel computing environments. The results show again the superiority of the proposed methods also in parallel environments.

Finally, we study the problem where not all machines have to be used and problems where not all jobs must be processed. Their mathematical formulations are studied first. Later, we present some algorithmic approaches for solving the first problem, as the second is shown to be readily solvable by mathematical programming.

The thesis closes with an assessment of all the achievements, publications and statements for future research lines.

Agradecimientos

En primer lugar, quiero agradecer a todos los miembros del SOA su paciencia explicándome todo lo necesario para el desarrollo de este estudio. Empezando por la programación en Delphi, que desconocía completamente, la explicación de los rudimentos de Linux y C++, el uso de distinto software, y un estar siempre todos dispuestos a echarme una mano y a atender todas mis preguntas sobre los temas más variados. Gracias y perdonar por las molestias.

Quiero agradecer a Ghirardi y Potts el habernos facilitado el código de su algoritmo RBS, y a Woclaw por el código de Partial, lo cuál nos ha ahorrado bastante trabajo y no pocos quebraderos de cabeza.

También quiero agradecer a mi familia el apoyo dado, al aguantarme esta temporada de largas jornadas de trabajo, y en especial a Celia, mi mujer, por escuchar mis explicaciones de algo que, en el mejor de los casos, le parecían harto extrañas.

Y por último, quiero darle mi más sincero agradecimiento a mi director de Tesis, que con infinita paciencia y dedicación fue explicándome todos los rudimentos de la investigación, la programación, la escritura científica, y largo etcétera, así como un apoyo moral constante y una disposición total a atenderme y guiarme en cualquier momento y de una forma extremadamente rápida. Sinceramente gracias.

ÍNDICE GENERAL

1. Introducción **1**
1.1. Programación de la producción 4
1.2. Notación y clasificación 6
1.3. Motivación 10
1.4. Objetivos 12
1.5. Estructura de la Tesis Doctoral 13

2. Máquinas paralelas **15**
2.1. Máquinas paralelas idénticas 18
2.2. Máquinas paralelas uniformemente relacionadas 20
2.3. Máquinas paralelas no relacionadas 21
2.3.1. Revisión bibliografía 23
2.3.2. Estado del arte 30
2.3.2.1. Partial de Mokotoff y Jimeno 31
2.3.2.2. RBS de Ghirardi y Potts 33
2.3.2.3. CPLEX 11.0 35

3. Algoritmos iniciales propuestos **37**
3.1. Base de los algoritmos propuestos 38
3.1.1. Solución inicial 41
3.1.2. Búsqueda local de inserción 42

3.1.3. Búsqueda local de intercambio 44
3.1.4. Modificación de la solución obtenida y criterio de parada 46
3.1.5. Ensamblado del algoritmo 47
3.2. Algoritmo iterativo voraz 51
3.2.1. Algoritmo iterativo voraz estándar (IGS) 52
3.2.2. Algoritmo iterativo voraz dirigido (IGD) 57
3.3. Búsquedas locales en vecindarios pequeños o restringidos (RLS) 63
3.3.1. Búsqueda local restringida-No al mismo sitio (NSP) . 63
3.3.2. Búsqueda local restringida-Virtual (Vir) 65
3.4. Instancias 70
3.5. Implementación de los métodos propuestos 73
3.6. Análisis computacional 75
3.6.1. Intervalo U(1,100) 80
3.6.2. Intervalo U(10,100) 85
3.6.3. Intervalo U(100,200) 90
3.6.4. Intervalo U(100,120) 95
3.6.5. Intervalo U(1000,1100) 100
3.6.6. Intervalo de trabajos correlacionados 105
3.6.7. Intervalo de máquinas correlacionadas 110
3.6.8. Estudio conjunto de todos los intervalos 115
3.7. Otras variantes probadas 120
3.7.1. Variantes de la solución inicial 121
3.7.2. Variantes de la búsqueda local por inserción 123
3.7.3. Variantes de la búsqueda local por intercambio 124
3.7.4. Variantes de la selección de trabajos para la modificación 126
3.7.5. Variantes de la reconstrucción/búsqueda 128
3.7.6. Variantes del ensamblado del algoritmo 129
3.8. Resumen de los algoritmos iniciales propuestos 130

4. Algoritmos de selección dirigida 133
4.1. Selección de máquinas 134
4.2. Selección de trabajos 140
4.3. Selección del número de elementos a modificar 141
4.4. Métodos mejorados 143
4.5. Algoritmo NVST-IG+ 144
4.6. Análisis computacional 146
4.6.1. Intervalo U(1,100) 147
4.6.2. Intervalo U(10,100) 149
4.6.3. Intervalo de los trabajos correlacionados 151
4.6.4. Intervalo máquinas correlacionadas 152
4.6.5. Intervalo U(100,200) 154
4.6.6. Intervalo U(100,120) 156
4.6.7. Intervalo U(1000,1100) 158
4.6.8. Todos los intervalos 160
4.7. Resumen de los algoritmos de selección dirigida 166

5. Heurísticas de reducción de tamaño 169
5.1. Métodos propuestos 170
5.1.1. Métodos de reducción de tamaño 175
5.1.2. Algoritmos serie de reducción de tamaño 179
5.1.3. Algoritmos paralelos de reducción de tamaño 181
5.2. Implementación de los métodos propuestos 182
5.3. Análisis computacional 183
5.3.1. Resultados de los algoritmos serie 184
5.3.2. Resultado de los algoritmos paralelos 189
5.4. Resumen de algoritmos de reducción de tamaño 192

6. Máquinas opcionales y selección de trabajos 195
6.1. Modelos matemáticos, restricciones y limitaciones 197
6.1.1. Modelos matemáticos 197

6.1.2. Restricciones más habituales 199
6.1.3. Limitaciones de los modelos 200
6.2. No todas las máquinas (NAM) 203
6.2.1. Ranking de máquinas 205
6.2.2. Método de las selecciones de máquinas 209
6.2.3. Algoritmos para la resolución de NAM 212
6.2.4. Análisis computacional 214
6.3. No todos los trabajos (NAJ) 229
6.4. Resumen de máquinas opcionales y selección de trabajos . . . 232

7. Conclusiones y líneas futuras de investigación 235

REFERENCIAS 241

A. Tablas de los algoritmos iniciales 251
A.1. Tabla completa para el intervalo U(1,100) 252
A.2. Tabla completa para el intervalo U(10,100) 259
A.3. Tabla completa para el intervalo U(100,200) 266
A.4. Tabla completa para el intervalo U(100,120) 273
A.5. Tabla completa para el intervalo U(1000,1100) 280
A.6. Tabla completa para trabajos correlacionados 287
A.7. Tabla completa para máquinas correlacionadas 294

B. Gráficos y tablas de interacción de los algoritmos iniciales 301
B.1. Interacciones en el intervalo U(1,100) 302
B.2. Interacciones en el intervalo U(10,100) 305
B.3. Interacciones en el intervalo U(100,200) 308
B.4. Interacciones en el intervalo U(100,120) 311
B.5. Interacciones en el intervalo U(1000,1100) 314
B.6. Interacciones en el intervalo de trabajos correlacionados . . . 317
B.7. Interacciones en el intervalo de máquinas correlacionadas . . . 320

C. Resultado de calibración de los algoritmos 323
C.1. Calibración de IGS 323
C.2. Calibración de IGD 324
C.3. Calibración de NSP 325
C.4. Calibración de Virtual 326

D. Tablas y ANOVAS de los algoritmos de reducción de tamaño 329
D.1. Tablas de algoritmos de reducción de tamaño 330
D.2. ANOVAS de algoritmos de reducción de tamaño 372

E. Tablas y ANOVAS para el problema de máquinas opcionales 395
E.1. Tablas para un 50 % de máquinas sin usar 396
E.2. Tablas para un 20 % de máquinas sin usar 410
E.3. Tablas para un 80 % de máquinas sin usar 424
E.4. Anovas para un 50 % de máquinas sin usar 438
E.5. Anovas para un 20 % de máquinas sin usar 446
E.6. Anovas para un 80 % de máquinas sin usar 454

F. Artículos aceptados y congresos 463

ÍNDICE DE FIGURAS

2.1. Diagrama de Gantt con la asignación de trabajos en máquinas paralelas. 16

3.1. Diagrama de flujo de los algoritmos propuestos. 39
3.2. Pseudo-código del algoritmo base para todos los métodos propuestos. 41
3.3. Procedimiento de Búsqueda Local de Inserción empleado en la fase VND. 44
3.4. Procedimiento de Búsqueda Local de Intercambio empleado en la fase VND. 46
3.5. Ejemplo de algoritmo iterativo voraz estándar (IGS): Selección de un primer trabajo a destruir de la máquina con $C_{\text{máx}}$. 53
3.6. Ejemplo de algoritmo iterativo voraz estándar (IGS): Selección de un segundo trabajo a destruir de una máquina al azar. . . . 54
3.7. Ejemplo de algoritmo iterativo voraz estándar (IGS): Reasignación del primer trabajo en la mejor máquina posible. 55
3.8. Ejemplo de algoritmo iterativo voraz estándar (IGS): Reasignación del otro trabajo en la mejor máquina posible. 56
3.9. Ejemplo de algoritmo iterativo voraz estándar (IGS): Estado final. 57

3.10. Ejemplo de algoritmo iterativo voraz dirigido (IGD): Selección de un primer trabajo a destruir de la máquina con mayor $C_{\text{máx}}$. 58
3.11. Ejemplo de algoritmo iterativo voraz dirigido (IGD): Selección de un segundo trabajo a destruir de la máquina con mayor $C_{\text{máx}}$ en ese momento. 59
3.12. Ejemplo de algoritmo iterativo voraz dirigido (IGD): Reubicación de uno de los trabajos en la mejor posición posible. . . . 60
3.13. Ejemplo de algoritmo iterativo voraz dirigido (IGD): Reubicación del otro trabajo en la mejor posición posible. 61
3.14. Ejemplo de algoritmo iterativo voraz dirigido (IGD): Estado final. 62
3.15. Ejemplo del algoritmo de búsqueda local en vecindarios pequeños o restringidos tipo No al mismo sitio (NSP): Proceso. . 64
3.16. Ejemplo del algoritmo de búsqueda local en vecindarios pequeños o restringidos tipo No al mismo sitio (NSP): Estado final. 65
3.17. Ejemplo 1 del algoritmo de búsqueda local en vecindarios pequeños o restringidos tipo Virtual: Proceso. 67
3.18. Ejemplo 1 del algoritmo de búsqueda local en vecindarios pequeños o restringidos tipo Virtual: Estado final. 68
3.19. Ejemplo 2 del algoritmo de búsqueda local en vecindarios pequeños o restringidos tipo Virtual: Proceso. 69
3.20. Ejemplo 2 del algoritmo de búsqueda local en vecindarios pequeños o restringidos tipo Virtual: Estado final. 70
3.21. Gráfico ANOVA de los algoritmos CPLEX, Partial,RBS, IGS, IGD, NSP y Vir para el intervalo de distribución de tiempos de proceso U(1,100). 85
3.22. Gráfico ANOVA de los algoritmos CPLEX, Partial, RBS, IGS, IGD, NSP y Vir para el intervalo de distribución de tiempos de proceso U(10,100). 90

3.23. Gráfico ANOVA de los algoritmos CPLEX, Partial, RBS, IGS, IGD, NSP y Vir para el intervalo de distribución de tiempos de proceso U(100,200). 95
3.24. Gráfico ANOVA de los algoritmos CPLEX, Partial, RBS, IGS, IGD, NSP y Vir para el intervalo de distribución de tiempos de proceso U(100,120). 100
3.25. Gráfico ANOVA de los algoritmos CPLEX, Partial, RBS, IGS, IGD, NSP y Vir para el intervalo de distribución de tiempos de proceso U(1000,1100). 105
3.26. Gráfico ANOVA de los algoritmos CPLEX, Partial, RBS, IGS, IGD, NSP y Vir para el intervalo de distribución de tiempos de proceso de trabajos correlacionados. 110
3.27. Gráfico ANOVA de los algoritmos CPLEX, Partial, RBS, IGS, IGD, NSP y Vir para el intervalo de distribución de tiempos de proceso de máquinas correlacionadas. 115
3.28. Gráfico ANOVA de los algoritmos CPLEX, Partial, RBS, IGS, IGD, NSP y Vir para todos los intervalos de distribución de tiempos de proceso estudiados. 118
3.29. Gráfico ANOVA de los algoritmos CPLEX, Partial, RBS, IGS, IGD, NSP y Vir para los intervalos de distribución de tiempos de proceso más estudiados en la literatura. 119

4.1. Distribución de probabilidad basada en la asignación de números triangulares de forma directa. 135
4.2. Distribución de probabilidad basada en la asignación de números triangulares de forma inversa. 136
4.3. Distribución de probabilidad basada por igual en triangulares directa e inversa. 137
4.4. Distribución de probabilidad asimétrica basada en distribuciones triangulares directa e inversa. 138

4.5. Distribución de probabilidad Uniforme para las máquinas con $C_{\text{máx}}$ y Triangular Inversa entre el resto. 139
4.6. Distribución de probabilidad Uniforme para las máquinas con $C_{\text{máx}}$ y Cuadrática entre el resto. 140
4.7. Diagrama de flujo del algoritmo NVST-IG+. 145
4.8. ANOVA del intervalo U(1,100). 148
4.9. ANOVA del intervalo U(10,100). 150
4.10. ANOVA del intervalo de trabajos correlacionados. 152
4.11. ANOVA del intervalo de máquinas correlacionados. 154
4.12. ANOVA del intervalo U(100,200). 156
4.13. ANOVA del intervalo U(100,120). 158
4.14. ANOVA del intervalo U(1000,1100). 160
4.15. Medias e intervalos de Tukey HSD con un 95 % de nivel de confianza para todos los algoritmos y todas las instancias estudiadas. 162
4.16. Gráfico para los diversos tiempos estudiados (en segundos) en los algoritmos CPLEX y NVST-IG+ y para todas las instancias. 166

5.1. Gráfico de los algoritmos serie para todas las instancias y 120 segundos de tiempo de parada 189
5.2. Gráfico para la interacción entre el tiempo de parada y los algoritmos paralelos probados, para todas las instancias e intervalos. 191
5.3. Algoritmos paralelos frente a serie: gráfico de medias para todas las instancias e intervalos y todos los tiempos de parada. 192

6.1. Orden de máquinas o ranking (NAM). 208
6.2. Método de selección de máquinas (NAM). 211
6.3. ANOVA para todos los intervalos con 50 % máquinas sin usar y 300 segundos. 222
6.4. ANOVA para todos los intervalos con 50 % máquinas sin usar y 60 segundos. 223

6.5. ANOVA para todos los intervalos con 20 % máquinas sin usar y 300 segundos. 224
6.6. ANOVA para todos los intervalos con 20 % máquinas sin usar y 60 segundos. 225
6.7. ANOVA para todos los intervalos con 80 % máquinas sin usar y 300 segundos. 226
6.8. ANOVA para todos los intervalos con 80 % máquinas sin usar y 60 segundos. 227
6.9. ANOVA con la interacción por % máquinas sin usar y algoritmo para todos los intervalos y 300 segundos. 228
6.10. ANOVA con la interacción por % máquinas sin usar y algoritmo para todos los intervalos y 60 segundos. 228
6.11. ANOVA del intervalo U(10,100) con 50 % trabajos realizados y 300 segundos. 231
6.12. ANOVA del intervalo Trabajos Correlacionados con 50 % trabajos realizados y 300 segundos. 232
6.13. ANOVA del intervalo U(100,200) con 50 % trabajos realizados y 300 segundos. 232

B.1. Gráfico de interacción de algoritmos frente número de trabajos, para la distribución de p_{ij} U(1,100). 302
B.2. Gráfico de interacción de algoritmos frente número de máquinas, para la distribución de p_{ij} U(1,100). 302
B.3. Gráfico de interacción de algoritmos frente número de trabajos, para la distribución de p_{ij} U(10,100). 305
B.4. Gráfico de interacción de algoritmos frente número de máquinas, para la distribución de p_{ij} U(10,100). 305
B.5. Gráfico de interacción de algoritmos frente número de trabajos, para la distribución de p_{ij} U(100,200). 308
B.6. Gráfico de interacción de algoritmos frente número de máquinas, para la distribución de p_{ij} U(100,200). 308

B.7. Gráfico de interacción de algoritmos frente número de trabajos, para la distribución de p_{ij} U(100,120). 311
B.8. Gráfico de interacción de algoritmos frente número de máquinas, para la distribución de p_{ij} U(100,120). 311
B.9. Gráfico de interacción de algoritmos frente número de trabajos, para la distribución de p_{ij} U(1000,1100). 314
B.10. Gráfico de interacción de algoritmos frente número de máquinas, para la distribución de p_{ij} U(1000,1100). 314
B.11. Gráfico de interacción de algoritmos frente número de trabajos, para tiempos de p_{ij} de trab. correlacionados. 317
B.12. Gráfico de interacción de algoritmos frente número de máquinas, para tiempos de p_{ij} de trab. correlacionados. 317
B.13. Gráfico de interacción de algoritmos frente número de trabajos, para tiempos de p_{ij} de máq. correlacionadas. 320
B.14. Gráfico de interacción de algoritmos frente número de máquinas, para tiempos de p_{ij} de máq. correlacionadas. 320

D.1. Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo U(1,100) y 15 segundos. 372
D.2. Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo U(1,100) y 30 segundos. 373
D.3. Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo U(1,100) y 60 segundos. 373
D.4. Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo U(1,100) y 120 segundos. 374
D.5. Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo U(1,100) y 240 segundos. 374
D.6. Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo U(1,100) y 300 segundos. 375
D.7. Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo U(10,100) y 15 segundos. 375

D.8. Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo U(10,100) y 30 segundos. 376
D.9. Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo U(10,100) y 60 segundos. 376
D.10. Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo U(10,100) y 120 segundos. 377
D.11. Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo U(10,100) y 240 segundos. 377
D.12. Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo U(10,100) y 300 segundos. 378
D.13. Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo Trabajos correlacionados y 15 segundos. . . . 378
D.14. Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo Trabajos correlacionados y 30 segundos. . . . 379
D.15. Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo Trabajos correlacionados y 60 segundos. . . . 379
D.16. Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo Trabajos correlacionados y 120 segundos. . . 380
D.17. Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo Trabajos correlacionados y 240 segundos. . . 380
D.18. Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo Trabajos correlacionados y 300 segundos. . . 381
D.19. Gráfico de medias de los algoritmos de reducción de tamaño (excepto 3J y 3J2M) para el intervalo Máquinas correlacionadas y 15 segundos. 381
D.20. Gráfico de medias de los algoritmos de reducción de tamaño (excepto 3J y 3J2M) para el intervalo Máquinas correlacionadas y 30 segundos. 382

D.21. Gráfico de medias de los algoritmos de reducción de tamaño (excepto 3J y 3J2M) para el intervalo Máquinas correlacionadas y 60 segundos. 382
D.22. Gráfico de medias de los algoritmos de reducción de tamaño (excepto 3J y 3J2M) para el intervalo Máquinas correlacionadas y 120 segundos. 383
D.23. Gráfico de medias de los algoritmos de reducción de tamaño (excepto 3J y 3J2M) para el intervalo Máquinas correlacionadas y 240 segundos. 383
D.24. Gráfico de medias de los algoritmos de reducción de tamaño (excepto 3J y 3J2M) para el intervalo Máquinas correlacionadas y 300 segundos. 384
D.25. Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo U(100,200) y 15 segundos. 384
D.26. Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo U(100,200) y 30 segundos. 385
D.27. Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo U(100,200) y 60 segundos. 385
D.28. Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo U(100,200) y 120 segundos. 386
D.29. Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo U(100,200) y 240 segundos. 386
D.30. Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo U(100,200) y 300 segundos. 387
D.31. Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo U(100,120) y 15 segundos. 387
D.32. Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo U(100,120) y 30 segundos. 388
D.33. Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo U(100,120) y 60 segundos. 388

D.34. Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo U(100,120) y 120 segundos. 389
D.35. Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo U(100,120) y 240 segundos. 389
D.36. Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo U(100,120) y 300 segundos. 390
D.37. Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo U(1000,1100) y 15 segundos. 390
D.38. Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo U(1000,1100) y 30 segundos. 391
D.39. Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo U(1000,1100) y 60 segundos. 391
D.40. Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo U(1000,1100) y 120 segundos. 392
D.41. Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo U(1000,1100) y 240 segundos. 392
D.42. Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo U(1000,1100) y 300 segundos. 393

E.1. ANOVA para el intervalo U(1,100) con 50 % máquinas quitadas y 300 segundos. 439
E.2. ANOVA para el intervalo U(1,100) con 50 % máquinas quitadas y 60 segundos. 439
E.3. ANOVA para el intervalo U(10,100) con 50 % máquinas quitadas y 300 segundos. 440
E.4. ANOVA para el intervalo U(10,100) con 50 % máquinas quitadas y 60 segundos. 440
E.5. ANOVA para el intervalo Trabajos Correlacionados con 50 % máquinas quitadas y 300 segundos. 441
E.6. ANOVA para el intervalo Trabajos Correlacionados con 50 % máquinas quitadas y 60 segundos. 441

E.7. ANOVA para el intervalo Máquinas Correlacionadas con 50 % máquinas quitadas y 300 segundos. 442
E.8. ANOVA para el intervalo Máquinas Correlacionadas con 50 % máquinas quitadas y 60 segundos. 442
E.9. ANOVA para el intervalo U(100,200) con 50 % máquinas quitadas y 300 segundos. 443
E.10. ANOVA para el intervalo U(100,200) con 50 % máquinas quitadas y 60 segundos. 443
E.11. ANOVA para el intervalo U(100,120) con 50 % máquinas quitadas y 300 segundos. 444
E.12. ANOVA para el intervalo U(100,120) con 50 % máquinas quitadas y 60 segundos. 444
E.13. ANOVA para el intervalo U(1000,1100) con 50 % máquinas quitadas y 300 segundos. 445
E.14. ANOVA para el intervalo U(1000,1100) con 50 % máquinas quitadas y 60 segundos. 445
E.15. ANOVA para el intervalo U(1,100) con 20 % máquinas quitadas y 300 segundos. 447
E.16. ANOVA para el intervalo U(1,100) con 20 % máquinas quitadas y 60 segundos. 447
E.17. ANOVA para el intervalo U(10,100) con 20 % máquinas quitadas y 300 segundos. 448
E.18. ANOVA para el intervalo U(10,100) con 20 % máquinas quitadas y 60 segundos. 448
E.19. ANOVA para el intervalo Trabajos Correlacionados con 20 % máquinas quitadas y 300 segundos. 449
E.20. ANOVA para el intervalo Trabajos Correlacionados con 20 % máquinas quitadas y 60 segundos. 449
E.21. ANOVA para el intervalo Máquinas Correlacionadas con 20 % máquinas quitadas y 300 segundos. 450

E.22. ANOVA para el intervalo máquinas Correlacionadas con 20 % máquinas quitadas y 60 segundos. 450
E.23. ANOVA para el intervalo U(100,200) con 20 % máquinas quitadas y 300 segundos. 451
E.24. ANOVA para el intervalo U(100,200) con 20 % máquinas quitadas y 60 segundos. 451
E.25. ANOVA para el intervalo U(100,120) con 20 % máquinas quitadas y 300 segundos. 452
E.26. ANOVA para el intervalo U(100,120) con 20 % máquinas quitadas y 60 segundos. 452
E.27. ANOVA para el intervalo U(1000,1100) con 20 % máquinas quitadas y 300 segundos. 453
E.28. ANOVA para el intervalo U(1000,1100) con 20 % máquinas quitadas y 60 segundos. 453
E.29. ANOVA para el intervalo U(1,100) con 80 % máquinas quitadas y 300 segundos. 455
E.30. ANOVA para el intervalo U(1,100) con 80 % máquinas quitadas y 60 segundos. 455
E.31. ANOVA para el intervalo U(10,100) con 80 % máquinas quitadas y 300 segundos. 456
E.32. ANOVA para el intervalo U(10,100) con 80 % máquinas quitadas y 60 segundos. 456
E.33. ANOVA para el intervalo Trabajos Correlacionados con 80 % máquinas quitadas y 300 segundos. 457
E.34. ANOVA para el intervalo Trabajos Correlacionados con 80 % máquinas quitadas y 60 segundos. 457
E.35. ANOVA para el intervalo Máquinas Correlacionadas con 80 % máquinas quitadas y 300 segundos. 458
E.36. ANOVA para el intervalo Máquinas Correlacionadas con 80 % máquinas quitadas y 60 segundos. 458

E.37. ANOVA para el intervalo U(100,200) con 80 % máquinas quitadas y 300 segundos. 459
E.38. ANOVA para el intervalo U(100,200) con 80 % máquinas quitadas y 60 segundos. 459
E.39. ANOVA para el intervalo U(100,120) con 80 % máquinas quitadas y 300 segundos. 460
E.40. ANOVA para el intervalo U(100,120) con 80 % máquinas quitadas y 60 segundos. 460
E.41. ANOVA para el intervalo U(1000,1100) con 80 % máquinas quitadas y 300 segundos. 461
E.42. ANOVA para el intervalo U(1000,1100) con 80 % máquinas quitadas y 60 segundos. 461

ÍNDICE DE TABLAS

2.1. Tabla de tiempos de proceso de ejemplo de asignación en máquinas paralelas. 16
2.2. Tabla resumen de la revisión bibliográfica. 29

3.1. Tabla de tiempos de proceso de ejemplo de ciclado: situación inicial. 48
3.2. Tabla de tiempos de proceso de ejemplo de ciclado tras la primera inserción. 49
3.3. Tabla de tiempos de proceso de ejemplo de ciclado tras la segunda inserción. 49
3.4. Tabla de tiempos de proceso de ejemplo de ciclado tras la fase de intercambio. 50
3.5. Tabla del porcentaje del número de instancias resueltas por CPLEX 11.0, tras 2 horas de tiempo de CPU como máximo con cada instancia, para todos los intervalos de tiempos de proceso. 79
3.6. Tabla de medias, error relativo porcentual medio y limites inferior y superior del intervalo de confianza del 95 % para el intervalo U(1,100). 81

3.7. Comparación general entre los algoritmos CPLEX, Partial, RBS, IGS, IGD, NSP y Vir para el intervalo de distribución de tiempos de proceso U(1,100). 82
3.8. Comparación entre los algoritmos agrupados por número de trabajos CPLEX, Partial, RBS, IGS, IGD, NSP y Vir para el intervalo de distribución de tiempos de proceso U(1,100). . . 83
3.9. Comparación entre los algoritmos agrupados por número de máquinas CPLEX, Partial, RBS, IGS, IGD, NSP y Vir para el intervalo de distribución de tiempos de proceso U(1,100). . . 84
3.10. Tabla de medias, error relativo porcentual medio y limites inferior y superior del intervalo de confianza del 95 % para el intervalo U(10,100). 86
3.11. Comparación general entre los algoritmos CPLEX, Partial, RBS, IGS, IGD, NSP y Vir para el intervalo de distribución de tiempos de proceso U(10,100). 87
3.12. Comparación entre los algoritmos agrupados por número de trabajos CPLEX, Partial, RBS, IGS, IGD, NSP y Vir para el intervalo de distribución de tiempos de proceso U(10,100). . . 88
3.13. Comparación entre los algoritmos agrupados por número de máquinas CPLEX, Partial, RBS, IGS, IGD, NSP y Vir para el intervalo de distribución de tiempos de proceso U(10,100). . . 89
3.14. Tabla de medias, error relativo porcentual medio y limites inferior y superior del intervalo de confianza del 95 % para el intervalo U(100,200). 91
3.15. Comparación general entre los algoritmos CPLEX, Partial, RBS, IGS, IGD, NSP y Vir para el intervalo de distribución de tiempos de proceso U(100,200). 92
3.16. Comparación entre los algoritmos agrupados por número de trabajos CPLEX, Partial, RBS, IGS, IGD, NSP y Vir para el intervalo de distribución de tiempos de proceso U(100,200). . 93

3.17. Comparación entre los algoritmos agrupados por número de máquinas CPLEX, Partial, RBS, IGS, IGD, NSP y Vir para el intervalo de distribución de tiempos de proceso U(100,200). . 94
3.18. Tabla de medias, error relativo porcentual medio y limites inferior y superior del intervalo de confianza del 95 % para el intervalo U(100,120). 96
3.19. Comparación general entre los algoritmos CPLEX, Partial, RBS, IGS, IGD, NSP y Vir para el intervalo de distribución de tiempos de proceso U(100,120). 97
3.20. Comparación entre los algoritmos agrupados por número de trabajos CPLEX, Partial, RBS, IGS, IGD, NSP y Vir para el intervalo de distribución de tiempos de proceso U(100,120). . 98
3.21. Comparación entre los algoritmos agrupados por número de máquinas CPLEX, Partial, RBS, IGS, IGD, NSP y Vir para el intervalo de distribución de tiempos de proceso U(100,120). . 99
3.22. Tabla de medias, error relativo porcentual medio y limites inferior y superior del intervalo de confianza del 95 % para el intervalo U(1000,1100). 101
3.23. Comparación general entre los algoritmos CPLEX, Partial, RBS, IGS, IGD, NSP y Vir para el intervalo de distribución de tiempos de proceso U(1000,1100). 102
3.24. Comparación entre los algoritmos agrupados por número de trabajos CPLEX, Partial, RBS, IGS, IGD, NSP y Vir para el intervalo de distribución de tiempos de proceso U(1000,1100). 103
3.25. Comparación entre los algoritmos agrupados por número de máquinas CPLEX, Partial, RBS, IGS, IGD, NSP y Vir para el intervalo de distribución de tiempos de proceso U(1000,1100). 104
3.26. Tabla de medias, error relativo porcentual medio y limites inferior y superior del intervalo de confianza del 95 % para trabajos correlacionados. 106

3.27. Comparación general entre los algoritmos CPLEX, Partial, RBS, IGS, IGD, NSP y Vir para el intervalo de distribución de tiempos de proceso de trabajos correlacionados. 107
3.28. Comparación entre los algoritmos agrupados por número de trabajos CPLEX, Partial, RBS, IGS, IGD, NSP y Vir para el intervalo de distribución de tiempos de proceso de trabajos correlacionados. 108
3.29. Comparación entre los algoritmos agrupados por número de máquinas CPLEX, Partial, RBS, IGS, IGD, NSP y Vir para el intervalo de distribución de tiempos de proceso de trabajos correlacionados. 109
3.30. Tabla de medias, error relativo porcentual medio y limites inferior y superior del intervalo de confianza del 95 % para máquinas correlacionadas. 111
3.31. Comparación general entre los algoritmos CPLEX, Partial, RBS, IGS, IGD, NSP y Vir para el intervalo de distribución de tiempos de proceso de máquinas correlacionadas. 112
3.32. Comparación entre los algoritmos agrupados por número de trabajos CPLEX, Partial, RBS, IGS, IGD, NSP y Vir para el intervalo de distribución de tiempos de proceso de máquinas correlacionadas. 113
3.33. Comparación entre los algoritmos agrupados por número de máquinas CPLEX, Partial, RBS, IGS, IGD, NSP y Vir para el intervalo de distribución de tiempos de proceso de máquinas correlacionadas. 114
3.34. Comparación entre los algoritmos CPLEX, Partial, RBS, IGS, IGD, NSP y Vir para todos los intervalos de distribución de tiempos de proceso estudiados. 116

3.35. Comparación entre los algoritmos CPLEX, Partial, RBS, IGS, IGD, NSP y Vir en tanto por cien de desviación respecto al mejor algoritmo. 117
3.36. Comparación de otras variantes probadas para la solución inicial. 122
3.37. Comparación de otras variantes probadas para la búsqueda local por inserción. 123
3.38. Comparación de otras variantes probadas para la búsqueda por intercambio: estrategias first, best, a tres bandas y combinaciones de ellas. 125
3.39. Comparación de otras variantes probadas para la búsqueda por intercambio: admisión de peores soluciones. 126
3.40. Comparación de otras variantes probadas para la selección de trabajos a destruir: número fijo vs tanto por cien. 127
3.41. Comparación de otras variantes probadas para la selección de trabajos a destruir: aumento progresivo de la destrucción si no se mejora el $C_{\text{máx}}$. 127
3.42. Comparación de otras mejoras probadas para la reconstrucción y búsqueda. 128
3.43. Comparación de otras mejoras probadas para el ensamblado del algoritmo: reinicio, sin intercambios, orden inverso de la inserción e intercambio, sin fase de modificación y combinación variantes más prometedoras. 130

4.1. Comparación entre los algoritmos CPLEX, Partial, RBS, IGS, NSP, Vir, IG+, NSP+ y Vir+ para el intervalo de distribución de tiempos de proceso $U(1, 100)$. 147
4.2. Comparación entre los algoritmos CPLEX, Partial, RBS, IGS, NSP, Vir, IG+, NSP+ y Vir+ para el intervalo de distribución de tiempos de proceso $U(10, 100)$. 149

4.3. Comparación entre los algoritmos CPLEX, Partial, RBS, IGS, NSP, Vir, IG+, NSP+ y Vir+ para el intervalo de distribución de tiempos de proceso de trabajos correlacionados. 151
4.4. Comparación entre los algoritmos CPLEX, Partial, RBS, IGS, NSP, Vir, IG+, NSP+ y Vir+ para el intervalo de distribución de tiempos de proceso de máquinas correlacionadas. 153
4.5. Comparación entre los algoritmos CPLEX, Partial, RBS, IGS, NSP, Vir, IG+, NSP+ y Vir+ para el intervalo de distribución de tiempos de proceso de $U(100, 200)$. 155
4.6. Comparación entre los algoritmos CPLEX, Partial, RBS, IGS, NSP, Vir, IG+, NSP+ y Vir+ para el intervalo de distribución de tiempos de proceso de $U(100, 120)$. 157
4.7. Comparación entre los algoritmos CPLEX, Partial, RBS, IGS, NSP, Vir, IG+, NSP+ y Vir+ para el intervalo de distribución de tiempos de proceso $U(1000, 1100)$. 159
4.8. Resumen de resultados para todos los intervalos y algoritmos. En negrita (cursiva) se representa el mejor (peor) resultado, respectivamente. 161
4.9. Resultados de las medias de los mejores algoritmos para diferentes tiempos de parada de CPU. En negrita (cursiva) se destacan los mejores (peores) resultados. 164

5.1. Tabla de instancias resueltas por CPLEX 11.0, tras 2 horas como máximo con cada instancia, así como sus medias, para todos los intervalos de tiempos de proceso. 171
5.2. Porcentaje de trabajos que no se asignan a ninguna de las tres máquinas más rapidas i_{1j}, i_{2j} o i_{3j} en la solucion de referencia dada por IBM-ILOG CPLEX 11.0 con un tiempo límite de 2 horas. 174
5.3. Ejemplo de valores de p_{ij} seleccionados por el método 3JB de reducción de tamaño. En negrita los valores seleccionados. . . 176

5.4. Ejemplo de valores seleccionados de p_{ij} por el método de reducción de tamaño 2MB. En negrita los valores seleccionados. 177
5.5. Ejemplo de valores de p_{ij} seleccionados por DIG para el método de reducción de tamaño. En negrita los valores seleccionados. 178
5.6. Desviación porcentual relativa media para todos los intervalos excepto máquinas correlacionadas para CPLEX, DIG, 3J y 3J2M. 185
5.7. Desviación porcentual relativa media para los algoritmos serie CPLEX, DIG, 3JD, 2JDi y 2JDi(90) con diferentes tiempos de CPU como criterio de parada. En negrita (cursiva) se representan los mejores (peores) valores, respectivamente. 187
5.8. Desviación porcentual relativa media para los algoritmos paralelo M-CPLEX, M-DIG y M-2JDi(90) con diferentes tiempos de CPU como criterio de parada. En negrita (cursiva) se representan los mejores (peores) valores, respectivamente. 190

6.1. Ejemplo NAM: Tabla inicial. 206
6.2. Ejemplo NAM: Tabla de tres menores valores por trabajo y cuarto menor valor en negrita. 206
6.3. Ejemplo NAM: Tabla de diferencia de los tres menores valores con el cuarto. $C_o = C'_i$ si $C'_i <> 0$ ó $C_o = C_i$ de la tabla 6.1 si $C'_i = 0$ 207
6.4. Tabla de los resultados de las calibraciones para los algoritmos NAM propuestos. Valores en segundos. 214
6.5. Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para todos los intervalos y 50 % de máquinas sin usar con 300 segundos. 216
6.6. Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para todos los intervalos y 50 % de máquinas sin usar con 60 segundos. 217

6.7. Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para todos los intervalos y 20 % de máquinas sin usar con 300 segundos. 218
6.8. Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para todos los intervalos y 20 % de máquinas sin usar con 60 segundos. 218
6.9. Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para todos los intervalos y 80 % de máquinas sin usar con 300 segundos. 219
6.10. Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para todos los intervalos y 80 % de máquinas sin usar con 60 segundos. 220
6.11. Tabla porcentual resumen para la media de todos los intervalos y 20 %, 50 % y 80 % de máquinas sin usar con 60 y 300 segundos. 221
6.12. Tabla porcentual de la media de las diferencias respecto a la cota minima para todos los intervalos y 50 % de trabajos realizados con 300 segundos. 230

A.1. Tabla completa para el intervalo U(1,100). 258
A.2. Tabla completa para el intervalo U(10,100). 265
A.3. Tabla completa para el intervalo U(100,200). 272
A.4. Tabla completa para el intervalo U(100,120). 279
A.5. Tabla completa para el intervalo U(1000,1100). 286
A.6. Tabla completa de trabajos correlacionados. 293
A.7. Tabla completa para máquinas correlacionadas. 300

B.1. Tabla de datos estadísticos agrupados por trabajos para el intervalo U(1,100). 303
B.2. Tabla de datos estadísticos agrupados por máquinas para el intervalo U(1,100). 304

B.3. Tabla de datos estadísticos agrupados por trabajos para el intervalo U(10,100). 306
B.4. Tabla de datos estadísticos agrupados por máquinas para el intervalo U(10,100). 307
B.5. Tabla de datos estadísticos agrupados por trabajos para el intervalo U(100,200). 309
B.6. Tabla de datos estadísticos agrupados por máquinas para el intervalo U(100,200). 310
B.7. Tabla de datos estadísticos agrupados por trabajos para el intervalo U(100,120). 312
B.8. Tabla de datos estadísticos agrupados por máquinas para el intervalo U(100,120). 313
B.9. Tabla de datos estadísticos agrupados por trabajos para el intervalo U(1000,1100). 315
B.10. Tabla de datos estadísticos agrupados por máquinas para el intervalo U(1000,1100). 316
B.11. Tabla de datos estadísticos agrupados por trabajos para trabajos correlacionados 318
B.12. Tabla de datos estadísticos agrupados por máquinas para trabajos correlacionados 319
B.13. Tabla de datos estadísticos agrupados por trabajos para máquinas correlacionadas 321
B.14. Tabla de datos estadísticos agrupados por máquinas para máquinas correlacionadas 322

C.1. Calibración del algoritmo IGS: primera parte. 324
C.2. Calibración del algoritmo IGS: segunda parte. 324
C.3. Calibración del algoritmo IGD. 325
C.4. Calibración del algoritmo NSP. 325
C.5. Calibración del algoritmo Vir: Primera parte. 326
C.6. Calibración del algoritmo Vir: Segunda parte. 327

D.1. Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo U(1,100) de los algoritmos de reducción de tamaño para 15 segundos como tiempo de parada. 330
D.2. Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo U(1,100) de los algoritmos de reducción de tamaño para 30 segundos como tiempo de parada. 331
D.3. Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo U(1,100) de los algoritmos de reducción de tamaño para 60 segundos como tiempo de parada. 332
D.4. Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo U(1,100) de los algoritmos de reducción de tamaño para 120 segundos como tiempo de parada.333
D.5. Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo U(1,100) de los algoritmos de reducción de tamaño para 240 segundos como tiempo de parada.334
D.6. Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo U(1,100) de los algoritmos de reducción de tamaño para 300 segundos como tiempo de parada.335
D.7. Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo U(10,100) de los algoritmos de reducción de tamaño para 15 segundos como tiempo de parada. 336
D.8. Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo U(10,100) de los algoritmos de reducción de tamaño para 30 segundos como tiempo de parada. 337
D.9. Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo U(10,100) de los algoritmos de reducción de tamaño para 60 segundos como tiempo de parada. 338
D.10. Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo U(10,100) de los algoritmos de reducción de tamaño para 120 segundos como tiempo de parada.339

D.11. Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo U(10,100) de los algoritmos de reducción de tamaño para 240 segundos como tiempo de parada. 340
D.12. Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo U(10,100) de los algoritmos de reducción de tamaño para 300 segundos como tiempo de parada. 341
D.13. Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo trabajos correlacionados de los algoritmos de reducción de tamaño para 15 segundos como tiempo de parada. 342
D.14. Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo trabajos correlacionados de los algoritmos de reducción de tamaño para 30 segundos como tiempo de parada. 343
D.15. Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo trabajos correlacionados de los algoritmos de reducción de tamaño para 60 segundos como tiempo de parada. 344
D.16. Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo trabajos correlacionados de los algoritmos de reducción de tamaño para 120 segundos como tiempo de parada. 345
D.17. Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo trabajos correlacionados de los algoritmos de reducción de tamaño para 240 segundos como tiempo de parada. 346
D.18. Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo trabajos correlacionados de los algoritmos de reducción de tamaño para 300 segundos como tiempo de parada. 347

D.19. Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo máquinas correlacionadas de los algoritmos de reducción de tamaño para 15 segundos como tiempo de parada. 348
D.20. Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo máquinas correlacionadas de los algoritmos de reducción de tamaño para 30 segundos como tiempo de parada. 349
D.21. Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo máquinas correlacionadas de los algoritmos de reducción de tamaño para 60 segundos como tiempo de parada. 350
D.22. Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo máquinas correlacionadas de los algoritmos de reducción de tamaño para 120 segundos como tiempo de parada. 351
D.23. Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo máquinas correlacionadas de los algoritmos de reducción de tamaño para 240 segundos como tiempo de parada. 352
D.24. Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo máquinas correlacionadas de los algoritmos de reducción de tamaño para 300 segundos como tiempo de parada. 353
D.25. Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo U(100,200) de los algoritmos de reducción de tamaño para 15 segundos como tiempo de parada. 354
D.26. Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo U(100,200) de los algoritmos de reducción de tamaño para 30 segundos como tiempo de parada. 355

D.27. Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo U(100,200) de los algoritmos de reducción de tamaño para 60 segundos como tiempo de parada. 356
D.28. Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo U(100,200) de los algoritmos de reducción de tamaño para 120 segundos como tiempo de parada. 357
D.29. Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo U(100,200) de los algoritmos de reducción de tamaño para 240 segundos como tiempo de parada. 358
D.30. Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo U(100,200) de los algoritmos de reducción de tamaño para 300 segundos como tiempo de parada. 359
D.31. Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo U(100,120) de los algoritmos de reducción de tamaño para 15 segundos como tiempo de parada. 360
D.32. Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo U(100,120) de los algoritmos de reducción de tamaño para 30 segundos como tiempo de parada. 361
D.33. Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo U(100,120) de los algoritmos de reducción de tamaño para 60 segundos como tiempo de parada. 362
D.34. Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo U(100,120) de los algoritmos de reducción de tamaño para 120 segundos como tiempo de parada. 363
D.35. Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo U(100,120) de los algoritmos de reducción de tamaño para 240 segundos como tiempo de parada. 364
D.36. Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo U(100,120) de los algoritmos de reducción de tamaño para 300 segundos como tiempo de parada. 365

D.37. Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo U(1000,1100) de los algoritmos de reducción de tamaño para 15 segundos como tiempo de parada. 366
D.38. Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo U(1000,1100) de los algoritmos de reducción de tamaño para 30 segundos como tiempo de parada. 367
D.39. Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo U(1000,1100) de los algoritmos de reducción de tamaño para 60 segundos como tiempo de parada. 368
D.40. Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo U(1000,1100) de los algoritmos de reducción de tamaño para 120 segundos como tiempo de parada. 369
D.41. Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo U(1000,1100) de los algoritmos de reducción de tamaño para 240 segundos como tiempo de parada. 370
D.42. Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo U(1000,1100) de los algoritmos de reducción de tamaño para 300 segundos como tiempo de parada. 371

E.1. Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo U(1,100) y 50 % de máquinas quitadas con 300 segundos. 396
E.2. Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo U(1,100) y 50 % de máquinas quitadas con 60 segundos. 397
E.3. Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo U(10,100) y 50 % de máquinas quitadas con 300 segundos. 398
E.4. Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo U(10,100) y 50 % de máquinas quitadas con 60 segundos. 399

E.5. Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo Trabajos Correlacionados y 50 % de máquinas quitadas con 300 segundos. . . . 400
E.6. Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo Trabajos Correlacionados y 50 % de máquinas quitadas con 60 segundos. . . . 401
E.7. Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo Máquinas Correlacionados y 50 % de máquinas quitadas con 300 segundos. . . . 402
E.8. Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo Máquinas Correlacionados y 50 % de máquinas quitadas con 60 segundos. . . . 403
E.9. Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo U(100,200) y 50 % de máquinas quitadas con 300 segundos. 404
E.10. Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo U(100,200) y 50 % de máquinas quitadas con 60 segundos. 405
E.11. Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo U(100,120) y 50 % de máquinas quitadas con 300 segundos. 406
E.12. Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo U(100,120) y 50 % de máquinas quitadas con 60 segundos. 407
E.13. Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo U(1000,1100) y 50 % de máquinas quitadas con 300 segundos. 408
E.14. Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo U(1000,1100) y 50 % de máquinas quitadas con 60 segundos. 409

E.15. Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo U(1,100) y 20 % de máquinas quitadas con 300 segundos. 410
E.16. Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo U(1,100) y 20 % de máquinas quitadas con 60 segundos. 411
E.17. Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo U(10,100) y 20 % de máquinas quitadas con 300 segundos. 412
E.18. Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo U(10,100) y 20 % de máquinas quitadas con 60 segundos. 413
E.19. Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo Trabajos Correlacionados y 20 % de máquinas quitadas con 300 segundos. . . . 414
E.20. Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo Trabajos Correlacionados y 20 % de máquinas quitadas con 60 segundos. . . . 415
E.21. Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo Máquinas Correlacionados y 20 % de máquinas quitadas con 300 segundos. . . . 416
E.22. Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo Máquinas Correlacionados y 20 % de máquinas quitadas con 60 segundos. . . . 417
E.23. Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo U(100,200) y 20 % de máquinas quitadas con 300 segundos. 418
E.24. Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo U(100,200) y 20 % de máquinas quitadas con 60 segundos. 419

E.25. Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo U(100,120) y 20 % de máquinas quitadas con 300 segundos. 420
E.26. Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo U(100,120) y 20 % de máquinas quitadas con 60 segundos. 421
E.27. Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo U(1000,1100) y 20 % de máquinas quitadas con 300 segundos. 422
E.28. Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo U(1000,1100) y 20 % de máquinas quitadas con 60 segundos. 423
E.29. Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo U(1,100) y 80 % de máquinas quitadas con 300 segundos. 424
E.30. Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo U(1,100) y 80 % de máquinas quitadas con 60 segundos. 425
E.31. Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo U(10,100) y 80 % de máquinas quitadas con 300 segundos. 426
E.32. Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo U(10,100) y 80 % de máquinas quitadas con 60 segundos. 427
E.33. Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo Trabajos Correlacionados y 80 % de máquinas quitadas con 300 segundos. . . . 428
E.34. Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo Trabajos Correlacionados y 80 % de máquinas quitadas con 60 segundos. . . . 429

E.35. Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo Máquinas Correlacionados y 80 % de máquinas quitadas con 300 segundos. . . . 430
E.36. Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo Máquinas Correlacionados y 80 % de máquinas quitadas con 60 segundos. . . . 431
E.37. Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo U(100,200) y 80 % de máquinas quitadas con 300 segundos. 432
E.38. Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo U(100,200) y 80 % de máquinas quitadas con 60 segundos. 433
E.39. Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo U(100,120) y 80 % de máquinas quitadas con 300 segundos. 434
E.40. Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo U(100,120) y 80 % de máquinas quitadas con 60 segundos. 435
E.41. Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo U(1000,1100) y 80 % de máquinas quitadas con 300 segundos. 436
E.42. Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo U(1000,1100) y 80 % de máquinas quitadas con 60 segundos. 437

CAPÍTULO 1

INTRODUCCIÓN

La programación de la producción es un proceso de toma de decisiones que desempeña un papel crucial en la fabricación así como en los sistemas de servicios e industrias. En el actual entorno competitivo, rápidamente cambiante, una programación efectiva se ha convertido en una necesidad para la supervivencia en el mercado. Las empresas tienen que cumplir con los plazos de producción y fechas de entrega comprometidos con los clientes y el no hacerlo puede derivar en una pérdida económica significativa y de confianza del cliente.

En términos generales, la programación de la producción se refiere a la asignación de unos recursos, generalmente limitados, a unas tareas, durante un tiempo. Es un proceso de toma de decisiones cuyo objetivo es la optimización de uno o más objetivos sometido a limitaciones o restricciones de toda índole. Los recursos y las tareas pueden tomar muchas formas. Por ejemplo, los recursos pueden ser máquinas en un taller, pistas de aterrizaje en un aeropuerto, los equipos de una obra de construcción, unidades de procesamiento por medio de ordenadores, etc. Las tareas correspondientes pueden adoptar las formas

siguientes: operaciones en un proceso de producción, despegues y aterrizajes en el aeropuerto, las etapas de una obra, la ejecución de programas de ordenador, etc. Además, cada tarea puede tener diferente tiempo de procesamiento, nivel de prioridad (o peso), o una fecha de vencimiento. El objetivo también puede tomar muchas formas. Uno de los objetivos es la reducción al mínimo del tiempo en el que se termina la última tarea (lo llamamos la minimización de makespan), otro la reducción al mínimo del número de tareas completadas después de la fecha de vencimiento, etc.

La programación de la producción comenzó a ser tomada en serio en la industria al principio del pasado siglo con los trabajos de Henry Gantt (Gantt, 1910 y Gantt, 1919) y otros pioneros que siguieron sus ideas. Sin embargo, fueron necesarios muchos años para que aparecieran las primeras publicaciones en la literatura de la investigación operativa. Algunas de estas primeras publicaciones aparecieron en la década de 1950. En concreto, tenemos los artículos de Smith (1956), Johnson (1954), y Jackson (1956). Durante la década de 1960 una parte importante de los trabajos se centraron en la programación tanto dinámica como entera para los problemas de programación de la producción. Después de que Richard Karp publicara su famoso artículo sobre teoría de la complejidad (Karp, 1975), la investigación en la década de 1970 se centró principalmente en la jerarquía de la complejidad de problemas de programación de la producción. En la década de 1980 se siguieron diferentes direcciones tanto en la parte teórica como en la práctica, con una atención creciente en la programación estocástica (Pinedo, 2008). Además, como los ordenadores personales empezaron a ser de amplia difusión en el ambiente empresarial, los sistemas de programación se estaban enfocando en el desarrollo de programas que pudieran utilizarse en la práctica.

La programación de la producción puede ser difícil, tanto desde un punto de vista teórico como práctico. Las dificultades teóricas son similares a las encontradas en otras ramas de la optimización combinatoria. Así, los resultados teóricos obtenidos para otros problemas de optimización se pueden aplicar, al

menos parcialmente, a los problemas de programación de la producción. Por desgracia, las dificultades encontradas durante la implementación son de un tipo completamente diferente. Están relacionadas con la modelización de los problemas de programación de la producción en el mundo real. En términos generales, se puede decir que cada problema de programación de la producción de la vida real es diferente, y requiere por lo tanto una modelización generalmente ad-hoc, que en la mayoría de los casos se tiene que hacer desde cero. Para una buena introducción en la teoría de la programación de la producción y sus aplicaciones pueden consultarse los trabajos de Conway, Maxwell y Miller (1967), Brucker (1995), Pinedo (2008) y Brucker (2005), entre muchos otros.

La planificación y programación de la producción es una herramienta importante en la toma de decisiones para la mayoría de las empresas tanto en el sector industrial como en el de servicios, así como para la mayoría de los entornos de procesamiento de información donde puede tener un gran impacto en la productividad de un proceso. En la industria, el fin de la programación de la producción es reducir al mínimo el tiempo de producción y los costos, teniéndose en cuenta que en una planta de producción se deben decidir cosas como lo que se debe hacer, cuándo, con qué personal, y/o con qué equipo. Del mismo modo, la programación de la producción de las empresas de servicios, tales como líneas aéreas y el transporte público, tiene como objetivo maximizar la eficiencia de las operaciones y reducir sus costos. Las modernas herramientas informaticas de programación de la producción superan ampliamente a los antiguos métodos de programación manual. Éstas proporcionan al planificador de producción unas poderosas interfaces gráficas que se puede utilizar para optimizar las cargas de trabajo en tiempo real en las diversas etapas de la producción (véase, p.e., Pinedo, 2009). Así, por ejemplo, las compañías aéreas podrían preferir minimizar el número de puertas del aeropuerto necesarias para sus aviones, con el fin de reducir los costos, y el software de programación puede permitir a los planificadores ver cómo puede hacerse esto mediante

el análisis de los horarios, el uso de las aeronaves, o el flujo de pasajeros. Las empresas utilizan la programación de la producción para planificar sus recursos humanos y materiales tanto en sentido temporal hacia atrás como hacia adelante. La programación de la producción hacia atrás es la planificación de las tareas con una fecha de vencimiento para determinar la fecha de inicio y/o cualquier variación de las necesidades requeridas, mientras que la programación hacia adelante es la planificación de las tareas con fecha de inicio para determinar la fecha de envío o la fecha de vencimiento. Véanse, por ejemplo los trabajos de Raghavan y Thompson (1987), Ovacik y Uzsoy (1996), Tsang y Voudouris (1997), Yu et al. (2002) y Pinedo (2009), de nuevo, entre muchos otros, ya que, como es bien conocido, la literatura existente sobre la programación de la producción es muy extensa.

1.1. Programación de la producción

En la programación de la producción hay dos fases claramente diferenciadas, la fase de secuenciación y la fase de programación. La secuenciación consiste en la asignación de recursos en el tiempo para la realización de un conjunto de tareas. Previa a la secuenciación, se supone que la planificación de la producción ya se ha realizado, es decir, se sabe qué productos se van a realizar, las cantidades de cada uno de ellos y la configuración, disposición y cantidad de recursos disponibles. Esta secuenciación es de vital importancia para las empresas. En el contexto de la presente Tesis Doctoral y sin pérdida de generalidad, nos referiremos a los recursos como máquinas y a cada pedido de la producción como trabajo. Así pues, la programación de la producción la podemos ver como una asignación de los trabajos a las máquinas (asignación), la ordenación de los trabajos asignados a cada máquina (secuenciación) y la obtención de los tiempos de inicio y finalización de cada trabajo (programación).

Procedamos a realizar una clasificación de los distintos problemas de

secuenciación atendiendo a la configuración productiva (Portmann, 1997 ó Pinedo, 2008):

- Problemas de una máquina: solo existe una máquina y cada trabajo solo tiene una operación. No hay problema de asignación pero si de secuenciación.
- Problemas de máquinas paralelas: existen dos o más máquinas dispuestas en paralelo, donde los trabajos solo tienen una única operación. En general, se tienen que asignar y secuenciar los trabajos en las máquinas, aunque dependerá del objetivo optimizar. Los problemas de máquinas paralelas pueden asimismo dividirse según la naturaleza de las máquinas en:
    - Máquinas paralelas idénticas: cada trabajo se procesa a la misma velocidad con independencia de la máquina a la que ha sido asignado.
    - Máquinas paralelas uniformes: algunas máquinas son capaces de procesar todos los trabajos más rápido (o lento) que las demás. A veces se las llama máquinas proporcionales.
    - Máquinas paralelas no relacionadas: la velocidad de proceso depende de la máquina y del trabajo que está procesando. Es un caso más general que engloba a los otros dos.
- Problemas de tipo taller. En estos problemas, existen varias etapas que se deben visitar para completar los trabajos. Adicionalmente, existe una relación de precedencia entre las etapas para cada trabajo. A esta relación se la denomina ruta. Los problemas de tipo taller se subdividen a la vez atendiendo a las características de las rutas en:
    - Taller abierto: no existe ninguna restricción en la ruta, por lo que las operaciones pueden efectuarse en cualquier orden.

- Taller de flujo: todos los trabajos tienen la misma ruta, es decir, existe una ordenación previa de etapas o máquinas que es la misma para todos los trabajos.
- Taller de trabajo: cada trabajo tiene su propia ruta a seguir dentro de las máquinas.

1.2. Notación y clasificación

Los problemas de programación de la producción vienen determinados por el número de trabajos y las operaciones a procesar, el número y tipo de máquinas, el flujo de las operaciones en las máquinas y los criterios de optimización, entre otros datos.
Se asume normalmente un número de trabajos y máquinas finito y determinista. Según Baker (1974) y Pinedo (2008), en los problemas de producción hay un conjunto N de trabajos, donde $N = 1, 2, ..., n$, que hay que procesar en un conjunto M de máquinas, $M = 1, 2, ..., m$. Utilizamos los subíndices j y k para referirnos a los trabajos y el subíndice i para las máquinas y donde p_{ij} representa el tiempo de proceso del trabajo j en la máquina i.

Para la clasificación de los diversos problemas de la producción se han propuesto numerosas notaciones, entre las que podemos destacar la de notación cuádruple empleada por Conway, Maxwell y Miller (1967), la algo más elaborada presentada por Rinnooy Kan (1976) y la más frecuentemente usada, introducida por Graham et al. (1979), basada en la tripleta $\alpha/\beta/\gamma$ y ampliada en trabajos como los de Blazewicz, Lenstra y Kan (1983), Lawler, Lenstra y Kan (1993), Blazewicz et al. (1996), Portmann (1997) y Pinedo (2008). Pasemos a detallar parte de esta notación con las aportaciones de los trabajos citados:

El campo α se subdivide a su vez en α_1 y α_2, donde α_2 representa el

número de máquinas fijado (en caso de no figurar dependerá del ejemplo a resolver) y α_1 puede tener los valores de:
1: una sola máquina.
P: máquinas paralelas idénticas.
Q: máquinas paralelas uniformes.
R: máquinas paralelas no relacionadas.
O: taller abierto.
F: taller de flujo.
J: taller de trabajo.

El campo β indica las características de los trabajos, las cuales son muy numerosas, por lo que los autores dividen este campo en varios subcampos. En el caso de no figurar de forma expresa, se entiende que no se da la característica indicada. Los subcampos más usuales son:
$pmtn$: se permite interrumpir las operaciones y continuarlas más tarde o en otra máquina (también se conoce como $prmp$).
$prec$: relaciones de precedencia.
r_j: tiempo en el cual el trabajo j esta listo para procesarse (*release date*).
S_{nsd}: existen tiempos de cambio o preparación en las máquinas antes de procesar un trabajo que no dependen de la secuencia. Se denotan por S_{ij}.
S_{sd}: existen tiempos de cambio o preparación en las máquinas antes de procesar un trabajo que sí dependen de la secuencia. Se denotan por S_{ijk}.
d_j: fecha de compromiso de realización de cada trabajo (*due date*). Se permite la finalización posterior pero con penalización.
$\overline{d_j}$: fecha de finalización obligatoria (*deadline*).
$prmu$: el orden de entrada de los trabajos es el mismo para todas las máquinas. Solo aplicable cuando $\alpha_1 = F$.
$brkdwn$: las máquinas están sujetas a averías o a periodos donde no pueden procesar trabajos.
nwt: sin esperas entre dos máquinas. A veces implica retrasar el inicio de las

operaciones de un trabajo.
$block$: bloqueo. Existe una capacidad (o buffer) finita para almacenar el producto en curso. Si se llena el buffer de una máquina, está quedará bloqueada por el último trabajo que ha procesado y no ha podido dejar en el buffer.
$recrc$: recirculación. Indica que un trabajo (o más) pueden necesitar ser procesados más de una vez por alguna (o varias) de las máquinas. Solo cuando $\alpha_1 = O, F, J$.
M_j: restricción de uso de máquinas. Implica que un trabajo (o más) solo puede ser procesado en determinadas máquinas.
$p_{ij} = p$: todos los tiempos de trabajo son iguales a p. Una variante común es hacer $p = 1$.

Finalmente, el campo γ nos da información acerca del criterio de optimización que se ha empleado. Antes definiremos una serie de conceptos y parámetros adicionales, que nos permitirán definir mejor las medidas de optimización (French, 1982).
a_j: periodo permitido para el proceso de trabajo. $a_j = d_j - r_j$ ó $a_j = \overline{d_j} - r_j$.
W_{ij}: tiempo de espera del trabajo j antes de empezar a procesarse en la máquina i.
W_j: tiempo total de espera del trabajo j.
C_i: tiempo de finalización de la maquina i.
$C_{\text{máx}}$: tiempo máximo de finalización de todos los trabajos o makespan. El makespan óptimo se denota por $C^*_{\text{máx}}$.
I_i: tiempo ocioso o sin procesar trabajos de la máquina i.
F_j: tiempo de flujo o cantidad de tiempo que el trabajo j permanece en el taller. $F_j = C_j - r_j$
L_j: holgura frente a la fecha de finalización de un trabajo. $L_j = C_j - d_j$ ó $L_j = C_j - \overline{d_j}$.
T_j: retraso de un trabajo j. $T_j = \text{máx}\{L_j, 0\}$.
E_j: adelanto de un trabajo j. $E_j = \text{máx}\{-L_j, 0\}$.

Procedamos a definir ahora algunos de los criterios de optimización más comunes:
$C_{\text{máx}}$: minimización del tiempo máximo de finalización de todos los trabajos o makespan.
$\overline{C}$: minimización del tiempo medio de finalización.
$F_{\text{máx}}$: minimización del tiempo máximo de flujo.
$\overline{F}$: minimización del tiempo medio de flujo.
$L_{\text{máx}}$: minimización de la máxima holgura.
$\overline{L}$: minimización de la holgura media.
$T_{\text{máx}}$: minimización del retraso máximo.
$\overline{T}$: minimización del retraso medio.
$E_{\text{máx}}$: minimización del adelanto máximo.
$\overline{E}$: minimización del adelanto medio.
N_T: minimización del número de trabajos retrasados.
$\sum_{j=1}^{n} E_j + T_j$: minimización de la suma de retrasos y adelantos.
$\sum_{i=1}^{m} I_i$: minimización de los tiempos ociosos.

Evidentemente, muchos de estos objetivos admiten una ponderación de los mismos sin más que añadir el correspondiente peso (denominado w_j) a sus componentes.

Como ejemplo de esta notación podemos ver $R4/pmtn/C_{\text{máx}}$ donde se tendría un problema de cuatro máquinas paralelas no relacionadas, en las que se permite interrumpir las operaciones y con el objetivo de minimizar el $C_{\text{máx}}$.

La notación de los problemas de máquinas paralelas con el objetivo de minimizar el $C_{\text{máx}}$ sería $P//C_{\text{máx}}$ para el caso de máquinas paralelas idénticas, $Q//C_{\text{máx}}$ para el caso de máquinas paralelas uniformemente relacionadas y $R//C_{\text{máx}}$ cuando las máquinas paralelas no están relacionadas.

1.3. Motivación

A la hora de iniciarse la presente Tesis Doctoral buscamos un problema de programación de la producción relativamente sencillo que tuviera amplia aplicación en distintos ámbitos y del que pudieran derivar problemas más complejos. Nos centramos en el problema máquinas paralelas, y dentro de él, en el de máquinas paralelas no relacionadas, ya que éste último es el más general de los tres problemas de máquinas paralelas. Este entorno, a pesar de parecer de carácter marcadamente teórico, tiene aplicaciones directas en el mundo empresarial, desde sencillos ejemplos como una empresa de reprografía hasta empresas cerámicas tan abundantes en nuestra comunidad. Observemos que en el sencillo caso de una empresa de reprografía donde se deban realizar distintos trabajos, de distintas características y se posean distintas máquinas, asignar los diferentes trabajos a las máquinas de forma que se finalice la totalidad de los trabajos lo antes posible, ya conlleva una tarea de programación de la producción. Para el caso de las empresas cerámicas, nos encontramos que algunas de ellas solo quieren secuenciar sus recursos más conflictivos: los hornos, así que toda su problemática inicial se reduce a una adecuada secuenciación de los trabajos en los hornos, dado que para el resto de la producción no tienen a priori mayor problema. Los distintos hornos tienen distintas características por lo que los trabajos que se efectúan en ellos tienen distintos tiempos de ejecución. Existen muchos otros casos donde el problema reside en una etapa crítica, la cual, secuenciada correctamente, proporciona una ganancia neta muy alta en todo el proceso. Evidentemente, una vez se han optimizado los hornos, se puede comprobar como se puede optimizar el resto de la producción, por lo que entramos en problemas más complejos como el del taller de flujo, por citar un ejemplo.

Una vez determinado el problema base sobre el que queríamos trabajar, pensamos no quedarnos solo ahí, si no dar una paso más e introducirnos en una variante del problema que consideramos interesante, la cual consiste en

secuenciar el problema en el caso de no tener que usar todas las máquinas o no realizar todos los trabajos. La novedad estriba en que el uso de todas las m máquinas disponibles es opcional y se relaja la restricción de que todos los n trabajos deben secuenciarse, es decir, se permite la selección de los trabajos. Este problema generalizado permite modelizar situaciones donde se pueden usar más o menos máquinas de las disponibles, como cuando se desea reducir o ampliar la capacidad de un taller, o donde se quiere seleccionar los trabajos a procesar para, por ejemplo, fabricar aquellos más rentables o menos problemáticos. En los tiempos actuales de crisis, donde la demanda puede disminuir para algunas empresas, el ser capaces de decidir qué máquinas de las que tenemos nos conviene parar y cuáles mantener activas para optimizar la carga de trabajo que tenemos, puede tener una alta repercusión económica que debería estar sustentada en pruebas objetivas y resultados fiables para tomar la decisión más adecuada. Es por ello que consideramos que la modelización de problemas donde la totalidad de máquinas y trabajos no sea una restricción, debería ser un tema a tener en cuenta.

Finalmente, dado el actual proceso de integración en el mismo ordenador de varios núcleos de procesamiento, del que los paquetes de software comerciales empiezan a hacer uso con la paralelización de tareas, creímos conveniente adentramos en este tipo de programación. Lo que queríamos, es que los algoritmos con mejores resultados que desarrolláramos, finalmente emplearan todos los núcleos que hubiera disponibles aunque fuera de una manera simple en aras de una iniciación en la programación paralela.

Con estas motivaciones, procedimos a marcarnos una serie de objetivos a alcanzar durante la realización de esta Tesis, que pasamos a detallar a continuación.

1.4. Objetivos

El objetivo principal de esta Tesis Doctoral es el desrrollar unos algoritmos sencillos, pero potentes, para la resolución del problema de máquinas paralelas no relacionadas con el objetivo de minimizar el $C_{\text{máx}}$. Con este mismo objetivo, se pretende realizar una generalización del mismo para el caso en el que no se empleen todas las máquinas o no se realicen todos los trabajos. Para ello, se llevará a cabo un estudio profundo de la literatura concerniente al problema tratado para poder estudiar los distintos métodos ya existentes para la resolución de este problema, así como el poder identificar el estado del arte concerniente a $R//C_{\text{máx}}$. Así mismo, se proponen una serie de instancias lo suficientemente representativas como para efectuar amplios tests computacionales y estadísticos en los que basar nuestros resultados y conclusiones. También se tiene en cuenta el comportamiento de un moderno solver comercial para la resolución de este tipo de problemas, dado que, como se verá, no se comparaban los distintos resultados de la literatura con los ofrecidos por un solver comercial. Una vez efectuado esto, nos planteamos desarrollar unos métodos cuanto menos competitivos con el estado del arte a la par que sencillos para una fácil implementación de los mismos. El lograr esto se antoja una labor árdua toda vez que hacía ya varios años que no se aportaba nada nuevo capaz de igualar siquiera el estado del arte. Una vez logrados unos algoritmos sencillos y potentes nos planteamos tratar de afinarlos lo máximo posible para que sean capaces de obtener aún mejores resultados. Tras esto, nos ponemos también como objetivo adentrarnos en el mundo de la programación paralela dado que los modernos ordenadores ya poseen varios núcleos integrados dentro del mismo chip y los más novedosos solvers hacen amplio uso de este hecho mediante la paralelización de tareas. Finalmente se plantea estudiar los casos en el que el número de máquinas es opcional o que la cantidad de trabajos es seleccionable.

1.5. Estructura de la Tesis Doctoral

En el capítulo 2 se hace una primera aproximación a los problemas de máquinas paralelas. Después de esta primera aproximación, tanto en su formato de máquinas paralelas idénticas, uniformemente relacionadas o no relacionadas, se pasa a profundizar en el problema fundamental de esta Tesis, es decir, el de las máquinas paralelas no relacionadas. Se procede a ello mediante un estudio profundo de la bibliografía existente sobre el problema y un análisis más detallado de los métodos que constituyen el estado del arte, con mención al solver comercial IBM-ILOG CPLEX como opción para la resolución de este problema.

En el capítulo 3 presentamos los primeros algoritmos basados en un bucle general con búsquedas locales en vecindarios cada vez mayores seguidos de una modificación parcial de la solución. Se introduce también el grupo de instancias que se emplearán a lo largo de toda la Tesis y se muestran los resultados de estos algoritmos enfrentados al estado del arte.

A lo largo del capítulo 4 se presentan nuevos métodos que modifican a los expuestos en el capítulo anterior con la idea de obtener una mejora significativa, para finalizar en un algoritmo híbrido que recoge las mejores características de todos los algoritmos propuestos.

En el capítulo 5 se propone un acercamiento al problema desde una perspectiva diferente, basada en una reducción de las dimensiones del problema. Así mismo se proponen varios algoritmos paralelos para la resolución del problema y se hace una amplia comparativa entre los mejores algoritmos propuestos.

El capítulo 6 trata de la introducción al problema de no emplear todas las máquinas o de no realizar todos los trabajos, es decir, el problema donde existen máquinas opcionales y selección de trabajos. Se da primeramente la formulación matemática de éstos problemas indicándose sus restricciones más habituales así como sus limitaciones, para acabar proponiendo unos algoritmos para la resolución de estos casos.

El capítulo 7 expone las conclusiones alcanzadas durante la presente Tesis Doctoral y las líneas futuras de investigación a desarrollar a resultas del trabajo aquí expuesto.
En el anexo A mostramos las tablas con los resultados, detallados para cada instancia, de los algoritmos iniciales propuestos así como del estado del arte. El anexo B muestra los gráficos y tablas de interacción de los distintos algoritmos con el número de trabajos y el número de máquinas. La calibración de los algoritmos propuestos viene reflejada en el anexo C, mientras que en el anexo D presentamos las tablas y gráficos estadísticos correspondientes a los algoritmos de reducción de tamaño propuestos en el capítulo 5. El anexo E muestra las tablas y gráficos referentes al problema de máquinas opcionales y selección de trabajos tratado en el capítulo 6. Finalmente en el anexo F se han incluido los artículos enviados y aceptados a diferentes revistas, presentaciones y publicaciones en congresos, resultado de las diversas investigaciones desarrolladas en la presente Tesis Doctoral.

CAPÍTULO 2

MÁQUINAS PARALELAS

Hemos visto en el capítulo 1 una primera definición de los distintos tipos de problemas de máquinas paralelas. En este capítulo procederemos a estudiar más en detalle estos problemas. En la presente Tesis Doctoral trataremos con los problemas de máquinas paralelas deterministas donde se conocen inicialmente el número de máquinas y de trabajos, así como sus características. Igualmente, la matriz de los tiempos de proceso de un trabajo j en una máquina i (p_{ij}) es conocida de antemano y determinista. También asumiremos que trabajamos con las restricciones de que cada trabajo se asignará a una sola máquina y no se podrá interrumpir una vez éste se ha iniciado. Todos los trabajos están disponibles en el tiempo cero. Tampoco ninguna máquina podrá procesar más de un trabajo a la vez, no sufren roturas ni paradas y siempre están disponibles. Nuestro objetivo sera la minimización del $C_{\text{máx}}$. Nótese, y como veremos en detalle más adelante, que con el objetivo de minimizar el $C_{\text{máx}}$, el problema se reduce a una asignación de los trabajos a las máquinas. Esto es debido a que el valor del C_i (tiempo de finalización) de cada máquina, una vez asignados todos los trabajos, es independiente de la secuencia.

Pongamos un ejemplo gráfico. Por ejemplo, tenemos la tabla de tiempos de proceso 2.1, donde se destacan en negrita y cursiva una asignación cualquiera de los trabajos a las distintas máquinas:

	J_1	J_2	J_3	J_4	J_5
M_1	***5***	10	***15***	45	15
M_2	15	***10***	35	20	60
M_3	45	55	5	***15***	***30***

Tabla 2.1 – Tabla de tiempos de proceso de ejemplo de asignación en máquinas paralelas.

La asignación de trabajos podría ser la siguiente:

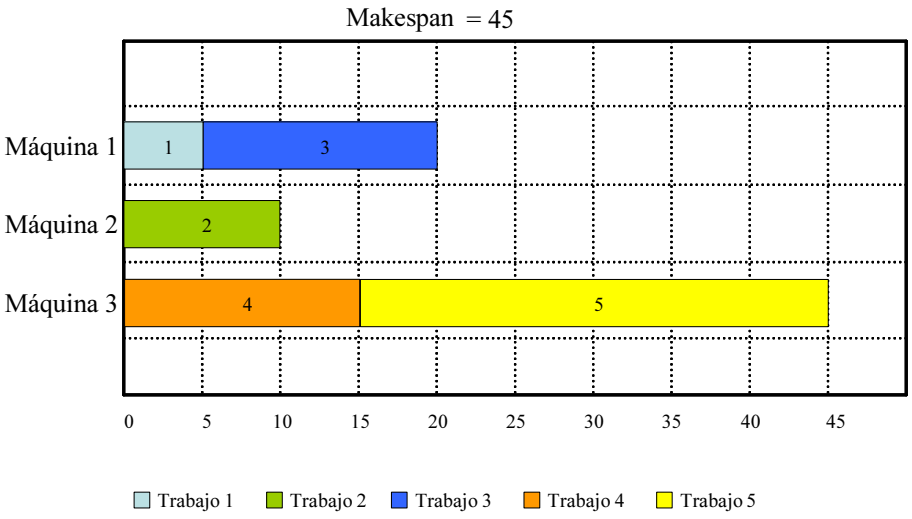

Figura 2.1 – Diagrama de Gantt con la asignación de trabajos en máquinas paralelas.

Por tanto, en nuestro caso concreto, el hallar el $C_{\text{máx}}$, se reduce a un

problema de asignación, ya que una vez asignado un trabajo a una máquina, no influye el orden en que éstos se procesen. Por ejemplo, fijémonos en la máquina 3, daría igual procesar primero el trabajo 4 y después el 5 tal y como aparece en la figura 2.1 o al revés, ya que el instante de finalización en la máquina $C_3 = C_{\text{máx}} = 45$ no cambiaría.
Nótese que esta propiedad del problema de máquinas paralelas es válida para el objetivo de minimizar el $C_{\text{máx}}$, dado que otros objetivos, como minimizar la tardanza media o la tardanza máxima, sí que dependen de la secuencia dentro de cada máquina. No obstante, estos aspectos quedan fuera del presente trabajo.

En cuanto a la dificultad que entrañan este tipo de problemas, mencionemos que en un principio y de una manera intencionadamente simplificada, se clasificaron los problemas de programación en fáciles (es decir, resolubles en tiempo polinómico $\mathcal{P}$) y difíciles (es decir, $\mathcal{NP}$-Hard). Posteriormente Schuurman y Woeginger (1999) profundizarón en esta clasificación de los problemas. Los problemas de máquinas paralelas sin interrupción pertenecen a la rama de los problemas de difícil resolución, ya que la inmensa mayoría de ellos son del tipo $\mathcal{NP}$-Hard. Por lo tanto, es poco probable (de hecho, a menos que $\mathcal{P} = \mathcal{NP}$) que exista un algoritmo de tiempo polinómico capaz de calcular un $C_{\text{máx}}$ óptimo.
Una forma habitual para el tratamiento de los problemas $\mathcal{NP}$-Hard no es la búsqueda de una solución óptima, sino una búsqueda de soluciones buenas o quasi-óptimas. Un algoritmo que de soluciones buenas se le denomina algoritmo de aproximación. Si se ejecuta en tiempo polinómico, entonces recibe el nombre de algoritmo de aproximación de tiempo polinomial. Un algoritmo de aproximación que devuelve una solución casi óptima, que es como mucho un factor ρ por encima de la solución óptima (donde $\rho > 1$ es un número real fijo) se llama un algoritmo ρ de aproximación, y el valor de ρ se llama factor de aproximación. A la familia de algoritmos de aproximación ($1 + \epsilon$, con $\epsilon > 0$)

se denomina algoritmos de régimen de tiempo polinomial de aproximación o PTAS, para abreviar. Si la complejidad del tiempo de un PTAS está también polinómicamente delimitada por $1/\epsilon$, entonces se le llama algoritmos de régimen de tiempo polinomial de aproximación completo o FPTAS. Un FPTAS es, esencialmente, la aproximación más fuerte posible, de tiempo polinomial, que podemos obtener para un problema $\mathcal{NP}$-Hard. Para profundizar más sobre los algoritmos de aproximación de tiempo polinómico para programación de la producción pueden consultarse los trabajos de Chrétienne et al. (1995) y de Hochbaum (1997).
Otra posibilidad de resolver problemas difíciles de programación es considerar algoritmos heurísticos con la esperanza de obtener resultados cercanos al óptimo. Sin embargo, mediante el uso de heurísticas, hay que tener presente que en ocasiones no se tiene garantía de la calidad de la solución y/o del tiempo de cálculo.

2.1. Máquinas paralelas idénticas

En primer lugar, consideremos el problema de programación con máquinas idénticas denotado por Graham et al. (1979) como $P//C_{\text{máx}}$. En este caso, el tiempo de proceso del trabajo $j \in N$ en cualquier máquina $i \in M$ es igual a p_j. Esto viene a significar que las máquinas son idénticas y que, por ejemplo tarda el mismo tiempo en procesar el trabajo la máquina 1 que la máquina 2. Este problema es de interés porque minimizar el makespan tiene el efecto de equilibrar la carga entre las diversas máquinas, lo que es importante en la práctica. Para este problema $P//C_{\text{máx}}$, en Garey y Johnson (1979) ya se demostró que pertenecía al tipo $\mathcal{NP}$-Hard. Incluso la versión de dos máquinas ($P2//C_{\text{máx}}$) es ya NP-Hard según Lenstra, Rinnooy Kan y Brucker (1977).
Durante las últimas tres décadas se han desarrollado muchos algoritmos de aproximación y heurísticas para el problema $P//C_{\text{máx}}$. Quizá uno de los primeros resultados sencillos para delimitar el peor caso vino dado por el

rendimiento del LS (lista de programación o *list scheduling*) dada por Graham (1966), $2 - \frac{1}{m}$, es decir, que la mejor solución del peor caso que se puede dar es dos menos $\frac{1}{m}$ veces la solución óptima. El algoritmo asigna cada trabajo a la máquina que en ese momento tiene una menor carga de trabajos asignados.
La regla LPT (el trabajo con el tiempo de proceso más alto se secuencia primero) asigna, en el tiempo cero, el trabajo j con el tiempo de proceso más largo a una máquina. Continúa asignando a las máquinas libres (o menos sobrecargadas) los trabajos con tiempo de proceso más largos. La heurística trata de dejar los trabajos con tiempos de procesado más cortos al final de la secuencia con la idea de equilibrar las cargas de trabajo de todas las máquinas.
En el caso de usar LPT, se mejora el mejor peor caso anterior, y que vendrá dado por un valor menor o igual a $\frac{4}{3} - \frac{1}{3m}$, tal y como se mostró en Graham (1969). Es más, demostraron que en el caso de dos máquinas idénticas, el valor del factor de aproximación sería de $\frac{7}{6}$. Seiden, Sgall y Woeginger (2000) demostraron que para el caso de tres máquinas este factor podía llevarse hasta $\frac{1}{6}(1 + \sqrt{37}) \approx 1,18$.
Posteriormente, Sahni (1976) desarrolló unos algoritmos más complicados que podían obtener resultados tan cercanos al óptimo como se desease, pero a costo de un crecimiento exponencial del tiempo de ejecución. Más tarde Hochbaum y Shmoys (1987) propusieron una mejora de estos algoritmos que redujeron en parte el tiempo de ejecución.
Posteriormente se propuso por Coffman, Garey y Johnson (1978) el algoritmo MF (ajustes múltiples), que posteriormente mejoró Friesen (1984). La idea consiste en encontrar (por búsqueda binaria) la capacidad más pequeña de un compartimiento (bin) tal que un grupo de m bins puedan acomodar todos los trabajos cuando los trabajos están tomados en orden no creciente de p_j. El conjunto de trabajos del j-ésimo bin será procesado en la máquina i. Si se realizan k intentos de embalaje, el algoritmo, ahora denominado MF_k, proporcionará una aproximación igual o menor que $1,2 + \frac{1}{2^k}$. Obsérvese que usando la búsqueda binaria, con tan solo siete iteraciones del algoritmo MF_k

obtendrá una desviación respecto al óptimo menor o igual que $1,2 + 0,001$. Estos valores fueron posteriormente mejorados por Langston (1981), quien modificó el algoritmo MF usando técnicas similares a las de Friesen (1984), llegando a valores de $\frac{72}{61}$.
Finalmente señalar que Dell'Amico y Martello (1995) consiguieron mejorar todo los algoritmos presentes hasta la fecha. Posteriormente Mokotoff (2004) presentó un trabajo basado en un branch and bound pero, tal y como le contestaron en Dell'Amico y Martello (2005), estaba lejos de alcanzar los excelentes resultados anteriores de estos autores. Más tarde, en el artículo de Dell'Amico et al. (2008) propusieron dos nuevos algoritmos, uno exacto y otro metaheurístico basado en un *scatter search*, con muy buenos resultados.

2.2. Máquinas paralelas uniformemente relacionadas

Para el caso de máquinas uniformemente relacionadas o $Q//C_{\text{máx}}$, la expresión de los tiempos de proceso viene dada por $p_{ij} = p_j/s_i$, donde s_i es la velocidad de la máquina i. Esto quiere decir que el tiempo de proceso en cada máquina viene afectado por un valor relativo a la velocidad de la misma. Por ejemplo si la máquina 2 es el doble de rápida que la máquina 1, el tiempo de proceso del trabajo 1, será la mitad en la máquina 2 que en la máquina 1. Nótese que este caso es una generalización del caso anterior, por lo que es un problema $\mathcal{NP}$-Hard.
En el artículo de Liu y Liu (1974) aparecen estudios sobre numerosas cuestiones acerca de las máquinas relacionadas. En Graham (1966) y Graham (1969) se generalizó el algoritmo LPT para hacerlo aplicable al problema $Q//C_{\text{máx}}$. Para ello, se asignaba cada trabajo en orden no creciente de p_j a la máquina que finalizara antes, es decir, se asignaba el trabajo j a la máquina i de forma que se minimice $\zeta_i + \frac{p_j}{s_i}$, donde ζ_i es la carga de la máquina i justo antes de la asignación del trabajo j. En general se demostraba que la diferencia con el óptimo es menor o igual a $2 - \frac{2}{m+1}$. En Gonzalez, Ibarra y Sahni (1977)

se dieron ejemplos para el algoritmo LPT donde la diferencia con el óptimo se aproximaba a $\frac{3}{2}$ cuando m tiende a infinito. También demostraron que, con dos máquinas uniformes (o relacionadas), cualquier ratio de velocidad q mayor que uno, el valor del factor de aproximación era de $\frac{1}{4}(1+\sqrt{17}) \approx 1,28$, donde q es la relación entre la máquina más rápida y la más lenta. Posteriormente, Epstein y Favrholdt (2005) proporcionaron un factor de aproximación exacto para el algoritmo LPT en función del ratio q. Para el caso general de m máquinas relacionadas, Friesen (1987) proporcionó, para el algoritmo LPT, un valor de aproximación comprendido entre 1,52 y 1,66.
Sin embargo, Friesen y Langston (1983) modificaron el algoritmo MF de Coffman, Garey y Johnson (1978) adaptándolo para el caso de máquinas uniformemente relacionadas y consiguieron un peor caso para MF menor 1,40. En el mismo articulo mostraron un peor caso de 1,34, por lo que concluyeron que el peor caso estaba comprendido entre 1,34 y 1,40 para $Q//C_{\text{máx}}$.
Hochbaum y Shmoys (1988) mostraron un algoritmo basado en un procedimiento de decisión en el que se comprobaba si existía una secuencia para una instancia donde todos los trabajos se completen en un tiempo T. Entonces el problema de decisión puede ser contemplado como un bin-packing, donde el T mínimo se calcula mediante un procedimiento de búsqueda binaria simple.
Tras el artículo de Epstein y Favrholdt (2005), hubieron algunos otros algoritmos destacables, aunque en su mayoría incluyeron más restricciones; así Li (2006) realizó una generalización del articulo Lin y Li (2004) para tiempos de proceso iguales; o el artículo de Dosa y Epstein (2008), donde dieron una cota inferior para el problema de máquinas paralelas uniformes donde las máquinas pueden ser interrumpidas.

2.3. Máquinas paralelas no relacionadas

Por último, tenemos el caso más general, en el que las máquinas no están relacionadas y los tiempos de proceso son p_{ij}, con lo que el tiempo de proceso

depende del trabajo j a realizar y de la máquina i donde se vaya a procesar. Siguiendo la clasificación de Graham et al. (1979) este problema para el objetivo de minimizar el $C_{\text{máx}}$ queda definido como $R//C_{\text{máx}}$.

Pasemos a definir matemáticamente este problema, mediante programación lineal entera mixta (MILP). El objetivo, es el ya mencionado, minimizar el makespan ($C_{\text{máx}}$), cuyo valor será siempre no negativo.

Las variables de decisión son las siguientes:

$$x_{ij} = \begin{cases} 1, & \text{si el trabajo } j \text{ se asigna a la máquina } i \\ 0, & \text{en caso contrario} \end{cases}$$

Función objetivo:

$$\text{mín } C_{\text{máx}} \tag{2.1}$$

s.a.:

$$\sum_{i=1}^{m} x_{ij} = 1 \quad \forall j \in N \tag{2.2}$$

$$\sum_{j=1}^{n} p_{ij} \cdot x_{ij} \leq C_{\text{máx}} \quad \forall i \in M \tag{2.3}$$

$$x_{ij} \in \{0, 1\} \quad \forall j \in N, \ \forall i \in M \tag{2.4}$$

El primer conjunto de restricciones (2.2) asegura que el trabajo j se procesa exactamente en una única máquina, ya que la suma la variable binaria x_{ij} de si se hace o no el trabajo j en la máquina i para todas las máquinas, ha de ser 1. El segundo conjunto de restricciones (2.3) fuerza a que el valor máximo de la suma de todos los trabajos asignados a cada máquina sea el del $C_{\text{máx}}$. Así, la suma del producto de la variable binaria (se hace o no el trabajo j en la máquina i) por el tiempo de proceso, nos dará lo que tarda cada máquina en acabar de procesar todos los trabajos que tenga asignados, y esta suma podrá

ser como máximo el valor del $C_{\text{máx}}$.
Existen otros modelos como los propuestos por Unlu y Mason (2010), pero el modelo expuesto es el más habitual en la literatura para el problema de $R//C_{\text{máx}}$.
Es fácil constatar que este problema es una generalización de los dos precedentes (máquinas idénticas y máquinas relacionadas), por lo que la complejidad de este problema es, así mismo, $\mathcal{NP}$-Hard.

Dado que la presente Tesis Doctoral se va a centrar en el problema de máquinas paralelas no relacionadas, haremos un estudio más profundo de la literatura existente sobre este problema para finalmente identificar el estado del arte.

2.3.1. Revisión bibliografía

Tal y como podemos observar en las revisiones publicadas sobre máquinas paralelas como la de Cheng y Sin (1990) o la de Mokotoff (2001), los artículos publicados en máquinas paralelas no relacionadas son escasos con respecto a los publicados en cualquiera de las otras dos variantes. No obstante, pasemos a estudiar los artículos más relevantes publicados con respecto al problema de máquinas paralelas no relacionadas.

Ya desde los finales de los años cincuenta, con el trabajo de McNaughton (1959), se viene estudiando el problema de máquinas paralelas no relacionadas Las primeras aproximaciones se realizaron mediante secuenciación de listas (*list scheduling*) o reglas de despacho, a las que se le aplicaba una regla para determinar la asignación, como la regla LPT (*longest processing time*) de Graham (1969), consistente en asignar primero los trabajos de tiempo de proceso más largos, que garantizaba un error para el peor caso de $2 - 1/\varepsilon$. Posteriormente Horowitz y Sahni (1976) propusieron algoritmos de programación dinámica y algunos métodos aproximados para el caso de $R2//C_{\text{máx}}$. Ibarra y Kim (1977)

presentaron cinco métodos aproximados para los casos de 2 y m-máquinas junto con la prueba de que la regla LPT tiene un error ligeramente más ajustado que el de Graham (1969) para el caso del $C_{\text{máx}}$ y grandes valores de m. De y Morton (1980) propusieron heurísticas que ofrecían resultados cercanos al óptimo para problemas relativamente pequeños. Davis y Jaffe (1981) propusieron un algoritmo aproximado con un error para el peor caso de $2\sqrt{m}$. Más tarde, Lenstra, Shmoys y Tardos (1990) mostraron otra heurística cuyo mejor peor caso tenia un ratio de 2. Es más, en este mismo articulo demostraron que, para la resolución del problema genérico $R//C_{\text{máx}}$, ningún algoritmo de tiempo polinómico puede obtener una solución con un ratio mejor que 3/2, a menos que $\mathcal{P} = \mathcal{NP}$. Esto implica que el peor de los casos estará a una vez y media del óptimo.

Gran parte del esfuerzo de investigación a lo largo de la literatura se ha concentrado en la idea de resolver la relajación lineal del modelo MILP presentado en la sección 2.3 transformando la restricción (2.4) y teniendo en cuenta la restricción (2.5) en su lugar.

$$x_{ij} \geq 0 \quad \forall j \in N, \forall i \in M \tag{2.5}$$

La solución óptima de la relajación del MILP se alcanza en la primera fase. En la segunda fase, un método de redondeo se aplica para obtener una solución factible del problema. Este método de redondeo o *rounding phase* puede ser tanto exacto como aproximado. Potts (1985) fue el primero en emplear esta técnica, la cual luego fue explotada y refinada, entre otros, por Lenstra, Shmoys y Tardos (1990), Shmoys y Tardos (1993) y más recientemente, por Shchepin y Vakhania (2005). En este último artículo, los autores mejoraron el hasta entonces mejor error conocido de Lenstra, Shmoys y Tardos (1990) de 2 a $2-1/m$. Nótese que en esta "aproximación de dos fases" a menudo se necesita un solver de programación lineal para la primera fase.

Los algoritmos metaheurísticos han proporcionado muy buenos resultados para el problema de $R//C_{\text{máx}}$. Hariri y Potts (1991) propusieron algunas heurísticas complementadas con métodos de búsqueda local mejorados que mostraron unos resultados prometedores. En 1993, Van De Velde propuso dos algoritmos, uno exacto y uno metaherístico de búsquedas locales iteradas, ambos basados en la relajación y dualidad del modelo MILP presentado anteriormente. Se probaron problemas de hasta 200 trabajos y 20 máquinas (200×20). El método exacto fue capaz de resolver instancias de hasta 200×2 o de hasta 50×20. El procedimiento de búsqueda local obtuvo desviaciones relativamente grandes respecto a las soluciones óptimas en muchos de los casos, especialmente para los valores más altos de m. Por lo que nosotros sabemos, Glass, Potts y Shade (1994) fueron los primeros en proponer algoritmos de tipo Genético (GA), Búsqueda Tabú (TS) y Recocido Simulado (SA). Bajo ciertas condiciones, los tres algoritmos (en sus versiones standard) proporcionan resultados similares, mientras que para tiempos de computación altos, los algoritmos propuestos GA y SA mostraron mejor comportamiento. Más tarde, Piersma y van Dijk (1996) presentaron un SA y un TS con inicializaciones provenientes de la heurística de Hariri y Potts (1991) y Davis y Jaffe (1981). Los algoritmos propuestos incluían una búsqueda local efectiva lo cual proporcionaba muy buenos resultados por aquel entonces.

En la literatura también encontramos algunos algoritmos exactos con excelentes resultados. Martello, Soumis y Toth (1997) propusieron un método Ramificación y Acotación (B&B) usando una cota inferior y algunas heurísticas para la cota superior. Los resultados superaron a aquellos proporcionados por Van De Velde (1993) con unos errores relativamente pequeños para instancias de hasta 80×20. A pesar de estos buenos resultados, las investigaciones han seguido por la parte de las metaheurísticas. Por ejemplo, Srivastava (1998) propuso un TS avanzado. Más tarde, Sourd (2001) presentó dos métodos basados en búsquedas en vecindarios grandes. El primero es una heurística de

exploración de árbol parcial y la segunda está también basada en la dualidad del modelo MILP empleado por Van De Velde (1993).

Una propuesta interesante fue la de llevada a cabo por Mokotoff y Chretienne (2002). Los autores desarrollaron un método de planos de corte el cual, básicamente, selecciona un subconjunto de restricciones del modelo MILP antes mencionado. Se obtiene una solución óptima de ese modelo simplificado. Las restricciones originales se comprueban. Si todas las restricciones son satisfechas, estamos ante una solución del problema original. Si algunas de las restricciones originales se violan, entonces una o más restricciones se añaden al modelo MILP simplificado y el modelo se resuelve otra vez. Nótese cuan diferente es este método del anteriormente mencionado de aproximación en dos fases. Esta técnica fue posteriormente refinada en Mokotoff y Jimeno (2002). En este artículo, se presenta un algoritmo denominado como "Partial". Partial esta basado en la metodología de Dillenberger et al. (1994), la cual fue usada más tarde por Mansini y Speranza (1999). En general, Partial está basado en resolver el modelo MILP relajado, para posteriormente seleccionar parte de las variables binarias (nótese la diferencia con el artículo de Mokotoff y Chretienne (2002) donde en lugar de una relajación en las variables, se empleaban menos restricciones). Esta novedosa metodología produce excelentes resultados, resolviendo instancias que llegan hasta 200×20 empleando poco tiempo de CPU, menos de 75 segundos, con un pequeño porcentaje de desviaciones relativas respecto al óptimo, menores del $2\,\%$. Ambos métodos propuestos en Mokotoff y Chretienne (2002) y en Mokotoff y Jimeno (2002) hacen un amplio uso de solvers comerciales en sus respectivas fases iniciales.

En un trabajo excelente, Woclaw (2006) llevó a cabo una reimplantación y una cuidadosa evaluación computacional de la mayoría (si no todos) los algoritmos existentes en la literatura para el problema de $R//C_{\text{máx}}$. El autor probó y evaluó varias heurísticas, metaheurísticas y métodos exactos en un

escenario similar y con el mismo grupo de instancias. De estos resultados, se pudo concluir que el método Partial de Mokotoff y Jimeno (2002) era el estado del arte.

Después de la publicación de la Tesis de Woclaw, Ghirardi y Potts (2005) publicarón un trabajo que mostraba excelentes resultados. Se propuso en este trabajo un algoritmo de *Recovering Beam Search* (RBS). La metodología de RBS ya había sido estudiada por Croce, Ghirardi y Tadei (2004), la cual a su vez se basaba en el conocido método de Beam Search (BS) de Ow y Morton (1988), entre otros. BS básicamente trunca el B&B permitiendo la exploración de sólo los w nodos más prometedores en cada nivel del árbol de búsqueda. w es el ancho de banda. RBS es una mejora de BS en el cual hay una fase de recuperación donde una solución dada s se comprueba para ver si es dominada por otra solución parcial s_0 en el mismo nivel del árbol. Si este es el caso, s se descarta y s_0 sustituye a s como la nueva solución parcial. Se probaron instancias de hasta 1000×50, con buenos resultados para tiempos de CPU que llegaron hasta los 150 segundos.

Otros métodos han aparecido recientemente. Por ejemplo, Gairing, Monien y Woclaw (2007) vuelven a retomar el enfoque de las dos fases. Los autores proponen un método muy rápido aunque con resultados que no pueden competir en cuanto a la calidad de las soluciones con aquellos de Mokotoff y Jimeno (2002) o Ghirardi y Potts (2005). Revisando detenidamente la literatura, parece ser que los métodos de Partial y RBS, propuestos en estos dos últimos artículos respectivamente, representan el actual estado del arte.

Queremos finalizar la revisión de la literatura con una observación. Hasta donde alcanza nuestro conocimiento, ningún autor compara los resultados obtenidos por las diferentes estrategias para resolver el problema de $R//C_{\text{máx}}$ contra un solver moderno y efectivo. Como más tarde mostraremos, el simple

modelo MILP mostrado en la sección 2.3 se resuelve de forma bastante efectiva con un solver comercial standard como por ejemplo IBM-ILOG CPLEX 11.0. el cual esta basado en las actuales metodologías de Ramificación y Corte.

Para finalizar, podemos resumir diciendo que Glass, Potts y Shade (1994), ya compararon los resultados de un algoritmo genético estándar (GA), un Recocido Simulado (SA) y una Búsqueda Tabú (TS). Posteriormente Piersma y van Dijk (1996) presentaron unos algoritmos de búsquedas locales que lo mejoraban. Finalmente Sourd (2001) presentó dos algoritmos basados en grandes vecindarios, uno basado en una exploración del árbol de búsqueda mediante heurísticas y otro basado en la aproximación dual de Van De Velde (1993) que superaba a todos ellos. Por otra parte, se encuentra, dentro de los algoritmos del tipo Ramificación y Acotación, el de Martello, Soumis y Toth (1997), que fue superado en el artículo de Ghirardi y Potts (2005). Finalmente destacar que Woclaw (2006) realizo un extenso trabajo de implementación de la mayoría de los algoritmos existentes a lo largo de la literatura, destacando entre todos ellos los buenos resultados de Mokotoff y Chretienne (2002). El propio Sourd (2001) nos remitió al estudio realizado por Woclaw (2006) cuando nos pusimos en contacto con él.

Por último, citar los recientes artículos de Fanjul-Peyro y Ruiz (2010a) y Fanjul-Peyro y Ruiz (2010b), que son el resultado de parte de lo desarrollado en esta Tesis Doctoral y que expondremos con detalle en capítulos posteriores.

El la tabla 2.2 podemos ver de forma resumida parte de está revisión bibliográfica.

Autor	Año	Método	Exacto
Graham	1969	LPT	no
Horowitz y Sahni	1976	PD y Heu.	si/no
Ibarra y Kim	1977	Heu.	no
De y Morton	1980	Heu.	no
Davis y Jaffe	1981	Heu.	no
Potts	1985	DF	no
Lenstra, Shmoys y Tardos	1990	DF	no
Hariri y Potts	1991	LS	no
Van De Velde	1993	R&D/LS	si/no
Glass, Potts y Shade	1994	GA/TS/SA	no
Piersma y van Dijk	1996	TS/SA	no
Martello, Soumis y Toth	1997	B&B	si
Srivastava	1998	TS	no
Sourd	2001	LLS	no
Mokotoff y Chretienne	2002	B&C	si
Mokotoff y Jimeno	2002	Partial	no
Ghirardi y Potts	2005	RBS	no
Shchepin y Vakhania	2005	DF	no
Gairing, Monien y Woclaw	2007	DF	no
CPLEX 12.1	2010	B&C	si
Fanjul-Peyro y Ruiz	2010	IG	no
Fanjul-Peyro y Ruiz	2011	SR	no

Exacto = si/no: algoritmos exacto y Heu. en el mismo artículo. LPT: longest processing time; PD: programación dinámica Heu.: heurística; DF: dos fases; LS: local search; R&D: relajación y dualidad; GA: genético; TS: búsqueda tabú; SA: recocido simulado; B&B: ramificación y acotación; LLS: búsqueda local en vencindarios grandes; B&C: ramificación y corte; RBS: recovery beam search; IG: iterated greedy; SR: size reduced.

Tabla 2.2 – Tabla resumen de la revisión bibliográfica.

En la literatura existen variantes del problema $R//C_{\text{máx}}$ en el cual se incluyen características específicas para este tipo de problemas, como pueden ser restricciones de precedencia, fechas de finalización, costes (pesos). No

obstante, los artículos relativos a estas variantes son menos numerosos que los dedicados al problema genérico, probablemente debido a la complejidad que se añade a los modelos de estas variantes. Comentaremos a continuación algunos de estos artículos, como el de Jansen y Porkolab (1999) donde considera el problema con coste de funcionamiento. Herrmann, Proth y Sauer (1997) consideraron el problema de máquinas paralelas no relacionadas con relaciones de precedencia y Lancia (2000) investigó el caso en el que hubiese fechas de disponibilidad, al igual que hiciera Adamopoulos y Pappis (1998). Armacost y Salem (2001) desarrollaron una metodología para resolver el problema con tiempos de preparación y las restricciones de elegibilidad de máquina. Respecto al problema en el que se incluyen roturas de máquinas aleatorias, se pude mencionar el articulo de Allahverdi y Mittenthal (1994), aunque en este caso, el objetivo trataba de minimizar el tiempo medio del taller de flujo. Más tarde, Fishkin, Jansen y Mastrolilli (2008) examinaron el problema de minimizar la función objetivo que era una suma ponderada del makespan y los costos totales. Sobre el problema de máquinas paralelas no relacionadas con tiempo de cambio dependientes de la secuencia podemos destacar el reciente artículo de Vallada y Ruiz (2010).

Ya que nuestro estudio se centra en el problema de $R//C_{\text{máx}}$, procederemos a realizar un estudio más exhaustivo del estado del arte para este problema en base a lo comentado anteriormente.

2.3.2. Estado del arte

En esta sección detallaremos algoritmos como Partial de Mokotoff y Jimeno (2002), RBS de Ghirardi y Potts (2005) y haremos una introducción a un solver comercial como IBM-ILOG CPLEX como una posible herramienta a tener en cuenta en la resolución de este problema.

2.3.2.1. Partial de Mokotoff y Jimeno

Los algoritmos que desarrollaron Mokotoff y Jimeno (2002), se basaron en la metodología usada por Dillenberger et al. (1994) y usada posteriormente por Mansini y Speranza (1999), como ya se comentó.

En general, se basan en resolver un subproblema del tipo MILP (*Mixed Integer Linear Programming*), donde algunas variables binarias x_{ij} pasan a ser continuas acotadas tanto inferior, por 0, como superiormente, por 1. El procedimiento comprueba si la solución óptima del LP (*Linear Programming*) hallada, corresponde a una solución entera. Si es así, el algoritmo para, y si no, el subconjunto de las variables empleadas se varía.

Como primer paso, se realiza una relajación del MILP, que queda como sigue:

funcion objetivo:

$$\text{mín } C_{\text{máx}} \tag{2.6}$$

s.a.:

$$\sum_{i=1}^{m} x_{ij} = 1 \quad \forall j \in N \tag{2.7}$$

$$\sum_{j=1}^{n} p_{ij} \cdot x_{ij} \leq C_{max} \quad \forall i \in M \tag{2.8}$$

$$0 \leq x_{ij} \leq 1 \quad \forall j \in N,\ \forall i \in M; \tag{2.9}$$

De la solución óptima de la resolución de la relajación del MILP, se crea una lista de variables x_{ij}. Todas las variables de decisión con valores positivos en la solución se ponen al principio de la lista, y a continuación el resto de variables de decisión.

Si hay k variables de decisión con valores positivos en la solución de LP, se construye un nuevo MILP con las primeras $\alpha \geq k$ variables de decisión de

la lista. Por tanto, el subproblema MILP se resuelve con $mn - \alpha$ variables binarias, siendo mn el número de variables binarias del problema MILP original.

Aunque la metodología empleada en Partial es similar a la usada por Hariri y Potts (1991) y Potts (1985), la diferencia esencial con ellos es que solo usan los datos de la solución dada por el LP para decidir qué variables deben ser eliminadas de la formulación del MILP. Ahora bien, hay dos aspectos importantes a decidir: el criterio de ordenación de la lista del subconjunto de las variables con valores no positivos y el número de variables binarias a mantener. En su articulo, señalan como mejores opciones a emplear, el ordenar la lista por orden no decreciente de coste reducido, y como criterio de número de variables binarias a mantener, usar una variable r de forma que $\alpha = k + [r(mn - k)]$. Desarrollan tres variantes: en la primera solo seleccionan aquellas variables, que tras la resolución del LP relajado, tengan valores positivos; en la segunda variante retienen, además de las anteriores, un número fijo del resto de variables, según los criterios explicados anteriormente de ordenación y variables a mantener; y la tercera variante, que es similar a la segunda, salvo en que el número del resto variables a retener, lo hacen variar en cada iteración, ya que si da un valor de cero en la resolución del LP se retira de la lista.
Finalmente, en el desarrollo y conclusiones del artículo, de las tres variantes del algoritmo que desarrollan, apuntan a que la segunda variante es la que da mejores resultados. Igualmente, hacen diversas pruebas para valores de $r = 0,15; 0,25; 0,75$ principalmente, y aunque los mejores resultados se obtienen con 0,75, concluyen que esta opción consume mucho tiempo de CPU en comparación con la mejora obtenida, inclinándose por tanto como mejor opción por un $r = 0,25$. Trabajan con diez instancias de cada combinación de $m = 3, 5, 8, 10, 15, 20$ y $n = 20, 30, 40, 50, 60, 80, 100, 200$ siendo por tanto la instancia más grande de 200×20; todas ellas con tiempos de proceso seleccionados al azar dentro del intervalo U(1,100). Emplean PC Pentium 500 y para las instancias más grandes dan errores del 1,2 % en 74 segundos,

comparando el resultado obtenido por su algoritmos con el de CPLEX usado durante 1000 segundos, el cual no alcanza el óptimo para ninguna de las 10 instancias de ese tamaño.

En Woclaw (2006) se demostraba, tras un exhaustivo análisis de casi todos los algoritmos existentes para el problema de $R//C_{\text{máx}}$, que el mejor algoritmo que había hasta ese momento era, por un margen significativo, el método Partial propuesto por Mokotoff y Jimeno (2002).

2.3.2.2. RBS de Ghirardi y Potts

En el artículo de Ghirardi y Potts (2005), los autores tratan de afinar el método RBS (*Recovering Beam Search*) empleado por Croce, Ghirardi y Tadei (2004). Este método está basado en el *Beam Search* (BS), y en los algoritmos de Ow y Morton (1988), que son producto de un B&B (*branch and bound*). El RBS consiste en truncar el B&B, en el cuál, solo los w nodos más prometedores de cada nivel del árbol de búsqueda se seleccionan como nodos desde los cuales partirán nuevas ramas; w se le llama también ancho de banda. Evidentemente, a más ancho de banda o amplitud del rayo (*Beam*), más nodos a evaluar y más lento va el algoritmo. El procedimiento de evaluación de cada nodo por cada nivel es lo que caracteriza a los BS. Una evaluación superficial es más rápida pero se pueden perder buenas soluciones, mientras que una evaluación profunda consume mucho tiempo.
Lo usual es hacer una evaluación en dos pasos; primero se hace una evaluación superficial (*filtering*) para seleccionar un grupo reducido de nodos a los que se les aplica, en un segundo paso, una evaluación más precisa. Entonces, los w mejores nodos se seleccionan.

El procedimiento tradicional del BS no puede recuperarse de las decisiones erróneas; una vez descartada una rama de un nodo, no se puede volver a recuperar. En contraste con el BS, el procedimiento de evaluación de nodos

del RBS usa tanto la cota superior como la inferior. Mientras que la cota inferior se usa exclusivamente con propósito de evaluación, la cota superior también proporciona posibles soluciones, dado que esta cota es una solución factible. La función de evaluación viene determinada por la siguiente expresión $V = (1 - \alpha)LB + \alpha UB$, donde LB y UB representan la cota inferior y superior, respectivamente y α es un parámetro de peso que se determina experimentalmente. Los valores de la LB los obtienen de la relajación Langragiana ya empleada por Martello, Soumis y Toth (1997); una vez obtenidos, se genera el valor de la UB, la cuál es el valor de la función objetivo de la solución factible derivada directamente de la relajación Langragiana.

En el método de RBS, el ancho de banda w es constante y normalmente pequeño ($w < 10$) para disminuir el tiempo de CPU. Una vez se tienen los w mejores nodos y las correspondientes w mejores soluciones parciales, la fase de recuperación (*Recovering*) comprueba si la solución s es dominada por otra solución parcial s_0 en el mismo nivel del árbol de búsqueda. Si es así, s es descartada y s_0 pasa a ser una nueva solución parcial, si no está ya incluida. Si ya está incluida, queda reservada para una posible reexaminación durante la fase de recuperación, hasta mantener exactamente w número de nodos. Esto permite recuperar algunas de las decisiones erróneas tomadas con anterioridad. No obstante, una solución parcial solo podrá ser sustituida por otra del mismo nivel de profundidad dentro del árbol de búsqueda, lo que mantendrá en un número polinómico los nodos explorados cuando también lo sea el número de niveles del árbol de búsqueda.

Finalmente, los autores marcan para los parámetros a determinar, que un buen α es el dado por $\alpha = 0{,}1$, mientras que para el ancho de banda w, prueban con $w = 1$ y con $w = 3$, quedando patente que se obtienen resultados bastante similares en cuanto a la desviación respecto a la mejor solución pero consumiendo mucho más tiempo en el caso de $w = 3$. En su artículo muestran un conjunto de 20 instancias diferentes para cada combinación de $m = 10, 20, 30, 40, 50$ y $n = 100, 200, 500, 1000$, llegando a instancias de

hasta 1000×50, generadas para los tiempos de proceso de los intervalos U(10,100), U(10,1000) y tiempos de proceso correlacionados. Para las pruebas emplearon ordenadores PC Pentium III/866 y se compararon con el algoritmo APPROX propuesto por Martello, Soumis y Toth (1997). Sus resultados mostraron errores del 1,4 % en 143 segundos para las instancias más grandes, con respecto a la cota inferior.

Los resultados de este algoritmo resultaron bastante prometedores, aunque desafortunadamente no se compararon con Partial de Mokotoff y Jimeno (2002), con lo que quedaba la duda de si sería capaz de superar al algoritmo que había demostrado dar los mejores resultados. En la presente Tesis Doctoral sí llevamos a cabo esta comparativa y la expondremos en el capítulo siguiente.

2.3.2.3. CPLEX 11.0

Para finalizar esta sección del estado del arte, hemos querido hacer una llamada de atención sobre CPLEX 11.0. Este software comercial se usa como solver en determinados pasos de alguno de los algoritmos que se han propuesto para la resolución del problema de máquinas paralelas no relacionadas, particularmente en los algoritmos de *rounding* o de dos fases, en los que la resolución de la relajación del problema resulta básica, como en Mokotoff y Jimeno (2002) o Shchepin y Vakhania (2005). Cuando se comprueba a la velocidad que resuelve CPLEX los problemas cuando las variables del problema no son binarias, no es de extrañar que la mayoría de los intentos actuales vayan por dirigidos a tratar de aprovechar esta ventaja.
En nuestro estudio, probamos a utilizar el CPLEX para la resolución directa del problema estándar; es decir con las variables de decisión binarias. La formulación del problema de máquinas paralelas no relacionadas, como ya se vio en la sección 2.3, es bastante simple, así que, tras analizar los primeros resultados obtenidos por el CPLEX 11.0, consideramos que, dado el buen comportamiento de este solver, debería ser parte de las mejores soluciones

a tener en cuenta. Es más, como se verá con detalle más adelante en el capítulo 3, la aplicación directa de este solver de carácter general, resulta mejor, en muchos casos, que algoritmos creados ad hoc para resolver este problema y que además se apoyan en el mismo CPLEX 11.0.
Por otra parte, hay que tener en cuenta que al ser CPLEX 11.0 un software comercial, resulta opaco para el usuario el desarrollo en detalle de los métodos que usa para resolver los distintos problemas. Por tanto, no nos resulta asequible el saber qué es lo que realmente sucede ni justificar porque en algunos casos va mejor y en otros peor. Con todo, lo que es innegable son los resultados que se obtienen de él, motivo por el cuál forma parte de nuestro estudio, tal y como se verá en el siguiente capítulo.

CAPÍTULO 3

ALGORITMOS INICIALES PROPUESTOS

Dada la complejidad de los algoritmos existentes para la resolución del problema de máquinas paralelas no relacionadas, así como la dependencia de algunos de ellos de solvers comerciales y la escasa robustez en algunos casos, se decidió buscar nuevos métodos que fueran sencillos y robustos, a la par que competitivos con el estado del arte.
En este capítulo se estudiarán en detalle cuatro nuevos algoritmos, introduciendo en primer lugar la parte algoritmica que es común a todos ellos. Cada algoritmo queda diferenciado del resto por el último de los pasos de la base algoritmica. Los dos primeros, llamados IGS e IGD, se basan en los llamados algoritmos iterativos voraces (*Iterated Greedy*), mientras que los dos últimos, llamados NSP y Vir, se basan en búsquedas locales de vecindarios pequeños o restringidos (*Restricted Local Search*). Continuaremos detallando las instancias empleadas y la implementación de todos los métodos evaluados. Se realizará un extenso análisis computacional y por último, se mencionarán las distintas variantes aplicadas y sus resultados.

3.1. Base de los algoritmos propuestos

Los algoritmos que hemos desarrollado para la resolución del problema de máquinas paralelas no relacionadas tienen una base común en su desarrollo, y luego una fase especifica que los hace bastante diferentes.

La base de los algoritmos empieza con la obtención de una solución inicial mediante un procedimiento muy rápido, seguida de dos búsquedas locales, una por inserción y otra por intercambio, donde en cada una de ellas se alcanza un óptimo local, y posteriormente se realiza la fase específica de cada algoritmo, para retornar a la fase de búsquedas locales.

Primero, haremos una aproximación al esquema general del algoritmo, para luego detallar en las sucesivas secciones cada una de las partes que lo componen. El procedimiento general es el siguiente:

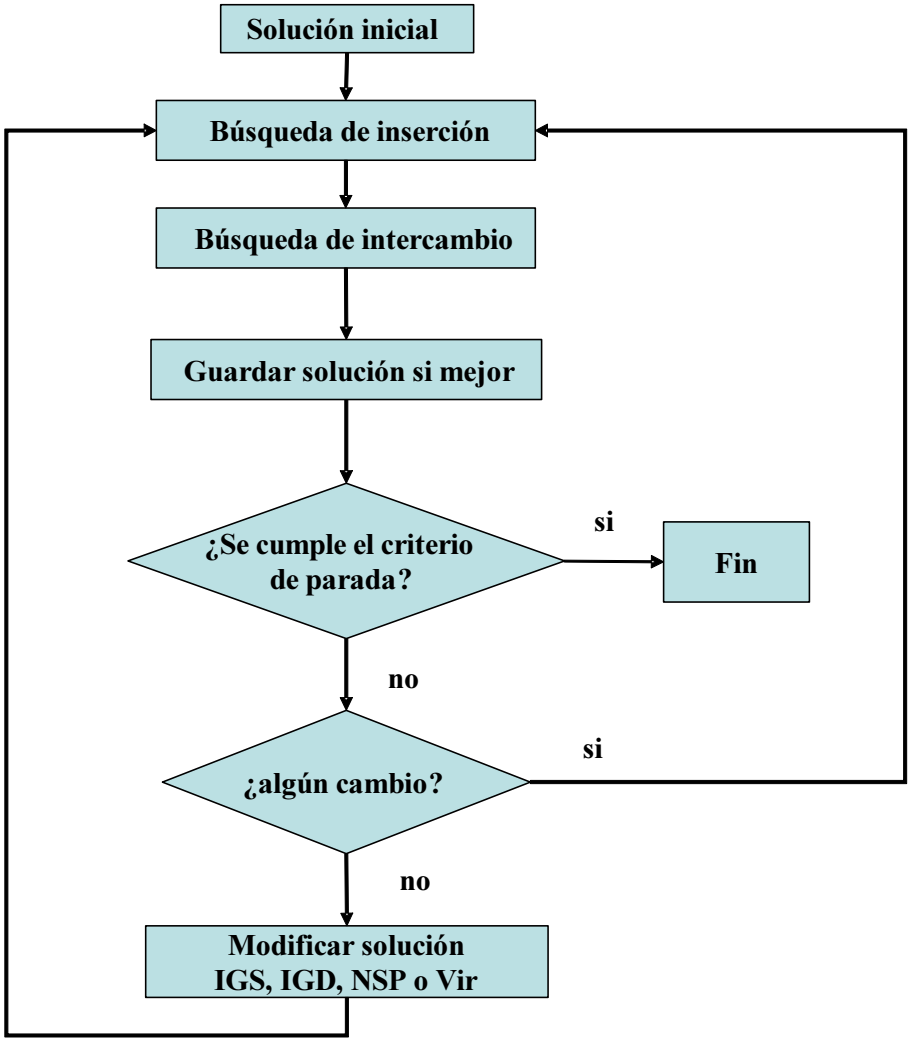

Figura 3.1 – Diagrama de flujo de los algoritmos propuestos.

1. Solución inicial basada en el menor tiempo de proceso de cada trabajo para todas las máquinas.

2. Búsqueda local por inserción: buscar, para cada trabajo j, qué máquina

hace que la suma del tiempo de proceso en ese trabajo en esa máquina, p_{ij}, más el tiempo de todos los trabajos que ya tiene asignados esa máquina C_i, sea mínimo, es decir la mejor opción (*best*). Esto puede provocar que el trabajo que antes estaba asignado a una máquina, sea reasignado a otra.

3. Búsqueda local por intercambio: busca el intercambiar un trabajo asignado a una máquina, por otro asignado a otra, siempre que $p_{i1,j1} + p_{i2,j2} > p_{i2,j1} + p_{i1,j2}$ ($i_1, i_2 \in M$, $j_1, j_2 \in N$, $j_1 \neq j_2$, $i_1 \neq i_2$) y no aumente el $C_{\text{máx}}$. Se intercambia con el primero que lo cumpla (*first*). Aquí, $p_{i1,j1}$ y $p_{i2,j2}$ representan los tiempos de proceso de los trabajos que intentamos intercambiar, y $p_{i2,j1}$ y $p_{i1,j2}$ los tiempos de proceso que tendrán tras el intercambio. Esta aplicación consecutiva de dos vecindarios se conoce como VNS (*Variable Neighborhood Search*) de Hansen y Mladenovic (2001).

4. Se guarda la solución como mejor si iguala o mejora la mejor encontrada hasta el momento.

5. Si se ha producido algún cambio en las búsquedas locales, se vuelve al paso 2.

6. Modificación de la solución obtenida (base de la variación de los distintos algoritmos) mediante IGS, IGD, NSP o Vir: por destrucción/reconstrucción o nueva búsqueda local restringida (diferente a las búsquedas locales anteriores).

7. Se vuelve a 2.

Finaliza, al acabar el tiempo asignado, o al cumplirse cualquier otro criterio de terminación.

Por expresarlo de una forma más rigurosa, la representación de la base de los algoritmos vendría dada por la figura 3.2.

```
procedure Base_de_algoritmos
    π := Inicialización de la solución;
    π_b := π;    % Mejor solución hallada hasta el momento
    while (criterio de finalización no satisfecho) do
        improved:=true;
        while(improved) do                          % Bucle VNS
            improved:=false;
            π' := Busqueda_Local_Inserción(π);      % hasta óptimo local
            π'' := Busqueda_Local_Intercambio(π'); % hasta óptimo local
            if π'' ≠ π then
                improved:=true;
                π := π'';
            endif
        endwhile
        if C_máx(π'') <= C_máx(π_b) then π_b := π'';
        π := Modificación_Solución(π_b);        % modificar por IG o RLS
    endwhile
end
```

Figura 3.2 – Pseudo-código del algoritmo base para todos los métodos propuestos.

Pasemos a detallar los diferentes pasos de la base de los algoritmos propuestos.

3.1.1. Solución inicial

La solución inicial de la que se parte es la obtenida al asignar cada trabajo j en la máquina i para la cual el tiempo de proceso p_{ij} es menor. Esta búsqueda tiene una complejidad computacional $\mathcal{O}(nm)$.
Esta solución es muy rápida de hallar y los experimentos demuestran que es igual o mejor solución inicial que otras más elaboradas, como ahora comentaremos.

Como ejemplo de soluciones iniciales probadas, podemos mencionar la

que se deriva de la anterior, pero en la que se mira cuál de los p_{ij} del trabajo j hace un tiempo de finalización C_i menor. Es decir, aquí se tiene en cuenta todos los demás trabajos que la máquina ya tiene asignados, para ver donde asignamos el trabajo de forma que el tiempo de finalización sea el menor posible. No obstante, hay que tener en cuenta que esta solución inicial es mucho más robusta ante distribuciones del p_{ij} muy asimétricas. Por ejemplo, si la máquina 1 tiene todos los tiempos de proceso $p_{1j} = 1$ ó igual al mínimo, con la primera opción de solución inicial, acabarían todos los trabajos asignados a la máquina 1, proporcionando una pobre solución inicial. En cualquier caso, dichos casos aberrantes son eso, aberrantes y poco esperables en la práctica. También, entre otras, se probó el no incluir una solución inicial, dado que la primera búsqueda local, la de inserción, no precisa necesariamente de una solución inicial previa. Este caso se probó por simplificación del algoritmo, pero tampoco proporcionó los mejores resultados. No obstante, se detallarán las variantes probadas en la sección 3.7.

3.1.2. Búsqueda local de inserción

La búsqueda local por inserción, representada por el pseudo-código de la figura 3.3, consiste en buscar, para cada trabajo j, en qué máquina deberíamos asignarlo para que la suma de todos los trabajos que esta tiene asignados más el tiempo de proceso del nuevo trabajo en esa máquina sea el mínimo posible. Es decir, se va trabajo por trabajo, y para cada trabajo, se va máquina por máquina viendo qué valor toma la suma total de los trabajos asignados a esa máquina (C_i) si se le añadiese el trabajo que se está estudiando en ese momento a esa máquina. El menor valor nos da la máquina donde hay que poner el trabajo. Nótese, que nuestro concepto de búsqueda local de inserción dista del concepto común donde normalmente se inserta un trabajo en todas las posibles posiciones de trabajos asignados a una máquina. Como hemos comentado en capítulos anteriores, esta búsqueda de inserción no tendría sentido para nuestro problema dado que la secuencia dentro de una máquina no tiene relevancia a

la hora de minimizar el *makespan*.

Una vez llegado al último trabajo, se comprueba si ha habido algún cambio a lo largo del proceso, es decir, si algún trabajo ha sido asignado a una máquina distinta de la que estaba antes de empezar el proceso. Si es así, se reinicia el proceso de búsqueda local de inserción. El proceso finaliza cuando no se produce ninguna variación con respecto a la solución anterior, por lo que todos los trabajos siguen asignados a las mismas máquinas. Se probaron otras variantes de esta forma de realizar la forma de inserción, no resultando dar mejores resultados, como se detallará en la sección 3.7.

Como cada uno de los n trabajos se intenta insertar en cada una de las m máquinas, incluyendo la máquina en donde estaba previamente, el número de pasos será de $n \cdot m$, es decir, la búsqueda local de inserción propuesta tiene una complejidad computacional de $\mathcal{O}(nm)$. Dichas combinaciones se harán todas, ya que se busca la mejor asignación para todos los trabajos. No obstante, debemos recordar que estamos hablando de simples sumas, con lo que la ejecución en tiempo de CPU es muy rápida.

```
procedure Busqueda_Local_Inserción
    π := solución entrante;
    improved := true;
    while(improved) do                    % hasta óptimo local
        improved := false;
        for j := 1 to n do
            i := máquina donde esta asignado el trabajo j;
            l = argmin_{h∈M/i} {C_h + p_hj};
            if (C_l + p_lj) < C_i then
                improved := true;
                π := extraer el trabajo j de la máquina i y asignar a máquina l;
            endif
        endfor
    endwhile
end
```

Figura 3.3 – Procedimiento de Búsqueda Local de Inserción empleado en la fase VND.

3.1.3. Búsqueda local de intercambio

Tal y como se puede apreciar en la figura 3.4, la búsqueda local por intercambio consiste en buscar una pareja de trabajos ya asignados, con tiempos de proceso $p_{i1,j1}$ y $p_{i2,j2}$, e intercambiarlos, con lo cual los tiempos de proceso que afectan a las nuevas asignaciones son, $p_{i2,j1}$ y $p_{i1,j2}$ y comprobar si mejora el $C_{\text{máx}}$. Por ejemplo, supongamos que el trabajo 1 esta asignado a la máquina 1 y el trabajo 2 a la máquina 2, ambos con sus tiempos de proceso correspondientes p_{11} y p_{22}. El intercambio supondría que el trabajo 1 pasaría a realizarlo la máquina 2, y que el trabajo 2 pasaría a realizarlo la máquina 1, es decir, los tiempos de proceso que afectan a la nueva asignación serían entonces p_{21} y p_{12}. El cambio será admisible, siempre y cuando no provocase un aumento de $C_{\text{máx}}$, ya que nos llevaría a un empeoramiento de la solución. Así mismo, el cambio estará permitido siempre y cuando la suma de los tiempos antes del cambio sea mayor que la suma de los tiempos después del mismo, es decir

cuando $p_{i1,j1} + p_{i2,j2} > p_{i2,j1} + p_{i1,j2}$. Esto nos asegura una reducción neta en los tiempos de proceso en las máquinas empleadas.

Llegados a este punto, queda por decidir qué estrategia seguir a la hora de realizar el cambio. Una sería permitir el primer cambio que cumpliera las restricciones arriba señaladas (*first*); la otra sería probar todos los intercambios posibles, y elegir de entre todos los cambios que cumplieran las restricciones, el que mejores valores proporcionara (*best*).
Los distintos experimentos realizados, que se detallarán en la sección 3.7, aconsejan elegir la estrategia *first*, que favorece sobre todo la rapidez del cálculo, añadiendo mayor variabilidad a los cambios. Es por tanto la estrategia seguida aquí. El intercambio se realiza para todos los trabajos hasta llegar al último. Una vez concluida esta parte, se comprueba si ha habido algún trabajo intercambiado entre dos máquinas cualesquiera. De ser así, se reinicia el proceso de búsqueda local de intercambio. El proceso finaliza cuando no se produce ninguna variación con respecto a la solución anterior, por lo que todos los trabajos siguen asignados a las mismas máquinas, llegando por tanto a un óptimo local con respecto a esta definición de vecindario.

Como cada uno de los n trabajos se intenta intercambiar con los restantes $n - 1$ trabajos, el número de combinaciones es de $n \cdot (n - 1)$, las cuales no se harán necesariamente todas, ya que se busca la primera asignación que mejore la suma de los intercambiados. Estamos hablando de simples sumas, con lo que la ejecución en tiempo de CPU es muy rápida, pero es más lento que el proceso anterior, el de inserción, ya que el número de trabajos suele ser muy superior al de las máquinas.

```
procedure Búsqueda_Local_Intercambio
   π := solución entrante;
   improved := true;
   while(improved) do              % hasta óptimo local
      improved := false;
      for j1 := 1 to n do
         for j2 := 1 to n do
            i1 := máquina donde el trabajo j1 está asignado;
            i2 := máquina donde el trabajo j2 está asignado;
            if j1 ≠ j2 and          % no intercambia con si mismo
               i1 ≠ i2 and          % no a la misma máquina
               p_{i1 j2} + p_{i2 j1} < p_{i1 j1} + p_{i2 j2} and    % balance positivo
               (C_{i1} + p_{i1 j2} − p_{i1 j1}) < C_máx and   % no empeorar C_máx
               (C_{i2} + p_{i2 j1} − p_{i2 j2}) < C_máx then
                     π := quitar el trabajo j1 de la máquina i1 y
                        el trabajo j2 de la máquina i2;
                     π := asignar el trabajo j1 a la máquina i2 y
                        el trabajo j2 a la máquina i1;
                     improved:= true;
                     exitfor        % estrategia primero que mejora
            endif
         endfor
      endfor
   endwhile
end
```

Figura 3.4 – Procedimiento de Búsqueda Local de Intercambio empleado en la fase VND.

3.1.4. Modificación de la solución obtenida y criterio de parada

Anteriormente hemos visto los bloques constructivos de todos los algoritmos propuestos. Ahora detallamos la modificación de la solución, que varía de uno a otro algoritmo. Vemos primero una panorámica general y luego el detalle.

Básicamente, en esta parte del algoritmo, se toma la solución hallada hasta

el momento, y se modifica; bien por un proceso de destrucción/reconstrucción de parte de la solución (algoritmo voraz iterativo), o bien por una nueva búsqueda local restringida o de pequeños vecindarios. Tanto en un caso como en el otro, el objetivo es una pequeña variación de la solución, suficiente para escapar de un óptimo local, con la que volveremos al ciclo inicial de las búsquedas locales de inserción e intercambio. Ambos casos serán detallados más adelante, en la sección 3.2 y en la sección 3.3, respectivamente.

El algoritmo finaliza cuando alcanza el tiempo marcado, o cualquier otro criterio de parada que quiera establecerse, como, por ejemplo, a un cierto número de iteraciones.

3.1.5. Ensamblado del algoritmo

Una vez estudiadas todas las partes de las que consta el algoritmo, vamos a pasar a pormenorizar en algunos detalles a tener en cuenta cuando el algoritmo se une para formar un todo. Hemos visto al principio de la sección 3.1 que en primer lugar se hace una búsqueda local por inserción, seguida de una de intercambio, y que dentro de cada una de ellas no se sale hasta alcanzar un óptimo local, es decir hasta que no le es posible mejorar la solución. Cuando se integran en un todo estas búsquedas, hay que controlar, al final de la segunda de ellas (búsqueda local por intercambio), que si ha habido algún cambio, se vuelva de nuevo a empezar el ciclo. A todo este conjunto se lo conoce como VNS (*Variable Neighborhood Search*) de Hansen y Mladenovic (2001), como ya hemos comentado al inicio de la sección 3.1. Dicho de otro modo, primero se hace una búsqueda local por inserción, después una búsqueda local por intercambio, si al finalizar ésta se constata que durante el proceso ha habido algún cambio, se empieza de nuevo con la primera, la búsqueda local por inserción. Por tanto este ciclo tipo VNS acaba cuando ya no se puede mejorar nada la solución a base de estas búsquedas locales. Todo esto es muy eficaz, dando muy buenos resultados, pero encierra un alto peligro de ciclado, ya

que una modificación realizada por una inserción, puede ser deshecha por un intercambio en la siguiente fase. Esto es debido a la manera que se aceptan los movimientos para el problema estudiado, sobre todo en la búsqueda local de intercambio.

Pongamos un ejemplo. Supongamos que tenemos la tabla 3.1 de tiempos de proceso, donde J_i representa los trabajos, M_i representa las máquinas, en negrita y cursiva se representan los trabajos asignados a las distintas máquinas y C_i es la suma de los tiempos de proceso de los trabajos asignados a cada máquina i.

	J_1	J_2	J_3	J_4	J_5	C_i
M_1	5	***2***	***12***	45	56	**14**
M_2	***3***	1	39	***17***	63	**20**
M_3	48	55	61	38	***25***	**25**

Tabla 3.1 – Tabla de tiempos de proceso de ejemplo de ciclado: situación inicial.

Podemos ver, en la tabla 3.1, que las variaciones que se van a producir por inserción e intercambio serán en los trabajos J_1 y J_2 ya que el resto tienen los valores más bajos ya asignados, así que nos centraremos en los cambios de estos trabajos.
Tras la primera inserción quedará tal y como aparece en la tabla 3.2; en ella se acepta la inserción del trabajo J_1 en la máquina M_1 ya que al principio estaba en la máquina M_2 con un C_2 de 20, y deja un C_2 de 17, subiendo el C_1 de 14 a 19, el cual es inferior al $C_{\text{máx}}$, que es 25. Además, de todas las máquinas, la inserción en la máquina M_1 es la que proporciona un C_i menor.

	J_1	J_2	J_3	J_4	J_5	C_i
M_1	***5***	***2***	***12***	45	56	**19**
M_2	3	1	39	***17***	63	**17**
M_3	48	55	61	38	***25***	**25**

Tabla 3.2 – Tabla de tiempos de proceso de ejemplo de ciclado tras la primera inserción.

Ahora se procedería a una segunda inserción. En este caso, la tabla 3.2 muestra como se inserta el trabajo J_2 desde la máquina M_1 a la máquina M_2, ya que el C_1 pasa de 19 a 17 y el C_2 no aumenta por encima del $C_{\text{máx}}$, siendo el menor C_i posible en esta inserción.

	J_1	J_2	J_3	J_4	J_5	C_i
M_1	***5***	2	***12***	45	56	**17**
M_2	3	***1***	39	***17***	63	**18**
M_3	48	55	61	38	***25***	**25**

Tabla 3.3 – Tabla de tiempos de proceso de ejemplo de ciclado tras la segunda inserción.

En este momento se finaliza la fase de inserciones al no poderse efectuar ningún cambio más, según las condiciones impuestas, y empieza la fase de intercambio (tabla 3.4). Aquí el intercambio se produce entre J_1 y J_2 ya que la suma de los tiempos asignados del trabajo J_1 en la máquina M_1 y del trabajo J_2 en la máquina M_2 $(5 + 1)$ es mayor que la suma de los del intercambio de hacer el trabajo J_1 en la máquina M_2 y hacer el trabajo J_2 en la máquina M_1 $(3 + 2)$ y no se supera el $C_{\text{máx}}$, que es 25. Esto nos devuelve a la situación inicial, con lo que, al haberse producido un cambio, y volver a empezar con las inserciones, nos llevaría a un ciclado del algoritmo.

	J_1	J_2	J_3	J_4	J_5	C_i
M_1	5	***2***	***12***	45	56	**14**
M_2	***3***	1	39	***17***	63	**20**
M_3	48	55	61	38	***25***	**25**

Tabla 3.4 – Tabla de tiempos de proceso de ejemplo de ciclado tras la fase de intercambio.

Se podría pensar que el ciclado sería evitable si no se permitieran unos tiempos de finalización C_i mayores que los precedentes antes del intercambio, pero esto supone el restringir unos posibles huecos en algunas máquinas que permiten alcanzar nuevos mínimos del $C_{\text{máx}}$ y condena la búsqueda local a malas soluciones.

Estos ciclos pueden ser de dos cambios (por inserción se hace un cambio y por intercambio se deshace), pero también pueden ser ciclos más amplios, como por ejemplo de 3, 4, etc. (por inserción se hace un cambio, por intercambio otro distinto, al volver a la inserción se hace un nuevo cambio y finalmente por intercambio se vuelve a la primera solución). Hay varias formas de tratar de evitar esto, como el mantener una lista de los cambios para evitar que se repitan, como en las listas tabú, pero tras algunas pruebas vimos que lo más rápido y sencillo era evitar los ciclos de dos, y añadir un contador que hiciera salir del bucle si alcanzaba cierto umbral, el cuál era bastante pequeño, ya que raramente se alcanzan los 10 ciclos tipo VNS antes de agotar ambas búsquedas locales. Otra cosa a tener en cuenta es, que cada vez que acaba esta fase, el algoritmo debe comprobar si la solución ha mejorado o igualado el mejor resultado que se tenía hasta el momento. De ser así, está pasa a ser la mejor solución y es la que pasará a la siguiente fase, la de modificación, antes de devolverla de nuevo a la fase de las dos búsquedas locales. En caso de que no mejore la mejor solución obtenida, se recuperara la mejor solución que había hasta ese momento.

Hay que hacer notar que, tras varias pruebas, se determinó que aceptar una solución que fuera igual que la que ya se tenía mejoraba bastante los resultados obtenidos. Esto es debido a que el algoritmo lleva a distintas posibles soluciones desde donde partir para encontrar una nueva mejor solución. Adicionalmente, también experimentamos con la idea de aceptar, temporalmente, soluciones peores, tal y como se hace en *Simulated Annealing* (SA). No obstante, todos estos experimentos, como ya detallaremos en la sección 3.7, no ofrecieron mejoras sustanciales.

En las siguientes secciones, pasamos a especificar las distintas formas de las modificaciones de las soluciones que dan lugar a los distintos algoritmos y sus variantes, no sin antes recordar, que una vez modificada la solución, se envía de nuevo a las fases previas, donde las búsquedas locales tanto de inserción como de intercambio mejoran la solución modificada.

3.2. Algoritmo iterativo voraz

Dentro de la fase de modificación de la solución, uno de los algoritmos probados ha sido el conocido como algoritmo iterativo voraz (en inglés *iterated greedy* o IG) desarrollado por Ruiz y Stützle (2007), basado en una destrucción parcial de la solución y una posterior reconstrucción. Se han probado dos tipos de algoritmos iterativos voraces, los cuales que se detallarán en las siguientes secciones; el estándar y una variación sobre este, basado en una selección más dirigida de los trabajos a destruir. En el algoritmo estándar, adaptado del propuesto por Ruiz y Stützle (2007), se escogen al azar una serie de trabajos que se eliminan, en la fase de destrucción, de la solución. En esta elección de trabajos a destruir, radica la diferencia entre ambas variantes que proponemos.

3.2.1. Algoritmo iterativo voraz estándar (IGS)

En este primer caso, se escogen al azar entre todos los trabajos, los que van a ser eliminados, con la única limitación de no repetirlos.

Acabada la fase de destrucción, se pasa a la fase de reconstrucción, en la cuál se toma la lista de todos los trabajos destruidos, y se procede a su reconstrucción buscando, para cada trabajo, en qué máquina i quedaría mejor asignado para dar una suma de tiempos de finalización menor. Es decir, para cada trabajo destruido j, se busca aquella máquina i que, tras toda la destrucción acaecida, hace que la suma de los tiempos de proceso de los trabajos que le quedan asignados, más el tiempo de proceso del trabajo j en la máquina i, p_{ij}, resulta menor. Los resultados preliminares mostraron un mejor resultado si se destruía un número fijo de trabajos, que si se eliminaba un tanto por cien de los trabajos, tal y como se detallará en la sección 3.7.4. Esto es debido a que el algoritmo trabaja mejor con un número determinado de combinaciones que debe hacer, a la hora de reconstruir la solución, que con un número variable en función del tamaño de los trabajos que intervienen. En este último caso, el número de combinaciones varían con el tamaño, por ejemplo, para un mismo 5 % de trabajos a destruir, en instancias de 100 trabajos representan 5 trabajos a destruir, pero en 1000 trabajos suponen 50 trabajos destruidos, lo que a la hora de recomponer la solución, lo hace más costoso. Igualmente se detectó que resultaba mejor que, del total de trabajos a destruir, se seleccionara una parte de ellos dentro de los asignados a la máquina que detentaba el $C_{\text{máx}}$ y la otra parte de entre el resto de los trabajos asignados a las otras máquinas. Para instancias con un menor número de trabajos que las instancias estudiadas, $n < 100$, se puede aplicar el principio simple de destruir el 30 % de los trabajos al azar. Por otra parte, el trabajo a eliminar de la máquina elegida, se selecciona de manera aleatoria. Es decir, una vez determinada la máquina de la que se va a destruir el trabajo, la selección del trabajo a destruir de entre los asignados a dicha máquina, se hace siguiendo una distribución uniforme entre ellos.

Veamos un ejemplo de todo el proceso; primero en la figura 3.5 se selecciona un primer trabajo a destruir, procedente de la máquina con mayor C_i, es decir, de la máquina con el $C_{\text{máx}}$ en este caso la máquina 3, ya que tiene un tiempo total de finalización de 30.

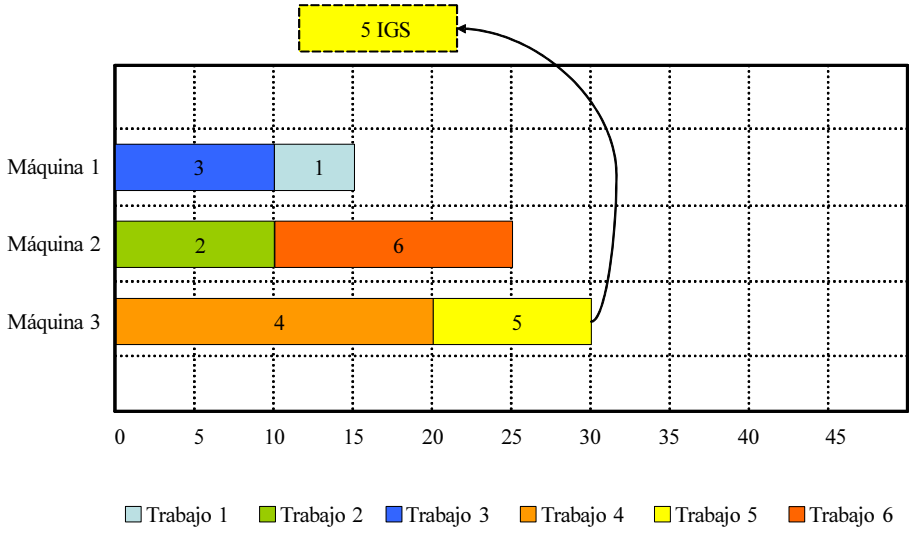

Figura 3.5 – Ejemplo de algoritmo iterativo voraz estándar (IGS): Selección de un primer trabajo a destruir de la máquina con $C_{\text{máx}}$.

La figura 3.6 representa la selección de un segundo trabajo a destruir, esta vez de una máquina elegida al azar. En el ejemplo el trabajo pertenece a la máquina 1.

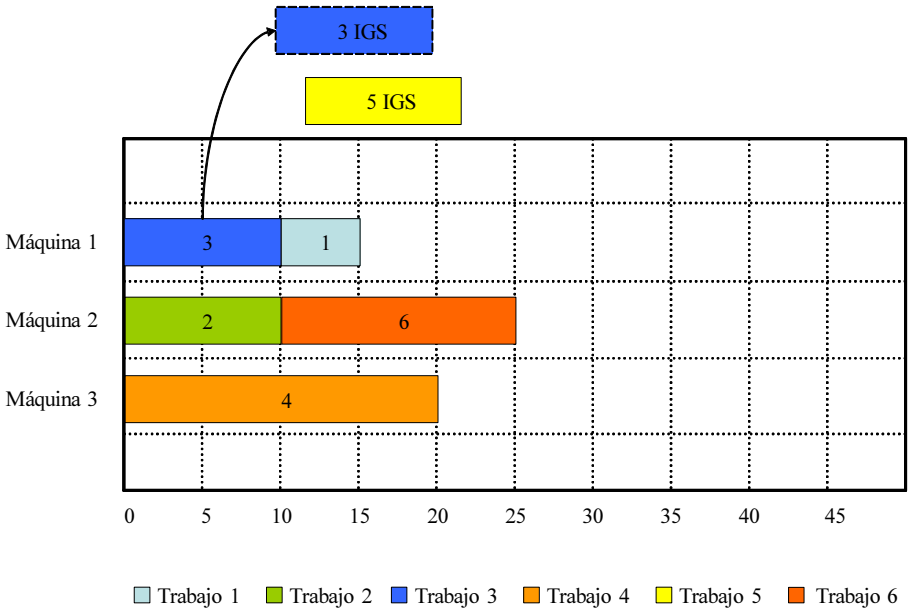

Figura 3.6 – Ejemplo de algoritmo iterativo voraz estándar (IGS): Selección de un segundo trabajo a destruir de una máquina al azar.

Una vez finalizada la fase de destrucción, se procede a la reconstrucción de la solución a base de buscar aquella máquina en donde los trabajos destruidos hacen menor el tiempo de C_i. En la figura 3.7 podemos ver como el trabajo 5 se asigna a la máquina 1 donde alcanza un $C_1 = 15$, mientras que asignarlo a cualquier otra máquina haría que cualquiera de sus C_i fueran superiores.

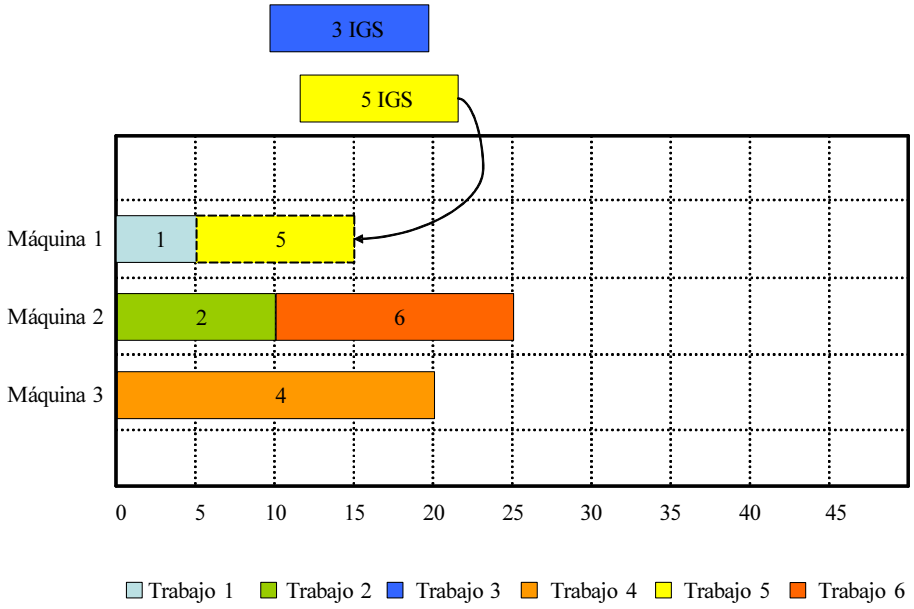

Figura 3.7 – Ejemplo de algoritmo iterativo voraz estándar (IGS): Reasignación del primer trabajo en la mejor máquina posible.

Se prosigue asignando el resto de trabajos destruidos buscando, de nuevo, el mejor valor de C_i para los trabajos a reasignar. En nuestro ejemplo, la figura 3.8 obtiene el mejor valor si el trabajo 3 es asignado a la máquina 1, donde llega hasta el valor de 25.

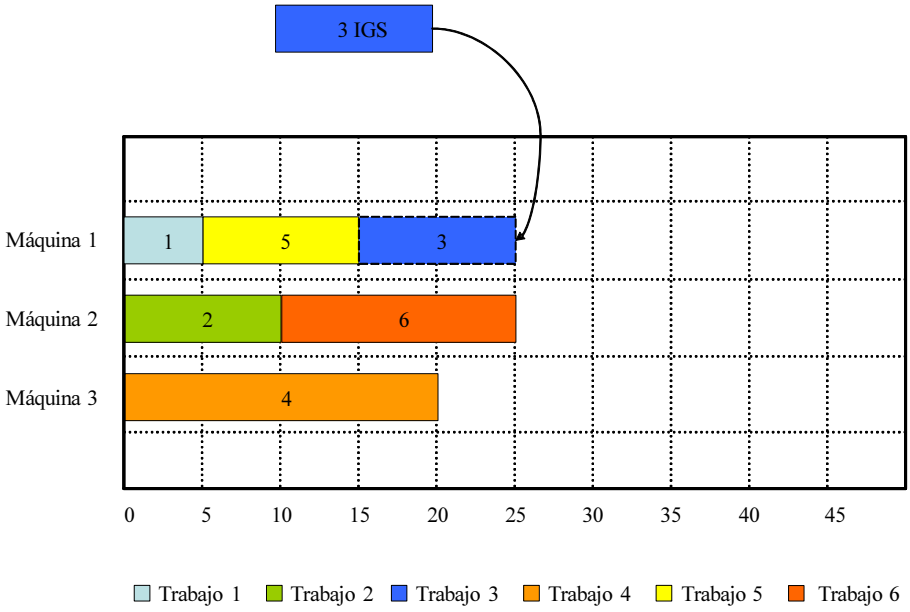

Figura 3.8 – Ejemplo de algoritmo iterativo voraz estándar (IGS): Reasignación del otro trabajo en la mejor máquina posible.

La situación final, tras las fases de destrucción y reconstrucción de la solución, está representada por la figura 3.9.

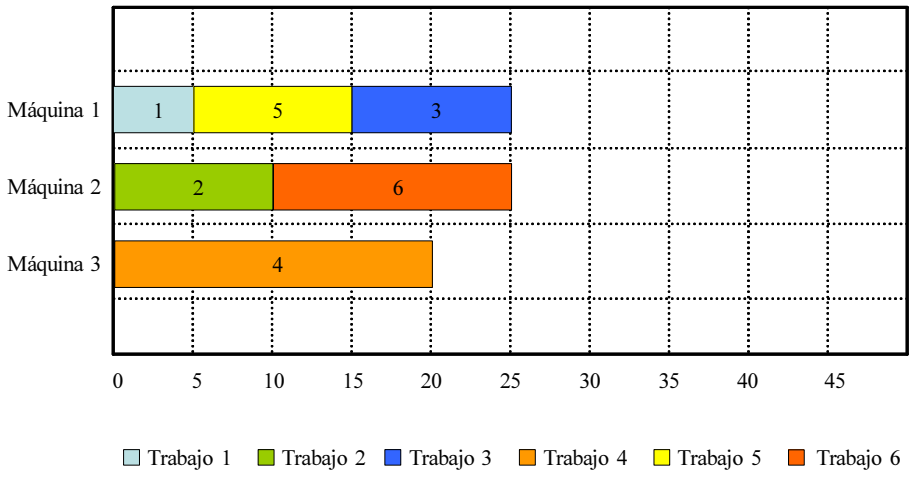

Figura 3.9 – Ejemplo de algoritmo iterativo voraz estándar (IGS): Estado final.

Dado que se han de fijar las variables de número de trabajos a destruir tanto del $C_{\text{máx}}$ como del resto, se calibró el algoritmo para ver su comportamiento y ajustar las variables, como se verá desarrollado en el anexo C, quedando estas fijadas en destruir 20 trabajos del $C_{\text{máx}}$ más 20 trabajos al azar.

3.2.2. Algoritmo iterativo voraz dirigido (IGD)

En este segundo caso, se trata de que la destrucción sea siempre de la máquina que en ese momento ostente el $C_{\text{máx}}$.

Como primer paso de la fase de destrucción, se selecciona un trabajo de los asignados a la máquina que ostenta el $C_{\text{máx}}$. En la figura 3.10 se destruye un trabajo de la máquina 3, la que en ese momento tiene el valor más alto de $C_i = 30$ ($C_{\text{máx}}$).

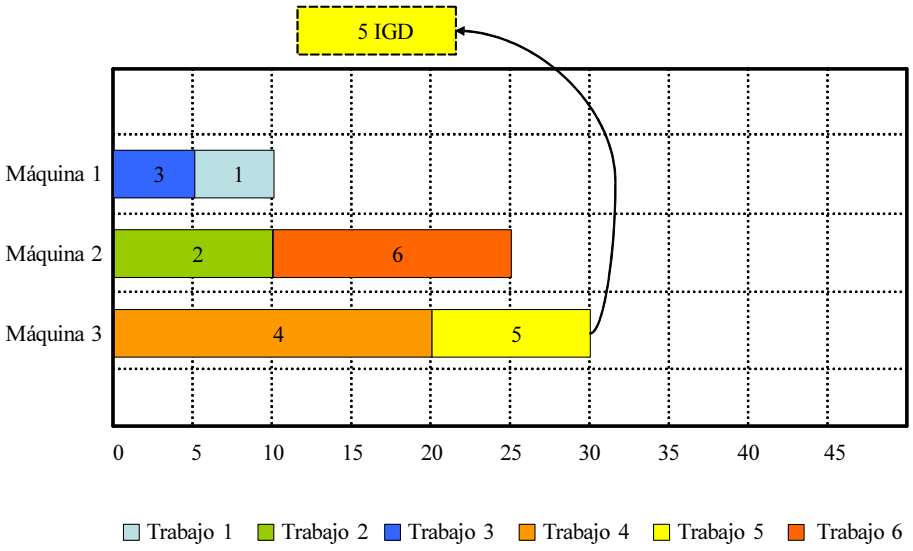

Figura 3.10 – Ejemplo de algoritmo iterativo voraz dirigido (IGD): Selección de un primer trabajo a destruir de la máquina con mayor $C_{\text{máx}}$.

El segundo trabajo a destruir saldría de la máquina que ahora, tras la destrucción del primer trabajo, ostentara el nuevo $C_{\text{máx}}$. Tal y como se puede apreciar en la figura 3.11, el segundo trabajo se selecciona de la máquina 2, la que en este momento es la máquina con el $C_{\text{máx}} = 25$.

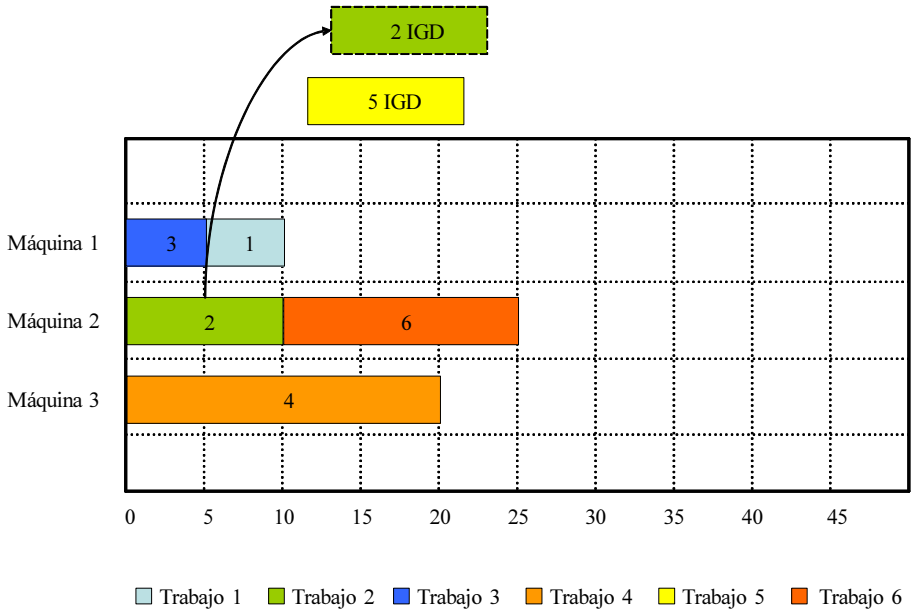

Figura 3.11 – Ejemplo de algoritmo iterativo voraz dirigido (IGD): Selección de un segundo trabajo a destruir de la máquina con mayor $C_{\text{máx}}$ en ese momento.

La figura 3.12 nos muestra como, tras la destrucción, la fase de reconstrucción asigna el primer trabajo destruido a la máquina en la que produciría un menor C_i, en este caso en la máquina 1.

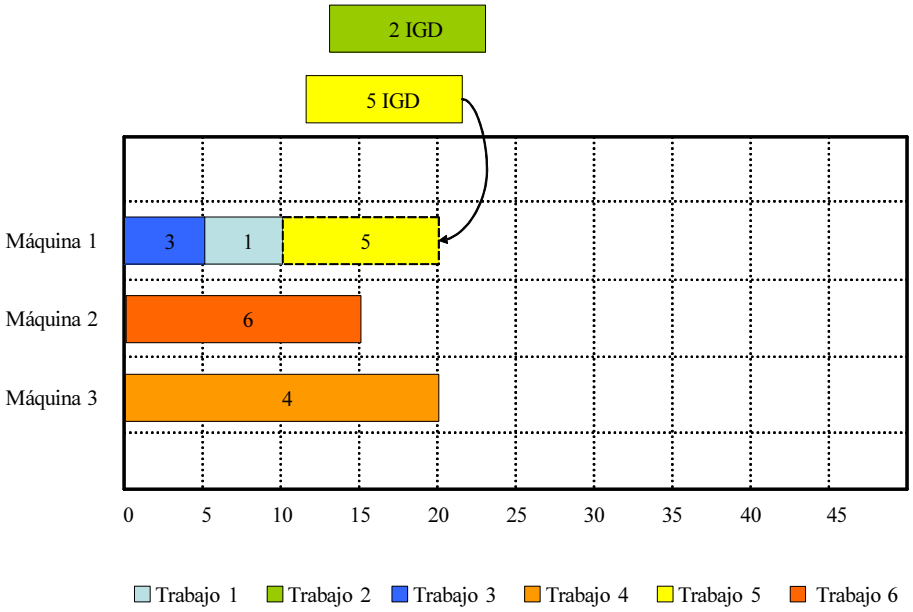

Figura 3.12 – Ejemplo de algoritmo iterativo voraz dirigido (IGD): Reubicación de uno de los trabajos en la mejor posición posible.

El segundo trabajo quedará asignado a la máquina 2, máquina en la que hará menor el C_i, en este caso 25, representado en la figura 3.13 .

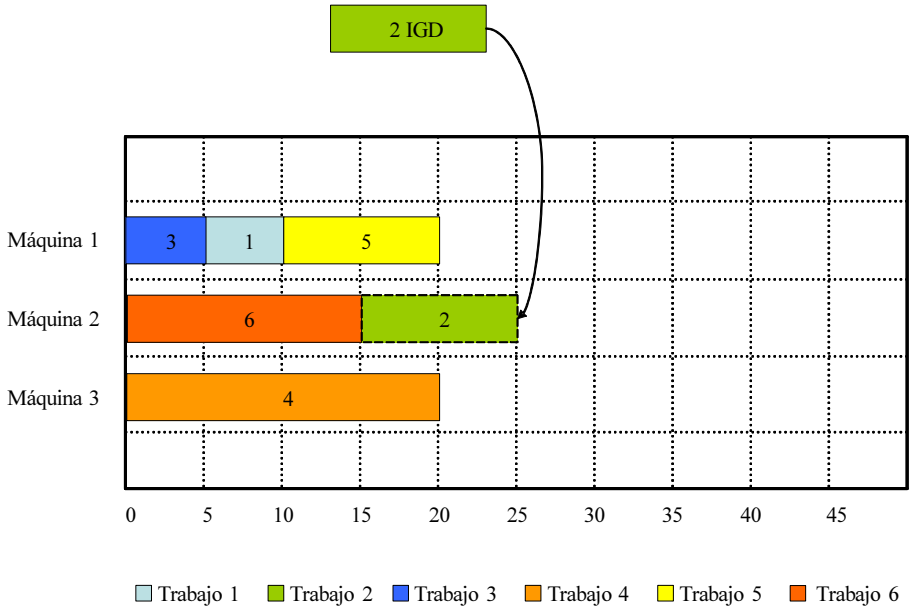

Figura 3.13 – Ejemplo de algoritmo iterativo voraz dirigido (IGD): Reubicación del otro trabajo en la mejor posición posible.

Por último, la figura 3.14 nos muestras el estado final tras la aplicación del algoritmo IGD.

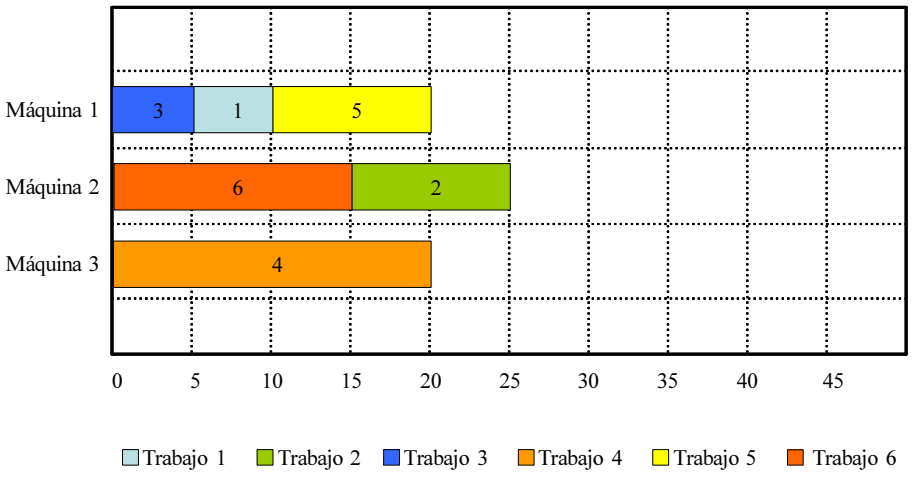

Figura 3.14 – Ejemplo de algoritmo iterativo voraz dirigido (IGD): Estado final.

Durante la fase de calibración, para la determinación de su comportamiento y la elección del número de trabajos a destruir, como se verá desarrollado en el anexo C, quedaron determinados en un total de 20 trabajos a destruir para el IG dirigido.

La fase de reconstrucción del IG dirigido es igual a la del IG estándar con la salvedad del orden en que se procede a reconstruir los trabajos. Para el IG estándar, los trabajos se reconstruyen en el mismo orden en el que se destruyen, es decir, primero los de la máquina de $C_{\text{máx}}$, y posteriormente el resto. Hay que intentar colocar los trabajos de la máquina del $C_{\text{máx}}$ lo mejor posible. En cambio, en el IG dirigido, los trabajos a eliminar son de máquinas que van ostentando un $C_{\text{máx}}$ parcial, por lo que, a la hora de reconstruir, hacemos una ordenación al azar, para ganar en variabilidad. Esto demostró resultar positivo a la hora de los resultados experimentales, como se verá en la sección 3.6.

3.3. Búsquedas locales en vecindarios pequeños o restringidos (RLS)

Las búsquedas locales en vecindarios pequeños o restringidos (*restricted local search* o RLS), se basan en la idea de modificar una solución, buscando, para un grupo reducido de trabajos, una variación de su asignación según cierto criterio de búsqueda. Esto es mucho más rápido que buscar en todo el vecindario completo. La variación en el criterio de búsqueda es lo que nos proporcionará los distintos algoritmos que vamos a estudiar.

Una de las diferencias fundamentales con el algoritmo iterativo voraz (IG), viene dada por el hecho de que en el IG se destruyen previamente todos los trabajos que intervendrán en la modificación de la solución, mientras que en las búsquedas locales en vecindarios pequeños o restringidos, las soluciones son siempre completas y todos los trabajos están presentes.

Dentro de estas RLS, hemos desarrollado dos tipos. La primera se basa en buscar la mejor máquina para un trabajo, siempre que ésta sea distinta a la máquina en la que estaba, y la segunda busca el asignar el trabajo a la mejor máquina sin haber quitado la asignación previa que tenía.

3.3.1. Búsqueda local restringida-No al mismo sitio (NSP)

En este tipo de modificación de la solución (*no same place* o NSP), se trata de elegir un trabajo para el cual se busca qué máquina, dentro de todas las máquinas posibles excepto donde ya está asignado, hace que la suma del tiempo de finalización de esta máquina C_i más el tiempo de proceso de ese trabajo en esa máquina p_{ij} sea el menor posible. Una vez hallada, se desasigna de donde estaba asignado, y se le da como nueva asignación la máquina de menor $C_i + p_{ij}$. Es de destacar que no hace falta desasignar previamente el trabajo a modificar, ya que es condición necesaria no asignarlo a la misma

máquina. Se procede nuevamente de este modo hasta alcanzar un número predeterminado de trabajos a modificar.

Veamos un ejemplo. La figura 3.15 podemos ver como el trabajo que se decide modificar, el trabajo 1, se prueba en todas las máquinas excepto donde ya estaba. El menor C_i se obtiene de su asignación a la máquina 2, donde alcanza el valor de 30, mientras que en la máquina 3 se llegaría hasta los 40. Nótese que el trabajo 1 no ha sido necesario desasignarlo de la máquina 1 ya que el algoritmo fuerza a no comprobar la asignación en esa máquina.

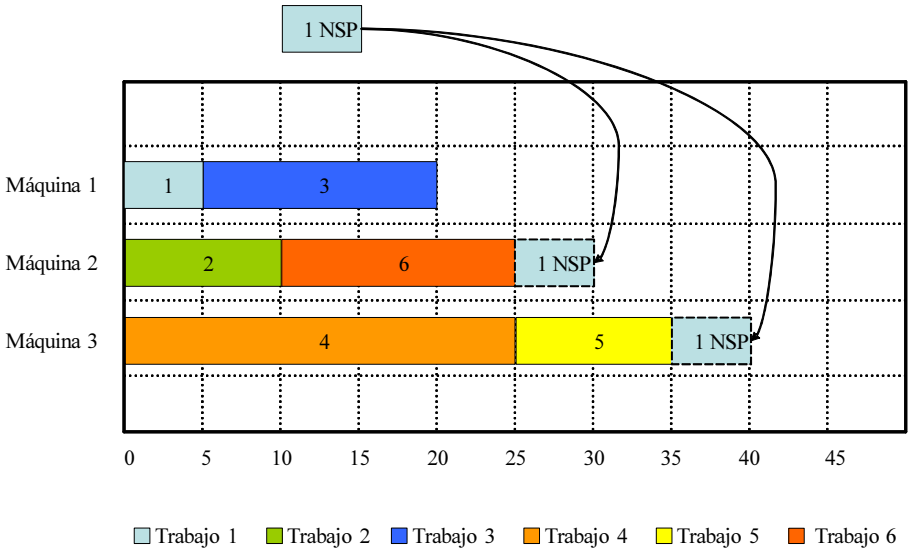

Figura 3.15 – Ejemplo del algoritmo de búsqueda local en vecindarios pequeños o restringidos tipo No al mismo sitio (NSP): Proceso.

Finalmente, se asigna el trabajo 1 a la máquina 2 y se desasigna de la máquina 1. Esto se realizaría con tantos trabajos como previamente se hubiese determinado.

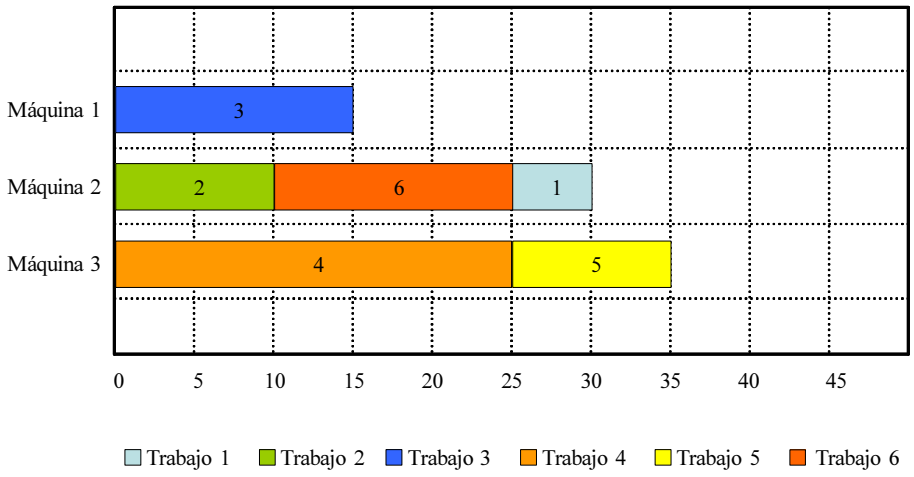

Figura 3.16 – Ejemplo del algoritmo de búsqueda local en vecindarios pequeños o restringidos tipo No al mismo sitio (NSP): Estado final.

Para la determinación del número de trabajos a modificar y así como para observar su comportamiento, se procedió a una calibración del algoritmo, como se verá en el anexo C, encontrando que, al igual que en casos anteriores, la mejor opción viene dada por la modificación de trabajos del $C_{\text{máx}}$ y otros al azar. En este caso se quedo fijados en 1 solo trabajo a modificar de la máquina del $C_{\text{máx}}$ y 1 solo trabajo del resto de las máquinas.

3.3.2. Búsqueda local restringida-Virtual (Vir)

Este último algoritmo, resulta ser de los cuatro algoritmos que se proponen, el menos ortodoxo de todos ellos. Se basa en la misma filosofía que el anterior, donde hay un número fijo de trabajos a modificar; para cada uno de ellos se busca qué máquina, dentro de todas las máquinas posibles, hace que la suma del tiempo de finalización de esta máquina C_i más el tiempo de proceso de ese trabajo en esa máquina p_{ij} sea el menor, pero en este caso incluyendo también la máquina donde ya estaba asignado. Una vez hallada, se desasigna

de donde estaba asignado, y se le da como nueva asignación la máquina de menor $C_i + p_{ij}$.

En este caso, hay que fijarse que la desasignación es posterior a la determinación de la mejor posición posible, y se comprobara también el resultado de $C_i + p_{ij}$ para la máquina en la que aún está asignado.

Dicho de otro modo, al no haber sido desasignado el trabajo, y permitir la asignación a la misma máquina donde originalmente está, es como si se tratara de asignar dos veces el mismo trabajo, uno real que aún esta asignado en la máquina original, y otro virtual que es el que se trata de secuenciar en todas las máquinas para buscar su mejor posición. Esto nos lleva a que, en ocasiones, el trabajo se vuelva asignar a la misma máquina a pesar de que no lo hemos quitado todavia, es decir, tiene asignado el trabajo real y el virtual. Evidentemente, al finalizar, se desasigna el trabajo real y se deja en trabajo virtual en la nueva asignación, que, en el caso de que coincidan en la misma máquina, no variara la solución original. Lo que esto nos proporciona son trabajos cuya asignación en una determinada máquina es tan buena que, aunque tuviera que repetirse el trabajo, se seguiría asignando a la misma máquina. Esto viene a suavizar el algoritmo anterior, donde muchas veces, y como se puede ver en los ejemplos usados, seleccionar un trabajo a quitar de una máquina y no permitir que vuelva a esa máquina, empeora, de facto, la solución inicial.

En el ejemplo de la figura 3.17 el trabajo 1 es el que vamos a modificar. Se busca la posición que hace menor el tiempo de finalización (C_i), sin haber desasignado previamente el trabajo 1 de la máquina 1. En este caso la mejor opción es colocarlo en la máquina 1, es decir, habría momentáneamente dos trabajos 1 en la máquina 1.

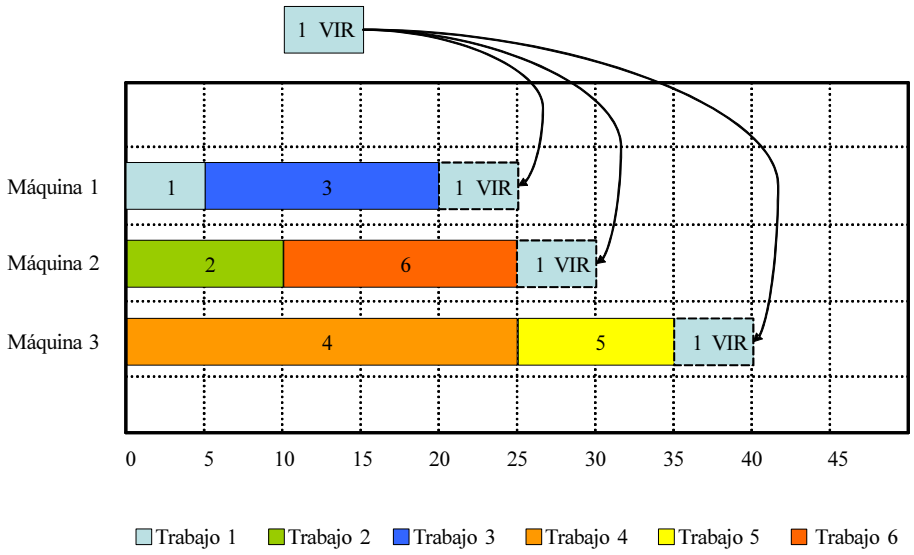

Figura 3.17 – Ejemplo 1 del algoritmo de búsqueda local en vecindarios pequeños o restringidos tipo Virtual: Proceso.

El resultado final que refleja la figura 3.18 muestra como el trabajo 1 se mantiene en la máquina 1 y se ha procedido a la desasignación del trabajo 1 original de dicha máquina, por lo que solo queda un trabajo 1. Para el objetivo del R//$C_{\text{máx}}$ resulta indiferente el orden de secuenciación en la máquina, como ya se comentó.

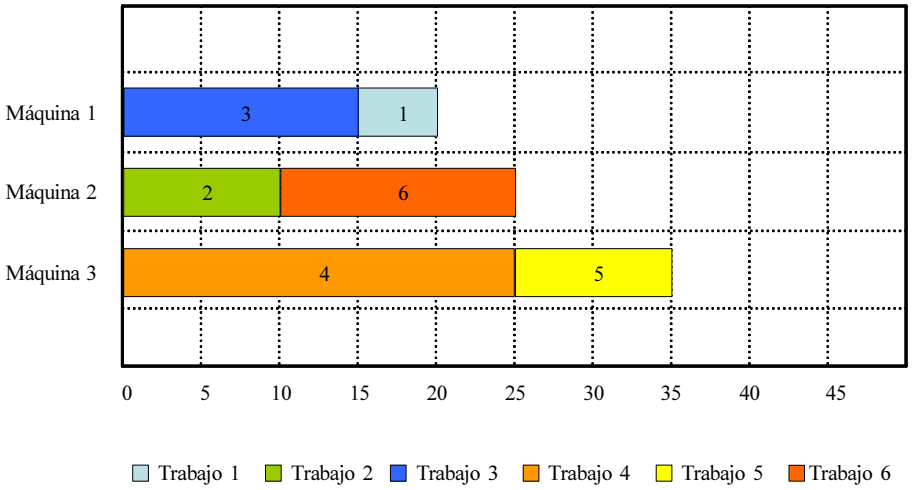

Figura 3.18 – Ejemplo 1 del algoritmo de búsqueda local en vecindarios pequeños o restringidos tipo Virtual: Estado final.

Veamos un segundo ejemplo donde el trabajo a modificar no se quede en la máquina original. En la figura 3.19 podemos ver como el trabajo a modificar en esta ocasión es el trabajo 2. En este caso, el trabajo 2 virtual sí que tiene una mejor posición en una máquina distinta (máquina 1) a donde originalmente esta asignado (máquina 2).

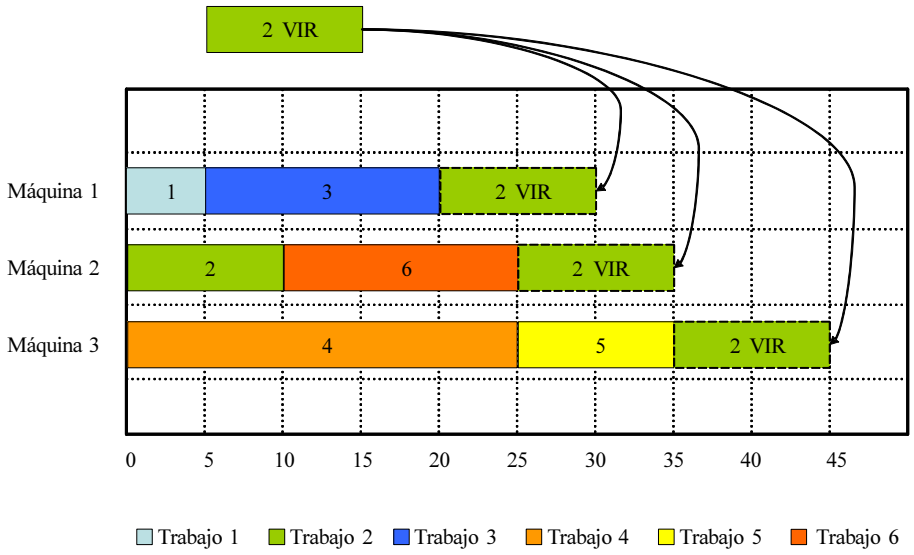

Figura 3.19 – Ejemplo 2 del algoritmo de búsqueda local en vecindarios pequeños o restringidos tipo Virtual: Proceso.

La situación final queda reflejada en la figura 3.20, donde se ha asignado el trabajo 2 a la máquina 1 y desasignado dicho trabajo de la máquina donde originalmente estaba (máquina 2).

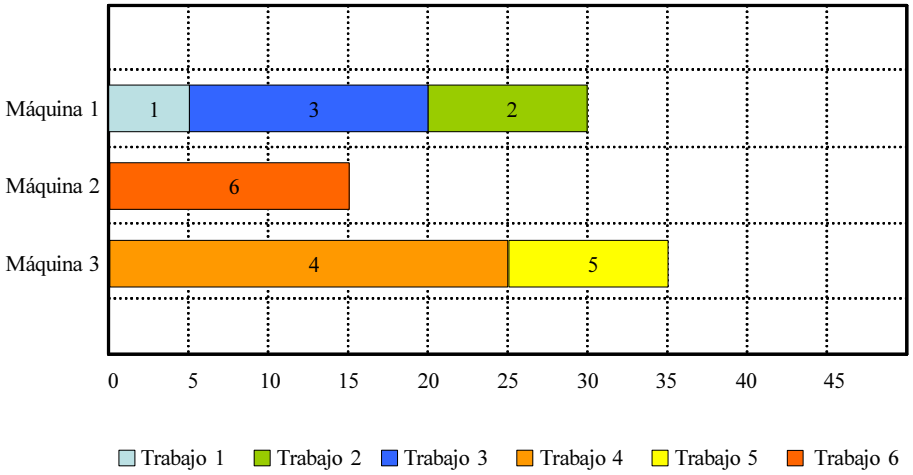

Figura 3.20 – Ejemplo 2 del algoritmo de búsqueda local en vecindarios pequeños o restringidos tipo Virtual: Estado final.

Dentro de la calibración, se observó que un número de modificaciones de 5 trabajos para la máquina con el $C_{\text{máx}}$ y otros 5 al azar, proporcionaba un buen comportamiento y resultados, como se verá en detalle en el anexo C.

3.4. Instancias

En general, cada autor genera sus propias instancias, como por ejemplo Martello, Soumis y Toth (1997), Sourd (2001) o Mokotoff y Jimeno (2002), pero posteriormente no son publicadas, además de no ser reutilizadas, por lo que para el presente estudio creamos un conjunto de instancias de los más amplios empleados en la literatura del problema de R//$C_{\text{máx}}$.

En general, casi todos los autores que han escrito sobre este problema usan tres tipos diferentes de intervalos:

- Los basados en una distribución uniforme, donde no hay ningún tipo de

relación entre las máquinas ni entre los trabajos, y los tiempos p_{ij} se asignan sacando un número al azar de entre los limites de esa distribución. Por ejemplo, en una distribución $U(1, 100)$, significaría que, los p_{ij} para los distintos trabajos y máquinas, podrían tomar cualquier valor entre 1 y 100.

- Los basados en una relación entre los trabajos, donde cada tiempo de proceso p_{ij} viene dado por la expresión $p_{ij} = b_j + d_{ij}$ donde b_j y d_{ij} son dos números generados al azar entre las distribuciones uniformes $U(1, 100)$ y $U(1, 20)$, respectivamente. Intuitivamente, se puede ver a b_j como el valor principal de lo que cuesta procesar el trabajo j, mientras que d_{ij} se puede ver como la desviación que se produce entre ese valor principal y su procesado en la máquina i.

- Los basados en una relación entre las máquinas, donde cada tiempo de proceso p_{ij} viene dado por la expresión $p_{ij} = a_i + c_{ij}$ donde a_i y c_{ij} son dos números generados al azar entre las distribuciones uniformes $U(1, 100)$ y $U(1, 20)$, respectivamente. El valor de a_i viene a representar el valor principal de lo que le cuesta realizar un trabajo a la máquina i, mientras que c_{ij} se puede ver como la desviación que se produce entre ese valor principal y su procesado cuando se realiza el trabajo j.

Dentro del número de máquinas y trabajos a considerar en las instancias, nos decantamos por los más usuales a partir de los últimos trabajos que constituían el estado del arte como Ghirardi y Potts (2005), los cuales optaron por usar 10, 20, 30, 40 y 50 máquinas y 100, 200, 500 y 1000 trabajos, es decir se llega hasta instancias de 1000×50, de lo más grande tratado en la literatura. Para cada combinación se crearon 10 instancias. Esto supone un total de 200 instancias por cada intervalo de distribución de tiempos de proceso p_{ij} que se

considere, como se detallará más adelante. Aunque en trabajos anteriores se usaron instancias más pequeñas, con unos números de trabajos y/o máquinas menores, los algoritmos actuales resuelven de forma óptima estas instancias pequeñas, por lo que no se han incluido en el presente estudio.

Existe, además, una cierta disparidad en los intervalos de tiempos de proceso empleados en los distintos artículos. En los trabajos más clásicos como Glass, Potts y Shade (1994) o Piersma y van Dijk (1996) se usaban mayoritariamente los intervalos de una distribución uniforme $U(1, 100)$, mientras que en los artículos más recientes Ghirardi y Potts (2005), Mokotoff y Jimeno (2002) o Martello, Soumis y Toth (1997) se extendía el uso del intervalo $U(10, 100)$. Por el contrario, sí coinciden en la distribución de tiempos para máquinas relacionadas y trabajos relacionados, donde los autores usan mayoritariamente las distribuciones de Glass, Potts y Shade (1994).

Con todo, al hacer las pruebas en el intervalo $U(1, 100)$ vimos que CPLEX 11.0 resolvía de forma óptima, imponiendo un límite de tiempo máximo de 2 horas por cada instancia, 176 de 200 instancias. En cambio, en el intervalo $U(10, 100)$, solo resolvía 52 de las 200. Esto nos llevó a probar nuevos intervalos de p_{ij}, para comprobar la dependencia entre los algoritmos y las distribuciones de p_{ij}. Además, la distribución $U(1, 100)$ es poco realista, ya que supone diferencias de hasta un 10.000 % en el tiempo de ejecución de un trabajo según la máquina; por ejemplo, el trabajo 1 puede hacerlo la máquina 1 en 1 hora, y la máquina 2 en 100 horas (4 días y 4 horas), y para el trabajo 2, lo contrario, la máquina 1 tardar más de 4 días y la máquina 2 solo 1 hora.

Así, se proponen, además de los consabidos $U(1, 100)$ y $U(10, 100)$, el $U(100, 200)$, por ser la misma dispersión de 100 unidades temporales, pero en otro intervalo, que suponen diferencias máximas de un 100 %. El intervalo $U(100, 120)$, con diferencias del 20 % y poca dispersión, para evaluar también este factor. Por último, $U(1000, 1100)$, por ser la misma dispersión de 100 unidades temporales, pero en otro intervalo, que suponen diferencias de un 10 %, además de evaluar valores temporales altos.

La generación de instancias se ha hecho con el algoritmo generador proporcionado por Woclaw (2006), y después se han usado esas mismas instancias para todos los algoritmos. Dichas instancias están disponibles en http://soa.iti.es. Por tanto se van a generar 10 instancias por cada combinación de 10, 20, 30, 40 y 50 máquinas y 100, 200, 500 y 1000 trabajos, es decir 200 instancias, en cada uno de los 7 intervalos temporales a estudiar, $U(1, 100)$, $U(10, 100)$, $U(100, 200)$, $U(100, 120)$, $U(1000, 1100)$, trabajos correlacionados y máquinas correlacionadas, lo que nos da un total de 1400 instancias a considerar.
Se trata de uno de los grupos de instancias más extenso que se ha generado para el estudio del problema de $R//C_{\text{máx}}$, con una gran variedad de intervalos, en los que se incluyen los intervalos más usuales de la literatura y que nos permitirán una buena evaluación del comportamiento de los distintos algoritmos a estudiar.

3.5. Implementación de los métodos propuestos

Todas las pruebas se han hecho en ordenadores tipo PC/AT con procesador Pentium 4 con 3.0 GHz y con 1 GB de memoria RAM empleando el sistema operativo Windows XP 32 bits SP3.

Pasemos a detallar como se han hecho las implementaciones de los distintos algoritmos con los que se han hecho comparaciones: Partial de Mokotoff y Jimeno (2002), RBS de Ghirardi y Potts (2005) y los algoritmos propuestos; así mismo, se ha empleado como elemento a incluir en las comparaciones el solver CPLEX 11.0.
El algoritmo de Partial de Mokotoff y Jimeno (2002), nos ha sido proporcionado amable y desinteresadamente por Andreas Woclaw, el cual lo implementó en C++ para trabajar en Linux, con un parámetro de $r = 0{,}5$ que demostró su mejor comportamiento en las pruebas iniciales. Hemos ejecutado el algoritmo en Linux Ubuntu con un acceso a una licencia de CPLEX 11.0 sobre Linux.

El algoritmo de RBS de Ghirardi y Potts (2005), ha sido cedido por Marco Ghirardi, esta desarrollado en C++ para Windows XP. Se generó un ejecutable a partir del código que nos proporcionó, el cual se ha hecho rodar en Windows XP. Se ha modificado el algoritmo para que lea el conjunto de instancias común a todos los algoritmos, ya que el original creaba instancias nuevas en cada ejecución.
CPLEX 11.0 se probó con la programación lineal que exige el problema, siendo cargado en el mismo grupo de ordenadores con sistema operativo Windows XP.

Los algoritmos propuestos: IGS, IGD, NSP y Vir, se han desarrollado en Delphi 2007 con el flag de optimización activado.
A diferencia de los anteriores, los algoritmos propuestos tienen un componente de azar que añade variabilidad a las soluciones pero que produce pequeñas variaciones en los resultados, por lo que se han hecho 5 réplicas para cada una de las 200 instancias en cada intervalo.

Hay que tener en cuenta que los algoritmos desarrollados en Delphi 2007 son del orden de un 20 % más lentos que los desarrollados en C++. No obstante, no se ha hecho ningún tipo de corrección en este sentido. Con respecto al tiempo de ejecución, cabe destacar que tanto los algoritmos propuestos como CPLEX 11.0, se pueden parar cuando se desee. El tiempo que hemos considerado suficiente para las pruebas iniciales ha sido de 15 segundos, dado que en este tiempo CPLEX 11.0 era capaz de resolver todas las instancias de 100 trabajos para el intervalo U(1,100). No obstante, en capítulos posteriores se evalúan los resultados de los mejores algoritmos para una amplia variedad de tiempos.
Partial se desarrolla en dos partes diferenciadas, la que obtiene una primera solución por relajación, tal y como se detalló en la sección 2.3.2.1, y la que desarrolla el algoritmo tomando nuevas variables de la lista y depurándolas.

Durante la primera parte no se puede interrumpir, así que los 15 segundos se contabilizan exclusivamente durante la segunda parte, lo que le lleva a tiempos que llegan a superar los 60 segundos para las instancias más grandes. Por su parte, RBS es un algoritmo determinista que finaliza cuando realiza toda su tarea, esto nos lleva desde tiempos muy breves para 100 trabajos (menos de 1 segundo) hasta tiempos muy elevados para 1000 trabajos (más de 100 segundos). Curiosamente, RBS da peores resultados para las instancias con menos trabajos (y menos tiempo de CPU) que para las instancias con más trabajos (y más tiempo de CPU). En realidad el crecimiento de tiempos es casi exponencial y pruebas realizadas para más trabajos disparaban los tiempos de resolución. No obstante lo anterior, no se ha hecho ningún tipo de corrección y se ha ejecutado cada algoritmo con las características mencionadas. Así mismo, las comparativas son en el mismo tipo de ordenador, con las mismas instancias y con el tiempo que se ha indicado para cada algoritmo.

3.6. Análisis computacional

En esta sección llevamos a cabo una comparativa entre el estado del arte (Partial, RBS y CPLEX 11.0) y los algoritmos propuestos, analizándolos en cada intervalo objeto de estudio, donde se expondrá un resumen de los resultados y el ANOVA (*ANalysis Of VAriance*) correspondiente, dejando como anexo las tablas completas. Seguidamente se hará un análisis comparativo del comportamiento global de los algoritmos para todos los intervalos.
Recordemos que los análisis se han hecho para los tiempos de proceso p_{ij}, comprendidos dentro de los intervalos $U(1, 100)$, $U(10, 100)$, $U(100, 200)$, $U(100, 120)$, $U(1000, 1100)$, trabajos correlacionados y máquinas correlacionadas, como ya se vio en la sección 3.4. Las comparativas se llevan a cabo para cada intervalo, en ellas mostramos una tabla con las medias, una tabla general con los resultados globales y otras dos tablas, una donde se agrupan por el número de trabajos y otra donde se agrupan por el número de máquinas.

En el anexo B se han incluido dos gráficos por cada intervalo de p_{ij}, que representan, una, la interacción de los distintos algoritmos y el número de trabajos, y la otra, la interacción de los distintos algoritmos y el número de máquinas. Estos gráficos nos ayudan a visualizar lo que los datos numéricos también nos indican.

En todas las tablas de la presente sección, los datos reflejan la desviación porcentual (*Relative Percentage Deviation*, RPD) , dada por la fórmula:

$$RPD(\%) = \frac{C_{\text{máx}}(k) - C^*_{\text{máx}}(k)}{C^*_{\text{máx}}(k)} \cdot 100 \tag{3.1}$$

Donde se hallara la desviación porcentual para cada instancia que hay de cada combinación del número de trabajos n y del número de máquinas m, siendo $C^*_{\text{máx}}(k)$ el valor obtenido por CPLEX 11.0 tras una máximo de 2 horas y $C_{\text{máx}}(k)$ el valor obtenido por el algoritmo en cuestión. En el caso de los algoritmos propuestos, este valor $C_{\text{máx}}(k)$, será la media de las 5 réplicas que se han hecho para cada instancia.
Así mismo, se realizan algunos test de estadísticos para garantizar que las diferencias observadas en las medias son estadísticamente significativas. Se lleva a cabo un Diseño de Experimentos (DOE, Montgomery, 2009) donde se estudia un solo factor, es decir, el tipo de algoritmo a 7 niveles y la desviación porcentual que se da en respuesta a esta variable. Teniendo en cuenta cada una de las instancias evaluadas, nos encontramos con 1.400 resultados para CPLEX, Partial y RBS y 7.000 resultados (cinco réplicas) para cada uno de los algoritmos propuestos. Esto nos da un total de 32.200 resultados. Los resultados de la DOE se analizaron mediante el Analisis de Varianza (ANOVA, *Analysis Of Variance*); esta colección de modelos estadísticos y sus procedimientos asociados, sirve para comparar si los valores de un conjunto de datos numéricos son significativamente distintos a los valores de otro u otros conjuntos de datos, basándose en la varianza global observada en los grupos de datos numéricos a comparar. Este análisis de varianza se utiliza

para realizar un contraste de hipótesis de la conclusión de que la media de un conjunto de datos es distinta de la media de otro conjunto. El ANOVA parte de algunos supuestos que han de cumplirse, la independencia de las observaciones, la distribución normal de los datos evaluados y la homocedasticidad u homogeneidad de las varianzas. En este tipo de análisis no solo se muestra la media, sino también el intervalo de variación en el que, con una probabilidad determinada, se encontrara dicho valor; en nuestro caso se fijó en un 95 %. Esto implica que, con una probabilidad de un 95 %, la media obtenida estará dentro del intervalo que aparece reflejado, y que si este intervalo coincide en parte con otro intervalo, estadísticamente no se puede decir de ellos que sean significativamente distintos.
En este análisis, se podrá apreciar como los intervalos de variación de los algoritmos propuestos son mas estrechos que el resto, debido a que para dichos algoritmos se promedian las 5 réplicas realizadas, dado el componente de azar que tienen, mientras que el resto de algoritmos proporcionan la misma solución cuantas veces repitas el experimento con las mismas instancias.

En la tablas de las medias se ha añadido el número de instancias evaluadas de cada uno de los algoritmos e intervalos (recordemos que para los algoritmos propuestos se han realizado 5 replicas por cada instancia), la media, el error standard y los límites inferior y superior del intervalo de confianza del 95 % de cada media.
En la tabla general también se han añadido los tiempos medios que tardan en resolver las instancias Partial y RBS, al ser los algoritmos que no se detienen a los 15 segundos, como si hacen los algoritmos propuestos y CPLEX 11.0. Recordemos que Partial puede superar los 60 segundos, mientras que RBS alcanza más de 100. Así mismo, en el código proporcionado por el autor del algoritmo RBS, si el tiempo de proceso era menos de 1 segundo, no se contabilizaba, y simplemente se indicaba como “menor de 1 seg”; por tanto, a efectos de cálculo, esto se ha reflejado con un valor de 0,5

segundos. Finalmente, a efectos comparativos, se ha añadido el tiempo que tarda el mejor algoritmo propuesto, Vir, en encontrar la solución final de las distintas instancias, dentro del intervalo de 15 segundos que tiene fijado para la finalización. Es decir, tras finalizar los 15 segundos, se observa la solución final que ha hallado, y se extrae en qué momento de esos 15 segundos ha hallado la solución dada al final. En las distintas tablas se podrá apreciar la velocidad de los algoritmos propuestos frente a Partial y RBS.
En las tablas generales de la presente sección solo se representarán las medias, no obstante, se pueden consultar las tablas completas en el Anexo A, donde se detallan los resultados para cada instancia, el $C_{\text{máx}}$ hallado por CPLEX 11.0 tras una máximo de 2 horas y el tiempo empleado por el mismo para alcanzar el óptimo o, en caso de no hallarlo, el *Gap* o la diferencia con la cota inferior (LB). La cota inferior hallada representa el valor por debajo del cual no será posible hallar una solución, es decir, nos ofrece un límite inferior para la posible solución óptima. La formula con la que calcularemos el Gap, que expresaremos en tanto por cien, vendrá dada por:

$$Gap(\,\%) = \frac{C_{\text{máx}} - LB}{C_{\text{máx}}} \cdot 100 \tag{3.2}$$

Hay que tener en cuenta que, aún después de 2 horas, CPLEX 11.0 no consigue hallar el óptimo para un gran número de instancias. Es más, el número de instancias que resuelve CPLEX 11.0 tras una máximo de 2 horas varía mucho según el intervalo de tiempos de proceso que se estudie. Un resumen del resultado del tanto por cien de las instancias resueltas queda reflejado en la tabla 3.5.

n	m	U(1,100)	U(10,100)	U(100,200)	U(100,120)	U(1000,1100)	jobcorre	maqcorre
100	10	100	100	100	80	30	0	100
100	20	100	50	80	70	40	0	90
100	30	100	50	0	50	0	0	90
100	40	100	90	10	90	0	0	90
100	50	100	90	20	100	100	0	100
200	10	100	30	30	60	0	0	40
200	20	100	10	10	40	0	0	40
200	30	100	0	0	0	0	0	50
200	40	100	10	20	100	20	0	70
200	50	100	10	10	90	0	0	70
500	10	90	40	0	80	0	0	20
500	20	70	20	0	30	0	0	10
500	30	80	0	0	0	0	0	30
500	40	70	0	0	20	0	0	10
500	50	80	0	0	40	0	0	50
1000	10	80	20	0	100	0	60	0
1000	20	50	0	0	30	0	0	0
1000	30	70	0	0	0	0	0	10
1000	40	90	0	0	0	0	0	30
1000	50	80	0	0	0	0	0	50
Instancias resueltas		88	26	14	49	9,5	3	47,5

U(x,y):Distribucion uniforme entre x e y; jobcorre:trabajos correlacionados; maqcorre:máquinas correlacionadas.

Tabla 3.5 – Tabla del porcentaje del número de instancias resueltas por CPLEX 11.0, tras 2 horas de tiempo de CPU como máximo con cada instancia, para todos los intervalos de tiempos de proceso.

En general, de las 1.400 instancias, CPLEX 11.0 es capaz de resolver un total de 474 instancias de forma óptima tras un máximo de 2 horas con cada instancia. En la tabla se puede apreciar también como la capacidad de CPLEX

11.0 para resolver de forma óptima las instancias al cabo de un máximo de 2 horas, va disminuyendo con el aumento del valor máximo y el mínimo que pueden adoptar los valores de p_{ij}, si se mantiene la dispersión de las unidades temporales, en este caso 100. En el caso de disminuir la dispersión de las unidades temporales, vuelve a mejorar el comportamiento de CPLEX, como se da en el caso de U(100,120), posiblemente por la disminución de la variedad de valores a elegir. También podemos observar como solo en el caso U(1,100) es capaz de resolver más del 50 % de las instancias, y que para bastantes casos no llega a resolver ni la cuarta parte de ellas.

En las siguientes secciones pasamos a detallar todos los resultados, desglosándolos para cada uno de los siete intervalos estudiados y finalizando con la presentación de un resumen de todos los intervalos en su conjunto. Se pueden ver más detalles de este estudio en el anexo B.

3.6.1. Intervalo U(1,100)

Ya comentamos que este intervalo era uno de los más usados en la literatura concerniente al problema $R//C_{\text{máx}}$. Bajo nuestro punto de vista, este intervalo, en el caso de máquinas paralelas, resulta poco realista por haber mucha diferencia (un 10.000 %) entre el menor y el mayor posible tiempo de proceso para un mismo trabajo. No obstante, realizamos las pruebas en este intervalo dado su amplio uso en la literatura para otros problemas, y fue el usado en los trabajos más clásicos de este problema, como Glass, Potts y Shade (1994) o Piersma y van Dijk (1996).

Observando las medias en las tablas 3.6 y 3.7, se puede apreciar el buen comportamiento de CPLEX 11.0 frente al resto de algoritmos, siendo el que proporciona peores resultados Partial, que es un algoritmo específico para este problema y que usa como solver el CPLEX 11.0. Puntualmente, se puede constatar el mal comportamiento de RBS para 100 trabajos y 10 máquinas

frente al resto, y el buen comportamiento si son 1000 los trabajos y 50 las máquinas, aunque en este caso el tiempo de resolución alcanza los 70,9 segundos frente a los 3,7 segundos de Vir. Es más, para 1000 trabajos y 10 máquinas es dos veces peor que Vir a pesar de emplear 72,3 segundos en resolver estas instancias frente a los 7,3 que emplea Vir. Así mismo, en este intervalo, los algoritmos propuestos obtienen buenos resultados, sobre todo Vir, aunque no son los mejores en este caso. Igualmente queda claro la rapidez de Vir con una media de resolución de 3,4 segundos frente a Partial con 18,6 o RBS con 19,9, así como un máximo de 7,3 segundos para Vir, frente a máximos de 72,3 segundos de RBS o de 63,3 segundos de Partial.

Algoritmo	Núm.	Media	Error Stnd.	Lim. Inf.	Lim. Sup.
CPLEX	200	1,88	0,16	1,58	2,19
Partial	200	2,88	0,16	2,57	3,19
RBS	200	2,03	0,16	1,72	2,34
IGS	1000	2,46	0,07	2,32	2,60
IGD	1000	2,85	0,07	2,71	2,98
NSP	1000	2,82	0,07	2,69	2,96
Vir	1000	2,30	0,07	2,16	2,44

Núm.= número de instancias evaluadas ($\times 5$ replicas para los algoritmos propuestos), Lim. Inf. y Sup. del intervalo del 95 %.

Tabla 3.6 – Tabla de medias, error relativo porcentual medio y limites inferior y superior del intervalo de confianza del 95 % para el intervalo U(1,100).

Instancias		Algoritmos (% de desviación)							Tiempo en seg		
n	m	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir	Partial	RBS	Vir
100	10	0,00	0,38	2,40	0,14	0,20	0,22	0,15	13,4	0,5	1,4
100	20	0,00	0,63	2,88	0,46	0,99	1,03	1,18	8,3	0,5	1,6
100	30	0,00	2,99	1,21	0,63	1,46	0,85	0,86	1,3	0,5	1,0
100	40	0,00	7,41	2,74	1,48	4,31	2,74	2,74	0,7	0,5	0,7
100	50	0,00	7,94	0,00	1,25	2,86	1,25	1,25	1	0,5	0,2
200	10	0,20	0,34	1,34	0,37	0,39	0,50	0,42	15,3	0,5	3,4
200	20	1,75	1,39	2,28	1,68	1,75	1,82	1,68	14,7	0,5	3,6
200	30	2,12	1,82	3,59	2,38	2,83	2,96	2,53	14,1	0,5	2,6
200	40	3,03	1,73	4,18	3,37	4,91	4,19	3,48	10,4	0,5	2,2
200	50	1,60	7,31	4,87	4,71	5,96	5,19	5,32	7	0,5	2,2
500	10	0,29	0,31	0,97	0,57	0,56	0,57	0,53	15,7	8,5	6,2
500	20	1,26	0,96	1,48	2,28	2,30	2,35	1,84	17,1	9,0	5,7
500	30	3,40	2,31	2,00	4,05	3,78	4,21	2,96	19,6	8,1	6,0
500	40	4,15	3,64	1,82	4,19	3,98	4,86	3,52	22,7	7,6	3,4
500	50	5,63	6,37	1,88	6,07	5,55	7,21	4,80	27	7,5	2,6
1000	10	0,19	0,54	0,73	0,42	0,41	0,41	0,38	16,9	72,3	7,3
1000	20	1,35	0,91	1,09	1,44	1,46	1,46	1,29	23	71,4	5,3
1000	30	2,36	2,03	1,57	3,25	3,09	3,42	2,70	32,6	70,8	4,6
1000	40	4,26	4,12	2,00	4,81	4,63	5,00	3,96	47,7	67,1	5,2
1000	50	6,05	4,50	1,57	5,66	5,51	6,21	4,38	63,3	70,9	3,7
	media	**1,88**	*2,88*	2,03	2,46	2,85	2,82	2,30	18,6	19,9	3,4

IGS:IG-Standard; IGD:IG-Dirigido; NSP:Búsqueda local restringida-No al mismo sitio; Vir:Búsqueda local restringida-Virtual

Tabla 3.7 – Comparación general entre los algoritmos CPLEX, Partial, RBS, IGS, IGD, NSP y Vir para el intervalo de distribución de tiempos de proceso U(1,100).

En la distribución por número de trabajos de la tabla 3.8, se puede apreciar como RBS da los mejores resultados con mayor número de trabajos, pero con tiempos de resolución, para 1000 trabajos, de alrededor de 70 segundos,

los cuales son muy superiores a los 15 segundos fijados para el resto de algoritmos. Por contra, para 200 trabajos, RBS da los peores resultados. Este comportamiento anómalo para 200 trabajos de RBS, así como más detalles, pueden consultarse en el anexo B.

Trabajos	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir
100	**0,00**	*3,87*	1,85	0,79	1,97	1,22	1,24
200	**1,74**	2,52	*3,25*	2,50	3,17	2,93	2,69
500	2,94	2,72	**1,63**	3,43	3,23	*3,84*	2,73
1000	2,84	2,42	**1,39**	3,11	3,02	*3,30*	2,54
Media	**1,88**	*2,88*	2,03	2,46	2,85	2,82	2,30

IGS:IG-Standard; IGD:IG-Dirigido; NSP:Búsqueda local restringida-No al mismo sitio; Vir:Búsqueda local restringida-Virtual

Tabla 3.8 – Comparación entre los algoritmos agrupados por número de trabajos CPLEX, Partial, RBS, IGS, IGD, NSP y Vir para el intervalo de distribución de tiempos de proceso U(1,100).

En la distribución por número de máquinas de la tabla 3.9, es de destacar el mal comportamiento de RBS para pocas máquinas, que en el caso de 10 máquinas llega a ser más de tres veces peor que cualquier otro algoritmo. En 20 máquinas sigue siendo el peor, para dar los mejores resultados para los números superiores de máquinas. Aquí se observa también un valor menor para 50 máquinas en RBS que para 40, siendo que el resto de algoritmos tiene una tendencia natural alcista según se incrementa el número de máquinas. Más detalles en el anexo B.

Máquinas	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir
10	**0,17**	0,39	*1,36*	0,38	0,39	0,43	0,37
20	1,09	**0,97**	*1,93*	1,46	1,62	1,67	1,50
30	**1,97**	2,29	2,09	2,58	2,79	*2,86*	2,26
40	2,86	4,23	**2,68**	3,46	*4,46*	4,20	3,43
50	3,32	*6,53*	**2,08**	4,42	4,97	4,96	3,94
Media	**1,88**	*2,88*	2,03	2,46	2,85	2,82	2,30

IGS:IG-Standard; IGD:IG-Dirigido; NSP:Búsqueda local restringida-No al mismo sitio; Vir:Búsqueda local restringida-Virtual

Tabla 3.9 – Comparación entre los algoritmos agrupados por número de máquinas CPLEX, Partial, RBS, IGS, IGD, NSP y Vir para el intervalo de distribución de tiempos de proceso U(1,100).

En el análisis ANOVA, se puede observar que las diferencias entre las medias observadas en la figura 3.21 no son estadísticamente significativas en los casos de CPLEX 11.0, ni el RBS ni el Vir, a pesar de tener mejor media el primero. Igualmente, IGS no es significativamente distinto de RBS y Vir, aunque sí lo es de CPLEX 11.0. El resto de algoritmos son distintos y con peores resultados que los tres primeros mencionados.
Por tanto, en este intervalo, los algoritmos con mejores resultados corresponden a CPLEX 11.0, RBS y Vir.

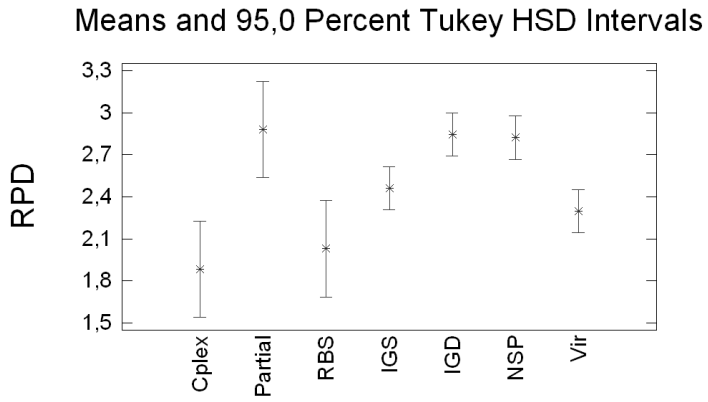

Figura 3.21 – Gráfico ANOVA de los algoritmos CPLEX, Partial,RBS, IGS, IGD, NSP y Vir para el intervalo de distribución de tiempos de proceso U(1,100).

En el presente intervalo U(1,100), CPLEX 11.0 resolvió, de forma óptima, 176 de las 200 instancias al cabo de un máximo de dos horas con cada instancia. Entre las instancias no resueltas, el *Gap* máximo alcanzó el 3,7 % y un valor medio de 1,49 %.

3.6.2. Intervalo U(10,100)

Este intervalo es el más usado en los últimos trabajos como los de Sourd (2001), Mokotoff y Jimeno (2002) o Ghirardi y Potts (2005). En él, las diferencias entre el máximo y el mínimo, son de un 1.000 %, que creemos aún están algo lejos de ser realistas para muchos problemas.

En las tablas 3.10 y 3.11 para este intervalo, Partial nos muestra una mejor media que el resto de algoritmos, aunque aquí, ya es fácil comprobar que los algoritmos propuestos de búsqueda local restringida o de pequeños vecindarios dan unos buenos resultados. Puntualmente, al igual que el caso anterior, RBS

da malos resultados para 100 trabajos y 10 máquinas y buenos para 1000 trabajos y 50 máquinas, aunque con valores de tiempo de resolución muy altos, dando peores valores RBS en el caso de 1000 trabajos y 10 máquinas a pesar de usar estos tiempos tan altos frente a Vir.

Algoritmo	Núm.	Media	Error Stnd.	Lim. Inf.	Lim. Sup.
CPLEX	200	1,64	0,07	1,51	1,77
Partial	200	1,31	0,07	1,18	1,44
RBS	200	1,87	0,07	1,74	2,01
IGS	1000	1,70	0,03	1,65	1,76
IGD	1000	2,27	0,03	2,21	2,33
NSP	1000	1,56	0,03	1,50	1,62
Vir	1000	1,45	0,03	1,39	1,51

Núm.= número de instancias evaluadas ($\times 5$ replicas para los algoritmos propuestos), Lim. Inf. y Sup. del intervalo del 95 %.

Tabla 3.10 – Tabla de medias, error relativo porcentual medio y limites inferior y superior del intervalo de confianza del 95 % para el intervalo U(10,100).

Instancias		Algoritmos (% de desviación)							Tiempo en seg		
n	m	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir	Partial	RBS	Vir
100	10	0,06	0,39	1,45	0,49	0,40	0,57	0,37	12,3	0,5	2,4
100	20	0,79	0,53	2,67	0,98	1,25	1,22	0,85	14,6	0,5	2,5
100	30	2,14	1,29	4,49	1,67	3,08	1,88	1,46	14,1	0,5	3,1
100	40	0,29	1,47	4,98	2,82	5,61	3,52	2,70	11,2	0,5	4,0
100	50	0,33	0,67	6,66	0,55	3,34	1,48	0,62	4,2	0,5	2,8
200	10	0,25	0,27	0,80	0,52	0,44	0,43	0,30	13,9	0,5	4,0
200	20	1,66	0,90	1,52	1,67	1,64	1,15	0,94	15,5	0,5	5,3
200	30	2,96	2,40	2,16	2,44	3,00	1,80	1,82	15,9	0,5	5,5
200	40	5,02	2,05	2,68	2,93	4,12	2,46	2,58	16,5	0,5	5,7
200	50	4,36	1,80	1,80	2,43	4,18	1,91	2,27	16,1	0,5	4,7
500	10	0,27	0,23	0,52	0,45	0,45	0,39	0,38	15,7	8,8	6,8
500	20	0,97	0,82	1,02	1,40	1,31	1,10	1,04	17,2	9,0	6,8
500	30	1,80	1,42	0,99	2,17	2,27	1,82	1,74	19,6	10,2	5,9
500	40	2,53	2,66	1,33	3,01	3,21	2,52	2,65	22,9	10,8	6,4
500	50	3,08	3,25	1,28	3,52	3,88	2,84	3,09	26,9	11,3	5,1
1000	10	0,19	0,73	0,39	0,26	0,24	0,23	0,22	17,3	72,4	5,7
1000	20	0,83	0,66	0,47	0,84	0,83	0,75	0,71	23	73,5	6,4
1000	30	1,38	0,98	0,64	1,44	1,47	1,25	1,24	32,8	75,6	6,0
1000	40	1,75	1,72	0,81	2,00	2,04	1,71	1,77	47,1	78,6	5,6
1000	50	2,05	2,05	0,83	2,48	2,67	2,18	2,23	65,4	90,5	4,9
	media	1,64	**1,31**	1,87	1,70	*2,27*	1,56	1,45	21,1	22,3	5,0

IGS:IG-Standard; IGD:IG-Dirigido; NSP:Búsqueda local restringida-No al mismo sitio; Vir:Búsqueda local restringida-Virtual

Tabla 3.11 – Comparación general entre los algoritmos CPLEX, Partial, RBS, IGS, IGD, NSP y Vir para el intervalo de distribución de tiempos de proceso U(10,100).

En la distribución por número de trabajos de la tabla 3.12, se vuelve apreciar como RBS da los mejores resultados con mayor número de trabajos, pero con tiempos de resolución, para 1000 trabajos, de más de 70 segundos, los

cuales son muy superiores a los 15 segundos fijados para el resto de algoritmos. Por contra, para 100 trabajos, RBS da los peores resultados con diferencia. También podemos observar como CPLEX da los mejores resultados para 100 trabajos pero los peores para 200 trabajos.

Trabajos	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir
100	**0,72**	0,87	*4,05*	1,30	2,74	1,73	1,20
200	*2,85*	**1,48**	1,79	2,00	2,68	1,55	1,58
500	1,73	1,68	**1,03**	2,11	*2,22*	1,73	1,78
1000	1,24	1,23	**0,63**	1,40	*1,45*	1,22	1,23
media	1,64	**1,31**	1,87	1,70	*2,27*	1,56	1,45

IGS:IG-Standard; IGD:IG-Dirigido; NSP:Búsqueda local restringida-No al mismo sitio; Vir:Búsqueda local restringida-Virtual

Tabla 3.12 – Comparación entre los algoritmos agrupados por número de trabajos CPLEX, Partial, RBS, IGS, IGD, NSP y Vir para el intervalo de distribución de tiempos de proceso U(10,100).

En la distribución por número de máquinas de la tabla 3.13, vuelve a destacar el mal comportamiento de RBS para pocas máquinas, que en el caso de 10 máquinas llega a ser dos veces peor que cualquier otro algoritmo. En 20 máquinas sigue siendo el peor. Los mejores resultados para cualquier número de máquinas, excepto 10, los obtuvo Partial.

Máquinas	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir
10	**0,19**	0,40	*0,79*	0,43	0,38	0,41	0,32
20	1,06	**0,73**	*1,42*	1,22	1,26	1,05	0,89
30	2,07	**1,52**	2,07	1,93	*2,46*	1,69	1,57
40	2,40	**1,97**	2,45	2,69	*3,75*	2,55	2,43
50	2,46	**1,94**	2,64	2,24	*3,52*	2,10	2,05
media	1,64	**1,31**	1,87	1,70	*2,27*	1,56	1,45

IGS:IG-Standard; IGD:IG-Dirigido; NSP:Búsqueda local restringida-No al mismo sitio; Vir:Búsqueda local restringida-Virtual

Tabla 3.13 – Comparación entre los algoritmos agrupados por número de máquinas CPLEX, Partial, RBS, IGS, IGD, NSP y Vir para el intervalo de distribución de tiempos de proceso U(10,100).

Con el análisis de ANOVA de la figura 3.22, podemos ver como siendo el mejor algoritmo Partial, no es significativamente distinto de Vir, e incluso llega a estar dentro del límite CPLEX. El algoritmo NSP queda muy cerca de estar dentro del intervalo de Partial. También podemos observar como RBS, IGS y NSP, no son significativamente distintos de CPLEX. Por tanto, en este intervalo, los algoritmos con mejores resultados serían Partial y Vir, teniendo también unos buenos resultados CPLEX y NSP.

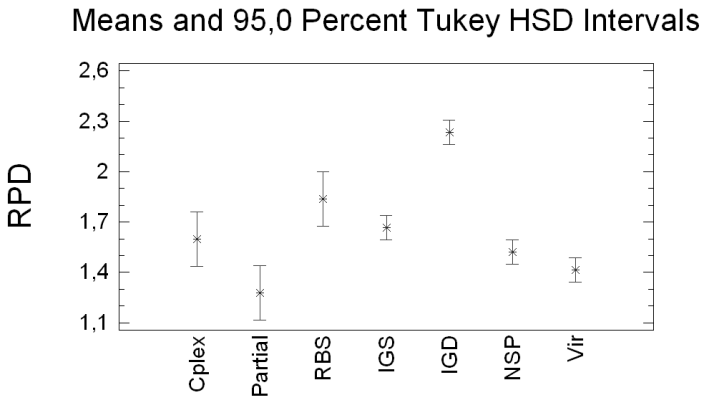

Figura 3.22 – Gráfico ANOVA de los algoritmos CPLEX, Partial, RBS, IGS, IGD, NSP y Vir para el intervalo de distribución de tiempos de proceso U(10,100).

En el presente intervalo $U(10, 100)$, CPLEX 11.0 resolvió, de forma óptima, 52 de las 200 instancias al cabo de un máximo de dos horas con cada instancia. Entre las instancias no resueltas, el *Gap* máximo alcanzó el 7,84 % y un valor medio de 1,33 %.

3.6.3. Intervalo U(100,200)

Aquí se introduce el primer intervalo novedoso. Se buscó un intervalo que tuviera la misma dispersión temporal (100 unidades), pero que representara una diferencia entre el máximo y mínimo del intervalo más pequeña y más acorde con los problemas reales que conocemos. Así pues, un intervalo de 100 a 200 cumple con estos requisitos, ya que nos lleva a diferencias del 100 % entre el mayor y menor valor de los tiempos de proceso para un mismo trabajo.

Las medias en este intervalo, que podemos ver en las tablas 3.14 y 3.15, ya dejan claro que los algoritmos propuestos dan los mejores resultados frente al

resto de métodos. Puntualmente, los algoritmos propuestos, excepto IGS, dan valores negativos para 200×30, lo que indica que obtiene mejores resultados en 15 segundos que CPLEX tras un máximo de 2 horas con cada instancia. El algoritmo NSP es el que mejores resultados da, batiendo a CPLEX, Partial o RBS por más del doble, seguido de Vir en cuanto a buenos resultados; es decir los algoritmos de búsqueda local restringida o de vecindarios pequeños son los de mejores resultados en este intervalo.

Algoritmo	Núm.	Media	Error Stnd.	Lim. Inf.	Lim. Sup.
CPLEX	200	0,76	0,03	0,71	0,81
Partial	200	1,05	0,03	1,00	1,10
RBS	200	0,81	0,03	0,76	0,86
IGS	1000	0,52	0,01	0,49	0,54
IGD	1000	0,49	0,01	0,46	0,51
NSP	1000	0,36	0,01	0,34	0,38
Vir	1000	0,45	0,01	0,43	0,47

Núm.= número de instancias evaluadas ($\times 5$ replicas para los algoritmos propuestos), Lim. Inf. y Sup. del intervalo del 95 %.

Tabla 3.14 – Tabla de medias, error relativo porcentual medio y limites inferior y superior del intervalo de confianza del 95 % para el intervalo U(100,200).

Instancias		Algoritmos (% de desviación)							Tiempo en seg		
n	m	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir	Partial	RBS	Vir
100	10	0,07	0,09	1,22	0,33	0,25	0,30	0,19	15,3	0,5	6,1
100	20	0,38	0,47	1,07	0,39	0,29	0,37	0,32	14,2	0,5	4,9
100	30	1,90	2,81	1,01	0,21	0,14	0,16	0,03	15,4	0,5	3,8
100	40	1,38	2,43	0,99	0,47	0,61	0,42	0,34	15,5	0,5	4,3
100	50	0,28	0,14	1,66	0,14	0,44	0,22	0,35	13,9	0,5	4,3
200	10	0,69	0,43	1,06	0,40	0,30	0,23	0,29	15,4	0,5	9,0
200	20	0,50	0,83	1,42	0,74	0,60	0,47	0,64	15,6	0,5	7,8
200	30	1,15	2,73	0,01	0,11	-0,04	-0,29	-0,12	16,1	0,5	7,4
200	40	1,01	1,06	0,89	0,83	0,72	0,48	0,83	16,5	0,7	7,0
200	50	0,85	0,68	1,12	0,95	0,86	0,55	0,96	16,4	1,0	6,4
500	10	0,29	0,23	0,49	0,26	0,19	0,13	0,18	15,8	8,9	10,2
500	20	1,05	0,66	1,04	0,75	0,64	0,58	0,67	17,3	9,5	9,2
500	30	0,91	1,47	0,57	0,78	0,74	0,56	0,66	19,8	10,1	8,2
500	40	0,48	2,15	0,32	0,54	0,49	0,35	0,43	23,3	11,0	7,2
500	50	0,69	0,98	0,69	0,84	0,84	0,58	0,80	27,4	11,8	8,7
1000	10	0,21	0,31	0,22	0,13	0,12	0,08	0,11	17,4	73,6	9,8
1000	20	0,80	0,67	0,64	0,50	0,46	0,38	0,46	23,5	75,6	11,3
1000	30	0,84	0,93	0,56	0,66	0,69	0,59	0,64	33,7	79,0	10,2
1000	40	0,98	1,18	0,70	0,69	0,70	0,58	0,66	48,6	84,6	10,4
1000	50	0,71	0,69	0,54	0,60	0,65	0,48	0,58	67,5	97,3	9,8
media		0,76	*1,05*	0,81	0,52	0,49	**0,36**	0,45	22,4	23,4	7,8

IGS:IG-Standard; IGD:IG-Dirigido; NSP:Búsqueda local restringida-No al mismo sitio; Vir:Búsqueda local restringida-Virtual

Tabla 3.15 – Comparación general entre los algoritmos CPLEX, Partial, RBS, IGS, IGD, NSP y Vir para el intervalo de distribución de tiempos de proceso U(100,200).

La tabla 3.16 de la distribución por número de trabajos, se pueden apreciar los buenos resultados de NSP en casi todas las agrupaciones de número de trabajos y los malos resultados de Partial. En general se aprecia como tanto

CPLEX como Partial y RBS quedan lejos del resto para todos los trabajos, e incluso RBS que para 1000 trabajos llega a tardar más de 97 segundos, no supera a NSP tras 15 segundos También podemos observar, como RBS y Partial dan peores resultados para 100 trabajos que para cualquier otra cantidad de trabajos.

Trabajos	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir
100	0,80	*1,19*	1,19	0,31	0,35	0,30	**0,25**
200	0,84	*1,14*	0,90	0,61	0,49	**0,29**	0,52
500	0,68	*1,10*	0,62	0,63	0,58	**0,44**	0,55
1000	0,71	*0,75*	0,53	0,52	0,52	**0,42**	0,49
media	0,76	*1,05*	0,81	0,52	0,49	**0,36**	0,45

IGS:IG-Standard; IGD:IG-Dirigido; NSP:Búsqueda local restringida-No al mismo sitio; Vir:Búsqueda local restringida-Virtual

Tabla 3.16 – Comparación entre los algoritmos agrupados por número de trabajos CPLEX, Partial, RBS, IGS, IGD, NSP y Vir para el intervalo de distribución de tiempos de proceso U(100,200).

En la distribución por número de máquinas mostrada en la tabla 3.17, vuelve a destacar el mal comportamiento de RBS para pocas máquinas, que en el caso de 10 máquinas llega a ser entre dos y más de tres veces peor que el resto de algoritmos. Para 20 máquinas sigue siendo el peor. También destacan los malos resultados de Partial para 30 o 40 máquinas respecto al resto. Los mejores resultados para cualquier número de máquinas, los obtuvo NSP.

Máquinas	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir
10	0,32	0,26	*0,75*	0,28	0,22	**0,19**	**0,19**
20	0,68	0,66	*1,04*	0,60	0,50	**0,45**	0,52
30	1,20	*1,98*	0,54	0,44	0,38	**0,26**	0,30
40	0,96	*1,70*	0,72	0,63	0,63	**0,46**	0,57
50	0,63	0,62	*1,00*	0,63	0,70	**0,46**	0,67
media	0,76	*1,05*	0,81	0,52	0,49	**0,36**	0,45

IGS:IG-Standard; IGD:IG-Dirigido; NSP:Búsqueda local restringida-No al mismo sitio; Vir:Búsqueda local restringida-Virtual

Tabla 3.17 – Comparación entre los algoritmos agrupados por número de máquinas CPLEX, Partial, RBS, IGS, IGD, NSP y Vir para el intervalo de distribución de tiempos de proceso U(100,200).

El análisis de ANOVA de la figura 3.23, muestra claramente como Partial, RBS y CPLEX son significativamente peores que todos los algoritmos propuestos. En este caso, el mejor algoritmo es NSP, seguidos de Vir, IGD e IGS, los cuales no se pueden considerar significativamente distintos entre sí.

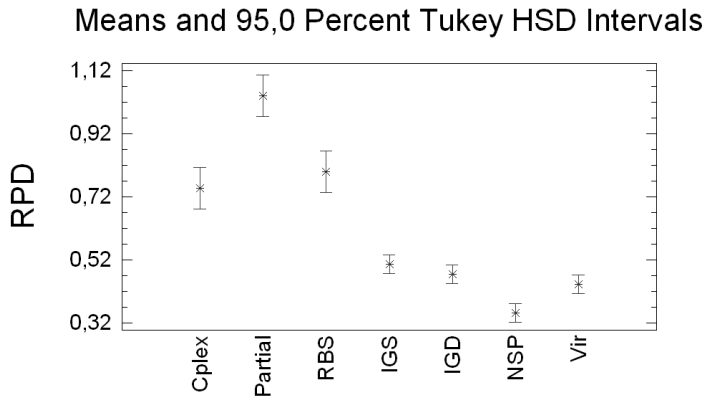

Figura 3.23 – Gráfico ANOVA de los algoritmos CPLEX, Partial, RBS, IGS, IGD, NSP y Vir para el intervalo de distribución de tiempos de proceso U(100,200).

En el presente intervalo $U(100, 200)$, CPLEX 11.0 resolvió, de forma óptima, 28 de las 200 instancias al cabo de un máximo de dos horas con cada instancia. Entre las instancias no resueltas, el *Gap* máximo alcanzó el 7,13 % y un valor medio de 1,02 %.

3.6.4. Intervalo U(100,120)

En este caso el intervalo varía entre unos límites que representan una diferencia de un 20 %, además de una dispersión menor entre los valores máximo y mínimo.

En este intervalo, las tablas 3.18 y 3.19 muestran como los algoritmos propuestos dan los mejores resultados, siendo los algoritmos de búsqueda local restringida o de pequeños vecindarios los que mejores resultados ofrecen. Puntualmente, RBS da valores negativos para 500×40 y 1000×30, lo que indica que obtiene mejores resultados que CPLEX tras un máximo de 2

horas con cada instancia, no obstante, sigue obteniendo malos resultados para 100×10. Partial resulta ser el de peores resultados, con una media de dos o más veces peor que el resto.

Algoritmo	Núm.	Media	Error Stnd.	Lim. Inf.	Lim. Sup.
CPLEX	200	0,15	0,01	0,14	0,17
Partial	200	0,33	0,01	0,31	0,34
RBS	200	0,13	0,01	0,11	0,14
IGS	1000	0,09	0,00	0,09	0,10
IGD	1000	0,10	0,00	0,09	0,11
NSP	1000	0,05	0,00	0,04	0,05
Vir	1000	0,08	0,00	0,07	0,08

Núm.= número de instancias evaluadas ($\times 5$ replicas para los algoritmos propuestos), Lim. Inf. y Sup. del intervalo del 95 %.

Tabla 3.18 – Tabla de medias, error relativo porcentual medio y limites inferior y superior del intervalo de confianza del 95 % para el intervalo U(100,120).

Instancias		Algoritmos (% de desviación)							Tiempo en seg		
n	m	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir	Partial	RBS	Vir
100	10	0,06	0,19	0,24	0,10	0,08	0,06	0,06	15,3	0,5	2,4
100	20	0,20	0,28	0,22	0,11	0,12	0,06	0,10	14,2	0,5	1,1
100	30	0,15	0,35	0,18	0,15	0,18	0,04	0,09	15,4	0,5	1,4
100	40	0,03	0,40	0,10	0,15	0,17	0,00	0,04	15,5	0,5	2,3
100	50	0,05	0,94	0,30	0,12	0,12	0,03	0,14	13,9	0,5	0,6
200	10	0,10	0,13	0,19	0,08	0,07	0,04	0,05	15,4	0,5	4,7
200	20	0,24	0,26	0,22	0,15	0,14	0,08	0,13	15,6	0,5	2,9
200	30	0,44	0,68	0,04	0,10	0,11	0,03	0,07	16,1	1,0	1,7
200	40	0,12	0,18	0,06	0,08	0,13	0,06	0,08	16,5	0,9	2,3
200	50	0,05	0,32	0,07	0,06	0,07	0,02	0,08	16,4	1,0	2,3
500	10	0,15	0,29	0,23	0,07	0,06	0,04	0,06	15,8	8,8	7,8
500	20	0,16	0,23	0,18	0,10	0,10	0,07	0,09	17,3	9,6	3,9
500	30	0,39	0,42	0,11	0,17	0,17	0,13	0,15	19,8	10,0	2,5
500	40	0,07	0,86	-0,02	0,06	0,06	0,01	0,05	23,3	11,1	2,2
500	50	0,11	0,10	0,04	0,09	0,10	0,05	0,08	27,4	11,1	1,9
1000	10	0,16	0,28	0,13	0,05	0,04	0,03	0,04	17,4	73,1	9,2
1000	20	0,25	0,26	0,16	0,08	0,07	0,06	0,08	23,5	75,9	8,9
1000	30	0,12	0,14	-0,05	0,07	0,08	0,05	0,07	33,7	75,9	6,6
1000	40	0,16	0,14	0,10	0,07	0,10	0,06	0,06	48,6	78,4	5,6
1000	50	0,07	0,10	0,03	0,03	0,05	0,02	0,03	67,5	83,7	4,9
	media	0,15	*0,33*	0,13	0,09	0,10	**0,05**	0,08	22,4	22,2	3,8

IGS:IG-Standard; IGD:IG-Dirigido; NSP:Búsqueda local restringida-No al mismo sitio; Vir:Búsqueda local restringida-Virtual

Tabla 3.19 – Comparación general entre los algoritmos CPLEX, Partial, RBS, IGS, IGD, NSP y Vir para el intervalo de distribución de tiempos de proceso U(100,120).

En la distribución por número de trabajos de la tabla 3.20, se pueden observar los buenos resultados de NSP en todas las agrupaciones de número de trabajos y los malos resultados de Partial. Se aprecia como tanto Partial como

RBS quedan lejos del resto para 100 trabajos, e incluso RBS que para 1000 trabajos, llega a tardar más de 83 segundos, tiene resultados 2 veces peor que NSP tras 15 segundos.

Trabajos	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir
100	0,10	*0,43*	0,21	0,13	0,13	**0,04**	0,09
200	0,19	*0,32*	0,12	0,09	0,10	**0,05**	0,08
500	0,18	*0,38*	0,11	0,10	0,10	**0,06**	0,09
1000	0,15	*0,19*	0,08	0,06	0,07	**0,04**	0,05
media	0,15	*0,33*	0,13	0,09	0,10	**0,05**	0,08

IGS:IG-Standard; IGD:IG-Dirigido; NSP:Búsqueda local restringida-No al mismo sitio; Vir:Búsqueda local restringida-Virtual

Tabla 3.20 – Comparación entre los algoritmos agrupados por número de trabajos CPLEX, Partial, RBS, IGS, IGD, NSP y Vir para el intervalo de distribución de tiempos de proceso U(100,120).

La tabla 3.21 en la que se refleja la distribución por número de máquinas, vuelve a destacar el mal comportamiento de Partial para cualquier número de máquinas y de RBS para pocas máquinas. CPLEX también es tres veces peor que NSP en casi todos los casos. Los mejores resultados para cualquier número de máquinas los obtuvo NSP.

Máquinas	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir
10	0,12	*0,22*	0,20	0,08	0,06	**0,04**	0,05
20	0,21	*0,25*	0,19	0,11	0,11	**0,07**	0,10
30	0,28	*0,40*	0,07	0,12	0,13	**0,06**	0,09
40	0,10	*0,40*	0,06	0,09	0,12	**0,03**	0,06
50	0,07	*0,37*	0,11	0,07	0,09	**0,03**	0,08
media	0,15	*0,33*	0,13	0,09	0,10	**0,05**	0,08

IGS:IG-Standard; IGD:IG-Dirigido; NSP:Búsqueda local restringida-No al mismo sitio; Vir:Búsqueda local restringida-Virtual

Tabla 3.21 – Comparación entre los algoritmos agrupados por número de máquinas CPLEX, Partial, RBS, IGS, IGD, NSP y Vir para el intervalo de distribución de tiempos de proceso U(100,120).

En este caso, el ANOVA de la figura 3.24 nos muestra que todos los algoritmos propuestos son significativamente mejores que el resto, destacando NSP, seguido de Vir que también es significativamente distinto del resto. Es decir, en este intervalo, el algoritmo de mejores resultados es NSP, y el segundo es Vir.

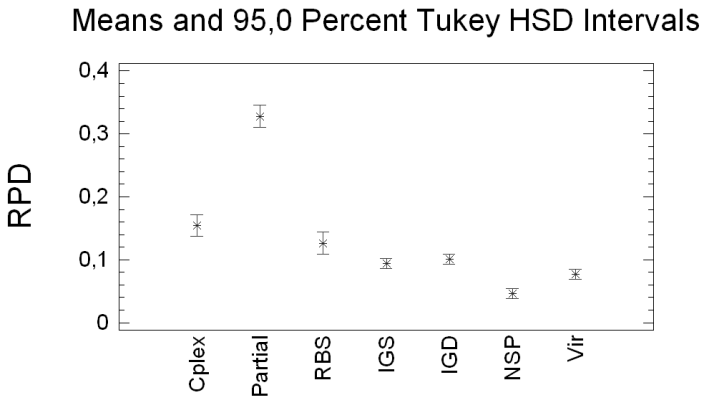

Figura 3.24 – Gráfico ANOVA de los algoritmos CPLEX, Partial, RBS, IGS, IGD, NSP y Vir para el intervalo de distribución de tiempos de proceso U(100,120).

En el presente intervalo $U(100, 120)$, CPLEX 11.0 resolvió, de forma óptima, 98 de las 200 instancias al cabo de un máximo de dos horas con cada instancia. Entre las instancias no resueltas, el *Gap* máximo alcanzó el 5,09 % y un valor medio de 0,68 %.

3.6.5. Intervalo U(1000,1100)

Finalmente, dentro de los intervalos de una distribución uniforme, se tomó un intervalo con una dispersión temporal de 100 unidades, pero que representa una diferencia entre el máximo y el mínimo pequeña, del orden de un 10 %.

En este intervalo, las tablas 3.22 y 3.23 nos muestran cómo los algoritmos propuestos dan los mejores resultados, siendo el algoritmo NSP el que mejores resultados ofrece. Puntualmente, todos los algoritmos, excepto CPLEX y Partial, dan valores negativos para 200×30 y Vir los da también en 100×30, lo que indica que obtienen mejores resultados que CPLEX tras un máximo de 2

horas con cada instancia. También destacan valores como 100×50 para RBS (0,77), 77 veces peor que el resto (0,01); para Partial, los resultados en 500×40 ó 200×30 son peores en un número de veces incluso superior. Partial resulta ser el de peores resultados, que junto con RBS se queda a unas 10 veces peor que NSP.

Algoritmo	Núm.	Media	Error Stnd.	Lim. Inf.	Lim. Sup.
CPLEX	200	0,08	0,01	0,07	0,09
Partial	200	0,23	0,01	0,22	0,24
RBS	200	0,18	0,01	0,17	0,19
IGS	1000	0,04	0,00	0,03	0,04
IGD	1000	0,03	0,00	0,03	0,04
NSP	1000	0,02	0,00	0,02	0,03
Vir	1000	0,03	0,00	0,02	0,03

Núm.= número de instancias evaluadas ($\times$5 replicas para los algoritmos propuestos), Lim. Inf. y Sup. del intervalo del 95 %.

Tabla 3.22 – Tabla de medias, error relativo porcentual medio y limites inferior y superior del intervalo de confianza del 95 % para el intervalo U(1000,1100).

Instancias		Algoritmos (% de desviación)							Tiempo en seg		
n	m	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir	Partial	RBS	Vir
100	10	0,06	0,04	0,14	0,04	0,03	0,03	0,02	15,3	0,5	6,7
100	20	0,04	0,07	0,18	0,03	0,02	0,03	0,02	14	0,5	5,4
100	30	0,17	0,36	0,23	0,02	0,00	0,01	-0,01	15,4	0,5	5,9
100	40	0,13	0,35	0,33	0,04	0,05	0,03	0,02	15,5	0,5	4,7
100	50	0,01	0,01	0,77	0,01	0,02	0,01	0,01	8	0,5	2,6
200	10	0,04	0,06	0,13	0,05	0,04	0,02	0,03	15,3	0,5	8,2
200	20	0,09	0,08	0,17	0,07	0,05	0,04	0,06	15,6	0,5	7,7
200	30	0,09	0,72	-0,03	-0,07	-0,09	-0,11	-0,09	16	0,9	7,4
200	40	0,10	0,09	0,29	0,05	0,05	0,02	0,05	16,6	1,0	6,1
200	50	0,07	0,11	0,52	0,06	0,05	0,03	0,07	17,3	1,0	6,9
500	10	0,06	0,03	0,09	0,03	0,03	0,02	0,02	15,8	8,2	10,0
500	20	0,10	0,04	0,09	0,05	0,04	0,03	0,04	17,3	9,1	8,9
500	30	0,02	0,94	0,03	0,04	0,03	0,01	0,03	19,9	10,0	8,6
500	40	0,03	1,19	0,03	0,04	0,04	0,02	0,03	23,3	11,0	7,5
500	50	0,08	0,09	0,23	0,07	0,07	0,05	0,07	26,1	12,0	6,9
1000	10	0,08	0,08	0,07	0,02	0,02	0,02	0,02	17,4	69,8	11,4
1000	20	0,10	0,07	0,08	0,04	0,04	0,03	0,04	23,5	71,3	9,9
1000	30	0,10	0,07	0,07	0,07	0,07	0,06	0,06	33,7	74,2	9,7
1000	40	0,08	0,08	0,09	0,05	0,05	0,04	0,05	48,7	81,5	10,3
1000	50	0,06	0,06	0,13	0,05	0,06	0,04	0,05	67,8	95,3	7,8
media		0,08	*0,23*	0,18	0,04	0,03	**0,02**	0,03	22,1	22,4	7,6

IGS:IG-Standard; IGD:IG-Dirigido; NSP:Búsqueda local restringida-No al mismo sitio; Vir:Búsqueda local restringida-Virtual

Tabla 3.23 – Comparación general entre los algoritmos CPLEX, Partial, RBS, IGS, IGD, NSP y Vir para el intervalo de distribución de tiempos de proceso U(1000,1100).

En la distribución por número de trabajos de la tabla 3.24, se pueden apreciar los buenos resultados de NSP en casi todas las agrupaciones de número de trabajos, al igual que Vir para 100 o 1000 trabajos, y los malos

resultados de Partial y RBS. Se aprecia como tanto Partial como RBS quedan lejos del resto para 100 y 200 trabajos, y Partial para 500. Incluso RBS, que para 1000 trabajos, llega a tardar más de 95 segundos, no supera a los algoritmos propuestos tras 15 segundos.

Trabajos	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir
100	0,08	0,16	*0,33*	0,03	0,02	0,02	**0,01**
200	0,08	0,21	*0,22*	0,03	0,02	**0,00**	0,02
500	0,06	*0,46*	0,09	0,05	0,04	**0,03**	0,04
1000	0,08	0,07	*0,09*	0,05	0,05	**0,04**	**0,04**
media	0,08	*0,23*	0,18	0,04	0,03	**0,02**	0,03

IGS:IG-Standard; IGD:IG-Dirigido; NSP:Búsqueda local restringida-No al mismo sitio; Vir:Búsqueda local restringida-Virtual

Tabla 3.24 – Comparación entre los algoritmos agrupados por número de trabajos CPLEX, Partial, RBS, IGS, IGD, NSP y Vir para el intervalo de distribución de tiempos de proceso U(1000,1100).

En la distribución por número de máquinas de la tabla 3.25, cabe destacar el mal comportamiento de Partial para 30 y 40 máquinas y de RBS para pocas máquinas. Los mejores resultados para cualquier número de máquinas, los obtuvo NSP, llegando a alcanzar valores negativos para 30 máquinas, lo que significa que da mejores resultados en 15 segundos que CLPEX 11.0 tras 2 horas.

Máquinas	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir
10	0,06	0,05	*0,11*	0,03	0,03	**0,02**	**0,02**
20	0,08	0,06	*0,13*	0,05	0,04	**0,03**	0,04
30	0,10	*0,52*	0,08	0,01	0,00	**-0,01**	0,00
40	0,08	*0,42*	0,19	0,04	0,05	**0,03**	**0,03**
50	0,06	0,07	*0,41*	0,05	0,05	**0,03**	0,05
media	0,08	*0,23*	0,18	0,04	0,03	**0,02**	0,03

IGS:IG-Standard; IGD:IG-Dirigido; NSP:Búsqueda local restringida-No al mismo sitio; Vir:Búsqueda local restringida-Virtual

Tabla 3.25 – Comparación entre los algoritmos agrupados por número de máquinas CPLEX, Partial, RBS, IGS, IGD, NSP y Vir para el intervalo de distribución de tiempos de proceso U(1000,1100).

Queda patente en la figura 3.25 del análisis del ANOVA que los resultados para Partial y RBS son claramente peores que el resto, y que CPLEX, aún mejorando a los anteriores, sigue siendo significativamente peor que todos los algoritmos propuestos. En este caso, los algoritmos propuestos quedan muy próximos unos de otros y prácticamente no hay diferencias significativas entre ellos. Para este intervalo, resultan ser los algoritmos con mejores resultados, NSP, Vir e IGD.

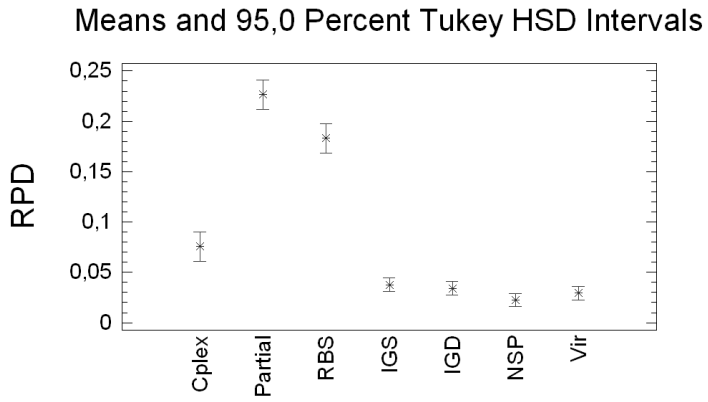

Figura 3.25 – Gráfico ANOVA de los algoritmos CPLEX, Partial, RBS, IGS, IGD, NSP y Vir para el intervalo de distribución de tiempos de proceso U(1000,1100).

En el presente intervalo $U(1000, 1100)$, CPLEX 11.0 resolvió, de forma óptima, 19 de las 200 instancias al cabo de un máximo de dos horas con cada instancia. Entre las instancias no resueltas, el *Gap* máximo alcanzo el 7,14 % y un valor medio de 0,77 %.

3.6.6. Intervalo de trabajos correlacionados

Recordemos que este intervalo es uno de los empleados habitualmente en la literatura, el cual no ha sufrido variación desde que se usara por primera vez. Se basa en una distribución $U(1, 100)$ para determinar el tiempo base de cada trabajo, y una distribución de $U(1, 20)$ para determinar como varía ese trabajo según se procese en cada una de las distintas máquinas.

En este intervalo, las tablas 3.26 y 3.27 muestran como RBS da los mejores resultados, con el algoritmo NSP dando resultados parejos. En este intervalo es donde se puede observar por primera y única vez un comportamiento

equivalente de RBS, con respecto al resto de algoritmos, en cuanto a valores medios similares, sin que se observe ningún pico para las instancias más pequeñas. Puntualmente se pueden destacar, los valores negativos obtenidos por RBS para algunos tipos de instancias, así como NSP en dos de esos mismos casos, lo que indica que obtienen mejores resultados que CPLEX tras un máximo de 2 horas con cada instancia. Partial, junto con CPLEX destacan por sus peores medias respecto a los mejores algoritmos.

Algoritmo	Núm.	Media	Error Stnd.	Lim. Inf.	Lim. Sup.
CPLEX	200	2,20	0,06	2,07	2,32
Partial	200	2,43	0,06	2,31	2,56
RBS	200	0,35	0,06	0,22	0,47
IGS	1000	1,08	0,03	1,03	1,14
IGD	1000	1,63	0,03	1,57	1,68
NSP	1000	0,54	0,03	0,49	0,60
Vir	1000	0,71	0,03	0,65	0,76

Núm.= número de instancias evaluadas ($\times 5$ replicas para los algoritmos propuestos), Lim. Inf. y Sup. del intervalo del 95 %.

Tabla 3.26 – Tabla de medias, error relativo porcentual medio y limites inferior y superior del intervalo de confianza del 95 % para trabajos correlacionados.

Instancias		Algoritmos (% de desviación)							Tiempo en seg		
n	m	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir	Partial	RBS	Vir
100	10	0,50	0,53	0,79	0,75	0,80	0,48	0,49	14,4	0,5	6,1
100	20	1,18	2,19	1,26	1,30	1,57	0,61	1,00	15,3	0,5	5,6
100	30	2,32	4,73	0,98	1,82	2,75	0,50	1,04	15,4	0,5	6,7
100	40	2,42	7,21	1,22	4,03	7,30	1,93	2,03	15,5	0,5	8,0
100	50	3,18	7,83	2,54	5,06	9,41	4,02	3,32	15,7	0,5	9,7
200	10	0,29	0,26	0,35	0,49	0,53	0,36	0,42	15,3	0,5	5,4
200	20	1,64	1,18	0,56	0,63	0,70	0,38	0,50	15,5	0,5	5,5
200	30	2,82	2,50	0,57	1,26	1,47	0,61	0,96	16	0,9	5,6
200	40	3,12	3,49	-0,24	1,34	1,75	0,11	0,79	16,5	1,0	6,8
200	50	4,97	5,09	-0,65	2,28	3,38	0,54	1,42	17,3	1,0	8,5
500	10	0,25	0,16	0,13	0,28	0,28	0,28	0,28	15,7	9,0	4,6
500	20	0,69	0,77	0,13	0,37	0,41	0,33	0,36	17,2	9,5	4,4
500	30	2,21	1,16	0,07	0,44	0,45	0,27	0,38	19,6	9,9	6,3
500	40	3,95	1,75	-0,06	0,35	0,38	0,04	0,25	22,8	10,2	6,4
500	50	4,83	3,34	-0,42	0,41	0,52	-0,10	0,18	26,7	10,8	6,9
1000	10	0,15	0,09	0,09	0,11	0,12	0,13	0,13	17,3	73,6	6,4
1000	20	0,70	0,47	0,05	0,27	0,28	0,25	0,27	23,1	72,4	3,1
1000	30	1,56	1,25	-0,02	0,21	0,21	0,16	0,18	32,7	76,2	4,6
1000	40	2,59	1,77	-0,11	0,12	0,11	0,03	0,08	46,4	74,8	6,3
1000	50	4,57	2,87	-0,28	0,14	0,14	-0,02	0,08	64,4	78,6	4,9
	media	2,20	*2,43*	**0,35**	1,08	1,63	0,54	0,71	22,1	21,6	6,1

IGS:IG-Standard; IGD:IG-Dirigido; NSP:Búsqueda local restringida-No al mismo sitio; Vir:Búsqueda local restringida-Virtual

Tabla 3.27 – Comparación general entre los algoritmos CPLEX, Partial, RBS, IGS, IGD, NSP y Vir para el intervalo de distribución de tiempos de proceso de trabajos correlacionados.

En la distribución por número de trabajos de la tabla 3.28, se pueden apreciar los buenos resultados de RBS en todas las agrupaciones de número

de trabajos, y los malos resultados de Partial para 100 trabajos y CPLEX para el resto. También podemos ver como CPLEX se queda muy por encima de los algoritmos con mejores resultados para los números de trabajos superiores. Así mismo se puede apreciar como RBS, para 500 y 1000 trabajos, da valores negativos, lo que implica que da mejores resultados que CPLEX tras 2 horas como máximo con cada instancia.

Trabajos	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir
100	1,92	*4,50*	**1,36**	2,59	4,37	1,51	1,58
200	*2,57*	2,51	**0,12**	1,20	1,57	0,40	0,82
500	*2,39*	1,44	**-0,03**	0,37	0,41	0,16	0,29
1000	*1,91*	1,29	**-0,05**	0,17	0,17	0,11	0,15
media	2,20	*2,43*	**0,35**	1,08	1,63	0,54	0,71

IGS:IG-Standard; IGD:IG-Dirigido; NSP:Búsqueda local restringida-No al mismo sitio; Vir:Búsqueda local restringida-Virtual

Tabla 3.28 – Comparación entre los algoritmos agrupados por número de trabajos CPLEX, Partial, RBS, IGS, IGD, NSP y Vir para el intervalo de distribución de tiempos de proceso de trabajos correlacionados.

En la distribución por número de máquinas de la tabla 3.32, cabe destacar el mal comportamiento de Partial y CPLEX para 30, 40 y 50 máquinas. Los mejores resultados los han obtenido NSP para 20 y 30 máquinas y RBS para 40 y 50; para 10 máquinas lo obtiene Partial a pesar de ser el peor de media.

Máquinas	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir
10	0,30	**0,26**	0,34	0,41	*0,43*	0,31	0,33
20	1,05	*1,15*	0,50	0,64	0,74	**0,39**	0,53
30	2,23	*2,41*	0,40	0,94	1,22	**0,38**	0,64
40	3,02	*3,56*	**0,20**	1,46	2,39	0,53	0,79
50	4,39	*4,78*	**0,30**	1,97	3,36	1,11	1,25
media	2,20	*2,43*	**0,35**	1,08	1,63	0,54	0,71

IGS:IG-Standard; IGD:IG-Dirigido; NSP:Búsqueda local restringida-No al mismo sitio; Vir:Búsqueda local restringida-Virtual

Tabla 3.29 – Comparación entre los algoritmos agrupados por número de máquinas CPLEX, Partial, RBS, IGS, IGD, NSP y Vir para el intervalo de distribución de tiempos de proceso de trabajos correlacionados.

Con el análisis de ANOVA de la figura 3.26, podemos ver que RBS no es significativamente distinto de NSP, quedando cerca de este último Vir. Por otra parte, quedan alejados del resto CPLEX y Partial, y algo menos IGS e IGD. Por tanto para este intervalo los algoritmos con mejores resultados son RBS y NSP, seguidos de Vir.

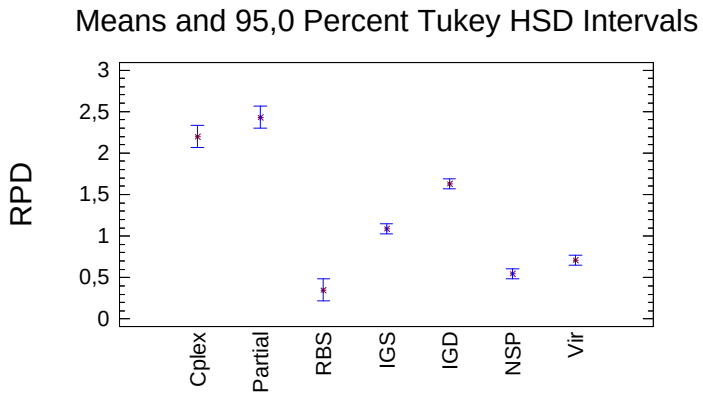

Figura 3.26 – Gráfico ANOVA de los algoritmos CPLEX, Partial, RBS, IGS, IGD, NSP y Vir para el intervalo de distribución de tiempos de proceso de trabajos correlacionados.

En el presente intervalo de trabajos correlacionados, CPLEX 11.0 resolvió, de forma óptima, 6 de las 200 instancias al cabo de un máximo de dos horas con cada instancia. Entre las instancias no resueltas, el *Gap* máximo alcanzó el 8,66 % y un valor medio de 1,49 %.

3.6.7. Intervalo de máquinas correlacionadas

Este último intervalo, se basa en una distribución $U(1, 100)$ para determinar el tiempo base de cada máquina, y una distribución $U(1, 20)$, para determinar la variación de este tiempo base según sea cada uno de los trabajos.

En este intervalo, las tablas 3.30 y 3.31 muestran como los algoritmos propuestos son los que proporcionan los mejores resultados, en especial los de búsqueda local restringida como Vir y NSP. Puntualmente se puede destacar los elevados valores de CPLEX y Partial para las instancias de 1000×50. RBS

destaca por su peor media respecto a los mejores algoritmos, que llega a ser 4 veces superior.

Algoritmo	Núm.	Media	Error Stnd.	Lim. Inf.	Lim. Sup.
CPLEX	200	1,13	0,04	1,05	1,22
Partial	200	0,94	0,04	0,86	1,03
RBS	200	2,36	0,04	2,28	2,45
IGS	1000	0,66	0,02	0,62	0,70
IGD	1000	0,67	0,02	0,64	0,71
NSP	1000	0,59	0,02	0,55	0,63
Vir	1000	0,58	0,02	0,54	0,62

Núm.= número de instancias evaluadas ($\times 5$ replicas para los algoritmos propuestos), Lim. Inf. y Sup. del intervalo del 95 %.

Tabla 3.30 – Tabla de medias, error relativo porcentual medio y limites inferior y superior del intervalo de confianza del 95 % para máquinas correlacionadas.

Instancias		Algoritmos (% de desviación)							Tiempo en seg		
n	m	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir	Partial	RBS	Vir
100	10	0,00	0,10	1,65	0,28	0,27	0,27	0,14	11,3	0,5	3,9
100	20	0,08	0,07	3,44	0,51	0,50	0,37	0,39	5,8	0,5	1,8
100	30	0,00	0,08	3,77	0,44	0,57	0,37	0,30	6,2	0,5	1,9
100	40	0,11	0,00	4,18	0,22	0,31	0,23	0,15	4,1	0,5	1,2
100	50	0,29	0,00	5,37	0,47	0,50	0,53	0,35	4,7	0,5	1,7
200	10	0,45	0,37	1,47	0,56	0,53	0,42	0,47	14,1	0,5	5,7
200	20	0,91	0,53	1,16	0,42	0,41	0,28	0,32	14,6	0,6	3,7
200	30	1,28	1,15	2,27	0,76	0,72	0,53	0,58	16,1	1,0	3,4
200	40	1,58	0,96	4,37	0,67	0,63	0,50	0,52	16,5	1,0	4,0
200	50	1,91	0,69	4,29	0,82	0,77	0,56	0,65	17,4	1,0	3,6
500	10	0,28	0,15	0,73	0,60	0,58	0,55	0,54	15,8	9,5	9,4
500	20	0,86	0,56	1,65	0,72	0,70	0,66	0,67	17,5	11,2	7,0
500	30	1,09	1,17	0,82	0,79	0,80	0,69	0,71	20,2	12,3	5,5
500	40	1,67	1,86	2,53	0,76	0,78	0,68	0,69	23,9	14,0	5,4
500	50	2,72	2,87	2,99	1,00	1,04	0,97	0,94	28,3	14,7	4,0
1000	10	0,27	0,20	0,34	0,64	0,62	0,60	0,59	17,7	73,8	10,1
1000	20	0,48	0,39	0,51	0,87	0,89	0,84	0,85	23,8	84,9	9,5
1000	30	0,96	1,19	1,20	1,02	1,05	1,03	1,02	34,6	92,1	7,1
1000	40	2,52	2,28	1,78	0,75	0,80	0,74	0,74	49,5	100,7	6,8
1000	50	5,19	4,23	2,75	0,93	0,99	0,96	0,95	69,7	112,9	6,8
	media	1,13	0,94	*2,36*	0,66	0,67	0,59	**0,58**	20,6	26,6	5,1

IGS:IG-Standard; IGD:IG-Dirigido; NSP:Búsqueda local restringida-No al mismo sitio; Vir:Búsqueda local restringida-Virtual

Tabla 3.31 – Comparación general entre los algoritmos CPLEX, Partial, RBS, IGS, IGD, NSP y Vir para el intervalo de distribución de tiempos de proceso de máquinas correlacionadas.

En la distribución por número de trabajos de la tabla 3.32, se pueden apreciar los buenos resultados de Vir y NSP en todas las agrupaciones de

número de trabajos, excepto para 100, que lo es Partial. Los peores resultados los obtiene RBS para todos los números de trabajos excepto para 1000, donde lo es CPLEX, aunque este último con 15 segundos exactos de tiempo de ejecución frente a los más de 69 segundos que alcanza Partial o más de 112 segundos de RBS.

Trabajos	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir
100	0,10	**0,05**	*3,68*	0,38	0,43	0,35	0,27
200	1,22	0,74	*2,71*	0,65	0,62	**0,46**	0,51
500	1,32	1,32	*1,75*	0,77	0,78	**0,71**	**0,71**
1000	*1,88*	1,66	1,31	0,84	0,87	0,84	**0,83**
media	1,13	0,94	*2,36*	0,66	0,67	0,59	**0,58**

IGS:IG-Standard; IGD:IG-Dirigido; NSP:Búsqueda local restringida-No al mismo sitio; Vir:Búsqueda local restringida-Virtual

Tabla 3.32 – Comparación entre los algoritmos agrupados por número de trabajos CPLEX, Partial, RBS, IGS, IGD, NSP y Vir para el intervalo de distribución de tiempos de proceso de máquinas correlacionadas.

En la distribución por número de máquinas de la tabla 3.33, cabe destacar el mal comportamiento de RBS para cualquier número de máquinas. Los mejores resultados los han obtenido NSP y Vir para 30 máquinas y Vir para 40 y 50; para 10 y 20 máquinas lo obtiene Partial. Por otra parte, CPLEX se aleja de los mejores resultados más del doble para 40 y 50 máquinas.

Máquinas	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir
10	0,25	**0,21**	*1,05*	0,52	0,50	0,46	0,43
20	0,58	**0,39**	*1,69*	0,63	0,63	0,54	0,56
30	0,83	0,90	*2,01*	0,76	0,79	**0,65**	**0,65**
40	1,47	1,27	*3,21*	0,60	0,63	0,54	**0,52**
50	2,53	1,95	*3,85*	0,81	0,82	0,75	**0,72**
media	1,13	0,94	*2,36*	0,66	0,67	0,59	**0,58**

IGS:IG-Standard; IGD:IG-Dirigido; NSP:Búsqueda local restringida-No al mismo sitio; Vir:Búsqueda local restringida-Virtual

Tabla 3.33 – Comparación entre los algoritmos agrupados por número de máquinas CPLEX, Partial, RBS, IGS, IGD, NSP y Vir para el intervalo de distribución de tiempos de proceso de máquinas correlacionadas.

El análisis de ANOVA nos muestra como los algoritmos propuestos son los mejor clasificados y que no son estadísticamente diferentes entre si, y sí lo son con respecto a los demás. También muestra como CPLEX y Partial no son estadísticamente distintos entre ellos, y que RBS queda realmente mal respecto a todos los demás. Curiosamente, en el anterior intervalo, trabajos correlacionados, era uno de los que mejor quedaba, lo que demuestra una gran inestabilidad, en cuanto a sus resultados, dependiendo de las posibles correlaciones de los tiempos de proceso p_{ij}. Tras este estudio, los algoritmos con mejores resultados son Vir, NSP, IGS e IGD.

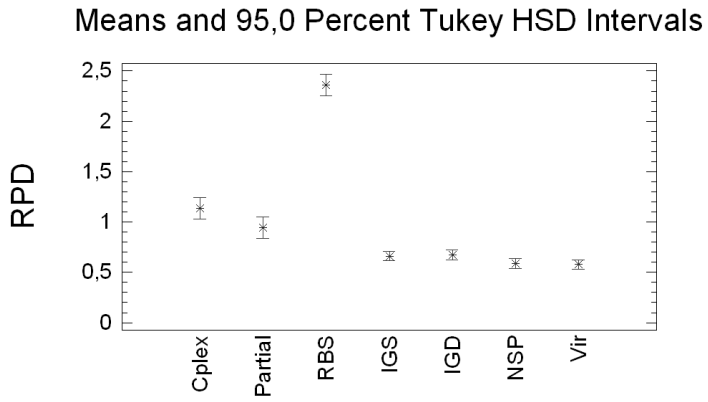

Figura 3.27 – Gráfico ANOVA de los algoritmos CPLEX, Partial, RBS, IGS, IGD, NSP y Vir para el intervalo de distribución de tiempos de proceso de máquinas correlacionadas.

En el presente intervalo máquinas correlacionadas, CPLEX 11.0 resolvió, de forma óptima, 95 de las 200 instancias al cabo de un máximo de dos horas con cada instancia. Entre las instancias no resueltas, el *Gap* máximo alcanzó el 1,64 % y un valor medio de 0,47 %.

3.6.8. Estudio conjunto de todos los intervalos

Pasamos a hacer una comparación de todos los intervalos objeto de este estudio, para poder ver cómo quedan en su conjunto. Empezamos por ver como quedan las medias de todos los promedios de los diferentes intervalos.

Intervalo	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir
U(1,100)	**1,88**	*2,88*	2,03	2,46	2,85	2,82	2,30
U(10,100)	1,64	**1,31**	1,87	1,70	*2,27*	1,56	1,45
U(100,200)	0,76	*1,05*	0,81	0,52	0,49	**0,36**	0,45
U(100,120)	0,15	*0,33*	0,13	0,09	0,10	**0,05**	0,08
U(1000,1100)	0,08	*0,23*	0,18	0,04	0,03	**0,02**	0,03
Job Corre	2,20	*2,43*	**0,35**	1,08	1,63	0,54	0,71
Maq Corre	1,13	0,94	*2,36*	0,66	0,67	0,59	**0,58**
Media	1,12	*1,31*	1,11	0,94	1,15	0,85	**0,80**

IGS:IG-Standard; IGD:IG-Dirigido; NSP:Búsqueda local restringida-No al mismo sitio; Vir:Búsqueda local restringida-Virtual

Tabla 3.34 – Comparación entre los algoritmos CPLEX, Partial, RBS, IGS, IGD, NSP y Vir para todos los intervalos de distribución de tiempos de proceso estudiados.

El la tabla 3.34 se puede apreciar que todos los algoritmos propuestos excepto IGD dan mejores resultados que Partial, CPLEX o RBS, en promedio, para el total de 200×7 instancias.

Tratemos de ver como quedan si calculamos el tanto por cien de diferencia entre el mejor algoritmo de cada intervalo y el resto de los algoritmos en ese intervalo. Esto nos dará una idea de como de lejos quedan los resultados que obtiene un algoritmo comparado con los resultados que se pueden hallar con el mejor algoritmo para ese intervalo.

Intervalo	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir
U(1,100)	**0,00**	53,10	7,84	30,66	51,18	49,97	22,05
U(10,100)	24,51	**0,00**	42,70	29,66	72,87	18,68	10,31
U(100,200)	109,89	189,41	124,11	42,86	34,22	**0,00**	24,98
U(100,120)	231,86	605,43	171,76	102,40	117,84	**0,00**	66,16
U(1000,1100)	238,28	914,20	719,73	68,40	52,52	**0,00**	31,48
Job Corre	530,77	598,32	**0,00**	211,20	367,42	56,36	103,13
Maq Corre	95,87	63,14	308,64	14,48	16,51	1,88	**0,00**
Media	175,88	346,23	196,40	71,38	101,79	18,13	36,87

IGS:IG-Standard; IGD:IG-Dirigido; NSP:Búsqueda local restringida-No al mismo sitio; Vir:Búsqueda local restringida-Virtual

Tabla 3.35 – Comparación entre los algoritmos CPLEX, Partial, RBS, IGS, IGD, NSP y Vir en tanto por cien de desviación respecto al mejor algoritmo.

Aquí podemos ver como Partial da las mayores diferencias, quedando de media a más de un 346 % respecto a los mejores algoritmos de cada intervalo; únicamente da los mejores resultados en el intervalo 10 a 100. En cualquier caso, Vir da un desviación en este intervalo de 10,31, lo cual no es mucho, especialmente si recordamos que el tiempo de CPU de Vir siempre es de 15 segundos, cuando Partial llega a usar hasta 65 segundos para las instancias mayores.

Los algoritmos de búsqueda local restringida o de pequeños vecindarios, son los que menores diferencias dan, y si además observamos su comportamiento para cada uno de los intervalos, podemos ver que no hay intervalos en los que tengan grandes diferencias con los mejores algoritmos, como sí les ocurre, en algún intervalo, al resto de algoritmos.

Finalmente, pasemos a ver el ANOVA de la figura 3.28 de todos los

algoritmos en su conjunto.

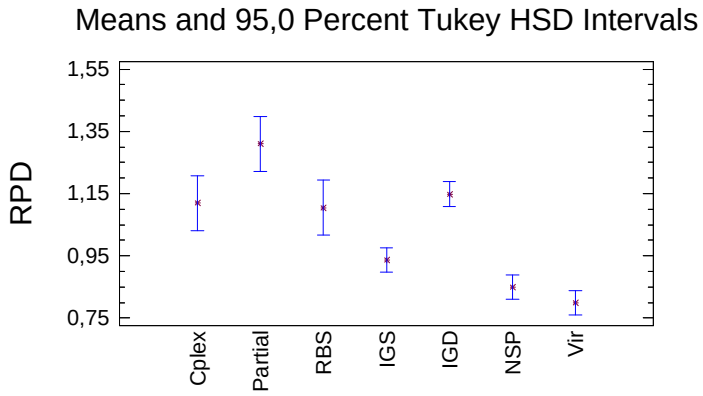

Figura 3.28 – Gráfico ANOVA de los algoritmos CPLEX, Partial, RBS, IGS, IGD, NSP y Vir para todos los intervalos de distribución de tiempos de proceso estudiados.

Se puede comprobar que los algoritmos con mejores resultados en su conjunto son Vir y NSP, los cuales no son significativamente distintos entre sí. Después queda el IGS que siendo peor que los anteriores, es significativamente mejor que el resto. De estos últimos, CPLEX, RBS e IGD no son estadísticamente diferentes; y Partial es el algoritmo que proporciona los peores resultados.

No obstante, y en aras de que no se nos tache de buscar intervalos extraños para justificar nuestros buenos resultados, aunque creemos que ya hemos justificado suficientemente su uso, haremos también la comprobación de todos los algoritmos con solo los intervalos más usados en la literatura, $U(1, 100)$, $U(10, 100)$, trabajos correlacionados y máquinas correlacionadas.

Procedemos, pues, a realizar un estudio parcial, en el que solo incluiremos

los intervalos más estudiados en la literatura anterior.

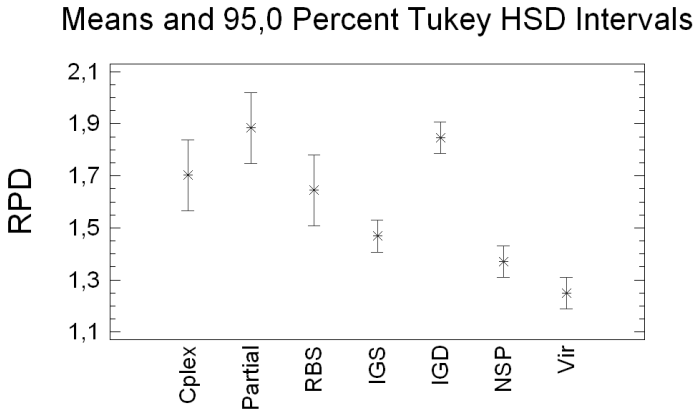

Figura 3.29 – Gráfico ANOVA de los algoritmos CPLEX, Partial, RBS, IGS, IGD, NSP y Vir para los intervalos de distribución de tiempos de proceso más estudiados en la literatura.

En esto caso, podemos ver que también Vir y NSP son los algoritmos con mejores resultados, y están en el limite de no ser estadísticamente distintos entre ellos, aunque sí son significativamente distintos del resto, excepto el IGS, que es comparable al NSP. RBS es comparable con IGS, pero ya no con NSP. El resto de algoritmos no son estadísticamente distintos del RBS, pero dan peores resultados de media. Por tanto, después de todos estos análisis, queda claro que los mejores algoritmos son aquellos que se derivan de las búsquedas locales restringidas o de pequeños vecindarios, es decir Vir y NSP. Por otra parte, Partial queda claramente superado por el resto de algoritmos. CPLEX y RBS son en cierta medida similares; en ocasiones dan muy buenos resultados en algunos intervalos puntuales, y en otros se quedan muy lejos del mejor algoritmo; además ya vimos que el factor tiempo aumenta de forma exponencial con el número de trabajos a resolver por parte de RBS. Así mismo,

en general RBS da los peores resultados cuando el número de máquinas o de trabajos es pequeño, con lo que sería otra cosa a tener en cuenta a la hora de emplear este algoritmo.
De los algoritmos propuestos, IGD se puede considerar un intento fallido de resolución, ya que, aunque da buenos resultados, y son comparables con el estado del arte, es el peor de los algoritmos propuestos, ya que es ampliamente superado por los otros tres.
Finalmente, IGS queda en general algo mejor que el estado del arte pero es superado claramente por Vir y NSP.
Entre Vir y NSP no queda claro cuál de los dos supera al otro; NSP queda peor en el intervalo $U(1, 100)$ (y significativamente distinto del mejor algoritmo), y Vir es significativamente distinto en dos intervalos en los cuales NSP es el algoritmo con mejores resultados, aunque es el segundo mejor algoritmo en ambas ocasiones. Además, no llegan a ser significativamente distintos cuando se evalúan para todo el conjunto de intervalos, ya sean los 7 intervalos objeto de este estudio, o los 4 intervalos más usados en la literatura.

Los algoritmos propuestos han demostrado ser los que proporcionan los mejores resultados, además, son algoritmos muy sencillos, con pocos parámetros, fáciles de programar y los resultados son fáciles de replicar, en contraposición con RBS y Partial que tienen una mayor complicación. Así mismo, por ser tan sencillos, son fáciles de extender a otros problemas, con lo que nos servirán para su uso en nuestro objetivo de no usar todas las máquinas y/o trabajos en el capítulo 6.

3.7. Otras variantes probadas

Hasta ahora hemos relatado como han quedado los algoritmos definitivos, después de numerosas pruebas e intentos de mejora. Algunas de las variantes intentadas más significativas ya se han adelantado, en aras de una mejor com-

prensión de las decisiones tomadas. No obstante, vamos a proceder a comentar las diversas variantes de los algoritmos que se han intentado y rechazado, bien por dar unos peores resultados, o por dar resultados similares pero aumentar el grado de complejidad de los algoritmos propuestos.

En general, solo incluiremos aquellos intentos de variantes que, tras unas pruebas preliminares aceptables, se les hicieron pruebas más en profundidad. Aquellos intentos que muy claramente se esperaba que no funcionasen y, efectivamente, no lo hicieron, no se incluyen.

Para comparar las distintas variantes, reflejaremos en tablas la media de la diferencia, en tanto por cien, del resultado obtenido por las distintas variantes del algoritmo frente al resultado por CPLEX 11.0 tras 2 horas con cada instancia. Para esta comparativa se usó un subgrupo de las 1.400 instancias propuestas, consistente en seleccionar sólo una instancia de las 10 instancias propuestas para cada combinación de n y m y los siete intervalos de tiempos de proceso considerados, es decir, un total de 140 instancias. Así mismo, el nombre de las distintas variantes lo destacaremos entre paréntesis y en negrita (**variantes**) para una mayor claridad de exposición.

3.7.1. Variantes de la solución inicial

Dentro de las posibles soluciones iniciales, ya vimos que habíamos adoptado la de asignar cada trabajo a la máquina más rápida, independientemente de la cantidad de trabajos que ésta máquina tuviera ya asignados (**menor**).

Si se tienen en cuenta todos los trabajos que ya tiene acumulados previamente cada máquina, se obtiene una solución inicial distinta (**acumulado**). Es decir, cada vez que se va a asignar un trabajo, se comprueba en qué máquina conseguirá un tiempo de finalización menor. De hecho, se obtiene una solución inicial diferente si el orden en el que se van considerando los trabajos es

diferente cada vez. Poe ejemplo si el trabajo J_1 se toma en primer lugar, se asignará a la máquina i que lo termine antes, pero si se toma en último lugar, es posible que esa máquina tenga ya un tiempo de finalización que, de asignarle el trabajo J_1, no sea el menor posible y se asigne a otra máquina.
Los resultados de aplicar esta solución inicial resultaron ser algo más pobres que la adoptada; ahora bien, en el caso de instancias preparadas con características especiales (por ejemplo, todos los tiempos de proceso de la primera máquina sean igual a uno), puede resultar interesante usar esta solución inicial, aunque esta situación no es la habitual y puede considerarse aberrante, como ya se comento en la sección 3.1.1.
Otra opción, más sencilla, pero que dio peores resultados, fue el no dar ninguna solución inicial (**ninguna**), ya que la primera búsqueda local que se emplea, la búsqueda local de intercambio, no precisa de una solución inicial.
Veamos la tabla de resultados de las pruebas de estas mejoras:

Menor	Acumulado	Ninguna
3,26	3,31	3,32

Menor: Toma el menor p_{ij} de cada trabajo; Acumulado: Toma el p_{ij} que hace menor el tiempo de finalización; Ninguna: No toma ninguna solución inicial.

Tabla 3.36 – Comparación de otras variantes probadas para la solución inicial.

Aquí se puede apreciar que hay poca diferencia entre cada tipo de solución inicial, siendo ligeramente mejor la aportada por tomar el menor p_{ij} de cada trabajo.

Otros intentos de soluciones iniciales fueron claramente peores y no se comentan, como por ejemplo dar una solución inicial generada totalmente al azar antes de las búsquedas locales.

3.7.2. Variantes de la búsqueda local por inserción

Para la búsqueda local por inserción, en un primer momento, se continuaba buscando el óptimo local mientras mejorara el $C_{\text{máx}}$ (**por mejora** $C_{\text{máx}}$), esto es, se seguía en tanto en cuanto, tras finalizar cada iteración de la búsqueda local de inserción, se hubiera mejorado algo el $C_{\text{máx}}$. En la versión definitiva, se continúa buscando el óptimo local mientras que se produzca algún cambio, es decir, mientras que se haya habido alguna inserción, independientemente de que se mejore el $C_{\text{máx}}$ (**por cambios**).
Otra variante fue realizar la inserción siguiendo la estrategia *first* (**inserfirst**), es decir, realizar la primera inserción que se pueda producir, en lugar de la *best*, la cual realiza la inserción que mejor resultado pueda aportar después de examinar todas las posibles inserciones.

La tabla 3.37 nos muestra como la variante de **por mejora** $C_{\text{máx}}$ resulta algo peor que la definitiva versión **por cambios**, donde se emplea la estrategia *best*, mientras que emplear para este caso la estrategia *first*, produce resultados muy pobres. La opción *first* da unos resultados tan lejanos a la estrategia *best* por que, en el primer caso, no se busca la mejor solución. En el caso de los algoritmos presentados, se necesita una parte del mismo que busque la mejor solución posible, no una variación de la misma para luego pasar a un refinado. Esta parte es la búsqueda local de inserción, por lo que se hace imperativo el que la búsqueda inserción encuentra la mejor solución posible dentro de su espacio local, tal y como demuestran los resultados.

Por cambios	Por mejora $C_{\text{máx}}$	Inserfirst
2,46	2,80	6,62

Por cambio: Insertar mientras se produzca alguna inserción; Por mejora $C_{\text{máx}}$: Insertar mientras se mejore el $C_{\text{máx}}$; Inserfirst: Inserción al primer valor;

Tabla 3.37 – Comparación de otras variantes probadas para la búsqueda local por inserción.

3.7.3. Variantes de la búsqueda local por intercambio

El la búsqueda local por intercambio, ya comentamos en la sección 3.1.3 que la estrategia *first* (**Ifirst**) daba un mejor resultado que la *best* (**Ibest**). Ahora analizaremos estos resultados. También se analizará la opción de realizar el intercambio, no a la mejor posible, si no escogiendo al azar entre la mejor posición posible y la segunda mejor posición posible (**entre 2**). Igualmente se trató de buscar la estrategia *first* pero variando el orden en que se buscaba el intercambio (**variando orden**); es decir, en lugar de intentar intercambiar del primero de los trabajos al último (por ejemplo, intentar intercambiar J_1, luego J_2, y así sucesivamente), se daba un orden al azar de los trabajos, y a partir de esa ordenación se hacia la búsqueda de intercambio según ese orden (por ejemplo, se intenta intercambiar primero J_5, luego J_3, y así sucesivamente en un orden de trabajos al azar.

Por último, se trató de probar la variante de hacer un intercambio entre tres trabajos, tipo de "máquina 1 hace el trabajo de máquina 2, máquina 2 el de máquina 3 y máquina 3 el de máquina 1". Este tipo de vecindario es mucho más extenso. Así, si los tiempos de proceso que se van a realizar son respectivamente $p_{i1,j1}$, $p_{i2,j2}$, y $p_{i3,j3}$, quedarían después de un intercambio a tres como $p_{i2,j1}$, $p_{i3,j2}$, y $p_{i1,j3}$. Este intercambio entre tres se probo tanto para la estrategia *best* (**3 B. best**), como para la estrategia *first* (**3 B.first**). También se probó, con el objeto de ahorrar tiempo de CPU, que este intercambio a tres se realizara solo con los trabajos que estaban asignados a la máquina con el $C_{\text{máx}}$ y cada 100 iteraciones que no se mejorara, es decir, después de realizar el intento de intercambio en 100 ocasiones sin variaciones, se intentaba un cambio a tres para la máquina que ostentara el $C_{\text{máx}}$.

Ifirst	Ibest	Entre 2	Variando orden	3 B. best	3 B. first	3 B. cada 100
3,26	3,61	3,62	3,37	5,30	3,33	3,28

Ifirst: Al primer valor que cumple las condiciones; Ibest: Al mejor valor que cumple las condiciones; Entre 2: Al azar entre los 2 mejores valores que cumplen las condiciones; Variando orden: Orden al azar de los trabajos en la estrategia first; 3 B. best: A 3 bandas en la mejor posición; 3 B. first: A 3 bandas en la primera posición; 3 B. cada 100: A 3 bandas cada 100 iteraciones que no se mejore;

Tabla 3.38 – Comparación de otras variantes probadas para la búsqueda por intercambio: estrategias first, best, a tres bandas y combinaciones de ellas.

En primer lugar, podemos ver como la estrategia *best* (**Ibest**) no mejora los resultados de la *first* (**Ifirst**), a pesar de consumir más tiempo de CPU, y no lo consigue mejorar ni aunque se elija entre los dos mejores (**Entre 2**). La estrategia *first* no mejora si tratamos de variar el orden en que se toman los trabajos para intercambiar (**Variando orden**). La búsqueda de vecindarios mayores (intercambio a tres) en ninguna de sus variantes tampoco consiguió mejorar la sencilla estrategia *first* (**3 B. best**, **3 B. first** ó **3 B. cada 100**).

Por otra parte, se trató, asimismo, de admitir peores soluciones después de estas fases de inserción e intercambio. La tabla 3.39 muestra que esto resultó ser contraproducente, quedando claro que, si bien era mejor admitir soluciones con un $C_{\text{máx}}$ igual a la mejor solución hallada hasta ese momento, admitir soluciones que elevaran, aunque fuera ligeramente, el valor del $C_{\text{máx}}$, producía peores resultados. El ejemplo mostrado en la tabla 3.39 recoge los resultados obtenidos por el mejor algoritmo propuesto, Vir, en su opción de no admitir peores soluciones destruyendo 5 trabajos de la máquina con $C_{\text{máx}}$ y 5 del resto de las máquinas al azar (**5+5**), los comparamos con admitir una solución peor, en un tanto por cien, que la mejor hallada hasta ese momento, para los valores de un 1 % (**1 % peores**), un 2 %, un 3 % y un 4 %. Se puede apreciar como el admitir peores soluciones no produce una mejora en las soluciones obtenidas.

5+5	1 % peores	2 % peores	3 % peores	4 % peores
2,30	2,42	2,43	2,52	2,54

5+5: Destrucción de 5 trabajos del $Cmax$ y 5 trabajos del resto; 1 % peores: Admitimos un 1 % de peores soluciones; e igualmente para 2 %,3 %,4 %.

Tabla 3.39 – Comparación de otras variantes probadas para la búsqueda por intercambio: admisión de peores soluciones.

3.7.4. Variantes de la selección de trabajos para la modificación

Dentro de este subapartado, vamos a estudiar las distintas variantes, tanto de la destrucción de parte de la solución como la variación por búsqueda restringida.

Ya comentamos que la mejor opción se nos presenta cuando modificamos un número fijo de trabajos, en lugar de un tanto por cien de los mismos. Esto viene dado por que el número de variaciones producidas en la modificación, influye mucho en la capacidad, por tiempo, de generar buenas soluciones por parte del algoritmo. Así, un número fijo de trabajos a modificar nos deja un tiempo relativamente bueno de interactuar con la solución modificada y variarla, mientras que un tanto por cien, en ocasiones varía muy poco la solución (caso de 100 trabajos) o, con ese mismo tanto por cien, la hace variar demasiado (caso de 1000 trabajos). En cambio, un número fijo de trabajos es el equivalente a un tanto por cien variable según el número de trabajos; por ejemplo, un número fijo de 20 trabajos, supone un 20 % de trabajos en las instancias de 100 trabajos, y un 2 % en las instancias con 1000 trabajos.

Veamos, como ejemplo, como quedaría una modificación de 5 trabajos del $C_{\text{máx}}$ y 5 trabajos al azar (**5+5**), respecto a modificar un 3 % de trabajos del $C_{\text{máx}}$ y un 3 % de trabajos al azar (**3 %+3 %**).

5+5	3 %+3 %
2,30	2,44

5+5: Destrucción de 5 trabajos del $Cmax$ y 5 trabajos del resto; 3 %+3 %: Destrucción de 3 % trabajos del $Cmax$ y 3 % trabajos del resto.

Tabla 3.40 – Comparación de otras variantes probadas para la selección de trabajos a destruir: número fijo vs tanto por cien.

Se puede apreciar en la tabla 3.40 que la modificación de un número de trabajos fijos, es mejor opción que una modificación similar pero en tantos por cien.

También se intentó un aumento progresivo de la destrucción si cada tras cierto número de iteraciones no mejoraba la mejor solución; esto es, que si después de k iteraciones, no se ha mejorado el $C_{\text{máx}}$, se procedía a aumentar en número de trabajos a modificar. Como se puede apreciar en la tabla 3.41 tanto si se aumentaba el número de trabajos a modificar de la máquina con el $C_{\text{máx}}$ (**M**$C_{\text{máx}}$), como si se aumentaba el número de trabajos modificados al azar (**al azar**), no se consiguió dar mejores resultados. Tampoco lo hizo una combinación de ambas (**combinados**).

Referencia	M$C_{\text{máx}}$	Al azar	Combinados
2,80	3,43	3,30	3,34

Referencia: Valor sin aplicar las variaciones; M$C_{\text{máx}}$: Aumento destrucción de los trabajos de la máquina con $C_{\text{máx}}$; Al azar: Aumento destrucción de los trabajos del resto de las máquinas; Combinados: Combinación del aumento de ambas destrucciones;

Tabla 3.41 – Comparación de otras variantes probadas para la selección de trabajos a destruir: aumento progresivo de la destrucción si no se mejora el $C_{\text{máx}}$.

Se puede apreciar que la diferencia de las distintas pruebas con la refe-

rencia, es decir, con el valor obtenido sin aplicar ninguna modificación, no ha producido mejores resultados.

3.7.5. Variantes de la reconstrucción/búsqueda

En el apartado de reconstrucción (en los IG) o de búsqueda de nuevas posiciones (en los RLS) para los trabajos designados para su modificación, se intento como opción, el asignar el trabajo a la máquina que menor valor de p_{ij} tuviera para ese trabajo (**menor valor**), sin tener en cuenta los trabajos que ya tenia asignados. Esta variante se evalúa frente a la opción que definitivamente se implemento, de dejarlo en la que mejor quedase (**referencia**); esto es, aquella que hacia menor la suma del C_i más su p_{ij} correspondiente. También se probó, para los algoritmos de IG, el buscar la reconstrucción en el sitio que mejor quedase, exceptuando donde ya estaba anteriormente (**IG-NSP**).

Los resultados de la tabla 3.42 muestran como no supone ninguna mejora el método (**menor valor**). Tampoco el no dejar que el IG reconstruya a la misma posición que ya estaba, ya que los IG hacen una destrucción previa de todos los trabajos seleccionados a destruir, lo que impedir la reconstrucción a la misma posición donde estaba previamente, crea un vedado importante y poco justificable a la hora de reconstruir.

Referencia	Menor valor	IG-NSP
2,80	3,53	3,26

Referencia: Valor sin aplicar las variaciones; Menor valor: menor valor de p_{ij} para cada trabajo; IG-NSP: Reconstrucción del IG al mejor sitio exceptuando el mismo sitio donde estaba;

Tabla 3.42 – Comparación de otras mejoras probadas para la reconstrucción y búsqueda.

3.7.6. Variantes del ensamblado del algoritmo

Una de las mejoras clásicas es la introducción de una reinicialización (*Restart*) después de que el algoritmo está un número determinado de iteraciones sin encontrar una mejor solución (**reinicio**). Esto provocaba mejores resultados al principio, pero una vez afinado el algoritmo a su versión definitiva, se pudo comprobar que era innecesario, ya que no variaba sustancialmente la solución el hecho de hacer o no la reinicialización. Nótese, que sin que se observe una mejora clara, preferimos eliminar todos estos aspectos, dado que el resultado es un algoritmo más claro, sencillo y eficiente. También se intentó el suprimir la fase de intercambio para observar si funcionaba mejor quitándole una de las partes que más tiempo consume del algoritmo (**sin inter.**), y por tanto dejando a las demás un número mayor de iteraciones. Incluso se probó el intercambio de la posición en que se realizan las búsquedas locales previas, es decir, realizar primero la búsqueda por intercambio y después la búsqueda por inserción (**orden cambiado**). Otra posible opción fue la de eliminar totalmente la fase de modificaciones (**sin modif.**), y hacer un reinicio después de cada finalización de las fases de búsquedas locales, es decir, se hacía una solución inicial, se realizaban las búsquedas locales de inserción e intercambio, y cuando llegaban al óptimo local, se guardaba la solución y se reiniciaba el proceso con una nueva solución inicial. En resumen, se hizo una VNS con especialización, quitando la parte original de los algoritmos propuestos. También se trató de hacer algunas combinaciones con las variantes más prometedoras de las que se han ido explicando a lo largo del capítulo (**mezcla**), como son el aumento de destrucciones cada ciertas iteraciones sin mejorar durante la fase de modificación de la solución, más seleccionar entre las dos mejores opciones en la fase de intercambio.

La tabla 3.43 nos muestra como el reinicio después de un número de iteraciones sin mejorar, no aportaba unos resultados más prometedores. El quitar la fase de intercambio produce unos resultados muy pobres, demostrando la

importancia de esta fase. Esto es, por otro lado, una de las claves del éxito del VNS; la complementariedad de vecindarios proporciona sinergias y resultados muy superiores a la búsqueda continua en un único vecindario. El intercambio en el orden de la ejecución de las búsquedas locales (**orden cambiado**), dio similares si no peores resultados, dado el tamaño de los distintos vecindarios. Asimismo, el retirar la parte de la modificación de la solución no llevó a mejorar los resultados, como tampoco lo hizo el tratar de mezclar algunas de las opciones que se habían propuesto.

Ref.	Reinicio	Sin Inter.	Orden cambiado	Sin modif.	Mezcla
3,26	3,85	4,27	3,43	3,75	3,47

Referencia: Valor sin aplicar las variaciones; Reinicio: Se reinicia cada 100 iteraciones sin mejorar; Sin intercambio: no se realiza la fase de intercambio ; Orden cambiado: se realiza primero el intercambio y luego la inserción; Sin modificaciones: no se realiza la fase de modificaciones; Mezcla: aumento de destrucción junto al mejor de dos.

Tabla 3.43 – Comparación de otras mejoras probadas para el ensamblado del algoritmo: reinicio, sin intercambios, orden inverso de la inserción e intercambio, sin fase de modificación y combinación variantes más prometedoras.

3.8. Resumen de los algoritmos iniciales propuestos

En este capítulo nos hemos centrado hasta ahora en ver el comportamiento de los mejores algoritmos publicados para el problema de máquinas paralelas no relacionadas, en aras a poder modificar su estructura para su empleo en nuevos problemas. Nos hemos encontrado con complejos algoritmos, que eran difíciles de modificar para adaptarlos a otros propósitos. También apreciamos que los algoritmos existentes eran muy sensibles al intervalo que tuvieran los tiempos de proceso.

En nuestra búsqueda de algoritmos más sencillos, hemos desarrollado cuatro algoritmos, no dependientes de ningún solver. Todos ellos los hemos com-

parado con CPLEX, Partial y RBS, el estado del arte. De esta comparación, quedó Partial, que usa el solver de CPLEX, como claramente un algoritmo ya superado. RBS y CPLEX dan resultados similares entre sí, pero el tiempo de resolución de RBS aumenta enormemente con el número de trabajos, dando, además, malos resultados cuando el número de máquinas o de trabajos es pequeño, por lo que parece ser CPLEX la mejor opción entre los tres algoritmos que había hasta ahora. Hablamos de la versión 11.0. No obstante, CPLEX solo resultó ser mejor que los algoritmos propuestos de búsqueda local restringida para el caso de U(1,100), e incluso en este caso no resultó ser estadísticamente distinto de Vir. De los algoritmos propuestos queda peor situado IGD, a pesar de ser comparable al estado del arte, seguido del IGS que lo mejora. Los mejores resultados con diferencia de todos los algoritmos estudiados hasta ahora son los de búsqueda local restringida, Vir y NSP.

En resumen, CPLEX se ha mostrado prácticamente mejor que los algoritmos que formaban el estado de arte, con lo que hay que tenerlo siempre presente a la hora de analizar, al menos, el problema de asignación estudiado. Por nuestra parte, hemos conseguido unos algoritmos sencillos, rápidos, y que mejoran todo lo publicado hasta la fecha, incluyendo a CPLEX. Además, en el resto de algoritmos influyen de manera considerable el intervalo de los tiempos de proceso y/o el número de máquinas y/o el número de trabajos, mientras que los algoritmos propuestos resultaron ser menos sensibles.
Por otra parte, el IGS ya ha sido empleado en problemas más complejos como el tratado por Vallada y Ruiz (2010), $R/S_{ijk}/C_{max}$ (*Sequence Dependent Setup Times*) con notable éxito, en tanto que CPLEX no es capaz de dar resultados más allá de instancias de 12×5 en estos problemas. Por tanto, se ha logrado el objetivo que teníamos fijado para este capítulo, encontrar unos algoritmos sencillos, rápidos y eficaces para este problema.

CAPÍTULO 4

ALGORITMOS DE SELECCIÓN DIRIGIDA

Hasta ahora hemos hecho un análisis en profundidad del $R//C_{\text{máx}}$, aportando una serie de métodos, que mediante unas búsquedas locales de inserción e intercambio, seguidas de una modificación de la solución por medio de técnicas sencillas pero eficaces, hemos logrado mejorar, de media, el estado del arte. Una vez establecida la naturaleza y características del problema tras este análisis exhaustivo, proseguiremos nuestro estudio tratando de refinar los métodos propuestos hasta ahora, motivado por algunas carencias que pasamos a detallar.

Los métodos propuestos para la modificación de la solución obtenida tras la fase de búsquedas locales, IGS, NSP y Vir, tienen un claro punto mejorable: la aleatoriedad de la elección de máquinas y trabajos, que se trató de subsanar en parte por medio del IGD, sin llegar a obtener un algoritmo lo suficientemente bueno, por lo que a partir de este capítulo descartamos este algoritmo. La selección de los trabajos asignados a las distintas máquinas consiste en una elección al azar de una máquina y una vez seleccionada,

se hace una nueva elección al azar de entre los trabajos asignados a esa máquina. La ventaja de esta aproximación viene dada por su simplicidad y una cierta independencia del conocimiento del tipo de problema, pero vamos a tratar de encontrar mejores soluciones con una selección más dirigida tanto de máquinas como de trabajos. Como ya se vio en las secciones 3.2 y 3.3, en las calibraciones iniciales de los algoritmos previos, parte de los trabajos a modificar debían pertenecer a la máquina que ostentara el $C_{\text{máx}}$, detalle que también debemos aprovechar para aumentar el rendimiento de los algoritmos. En los siguientes apartados vamos a mostrar diversas técnicas que proponemos para quitarle aleatoriedad a la selección de máquinas y trabajos con el ánimo de mejorar los resultados obtenidos hasta ahora.

4.1. Selección de máquinas

Ya se mostró que el seleccionar trabajos a modificar dentro de la máquina que genera el $C_{\text{máx}}$ daba buenos resultados, ahora bien, puede darse el caso, no infrecuente, de que varias máquinas a la vez tengan un valor de $C_i = C_{\text{máx}}$, además, habrá que comprobar el efecto, no solo de las máquinas con $C_{\text{máx}}$, si no del resto de las máquinas y sus distintos valores de C_i. Es decir debemos averiguar qué efecto tienen los distintos valores de C_i en la selección de máquinas. Para ello estudiaremos una serie de distribuciones de probabilidad en función de los distintos valores C_i de las máquinas, yendo desde el mayor valor de $C_i = C_{\text{máx}}$ hasta el menor valor de $C_i = C_{\text{mín}}$.

Las pruebas iniciales se centraron en seleccionar un porcentaje de trabajos de la máquina con $C_{\text{máx}}$ y un porcentaje variable entre las siguientes máquinas con mayor C_i, lo que nos llevó a buscar una probabilidad de selección de máquina que descendiera con el valor de C_i. Es decir, que fuera más probable seleccionar la máquina con mayor C_i que la máquina con menor C_i. Realizados unos primeros ensayos, se optó por una distribución basada en los números

triangulares tal y como se representa en el figura 4.1.

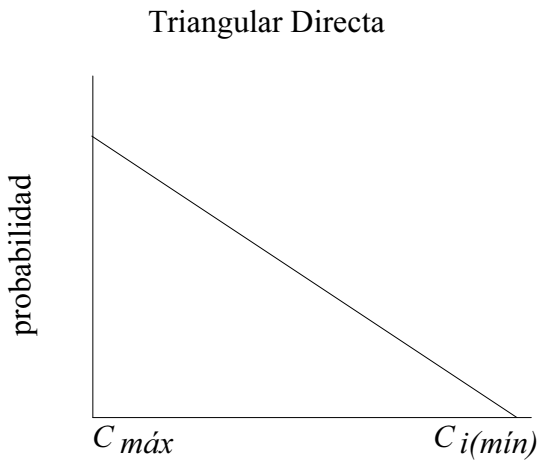

Figura 4.1 – Distribución de probabilidad basada en la asignación de números triangulares de forma directa.

El empleo de una distribución según los números triangulares se basa en estos números, donde cada uno de ellos es el resultado de la suma del anterior número más el número de iteración de que se trate. Así pues, se empieza por el 1 y le sigue, en la iteración 2, la suma del anterior número (el 1) más la iteración de que se trata (la 2º iteración) dando como resultado de la suma 3. La serie vendría dada por 1, 3, 6, 10, 15,... Estos números en realidad son la suma de los h primeros números enteros, donde h representa la posición que queremos hallar o expresado matemáticamente:

$\sum_{i=1}^{h} h = \frac{(h)\cdot(h+1)}{2}$

Estos números se asignan a las distintas máquinas en función de la secuencia ordenada de máquinas según su C_i. Para elegir la máquina en la que seleccionaremos el trabajo a modificar, generamos un número al azar entre 1 y el máximo valor obtenido en la serie. La máquina que tenga el valor superior más cercano al obtenido es la seleccionada. Por ejemplo, supongamos que

tenemos 5 máquinas, luego sus indices serían 1, 3, 6, 10 y 15; al generar un valor al azar entre 1 y 15 obtenemos un 7. En este caso, el valor superior más cercano al 7 es el 10 y la máquina en la posición cuarta de la lista sería la elegida. Como se puede apreciar, la quinta máquina será elegida para los valores generados al azar iguales a 11, 12, 13, 14 y 15, mientras que la primera máquina solo sería elegida si el número obtenido al azar entre 1 y 15 fuera el 1; es decir, la máquina quinta tiene 5 veces más probabilidades de ser elegida que la primera máquina.

Para el caso de la distribución basada en los números triangulares ordenados de forma que, a la máquina con mayor C_i, le corresponda el mayor valor de los números triangulares empleados según el número de máquinas del problema, la denominamos triangular directa, tal y como muestra la figura 4.1. En el caso opuesto, que la máquina con menor C_i reciba el valor más alto de los números triangulares, la denominaremos triangular inversa, representada en la figura 4.2.

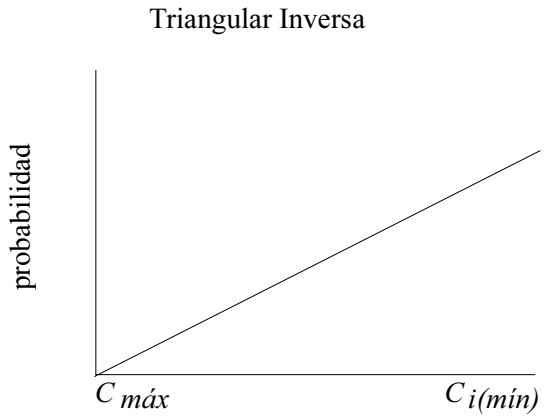

Figura 4.2 – Distribución de probabilidad basada en la asignación de números triangulares de forma inversa.

Con la distribución triangular directa buscamos el dar aun más importancia a valor de las máquinas con C_i elevado, además de la máquina que tuviera

el $C_{\text{máx}}$. Los resultados de las pruebas preliminares mostraron una ligera mejora respecto a los anteriores métodos. Sin embargo, al hacer pruebas con la distribución triangular inversa, los resultados fueron peores a los obtenidos anteriormente, dejando patente la importancia de seleccionar las máquinas con un sesgo hacia las máquinas de mayor C_i. No obstante lo anterior, se probó a emplear una nueva distribución en la selección de máquinas, consistente en, una vez ordenadas las máquinas de mayor a menor valor de C_i, elegir entre la primera mitad de las máquinas según una distribución triangular directa y entre la segunda mitad, la de menores C_i, según una distribución triangular inversa, tal y como se muestra en la figura 4.3. A esta distribución la denominamos piramidal inversa.

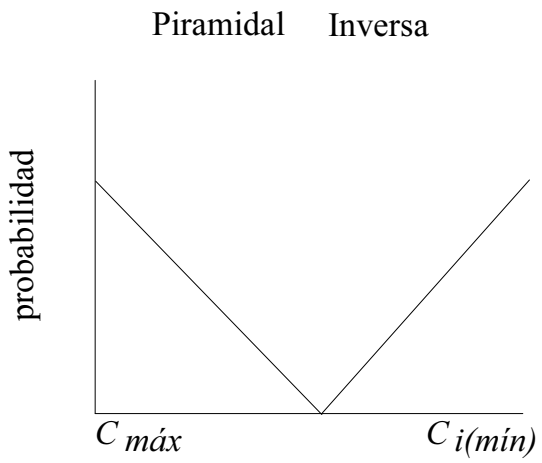

Figura 4.3 – Distribución de probabilidad basada por igual en triangulares directa e inversa.

Este caso mejoraba los resultados de las anteriores distribuciones. Es decir, que una vez al algoritmo ha salido de de las fases de búsquedas locales de inserción e intercambio, con la mayoría de los trabajos bastante bien ajustados para llegar a soluciones de gran calidad, se necesita buscar “sitio” para los trabajos asignados a las máquinas con mayor C_i, y estos probables “sitios”

deben estar en las máquinas con menor C_i. Así, seleccionando máquinas con C_i elevado y máquinas con C_i bajo, tratamos de buscar una mejor asignación a los trabajos mediante una homogeneización de tiempos entre las máquinas. No obstante, la distribución triangular directa e inversa ya nos habían mostrado el sesgo que había hacia la selección de máquinas con mayor C_i. Por tanto, como refinamiento del la distribución piramidal inversa llegamos a la distribución piramidal inversa asimétrica como se muestra en la figura 4.4 mediante la variación del punto de inflexion de la distribución piramidal.

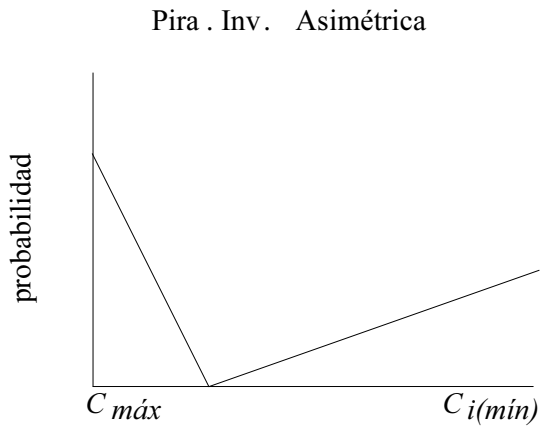

Figura 4.4 – Distribución de probabilidad asimétrica basada en distribuciones triangulares directa e inversa.

Esto nos llevó a valores aún mejores que la distribución piramidal inversa. Ahora bien, hasta ahora se ha soslayado la cuestión de que varias máquinas podrían tener el valor de $C_{\text{máx}}$ simultáneamente, en este caso tendría el mismo valor la selección de cualquiera de estas máquinas, es más, al avanzar en el cálculo del algoritmo, el caso más corriente es que se vayan homogeneizando los valores de C_i y nos encontremos con varias máquinas que ostentan el $C_{\text{máx}}$. Así pues, finalmente se comprobó que los mejores resultados se obtenían de una selección uniforme entre todas las máquinas que tenían el valor del $C_{\text{máx}}$ y una triangular inversa entre el resto de las máquinas, como se muestra en la

figura 4.5.

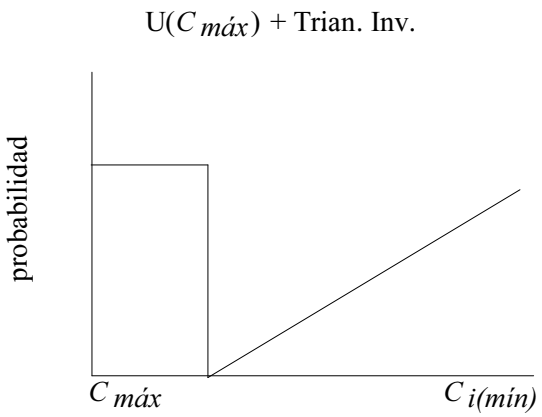

Figura 4.5 – Distribución de probabilidad Uniforme para las máquinas con $C_{\text{máx}}$ y Triangular Inversa entre el resto.

Esta última distribución, que viene a incorporar todas las observaciones relevantes, resultó ser la que mejores resultados ofreció. Se realizaron muy diversas pruebas e intentos de mejorar la distribución de probabilidades como la de hacer variar la pendiente de la distribución triangular inversa o de variar la distribución por otra de mayor incremento de la probabilidad como la cuadrática que puede verse en la figura 4.6, pero no se obtuvieron mejores resultados que los ya obtenidos o incluso empeoraron en algunos casos.

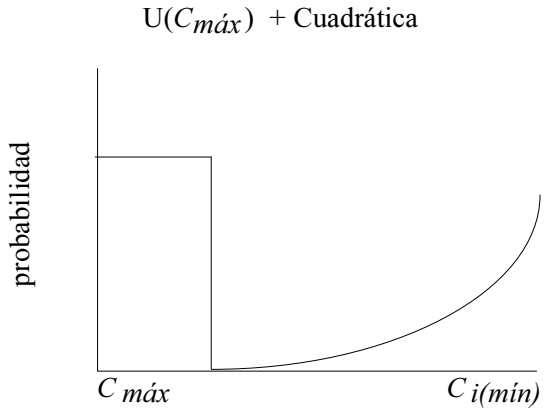

Figura 4.6 – Distribución de probabilidad Uniforme para las máquinas con $C_{\text{máx}}$ y Cuadrática entre el resto.

4.2. Selección de trabajos

Una vez se ha seleccionado una máquina, dentro de ella hay que seleccionar un trabajo entre los que tiene asignados para modificar su asignación. La selección puramente al azar, a pesar de los buenos resultados obtenidos, no parece una aproximación muy inteligente, así que se procedió a buscar una forma más dirigida de elección del trabajo a modificar su asignación entre los asignados a la máquina previamente seleccionada. Tras diversos intentos de formas de selección de trabajo, la forma más prometedora vino dada tras buscar qué trabajos de entre los asignados tienen tiempos de proceso más cortos en otras máquinas. Es decir, se comprueba para cada uno de los trabajos asignados a una máquina preseleccionada, cuál de esos trabajos podría ser realizado más rápido en otra máquina. Más formalmente, tenemos la máquina i que ha sido previamente seleccionada y J_i representa al grupo de trabajos asignados a la máquina i, siendo k en número total de trabajos asignados a la misma. Por tanto, $J_i = \{j_1, j_2, \ldots, j_k\}$. Entonces, entre los k trabajos, seleccionamos el trabajo p que tiene la mayor diferencia de tiempo de proceso con las otras

máquinas. Es decir, seleccionamos el trabajos p de acuerdo a la siguiente expresión:

$$p = \underset{j_k \in J_i}{\operatorname{argmax}} \left\{ \underset{\substack{l \in M \\ i \neq l}}{\text{máx}} \left(p_{j_k,i} - p_{j_k,l} \right) \right\} \tag{4.1}$$

En realidad lo que se está haciendo es simplemente buscar entre los trabajos asignados a la máquina en cuestión, cuál es el que tendría más posibilidades de, al ser quitado, encontrar una mejor colocación, ya que es el que puede ser ubicado, a priori, en una máquina que lo procesará más rápido. Esta forma de selección de trabajo es, evidentemente, determinista.

4.3. Selección del número de elementos a modificar

Queda pendiente la cuestión del número de trabajos que debemos retirar de las máquinas con $C_{\text{máx}}$ y cuántos del resto de las máquinas. Hay que tener en cuenta que se pueden producir diversas situaciones. Así, nos podemos encontrar desde que una sola máquina está generando el $C_{\text{máx}}$ hasta que la totalidad de las máquinas tienen el mismo C_i, por lo tanto, todas las máquinas tendrían el $C_{\text{máx}}$. A resultas de estas posibilidades, optamos por calcular una media de los trabajos que podrían ser quitados de cada máquina. Es decir, si tenemos 10 máquinas y queremos retirar $d = 20$ trabajos en total, aproximadamente habrá que retirar una media de 2 trabajos por máquina. No obstante, dado que hemos mostrado que conviene tener cierto sesgo hacia las máquinas con $C_{\text{máx}}$ habrá que, por una parte, separar el número de trabajos a retirar de las máquinas con $C_{\text{máx}}$, aumentado respecto a la media, y por otra parte el número de trabajos a retirar del resto de máquinas. Así, si marcamos como un incremento respecto a la media de las máquinas con $C_{\text{máx}}$ del 20 % significará que de los trabajos a retirar del resto de las máquinas, un 20 % de ellos serán retirados de las máquinas con $C_{\text{máx}}$, además de los que ya les correspondían por media.

Pongamos un ejemplo. Supongamos que tenemos 10 máquinas y queremos retirar $d = 20$ trabajos en total, esto supondría una media de 2 trabajos por máquina. Si, por ejemplo, hay 4 máquinas con $C_{\text{máx}}$, habrá 6 máquinas con C_i distinto al $C_{\text{máx}}$. Así tendríamos que habría que seleccionar $4 \times 2 = 8$ trabajos entre las máquinas con $C_{\text{máx}}$ y $6 \times 2 = 12$ entre el resto de las máquinas. Si marcamos un sesgo hacia las máquinas con $C_{\text{máx}}$ de un 20 % tendríamos que el 20 % de 12 es 2,4 que sera redondeado siempre hacia el entero superior, con lo que quedaría 3. Finalmente, el número de trabajos a seleccionar quedaría así: entre las máquinas con $C_{\text{máx}}$, 8 que les correspondía de media, más 3 de incremento por sesgo, igual a 11, y del resto de máquinas, que inicialmente tenían 12 que les correspondían de media, menos 3 por sesgo hacia las máquinas con $C_{\text{máx}}$, daría un total de 9. Resumiendo, el ejemplo queda así:

- Número de máquinas de la instancia m = 10.
- Número de trabajos a seleccionar para modificar su asignación d = 20.
- Media teórica de trabajos a seleccionar de cada máquina, media = 20/10 = 2 trabajos por máquina.
- Sesgo hacia las máquinas con $C_{\text{máx}}$, aumentar en un 20 %.
- Máquinas que ostentan el $C_{\text{máx}}$= 4; resto de máquinas con C_i distinto al $C_{\text{máx}}$= 6.
- Número de trabajos a seleccionar de máquinas con $C_{\text{máx}}$ según media = 4×20/10 = 8.
- Número de trabajos a seleccionar de máquinas con C_i distinto al $C_{\text{máx}}$ según media = 6×20/10 = 12.
- Sesgo del 20 % hacia máquinas con $C_{\text{máx}}$ = 12×20 % = 2,4; con redondeo = 3.
- Número final de trabajos para máquinas con $C_{\text{máx}}$ según media más sesgo = 8 + 3 = 11.

- Número final de trabajos para máquinas con C_i distinto del $C_{\text{máx}}$ según media menos sesgo = 12 - 3 = 9.

4.4. Métodos mejorados

Las mejoras en la selección de máquinas y de trabajos para una selección más efectiva de los trabajos a modificar su asignación que hemos explicado hasta ahora, se incluyeron en los métodos propuestos del capítulo 3. Recordemos que seguimos teniendo el mismo bucle de VND y las mismas técnicas de selección de trabajos IGS, NSP y Vir, a las que les incluiremos ahora los nuevos métodos de selección de trabajos expuestos, que mejoran a los anteriores. Las denominaciones de las nuevas variantes de los algoritmos se harán añadiéndole un signo más (+) a las antiguas denominaciones. Así, el IGS pasará a ser IG+, dado que será el único IG que evaluaremos a partir de ahora, el NSP será NSP+ y Vir pasará a Vir+. Como consecuencia de la variación introducida en los algoritmos, tendremos que recalibrar el número total de trabajos a modificar su asignación (d) y el sesgo a favor de las máquinas con $C_{\text{máx}}$. Tras unas breves calibraciones quedaron determinados estos valores como sigue:

- IG+: $d = 15$, incremento de trabajos sobre la media para máquinas con $C_{\text{máx}}$=20 %
- NSP+: $d = 5$, incremento de trabajos sobre la media para máquinas con $C_{\text{máx}}$=20 %
- Vir+: $d = 10$, incremento de trabajos sobre la media para máquinas con $C_{\text{máx}}$=40 %

Cabe significar que, al igual que en el caso de los algoritmos originales (IG, NSP y Vir), en los nuevos algoritmos (IG+, NSP+ y Vir+), el número total de trabajos a modificar es independiente del tamaño de la instancia considerada.

4.5. Algoritmo NVST-IG+

Finalmente, presentamos un algoritmo que reúne todas las características de los algoritmos desarrollados hasta ahora, integrándolo en un solo algoritmo, un algoritmo híbrido. Tenemos que tener en cuenta que NSP y Vir son casos especiales de búsquedas locales, así pues se podrían integrar dentro de ciclo del VND junto con las búsquedas locales de inserción e intercambio. La forma de integrar estas búsquedas locales restringidas (NSP y Vir) dentro del ciclo VND, siguiendo a Hansen y Mladenovic (2001), se lleva a cabo teniendo en cuenta que los vecindarios se examinan del de menor al de mayor tamaño. No obstante, se probaron todas las combinaciones posibles de ordenar estas cuatro búsquedas locales para comprobar la validez del argumento de Hansen y Mladenovic (2001), demostrándose, una vez más, la validez de sus afirmaciones. Así pues, la mejor combinación de búsquedas locales es siguiendo el orden ascendente por tamaño de vecindario, es decir, primero emplearemos el NSP, seguido de Vir y seguimos con la inserción y el intercambio; finalizaremos este nuevo algoritmo híbrido propuesto, empleando como modificador de la solución el IG+. Es decir, el procedimiento queda así:

1. Ciclo VND hasta óptimo local
   - *a*) Una única iteración de NSP y Vir para todos los trabajos, sin empeorar el valor del $C_{\text{máx}}$.
   - *b*) Búsquedas locales de inserción e intercambio, ambas hasta óptimo local.
2. Procedimiento IG+ como modificador de la solución.

Gráficamente podemos ver que el algoritmo NVST-IG+ vendría representado por la figura 4.7.

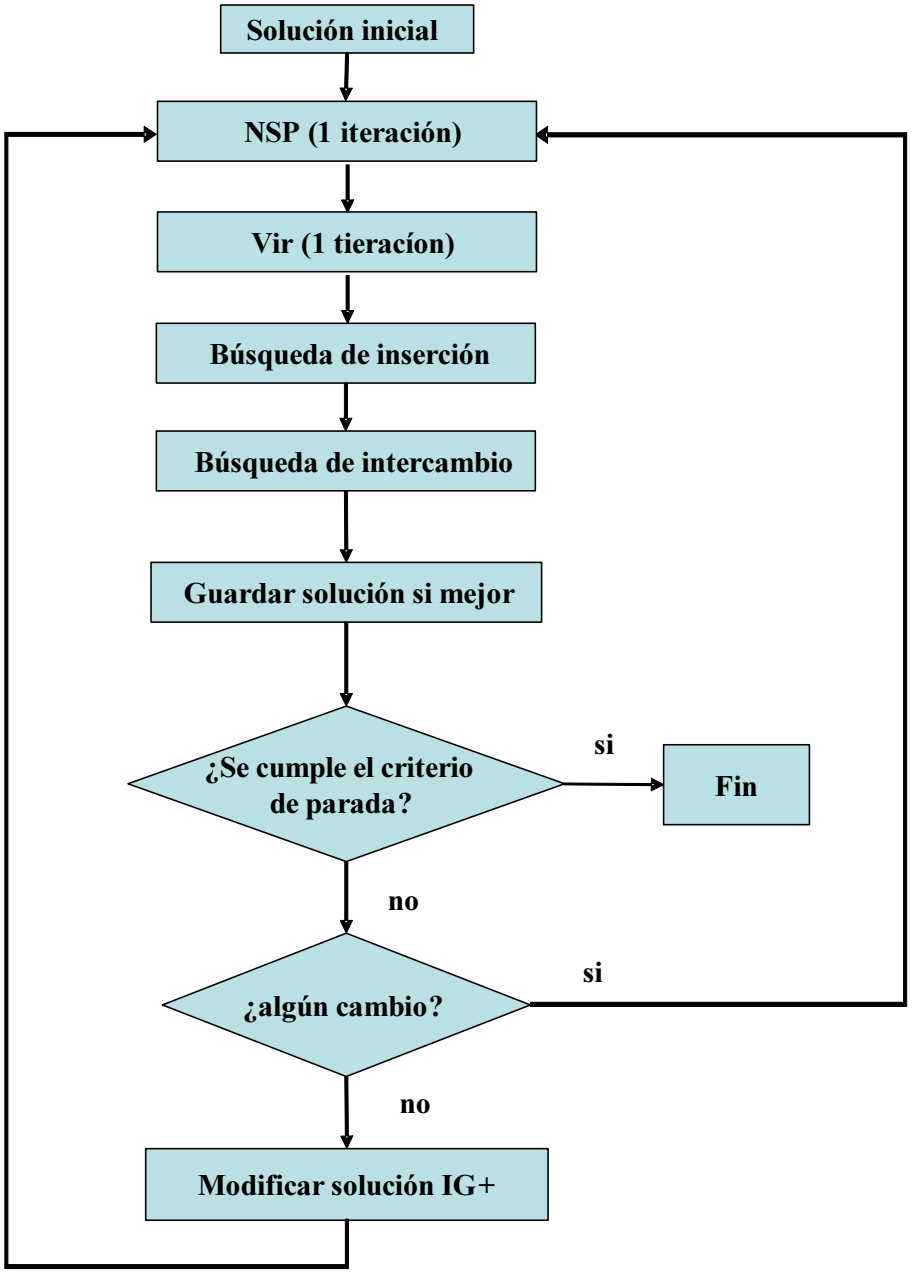

Figura 4.7 – Diagrama de flujo del algoritmo NVST-IG+.

Evidentemente, las búsquedas locales restringidas (NSP y Vir) integradas en el ciclo, solo se pueden aplicar en una única iteración, ya que si no se ciclarían. Por ejemplo, en la primera iteración de NSP se fuerza a que los trabajos seleccionados se coloquen en la mejor opción excepto donde ya estaban; una segunda iteración muy probablemente los devolvería al mismo sitio donde estaban inicialmente, produciendo un ciclado del algoritmo. Igualmente, a las búsquedas locales restringidas se les obliga a que no empeoren el valor del $C_{\text{máx}}$. A pesar de todo lo indicado hasta ahora, es importante remarcar que aún sigue siendo un algoritmo básicamente sencillo, ya que se compone de un bucle VND que incluye cuatro búsquedas locales (**N**SP, **V**IR, in**S**erción e in**T**ercambio) más un modificador de la solución tipo IG+. Denominaremos a este nuevo algoritmo NVST-IG+. Este algoritmo se implementa fácilmente y básicamente se compone de una serie de operaciones sencillas lo que hace que el algoritmo sea muy rápido en cada iteración.

4.6. Análisis computacional

Como en anteriores secciones, empleamos las mismas 1.400 instancias agrupadas en los siete intervalos ya mencionados, $U(1, 100)$, $U(10, 100)$, trabajos correlacionadas, máquinas correlacionadas, $U(100, 200)$, $U(100, 120)$ y $U(1000, 1100)$ y constituidas por 10 instancias de cada combinación de $m = \{10, 20, 30, 40, 50\}$ y $n = \{100, 200, 500, 1000\}$. Así mismo, se emplearan el mismo grupo de 8 ordenadores tipo PC/AT con procesador Pentium 4 con 3.0 GHz y con 1 GB de memoria RAM empleando el sistema operativo Windows XP 32 bits SP3.

Se presentan a continuación los resultados de los algoritmos estudiados hasta ahora, es decir, los 3 representantes del estado del arte (CPLEX, Partial y RBS) y siete algoritmos de los presentados en esta Tesis Doctoral hasta el momento (IG, NSP, Vir, IG+, NSP+, Vir+ y NVST-IG+), para finalmente

presentar un resumen de los mismos así como algunos análisis estadísticos. Se muestran todos los resultados juntos para una mejor comparación entre los datos ya obtenidos y las nuevas mejoras.

4.6.1. Intervalo U(1,100)

En la tabla 4.1 se pueden observar los resultados de todos los algoritmos estudiados. Como es habitual, el tiempo de parada es de 15 segundos para todos los algoritmos que pueden pararse a tiempo, lo que implica que sean excepciones Partial que ya, como se explicó, se para a 15 segundos en su segunda fase y RBS que no puede interrumpirse. Esto nos lleva a que Partial y RBS lleguen hasta tiempos muy superiores a 15 segundos para algunos casos.

n	m	CPLEX	Partial	RBS	IGS	NSP	Vir	IG+	NSP+	Vir+	NVST-IG+
100	10	0,00	0,38	2,40	0,14	0,22	0,15	0,13	0,52	0,29	0,17
100	20	0,00	0,63	2,88	0,46	1,03	1,18	0,39	1,68	1,21	0,73
100	30	0,00	2,99	1,21	0,63	0,85	0,86	0,63	1,07	0,74	0,00
100	40	0,00	7,41	2,74	1,48	2,74	2,74	1,24	1,81	2,74	0,91
100	50	0,00	7,94	0,00	1,25	1,25	1,25	1,00	1,25	1,25	0,00
200	10	0,20	0,34	1,34	0,37	0,50	0,42	0,35	0,58	0,46	0,26
200	20	1,75	1,39	2,28	1,68	1,82	1,68	0,94	1,94	1,51	0,98
200	30	2,12	1,82	3,59	2,38	2,96	2,53	1,79	2,65	2,29	2,02
200	40	3,03	1,73	4,18	3,37	4,19	3,48	2,89	3,49	3,01	2,16
200	50	1,60	7,31	4,87	4,71	5,19	5,32	4,55	5,17	5,14	3,69
500	10	0,29	0,31	0,97	0,57	0,57	0,53	0,45	0,53	0,49	0,37
500	20	1,26	0,96	1,48	2,28	2,35	1,84	1,41	1,73	1,61	1,20
500	30	3,40	2,31	2,00	4,05	4,21	2,96	2,18	2,61	2,43	1,83
500	40	4,15	3,64	1,82	4,19	4,86	3,52	1,97	2,33	2,23	1,46
500	50	5,63	6,37	1,88	6,07	7,21	4,80	4,51	4,80	4,21	2,78
1000	10	0,19	0,54	0,73	0,42	0,41	0,38	0,34	0,37	0,35	0,31
1000	20	1,35	0,91	1,09	1,44	1,46	1,29	0,87	1,01	0,98	0,83
1000	30	2,36	2,03	1,57	3,25	3,42	2,70	2,01	2,18	2,08	1,87
1000	40	4,26	4,12	2,00	4,81	5,00	3,96	2,85	2,93	2,88	2,45
1000	50	6,05	4,50	1,57	5,66	6,21	4,38	3,17	3,44	3,13	2,70
Media		1,88	*2,88*	2,03	2,46	2,82	2,30	1,68	2,10	1,95	**1,34**

Tabla 4.1 – Comparación entre los algoritmos CPLEX, Partial, RBS, IGS, NSP, Vir, IG+, NSP+ y Vir+ para el intervalo de distribución de tiempos de proceso $U(1, 100)$.

En esta tabla 4.1 podemos constatar de nuevo el buen comportamiento de CPLEX frente a otros algoritmos como Partial que usan al propio CPLEX como solver. Igualmente, se puede observar como las mejoras introducidas en los algoritmos propuestos nos proporcionan una mejora sustancial en sus resultados. Así los algoritmos IG+, NSP+ y Vir+ superan los resultados previamente obtenidos por sus homólogos sin las formas de selección más dirigidas que hemos desarrollado. En particular, podemos observar como, de media, el algoritmo IG+ ya supera el mejor resultado obtenido hasta la fecha, que estaba representado por CPLEX. Incluso en este caso vemos como la mayor desviación que daba CPLEX llegaba al 6 % mientras que IG+ apenas si supera el 4,5 %. No obstante lo anterior, el algoritmo híbrido NVST-IG+ da, con diferencia, los mejores resultados, no llegando a un error mayor que el 3,7 % en el peor de los casos, y con una media del 1,34 % es por tanto el mejor algoritmo en este intervalo.

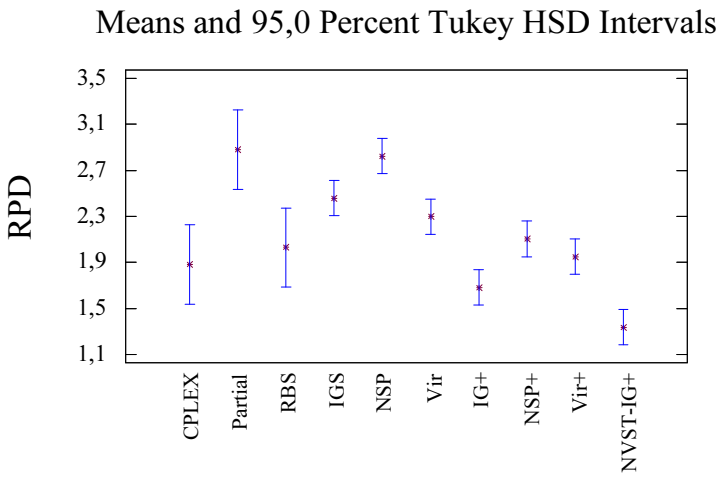

Figura 4.8 – ANOVA del intervalo U(1,100).

El la figura 4.8 podemos ver como los algoritmos de selección dirigida propuestos (IG+, NSP+ y Vir+) muestran ser estadísticamente superiores de

sus algoritmos de procedencia (IGS, NSP y Vir), aunque no lo son del estado del arte. Sin embargo, el algoritmo híbrido NVST-IG+ ya se muestra estadísticamente mejor que cualquier otro algoritmo estudiado.

4.6.2. Intervalo U(10,100)

En la tabla 4.2 podemos apreciar los resultados obtenidos en el intervalos U(10,100). Recordemos que en este intervalo Partial era el que mejores resultados proporcionaba, aunque los primeros algoritmos propuestos ya se acercaban a los mejores valores.

n	m	CPLEX	Partial	RBS	IGS	NSP	Vir	IG+	NSP+	Vir+	NVST-IG+
100	10	0,06	0,39	1,45	0,49	0,57	0,37	0,44	0,56	0,50	0,36
100	20	0,79	0,53	2,67	0,98	1,22	0,85	0,74	1,01	0,85	0,60
100	30	2,14	1,29	4,49	1,67	1,88	1,46	0,78	1,38	0,99	0,43
100	40	0,29	1,47	4,98	2,82	3,52	2,70	2,00	2,12	2,24	0,94
100	50	0,33	0,67	6,66	0,55	1,48	0,62	0,33	0,62	0,62	0,13
200	10	0,25	0,27	0,80	0,52	0,43	0,30	0,27	0,36	0,28	0,21
200	20	1,66	0,90	1,52	1,67	1,15	0,94	0,71	1,01	0,87	0,57
200	30	2,96	2,40	2,16	2,44	1,80	1,82	1,25	1,55	1,48	0,68
200	40	5,02	2,05	2,68	2,93	2,46	2,58	1,23	1,98	1,48	0,82
200	50	4,36	1,80	1,80	2,43	1,91	2,27	1,08	1,52	1,48	0,16
500	10	0,27	0,23	0,52	0,45	0,39	0,38	0,28	0,36	0,30	0,24
500	20	0,97	0,82	1,02	1,40	1,10	1,04	0,75	0,88	0,79	0,64
500	30	1,80	1,42	0,99	2,17	1,82	1,74	1,08	1,32	1,27	1,07
500	40	2,53	2,66	1,33	3,01	2,52	2,65	1,57	1,96	1,90	1,56
500	50	3,08	3,25	1,28	3,52	2,84	3,09	1,86	2,19	2,20	1,80
1000	10	0,19	0,73	0,39	0,26	0,23	0,22	0,18	0,19	0,19	0,16
1000	20	0,83	0,66	0,47	0,84	0,75	0,71	0,53	0,62	0,58	0,51
1000	30	1,38	0,98	0,64	1,44	1,25	1,24	0,87	1,01	0,96	0,98
1000	40	1,75	1,72	0,81	2,00	1,71	1,77	1,27	1,42	1,37	1,38
1000	50	2,05	2,05	0,83	2,48	2,18	2,23	1,62	1,83	1,79	1,81
Media		1,64	1,31	*1,87*	1,70	1,56	1,45	0,94	1,19	1,11	**0,75**

Tabla 4.2 – Comparación entre los algoritmos CPLEX, Partial, RBS, IGS, NSP, Vir, IG+, NSP+ y Vir+ para el intervalo de distribución de tiempos de proceso $U(10, 100)$.

Se puede apreciar como en este intervalo los algoritmos de selección dirigida son capaces de superar de media a cualquier algoritmo de los estudiados

hasta ahora. Claramente, se ve como IG+ supera a Partial, el mejor algoritmo para este intervalo hasta la fecha. De nuevo el mejor comportamiento lo ofrece NVST-IG+ que proporciona no solo los mejores resultados de media, sino que incluso obtiene los mejores resultados en la mayoría de cada una de las combinaciones de instancias objeto de estudio, con un máximo error del 1,81 % para las instancias más grandes. Así podemos ver como NVST-IG+ es un 219 % mejor que CPLEX, un 175 % mejor que Partial y cerca de un 249 % mejor que RBS.

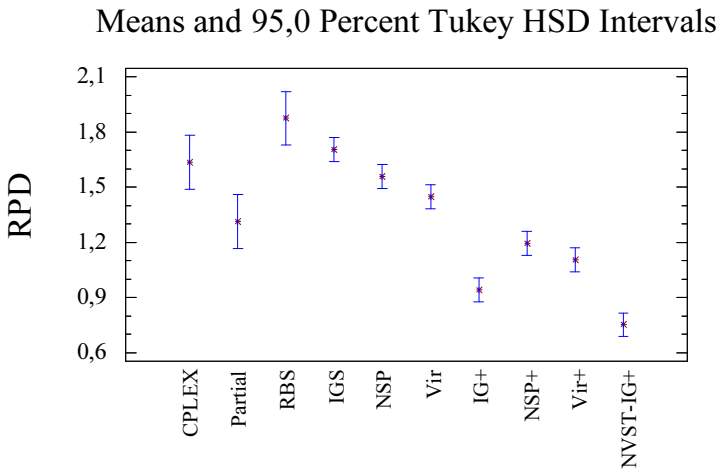

Figura 4.9 – ANOVA del intervalo U(10,100).

La figura 4.9 muestra como el algoritmo de selección dirigida IG+ ya es significativamente distinto a Partial, el algoritmo con mejor comportamiento hasta el momento en este intervalo. También se puede apreciar como todos los algoritmos de selección dirigida son significativamente distintos tanto de CPLEX como de RBS. Aquí se ve como el último algoritmo propuesto, NVST-IG+ de nuevo muestra una diferencia significativa con el resto de algoritmos considerados.

4.6.3. Intervalo de los trabajos correlacionados

La tabla 4.3 muestra como los algoritmos de selección dirigida se acercan ya mucho a los valores del mejor algoritmo para este intervalo que viene representado por RBS. Recordemos que al algoritmo RBS no se le fija un tiempo determinado de CPU, si no se detiene cuando finaliza con cada instancia, así que llega a superar los 100 segundos para las instancias más grandes frente a los 15 segundos que tienen el resto de los algoritmos con excepción de Partial. Aún así podemos apreciar como NSP+ obtiene valores proximos a RBS y el algoritmo NVST-IG+ tiene resultados similares.

n	m	CPLEX	Partial	RBS	IGS	NSP	Vir	IG+	NSP+	Vir+	NVST-IG+
100	10	0,50	0,53	0,79	0,75	0,48	0,49	0,38	0,31	0,37	0,35
100	20	1,18	2,19	1,26	1,30	0,61	1,00	0,88	0,50	0,75	0,78
100	30	2,32	4,73	0,98	1,82	0,50	1,04	0,95	0,44	1,01	0,71
100	40	2,42	7,21	1,22	4,03	1,93	2,03	2,41	1,39	2,54	2,24
100	50	3,18	7,83	2,54	5,06	4,02	3,32	2,34	1,60	2,92	2,17
200	10	0,29	0,26	0,35	0,49	0,36	0,42	0,24	0,22	0,25	0,22
200	20	1,64	1,18	0,56	0,63	0,38	0,50	0,37	0,33	0,41	0,37
200	30	2,82	2,50	0,57	1,26	0,61	0,96	0,80	0,65	0,80	0,64
200	40	3,12	3,49	-0,24	1,34	0,11	0,79	0,62	0,36	0,69	0,33
200	50	4,97	5,09	-0,65	2,28	0,54	1,42	1,27	0,91	1,27	0,94
500	10	0,25	0,16	0,13	0,28	0,28	0,28	0,14	0,15	0,16	0,14
500	20	0,69	0,77	0,13	0,37	0,33	0,36	0,27	0,26	0,26	0,22
500	30	2,21	1,16	0,07	0,44	0,27	0,38	0,29	0,25	0,31	0,17
500	40	3,95	1,75	-0,06	0,35	0,04	0,25	0,14	0,10	0,16	0,02
500	50	4,83	3,34	-0,42	0,41	-0,10	0,18	0,06	0,02	0,10	-0,10
1000	10	0,15	0,09	0,09	0,11	0,13	0,13	0,08	0,09	0,09	0,09
1000	20	0,70	0,47	0,05	0,27	0,25	0,27	0,22	0,22	0,23	0,20
1000	30	1,56	1,25	-0,02	0,21	0,16	0,18	0,16	0,14	0,16	0,13
1000	40	2,59	1,77	-0,11	0,12	0,03	0,08	0,05	0,03	0,05	0,01
1000	50	4,57	2,87	-0,28	0,14	-0,02	0,08	0,05	0,01	0,06	-0,02
Medio		2,20	*2,43*	**0,35**	1,08	0,54	0,71	0,59	0,40	0,63	0,48

Tabla 4.3 – Comparación entre los algoritmos CPLEX, Partial, RBS, IGS, NSP, Vir, IG+, NSP+ y Vir+ para el intervalo de distribución de tiempos de proceso de trabajos correlacionados.

El la figura 4.10 se puede apreciar que a pesar de obtener RBS mejores

valores de media, no es significativamente distinto de la mayoría de los algoritmos de selección dirigida o del híbrido NVST-IG+. Para este caso es de destacar los pobres resultados que muestran tanto CPLEX como Partial respecto al resto de algoritmos.

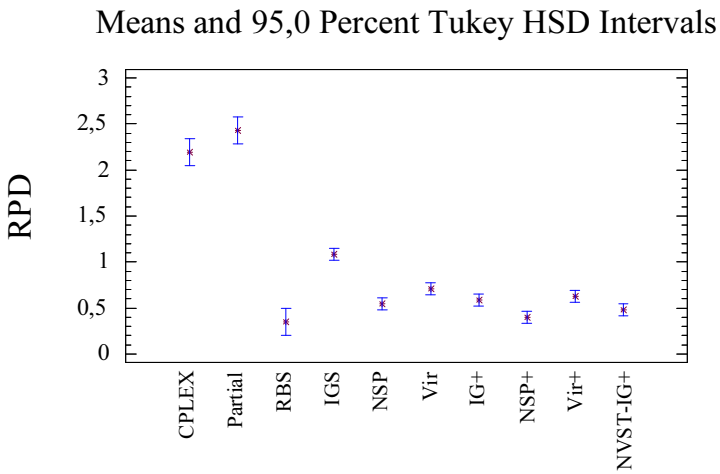

Figura 4.10 – ANOVA del intervalo de trabajos correlacionados.

4.6.4. Intervalo máquinas correlacionadas

La tabla 4.4 muestra los resultados para el intervalo de máquinas correlacionadas. Aquí ya se puede apreciar como mientras en el anterior intervalo de trabajos correlacionados RBS tenia un muy buen comportamiento, en este intervalo refleja los peores resultados con diferencia. Por parte de los algoritmos propuestos, cualquiera de ellos, muestra mejores valores de media que el estado del arte, siendo de el algoritmo NVST-IG+ el que presenta mejor media.

n	m	CPLEX	Partial	RBS	IGS	NSP	Vir	IG+	NSP+	Vir+	NVST-IG+
100	10	0,00	0,10	1,65	0,28	0,27	0,14	0,24	0,23	0,21	0,25
100	20	0,08	0,07	3,44	0,51	0,37	0,39	0,44	0,37	0,37	0,43
100	30	0,00	0,08	3,77	0,44	0,37	0,30	0,40	0,32	0,32	0,25
100	40	0,11	0,00	4,18	0,22	0,23	0,15	0,20	0,15	0,20	0,15
100	50	0,29	0,00	5,37	0,47	0,53	0,35	0,41	0,29	0,35	0,32
200	10	0,45	0,37	1,47	0,56	0,42	0,47	0,52	0,49	0,49	0,53
200	20	0,91	0,53	1,16	0,42	0,28	0,32	0,33	0,31	0,31	0,29
200	30	1,28	1,15	2,27	0,76	0,53	0,58	0,55	0,48	0,44	0,50
200	40	1,58	0,96	4,37	0,67	0,50	0,52	0,55	0,42	0,46	0,49
200	50	1,91	0,69	4,29	0,82	0,56	0,65	0,56	0,51	0,50	0,33
500	10	0,28	0,15	0,73	0,60	0,55	0,54	0,69	0,64	0,64	0,65
500	20	0,86	0,56	1,65	0,72	0,66	0,67	0,71	0,71	0,69	0,72
500	30	1,09	1,17	0,82	0,79	0,69	0,71	0,75	0,72	0,69	0,68
500	40	1,67	1,86	2,53	0,76	0,68	0,69	0,69	0,70	0,67	0,64
500	50	2,72	2,87	2,99	1,00	0,97	0,94	0,81	0,81	0,83	0,77
1000	10	0,27	0,20	0,34	0,64	0,60	0,59	0,61	0,63	0,61	0,59
1000	20	0,48	0,39	0,51	0,87	0,84	0,85	0,88	0,88	0,85	0,88
1000	30	0,96	1,19	1,20	1,02	1,03	1,02	0,97	1,00	1,00	0,98
1000	40	2,52	2,28	1,78	0,75	0,74	0,74	0,73	0,74	0,71	0,69
1000	50	5,19	4,23	2,75	0,93	0,96	0,95	0,85	0,89	0,85	0,82
Media		1,13	0,94	*2,36*	0,66	0,59	0,58	0,60	0,56	0,56	**0,55**

Tabla 4.4 – Comparación entre los algoritmos CPLEX, Partial, RBS, IGS, NSP, Vir, IG+, NSP+ y Vir+ para el intervalo de distribución de tiempos de proceso de máquinas correlacionadas.

La figura 4.11 nos muestra como todos los algoritmos propuestos son significativamente distintos al estado del arte. Igualmente queda patente la diferencia de resultados para este intervalo del RBS, que queda muy lejos de los valores obtenidos por resto de algoritmos.

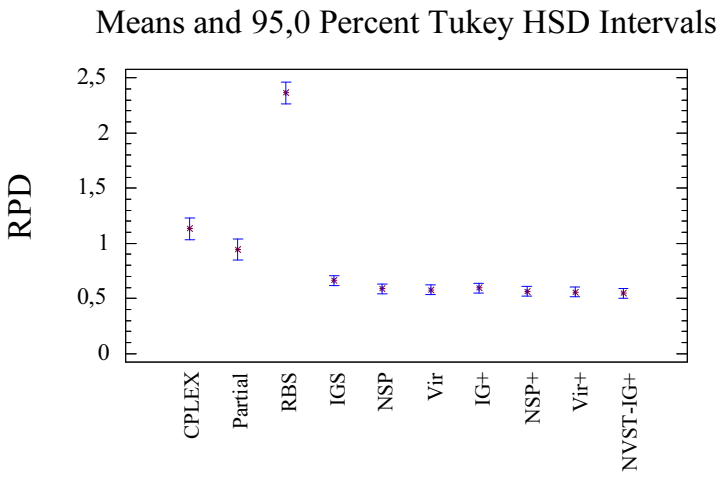

Figura 4.11 – ANOVA del intervalo de máquinas correlacionados.

4.6.5. Intervalo U(100,200)

El intervalo U(100,200) es el primero de los nuevos intervalos diseñados para estos experimentos. La tabla 4.5 muestra los resultados en este intervalo.

n	m	CPLEX	Partial	RBS	IGS	NSP	Vir	IG+	NSP+	Vir+	NVST-IG+
100	10	0,07	0,09	1,22	0,33	0,30	0,19	0,26	0,29	0,24	0,26
100	20	0,38	0,47	1,07	0,39	0,37	0,32	0,32	0,43	0,35	0,30
100	30	1,90	2,81	1,01	0,21	0,16	0,03	-0,01	0,02	-0,01	-0,10
100	40	1,38	2,43	0,99	0,47	0,42	0,34	0,22	0,16	0,18	0,07
100	50	0,28	0,14	1,66	0,14	0,22	0,35	0,10	0,09	0,36	0,11
200	10	0,69	0,43	1,06	0,40	0,23	0,29	0,33	0,38	0,35	0,31
200	20	0,50	0,83	1,42	0,74	0,47	0,64	0,50	0,56	0,59	0,49
200	30	1,15	2,73	0,01	0,11	-0,29	-0,12	-0,22	-0,22	-0,20	-0,30
200	40	1,01	1,06	0,89	0,83	0,48	0,83	0,43	0,48	0,66	0,42
200	50	0,85	0,68	1,12	0,95	0,55	0,96	0,54	0,58	0,85	0,52
500	10	0,29	0,23	0,49	0,26	0,13	0,18	0,21	0,23	0,22	0,21
500	20	1,05	0,66	1,04	0,75	0,58	0,67	0,71	0,76	0,77	0,69
500	30	0,91	1,47	0,57	0,78	0,56	0,66	0,59	0,57	0,59	0,55
500	40	0,48	2,15	0,32	0,54	0,35	0,43	0,31	0,30	0,33	0,28
500	50	0,69	0,98	0,69	0,84	0,58	0,80	0,51	0,59	0,64	0,51
1000	10	0,21	0,31	0,22	0,13	0,08	0,11	0,12	0,11	0,13	0,11
1000	20	0,80	0,67	0,64	0,50	0,38	0,46	0,46	0,50	0,49	0,40
1000	30	0,84	0,93	0,56	0,66	0,59	0,64	0,58	0,59	0,61	0,46
1000	40	0,98	1,18	0,70	0,69	0,58	0,66	0,57	0,63	0,61	0,58
1000	50	0,71	0,69	0,54	0,60	0,48	0,58	0,49	0,51	0,54	0,47
Media		0,76	*1,05*	0,81	0,52	0,36	0,45	0,35	0,38	0,42	**0,32**

Tabla 4.5 – Comparación entre los algoritmos CPLEX, Partial, RBS, IGS, NSP, Vir, IG+, NSP+ y Vir+ para el intervalo de distribución de tiempos de proceso de $U(100, 200)$.

Para este intervalo, ya se ve como todos los algoritmos propuestos producen mejores resultados que el estado de el arte, a pesar de que tanto Partial como RBS emplean más de los 15 segundos utilizados por el resto de algoritmos, para los cuales recordemos que se empleó para su codificación Delphi, más lento que C++ empleado en la codificación de Partial o RBS. Es decir, a pesar de actuar en circunstancias más desfavorables, consiguen sacar mejores resultados.

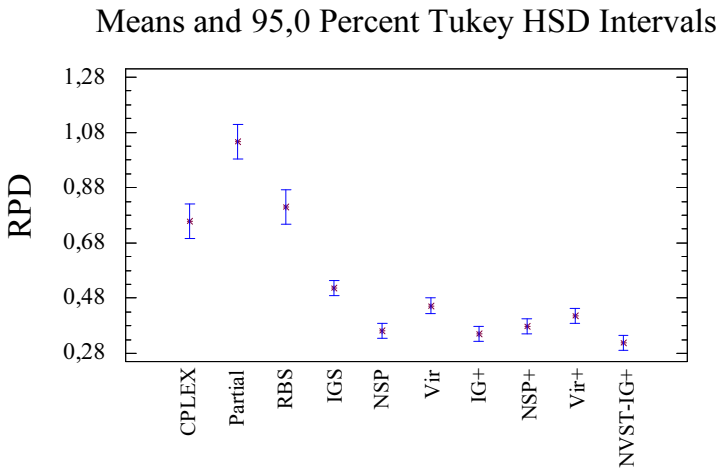

Figura 4.12 – ANOVA del intervalo U(100,200).

La figura 4.12 muestra como claramente todos los algoritmos propuestos son significativamente mejores que el estado del arte, con especial comportamiento por parte de algoritmos como NSP, IG+, NSP+ y NVST-IG+.

4.6.6. Intervalo U(100,120)

Las medias mostradas en la tabla 4.6 muestran que las diferencias de los algoritmos con la referencia son ya cercanas a cero. No obstante, se puede apreciar como los algoritmos propuestos siguen mejorando los valores dados por el estado del arte. Los mejores métodos son el IG+ y el NVST-IG+ con medias del 0,04 %. El mejor método del estado del arte es el RBS con un 0,13 % lo que significa que los mejores algoritmos propuestos son un 325 % mejores que este.

n	m	CPLEX	Partial	RBS	IGS	NSP	Vir	IG+	NSP+	Vir+	NVST-IG+
100	10	0,06	0,19	0,24	0,10	0,06	0,06	0,07	0,09	0,09	0,08
100	20	0,20	0,28	0,22	0,11	0,06	0,10	0,06	0,06	0,06	0,05
100	30	0,15	0,35	0,18	0,15	0,04	0,09	0,00	0,00	0,01	0,00
100	40	0,03	0,40	0,10	0,15	0,00	0,04	0,00	0,00	0,00	0,00
100	50	0,05	0,94	0,30	0,12	0,03	0,14	0,01	0,04	0,09	0,00
200	10	0,10	0,13	0,19	0,08	0,04	0,05	0,06	0,06	0,07	0,06
200	20	0,24	0,26	0,22	0,15	0,08	0,13	0,08	0,09	0,11	0,09
200	30	0,44	0,68	0,04	0,10	0,03	0,07	0,03	0,02	0,03	0,01
200	40	0,12	0,18	0,06	0,08	0,06	0,08	0,03	0,04	0,07	0,03
200	50	0,05	0,32	0,07	0,06	0,02	0,08	0,02	0,02	0,02	0,01
500	10	0,15	0,29	0,23	0,07	0,04	0,06	0,08	0,10	0,09	0,09
500	20	0,16	0,23	0,18	0,10	0,07	0,09	0,07	0,08	0,09	0,07
500	30	0,39	0,42	0,11	0,17	0,13	0,15	0,12	0,11	0,11	0,10
500	40	0,07	0,86	-0,02	0,06	0,01	0,05	-0,02	-0,01	-0,02	-0,02
500	50	0,11	0,10	0,04	0,09	0,05	0,08	0,01	0,03	0,03	0,01
1000	10	0,16	0,28	0,13	0,05	0,03	0,04	0,07	0,07	0,07	0,07
1000	20	0,25	0,26	0,16	0,08	0,06	0,08	0,08	0,09	0,09	0,08
1000	30	0,12	0,14	-0,05	0,07	0,05	0,07	0,05	0,05	0,05	0,03
1000	40	0,16	0,14	0,10	0,07	0,06	0,06	0,05	0,06	0,05	0,05
1000	50	0,07	0,10	0,03	0,03	0,02	0,03	0,01	0,02	0,01	0,01
Media		0,15	*0,33*	0,13	0,09	0,05	0,08	**0,04**	0,05	0,06	**0,04**

Tabla 4.6 – Comparación entre los algoritmos CPLEX, Partial, RBS, IGS, NSP, Vir, IG+, NSP+ y Vir+ para el intervalo de distribución de tiempos de proceso de $U(100, 120)$.

En el ANOVA de la figura 4.13 podemos observar como las diferencias entre el estado del arte y todos los algoritmos propuestos son estadísticamente significativas. Cabe destacar igualmente los pobres resultados ofrecidos por Partial.

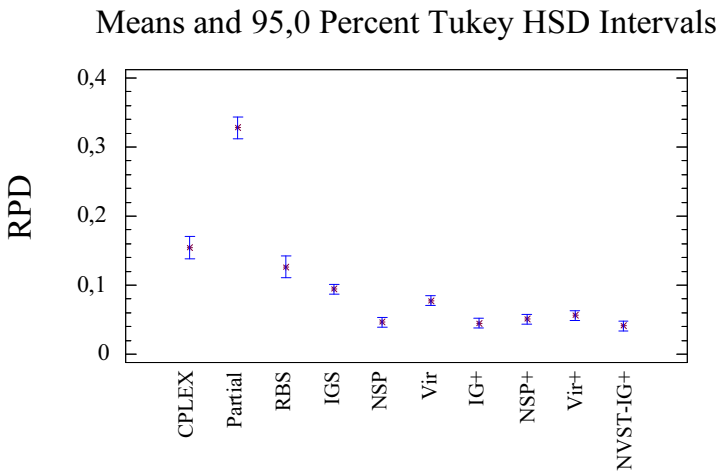

Figura 4.13 – ANOVA del intervalo U(100,120).

4.6.7. Intervalo U(1000,1100)

En este último intervalo nos muestra en la tabla 4.7 valores ya muy próximos al cero de media respecto a la referencia. Aún así, las medias mostradas por los algoritmos propuestos superan a las obtenidas por el estado del arte. Aquí IG+ y NVST-IG+ vuelven a dar muy buenos resultados con un 0,02 % mientras que el mejor resultado del estado del arte viene dado por CPLEX con un 0,08 %, es decir, un 400 % peor.

n	m	CPLEX	Partial	RBS	IGS	NSP	Vir	IG+	NSP+	Vir+	NVST-IG+
100	10	0,06	0,04	0,14	0,04	0,03	0,02	0,03	0,04	0,03	0,03
100	20	0,04	0,07	0,18	0,03	0,03	0,02	0,01	0,03	0,03	0,02
100	30	0,17	0,36	0,23	0,02	0,01	-0,01	-0,01	-0,01	-0,01	-0,02
100	40	0,13	0,35	0,33	0,04	0,03	0,02	0,01	0,01	0,01	0,00
100	50	0,01	0,01	0,77	0,01	0,01	0,01	0,00	0,00	0,02	0,00
200	10	0,04	0,06	0,13	0,05	0,02	0,03	0,04	0,05	0,04	0,04
200	20	0,09	0,08	0,17	0,07	0,04	0,06	0,04	0,06	0,06	0,05
200	30	0,09	0,72	-0,03	-0,07	-0,11	-0,09	-0,10	-0,10	-0,10	-0,11
200	40	0,10	0,09	0,29	0,05	0,02	0,05	0,02	0,04	0,04	0,02
200	50	0,07	0,11	0,52	0,06	0,03	0,07	0,03	0,04	0,05	0,03
500	10	0,06	0,03	0,09	0,03	0,02	0,02	0,03	0,04	0,04	0,03
500	20	0,10	0,04	0,09	0,05	0,03	0,04	0,04	0,04	0,04	0,03
500	30	0,02	0,94	0,03	0,04	0,01	0,03	0,02	0,02	0,02	0,01
500	40	0,03	1,19	0,03	0,04	0,02	0,03	0,02	0,02	0,02	0,01
500	50	0,08	0,09	0,23	0,07	0,05	0,07	0,04	0,05	0,06	0,04
1000	10	0,08	0,08	0,07	0,02	0,02	0,02	0,03	0,04	0,03	0,03
1000	20	0,10	0,07	0,08	0,04	0,03	0,04	0,05	0,05	0,05	0,05
1000	30	0,10	0,07	0,07	0,07	0,06	0,06	0,06	0,06	0,06	0,05
1000	40	0,08	0,08	0,09	0,05	0,04	0,05	0,04	0,04	0,04	0,04
1000	50	0,06	0,06	0,13	0,05	0,04	0,05	0,04	0,04	0,05	0,04
Media		0,08	*0,23*	0,18	0,04	**0,02**	0,03	**0,02**	0,03	0,03	**0,02**

Tabla 4.7 – Comparación entre los algoritmos CPLEX, Partial, RBS, IGS, NSP, Vir, IG+, NSP+ y Vir+ para el intervalo de distribución de tiempos de proceso $U(1000, 1100)$.

La figura 4.14 muestra las diferencias significativas que existen entre el estado del arte y cualquiera de los algoritmos propuestos, aunque apenas si hay diferencias apreciables entre estos últimos.

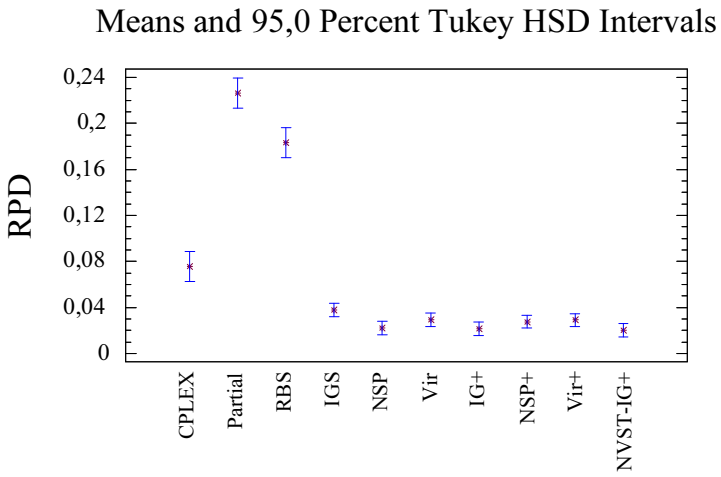

Figura 4.14 – ANOVA del intervalo U(1000,1100).

4.6.8. Todos los intervalos

Para poder tener un cuadro más completo de todos los resultados obtenidos en los intervalos estudiados, en la tabla 4.8 mostramos un resumen de dichos resultados.

Intervalo	CPLEX	Partial	RBS	IGS	NSP	Vir	IG+	NSP+	Vir+	NVST-IG+
$U(1,100)$	1,88	*2,88*	2,03	2,46	2,82	2,3	1,68	2,10	1,95	**1,34**
$U(10,100)$	1,64	1,31	*1,87*	1,7	1,56	1,45	0,94	1,19	1,11	**0,75**
$U(100,200)$	0,76	*1,05*	0,81	0,52	0,36	0,45	0,35	0,38	0,42	**0,32**
$U(100,120)$	0,15	*0,33*	0,13	0,09	0,05	0,08	**0,04**	0,05	0,06	**0,04**
$U(1000,1100)$	0,08	*0,23*	0,18	0,04	**0,02**	0,03	**0,02**	0,03	0,03	**0,02**
jobcorre	2,20	*2,43*	**0,35**	1,08	0,54	0,71	0,59	0,40	0,63	0,48
machcorre	1,13	0,94	*2,36*	0,66	0,59	0,58	0,60	0,56	0,56	**0,55**
Media	1,12	*1,31*	1,10	0,94	0,85	0,80	0,60	0,67	0,68	**0,50**
M, usuales	1,71	*1,89*	1,65	1,48	1,38	1,26	0,95	1,07	1,06	**0,78**

M.usuales= Media de los intervalos más usuales: $U(1,100)$; $U(10,100)$; Trabajos Correlacionados; máquinas Correlacionadas

Tabla 4.8 – Resumen de resultados para todos los intervalos y algoritmos. En negrita (cursiva) se representa el mejor (peor) resultado, respectivamente.

Se han presentado dos medias globales. La primera refleja la media de los siete intervalos estudiados, es decir una media de la desviación relativa respecto a los valores obtenidos por CPLEX al cabo de dos horas con cada una de la 1.400 instancias. La segunda media es la realizada sobre los cuatro intervalos más usados en la literatura, es decir U(1,100), U(10,100), trabajos correlacionados y máquinas correlacionadas. Se han mostrado ambos resultados para evitar que se sospeche de cierto sesgo al emplear los nuevos intervalos propuestos en beneficio de mejores resultados propios.

Viendo las medias obtenidas podemos comprobar como cualquiera de los algoritmos propuestos obtienen mejores resultados que el estado del arte, tanto en el caso de los siete intervalos estudiados, como en las medias sobre los intervalos más clasicos. En este punto queremos recordar que los algoritmos propuestos son esencialmente simples y que no emplean ningún solver comercial. También hemos visto como solo en el caso de trabajos correlacionados, RBS conseguía mejorar los resultados de NVST-IG+, aunque en este caso la media de tiempo que emplea RBS para resolver las instancias se eleva hasta los 25 segundos de tiempo de CPU, mientras que NVST-IG+ solo emplea

15 segundos Así pues, posteriormente veremos los resultados de todos los algoritmos estudiados para distintos tiempos de CPU.

Se muestra ahora un análisis estadístico basado en el ANOVA para comprobar si las diferencias observadas son estadísticamente significativas. En total se evalúan, para los métodos de CPLEX, Partial y RBS , 1.400 resultados respectivamente, proporcionados por las 200 instancias de cada uno de los siete intervalos de tiempos de proceso objeto de estudio, más los 7.000 resultados de cada uno de los algoritmos propuestos, ya que se hicieron 5 réplicas de cada uno, con lo que nos lleva a un total de 53.200 resultados a estudiar. La figura 4.15 nos muestra un gráfico de medias con el resumen del estudio.

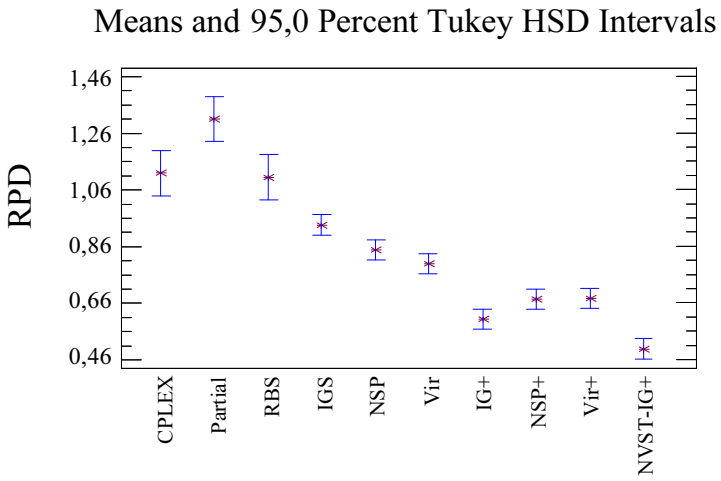

Figura 4.15 – Medias e intervalos de Tukey HSD con un 95 % de nivel de confianza para todos los algoritmos y todas las instancias estudiadas.

Como se puede apreciar, los algoritmos de selección dirigida son estadísticamente diferentes de sus homólogos no mejorados. NSP+ y Vir+ se pueden considerar equivalentes, mientras que IG+ ya empieza a diferenciarse de ellos.

Finalmente, podemos apreciar como NVST-IG+ es estadísticamente mejor que el resto de los algoritmos propuestos o así como del estado del arte.

Hasta ahora los resultados mostrados por los siete algoritmos propuestos y CPLEX se han obtenido parando a 15 segundos de tiempo de CPU. Recordemos que Partial ha sido parado a los 15 segundos de su segunda fase, lo que nos lleva a tiempos superiores a 15 segundos. Igualmente RBS supera los 15 segundos de tiempo dado el carácter constructivo de este algoritmo, por lo que se ha de dejar que finalice, lo que para las instancias más grandes eleva mucho este tiempo, siendo menor para las instancias pequeñas, y dándonos una media, para la amplitud de rayo recomendado por sus autores, de 25 segundos. Quedaría pendiente la cuestión de si los resultados obtenidos por el mejor algoritmo propuesto, NVST-IG+, mantienen estos buenos resultados para tiempos de parada de CPU mayores, o incluso menores. Para resolver esta cuestión, hemos llevado a cabo una serie de experimentos adicionales donde se han ajustado los tiempos de parada de CPU a 5, 25, 300, 600, 1800, 3600 y 7200 segundos. Estos tiempos de nuevo se ajustan exactamente en el caso del algoritmo propuesto NVST-IG+ y CPLEX. Para Partial estos nuevos tiempos se usan para la segunda fase de este algoritmo. En el caso de Partial solo se ha llegado hasta 300 segundos debido a que las diferencias, desfavorables para Partial, con los restantes algoritmos, iban aumentando conforme aumentaban los tiempos. Igualmente, ha sido seleccionado el tiempo de 25 segundos por ser, aproximadamente, la media del tiempo que emplea RBS en resolver todas las instancias con la amplitud de rayo recomendado por los autores. En el caso de RBS, se evaluó hasta 300 segundos aumentando la correspondiente amplitud de rayo.

Tiempo	Algoritmo	$U(1,100)$	$U(10,100)$	$U(100,200)$	$U(100,120)$	$U(1000,1100)$	Tracorre	Maqcorre	Media
5	Partial	*3,93*	2,25	*1,22*	*0,36*	*0,27*	2,89	1,49	1,77
	CPLEX	3,45	*2,53*	0,98	0,20	0,09	*3,92*	*2,70*	*1,98*
	NVST-IG+	**1,78**	**1,06**	**0,41**	**0,06**	**0,03**	**0,71**	**0,66**	**0,67**
25	Partial	*2,45*	1,22	0,69	*0,28*	*0,18*	*2,12*	0,73	*1,10*
	CPLEX	1,41	1,42	0,58	0,12	0,04	1,75	0,69	0,86
	NVST-IG+	**1,20**	**0,61**	**0,27**	**0,04**	**0,02**	0,38	**0,51**	**0,43**
	RBS	2,03	*1,87*	*0,81*	0,13	*0,18*	**0,35**	*2,36*	*1,10*
300	Partial	*1,59*	0,33	0,20	*0,21*	*0,09*	*1,23*	**0,10**	*0,54*
	CPLEX	**0,35**	0,29	0,24	0,04	0,02	0,45	0,12	0,22
	NVST-IG+	0,67	**0,19**	**0,14**	**0,02**	**0,00**	0,05	0,38	**0,21**
	RBS	0,87	*0,85*	*0,39*	0,07	0,08	**-0,20**	*1,17*	0,46
600	CPLEX	**0,24**	*0,20*	*0,17*	*0,03*	*0,01*	*0,27*	**0,07**	**0,14**
	NVST-IG+	*0,58*	**0,09**	**0,10**	**0,01**	**0,00**	**-0,02**	*0,35*	*0,16*
1800	CPLEX	**0,12**	*0,11*	*0,08*	*0,02*	*0,01*	*0,12*	**0,03**	**0,07**
	NVST-IG+	*0,53*	**0,02**	**0,07**	**0,01**	**-0,01**	**-0,11**	*0,32*	*0,12*
3600	CPLEX	**0,05**	*0,06*	**0,02**	*0,01*	*0,00*	*0,07*	**0,01**	**0,03**
	NVST-IG+	*0,49*	**-0,03**	*0,05*	**0,00**	**-0,01**	**-0,2**	*0,29*	*0,09*
7200	CPLEX	**0,00**	*0,00*	**0,00**	**0,00**	*0,00*	*0,00*	**0,00**	**0,00**
	NVST-IG+	*0,38*	**-0,07**	*0,03*	**0,00**	**-0,01**	**-0,26**	*0,27*	*0,05*

Tabla 4.9 – Resultados de las medias de los mejores algoritmos para diferentes tiempos de parada de CPU. En negrita (cursiva) se destacan los mejores (peores) resultados.

Podemos apreciar que NVST-IG+ es el mejor método cuando se ejecuta con tiempos de CPU extremadamente cortos (5 segundos) con una desviación media total del 0,67 %. Esto es un 296 % mejor que CPLEX y un 264 % mejor que Partial. Estos resultados creemos que son dignos de mención ya que en solo 5 segundos se alcanza una buena media de desviación y sin usar ningún solver. Para 25 segundos y 300 segundos NVST-IG+ sigue siendo el mejor método de media. Para 25 segundos, tiempo medio de ejecución de RBS, vemos como sigue quedando, de media, muy lejos de NVST-IG+ o de CPLEX, mientras que Partial es el peor método para estos tiempos. Como se puede apreciar, todas las evaluaciones previas que se hicieron para 15 segundos se pueden aplicar para menores o mayores tiempos de CPU hasta los 300 segundos.
Para los tiempos de proceso aún mayores, de 10, 30, 60 y 120 minutos, solo hemos comparado los dos mejores métodos: CPLEX y NVST-IG+. De hecho, con más tiempo, CPLEX tiene que ser obviamente mejor, dado que CPLEX aplica un Branch and Cut estado del arte, un algoritmo exacto. Es de esperar que con un tiempo arbitrariamente largo sería capaz de encontrar la solución óptima para todas las instancias. Sin embargo, es de destacar como en los intervalos de U(1000,1100) y trabajos correlacionados, NVST-IG+ es capaz de mejorar, en tan solo 10 minutos, los resultados obtenidos por CPLEX con hasta 2 horas con cada instancia, lo cual es muy reseñable.

Realmente evaluamos CPLEX frente NVST-IG+ para más tiempos de parada diferentes, que incluiremos en un último estudio. Realizamos un ANOVA de dos factores cuyo gráfico de interacción de las medias se muestra en la figura 4.16. De nuevo debemos recordar que el gráfico representa la media de todos los intervalos. En el gráfico se aprecia como, hasta los 60 segundos, es mejor usar NVST-IG+ que CPLEX. A partir de los 120 segundos es estadísticamente equivalente el emplear CPLEX o NVST-IG+. Sin embargo, se puede apreciar como la curva de CPLEX cae por debajo de la curva de NVST-IG+ a partir

de los 600 segundos. No obstante, para algún intervalo especifico y algunas instancias, NVST-IG+ tiene mejores resultados.

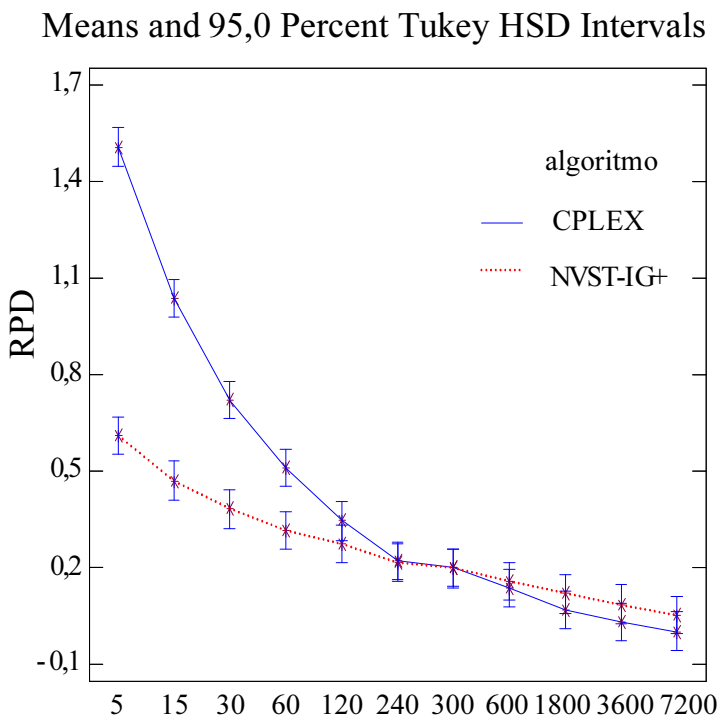

Figura 4.16 – Gráfico para los diversos tiempos estudiados (en segundos) en los algoritmos CPLEX y NVST-IG+ y para todas las instancias.

4.7. Resumen de los algoritmos de selección dirigida

En este capítulo hemos tratado de mejorar el funcionamiento de los algoritmos propuestos inicialmente, eliminando parte del azar que existía en la elección de los trabajos que tenían que ser seleccionados para su modificación por los distintos métodos. Se ha hecho un estudio sobre la preferencia de elegir

máquinas con $C_{\text{máx}}$ pero observando también la importancia de elegir las máquinas con menor C_i, llegando a una distribución uniforme en la selección entre las máquinas con $C_{\text{máx}}$ y una distribución siguiendo la numeración triangular inversa para la selección entre las restantes máquinas, con un sesgo hacia las máquinas con $C_{\text{máx}}$. Una vez seleccionada la máquina de la que hay que extraer el trabajo, la elección del trabajo es determinista siguiendo el criterio de quitar el trabajo asignado a esa máquina que mayor diferencia presente entre el tiempo de proceso de ese trabajo con el resto de tiempos de proceso del mismo trabajo en las otras máquinas. Estos métodos de selección de trabajos a ser modificados se aplicaron a los algoritmos IGS, NSP y Vir para llegar a los algoritmos IG+, NSP+ y Vir+. Finalmente, se integró dentro de la fase de búsquedas locales los algoritmos NSP y Vir limitándolos a una sola pasada y sin permitir empeorar el $C_{\text{máx}}$, manteniendo como modificador de la solución el IG+, con lo que se obtuvo el algoritmo híbrido NVST-IG+.
Los resultados obtenidos por los algoritmos modificados mostraron unas mejoras estadísticamente significativas y un distanciamiento claro sobre el estado del arte en prácticamente todos los casos. El estudio se ha completado para diversos intervalos de tiempos de parada de CPU que van desde los 5 segundos hasta las 2 horas, lo cual unido al hecho de estudiar 7 intervalos diferentes, con 200 instancias por intervalo y a hacer las 5 réplicas correspondientes para los algoritmos no deterministas, nos han llevado a superar los 4 años de tiempo de CPU para realizar todas las pruebas hasta ahora reseñadas.
Pensamos que los resultados obtenidos hasta aquí son notablemente buenos teniendo en cuenta la simplicidad de los algoritmos mostrados. Quedó patente en nuestro estudio que habrá que tener en cuenta las distintas versiones CPLEX tanto para este problema en particular como otros problemas que se puedan abordar donde sea factible el empleo de CPLEX con buenos resultados.

Quedan pendientes ahora algunas cuestiones que abordaremos en el siguiente capítulo. Por una parte, hasta ahora hemos empleado técnicas conoci-

das, como las búsquedas locales e iterated greedy y hemos realizado algunas variaciones novedosas como las búsquedas locales restringidas tipo NSP y Vir, a las que hemos añadido una selección de trabajos a modificar dentro del IGS de forma más inteligente que nos proporcionaron una algoritmo híbrido que mejora con el tiempo. Pero se nos puede tachar de no emplear una técnica realmente novedosa e igualmente de no emplear los modernos ordenadores con varios núcleos con la posible paralelización de tareas. También puede quedar la duda de si nuevas versiones de CPLEX serán capaces de superar cualquier algoritmo que se presente para este problema, dada la inherente simplicidad de su modelo matemático. Con estas cuestiones en mente y todo lo visto hasta este momento, procedemos a atacar el problema desde un ángulo diferente.

CAPÍTULO 5

HEURÍSTICAS DE REDUCCIÓN DE TAMAÑO

En capítulos anteriores se han propuesto nuevos algoritmos para el problema de máquinas paralelas con el objetivo de minimizar el makespan. Los experimentos llevados a cabo demuestran, en las condiciones probadas y con las limitaciones esperables de cualquier experimento de este tipo, que los algoritmos presentados pueden considerarse estado del arte. No obstante, como ya planteábamos al final de anterior capítulo, quedaban algunas cuestiones que abordar como el plantear unos algoritmos desde un punto de vista completamente distinto. Esto es necesario a fin emplear métodos nuevos y complementar el trabajo realizado con técnicas conocidas, como las búsquedas locales y el IG, a pesar de haber introducido las búsquedas restringidas y de hacer hecho una hibridación del todo esto en el algoritmo NVST-IG+ con acierto suficiente para superar al estado del arte. Además, la progresión de CPLEX a la versión 11.1, que se anunciaba con un aumento significativo en la resolución de problemas de asignación, nos hacía plantearnos si esta versión mejoraría notablemente sus anteriores resultados. Quedaba pendiente, así mismo, la posible paralelización del problema, dado que los modernos ordenadores suelen

poseer varios procesadores integrados, y las nuevas versiones de los solver suelen estar adaptadas a esta circunstancia. Por todo ello, nos planteamos en este parte de la Tesis, encontrar unos algoritmos basados en una reducción de tamaño de problema de asignación original, con la idea de considerar, de entre todas las posibles asignaciones de los trabajos a las máquinas, sólo unas pocas de las mejores asignaciones.

5.1. Métodos propuestos

Recordemos que las soluciones de referencia para este problema vienen dadas por el empleo de CPLEX 11.0 durante 2 horas con cada una de las 1.400 instancias pertenecientes a los 7 intervalos diferentes objeto de este estudio. La tabla 5.1 nos vuelve a mostrar de nuevo los resultados de este estudio, en el que hemos añadido el máximo gap o diferencia frente a la cota inferior y la media del gap, para cada intervalo estudiado.

n	m	U(1,100)	U(10,100)	U(100,200)	U(100,120)	U(1000,1100)	jobcorre	maqcorre	Media
100	10	100	100	100	80	30	0	100	72,90
100	20	100	50	80	70	40	0	90	61,40
100	30	100	50	0	50	0	0	90	41,40
100	40	100	90	10	90	0	0	90	54,30
100	50	100	90	20	100	100	0	100	72,90
200	10	100	30	30	60	0	0	40	37,10
200	20	100	10	10	40	0	0	40	28,60
200	30	100	0	0	0	0	0	50	21,40
200	40	100	10	20	100	20	0	70	45,70
200	50	100	10	10	90	0	0	70	40,00
500	10	90	40	0	80	0	0	20	32,90
500	20	70	20	0	30	0	0	10	18,60
500	30	80	0	0	0	0	0	30	15,70
500	40	70	0	0	20	0	0	10	14,30
500	50	80	0	0	40	0	0	50	24,30
1000	10	80	20	0	100	0	60	0	37,10
1000	20	50	0	0	30	0	0	0	11,40
1000	30	70	0	0	0	0	0	10	11,40
1000	40	90	0	0	0	0	0	30	17,10
1000	50	80	0	0	0	0	0	50	18,60
Inst. resueltas		88	26	14	49	9,50	3	47,50	33,86
Máximo gap		3,70	7,84	7,13	5,09	7,14	8,66	1,64	5,89
Gap medio		1,25	1,33	1,02	0,68	0,77	1,49	0,47	1,00

U(x,y):Distribucion uniforme entre x e y; jobcorre:trabajos correlacionados; maqcorre:máquinas correlacionadas.

Tabla 5.1 – Tabla de instancias resueltas por CPLEX 11.0, tras 2 horas como máximo con cada instancia, así como sus medias, para todos los intervalos de tiempos de proceso.

Como se puede apreciar, cerca del 34 % de las instancias se pueden resolver de forma óptima. A pesar de de no ser un gran número de ellas, dentro de las instancias no resueltas, podemos apreciar que la media del gap es alrededor del 1 %. No obstante, nos estamos refiriendo a resultados obtenidos tras dos largas horas de tiempo de CPU. Además, para algunos grupos y tamaños de

instancias, el máximo gap llega a alcanzar valores próximos al 9 %. Aún más sorprendente es encontrar que, en algunos casos, CPLEX puede resolver el 100 % de las instancias de hasta 200×50. Incluso nos podemos encontrar que CPLEX es capaz de resolver, de forma óptima, algunas instancias 1000×50 (es decir, de 50.000 variables binarias) en dos horas de CPU. Este comportamiento inesperadamente bueno de CPLEX se plantea como un verdadero desafío contra el anterior estado del arte. Ya hemos visto como CPLEX era muy competitivo (si no mejor) que los mejores métodos que existían en la literatura hasta este momento. Solo los últimos algoritmos propuestos desbancaban claramente el comportamiento de CPLEX.
Cuando se estudian las soluciones que se obtienen para las distintas instancias, se observa, dentro de las distribuciones uniformes, la escasa variación de éstas para cada grupo de instancias con el mismo número de trabajos y máquinas. Por ejemplo, podemos observar que para 100 trabajos y 10 máquinas en la distribución U(1,100), los valores del $C_{\text{máx}}$ de las 10 instancias generadas, oscilan muy poco alrededor de 99. De hecho, hallamos una sencilla fórmula empírica que nos permite hallar de forma aproximada el resultado de el problema $R//C_{\text{máx}}$ para las distribuciones uniformes. Si consideramos una distribución uniforme $U(b,t)$ de tiempos de proceso p_{ij}, en una instancia de n trabajos y m máquinas, la fórmula aproximada para obtener su $C_{\text{máx}}$ sería:

$$C_{\text{máx}} \simeq \frac{n}{m^2} \cdot (t-b) + \frac{n}{m} \cdot b \tag{5.1}$$

Se llega a esta fórmula tras observar, en el intervalo U(1,100), como existe una relación directa entre el aumento en el número de trabajos y el aumento del $C_{\text{máx}}$ y una relación inversa y exponencial con respecto al incremento en el número de máquinas, todo ello multiplicado por un factor próximo al 100. Esto nos dio el primer sumando de la ecuación. Cuando se amplió el estudio al resto de intervalos uniformes, hubo que incluir un término corrector direc-

tamente proporcional al número de trabajos y a la cota inferior del intervalo, e inversamente proporcional al número de máquinas. Se puede llegar fácilmente a una fórmula aproximada para la solución del $R//C_{\text{máx}}$ para una distribución uniforme, por que, entre otros motivos, para instancias muy grandes, la distribución uniforme distribuye, de media, valores similares entre cada trabajo y cada máquina.

Con todo, un análisis más detallado de los resultados nos lleva a constatar un sorprendente detalle: la mayoría de los trabajos se asignan a las máquinas que los procesan más rápido. Esto es predecible en tanto en cuanto nos encontramos ante un problema de asignación. Sin embargo, lo sorprendente del caso fue el constatar lo abrumadoramente consistente que resultó esta apreciación. En la inmensa mayoría de las ocasiones, cada trabajo se asigna o bien a la máquina más rápida para ese trabajo, o bien a la segunda más rápida, o bien a la tercera más rápida. En otras palabras, de entre las $n \cdot m$ variables binarias del modelo presentado en la sección 2.3, los trabajos generalmente se asignan a las máquinas i_{1j}, i_{2j} o i_{3j} donde $i_{1j} = argmin_{i \in M}\, p_{ij}$, $i_{2j} = argmin_{i \in M/i_{1j}}\, p_{ij}$ e $i_{3j} = argmin_{i \in M/i_{1j}, i_{2j}}\, p_{ij}$, es decir, la máquina que hace mínima la condición expuesta. Así pues, si los trabajos raramente se asignan a una máquina más lenta que las tres más rápidas, una pregunta acude rápidamente a la mente ¿es necesario definir todas las restantes variables? Un modelo MILP "reducido", considerando solo estas "tres máquinas más rápidas" tendrá $n \cdot 3$ variables binarias. Esto significa que si una instancia de 1000×50 necesita 50.000 variables binarias, usando solamente esas tres máquinas más rápidas para cada trabajo, todo lo que necesitaremos serán 3.000 variables binarias, es decir, 16,67 veces menos variables. Por supuesto, esta reducción de tamaño tiene un alto precio. Lo primero es que la optimalidad de la solución no puede garantizarse, dado que no se emplean todas las variables del problema. En segundo lugar, esta reducción de tamaño directa no se espera que funcione bien en todos los casos.

Veamos cómo de acertada es la hipótesis de las tres máquinas más rápidas. Para cada una de las 1.400 instancias previas y para las soluciones de referencia obtenidas por CPLEX con 2 horas de tiempo como criterio de terminación, procedemos a contar cuántos de los trabajos que no se han asignando a ninguna de las tres máquinas más rápidas, i_{1j}, i_{2j} ó i_{3j}. Los resultados se muestran en la tabla 5.2. Cada casilla de esta tabla representa el porcentaje de trabajos que en una instancia no ha sido asignados a ninguna de las tres máquinas más rápidas en la solución final de la mencionada referencia.

n	m	$U(1,100)$	$U(10,100)$	$U(100,200)$	$U(100,120)$	$U(1000,1100)$	jobcorre	maqcorre
100	10	0,30	0,50	0,80	1,30	1,10	0,70	30,20
	20	1,40	1,20	4,60	1,50	2,80	4,40	35,40
	30	3,80	4,60	18,00	24,40	31,20	10,90	41,30
	40	3,50	7,30	20,40	11,10	24,70	17,90	43,80
	50	5,50	10,20	6,60	1,50	4,80	32,40	41,30
200	10	0,20	0,10	0,30	0,30	0,50	0,30	31,90
	20	0,40	0,65	1,40	0,55	1,30	2,20	40,30
	30	0,80	1,65	6,80	17,75	10,50	4,75	44,05
	40	1,20	1,85	1,85	0,60	3,20	9,90	45,75
	50	2,15	3,65	4,30	0,25	4,35	12,60	46,15
500	10	0,04	0,00	0,06	0,10	0,08	0,02	32,82
	20	0,08	0,12	0,30	0,20	0,24	0,18	43,14
	30	0,32	0,12	2,34	13,46	8,80	0,80	44,48
	40	0,52	0,28	4,46	11,00	10,12	2,10	44,30
	50	0,36	0,74	0,84	0,28	0,84	4,20	45,10
1000	10	0,02	0,01	0,00	0,02	0,02	0,00	29,66
	20	0,04	0,02	0,19	0,13	0,12	0,06	37,65
	30	0,05	0,09	0,74	1,78	12,57	0,25	39,84
	40	0,13	0,09	0,29	0,16	0,17	0,86	46,67
	50	0,20	0,11	0,45	0,10	0,22	1,33	45,14
Media(%)		1,05	1,66	3,74	4,32	5,88	5,29	40,45

Tabla 5.2 – Porcentaje de trabajos que no se asignan a ninguna de las tres máquinas más rapidas i_{1j}, i_{2j} o i_{3j} en la solucion de referencia dada por IBM-ILOG CPLEX 11.0 con un tiempo límite de 2 horas.

En la tabla 5.2 podemos observar como en el intervalo donde los tiem-

pos de proceso están distribuidos al azar de forma uniforme en el intervalo $[1, 100]$, es decir, $U(1, 100)$, solo el 1 % de los trabajos se asignan, de media, a máquinas distintas a las tres más rápidas. Exceptuando la última columna, la de máquinas correlacionadas, ningún intervalo excede el 6 %. Tomemos por ejemplo la instancia más grande, 1000×50, y el intervalo $U(1, 100)$. Podemos comprobar como, sobre la media de estas 10 instancias, solo el 0,2 % de los trabajos se asignan fuera del rango de las tres máquinas más rápidas. Esto significa que solamente 2 trabajos, de los 1.000 que tiene cada una de las instancias, son asignados a máquinas más lentas.
La última columna, la de máquinas correlacionadas, se observa claramente como los trabajos no se asignan solo a las máquinas más rápidas, ya que, en este tipo de intervalo, hay máquinas que son rápidas para todos los trabajos y así, en las soluciones de referencia, no todos los trabajos acaban en las tres máquinas más rápidas, ya que estas se sobrecargan rápidamente.

Con estos datos de la tabla 5.2, vemos factible que proponer un modelo de reducción de tamaño considerando solo las tres máquinas más rápidas para cada trabajo, puedan funcionar bien y proporcionar bajos valores de $C_{\text{máx}}$. Esto resulta ser cierto para la gran mayoría de los casos, como veremos en las siguiente secciones.

5.1.1. Métodos de reducción de tamaño

En las secciones previas hemos expuesto como aparentemente no todas las $n \cdot m$ variables binarias x_{ij} del modelo MILP son aparentemente necesarias para alcanzar soluciones de alta calidad. Ahora presentamos algunos métodos simples que nos permiten seleccionar un subgrupo de máquinas reducido para cada trabajo antes de usar el solver. La primera aproximación es usar los k mejores (menores) tiempos de proceso p_{ij} para cada trabajo, es decir, para cada trabajo forzaremos a que solo estén disponibles las k máquinas más

rápidas para poder ser asignado. Nótese que nos referimos a las variables binarias seleccionadas x_{ij} o a los valores p_{ij} indistintamente. En este caso, los $k \cdot n$ valores más pequeños de p_{ij} se seleccionan. Llamamos a este método de reducción de tamaño basado en los trabajos (*job-based size reduction*) o kJB.

La tabla 5.3 muestra un ejemplo de 3JB para una instancia de 10 trabajos y 5 máquinas, donde los tres menores valores de p_{ij} se seleccionan dentro de cada trabajo.

	J1	J2	J3	J4	J5	J6	J7	J8	J9	J10
M1	5	9	**3**	6	**4**	**3**	6	**5**	7	**3**
M2	8	9	7	6	6	4	**5**	**1**	**1**	4
M3	**1**	**1**	**4**	**3**	8	**3**	7	**4**	8	**1**
M4	**2**	**8**	**3**	**4**	**2**	**3**	**1**	7	**1**	6
M5	**3**	**7**	8	**5**	**2**	8	**4**	9	**5**	**3**

Tabla 5.3 – Ejemplo de valores de p_{ij} seleccionados por el método 3JB de reducción de tamaño. En negrita los valores seleccionados.

Como concluimos de la tabla 5.2, este método selecciona valores apropiados en la mayoría de los casos pero es una mala elección para el intervalo de máquinas correlacionadas. Por ello, probamos un método que selecciona los p_{ij} entre las máquinas en lugar de entre los trabajos. Para ello, primero calculamos la media teórica de trabajos que pueden asignarse a cada máquina, $a = n/m$, y usamos el valor de a para determinar cuantos valores de p_{ij} seleccionaremos de cada máquina. Sin embargo, algunas máquinas puede que le correspondan en realidad más trabajos de los que la media teórica les otorga. Para tener esto en cuenta, seleccionamos $l \cdot a$ valores de p_{ij} para cada máquina, y el número total de valores seleccionados de p_{ij} con este método es entonces de $l \cdot a \cdot m$. Denominamos a este método como reducción de tamaño basado

en máquinas (*machine-based size reduction*) o lMB en breve. Un ejemplo de 2MB se puede observar en la tabla 5.4. Nótese que para este ejemplo $a = 10/5 = 2$ y $l = 2$ por lo que en realidad, cuatro valores se seleccionan de cada fila (máquina) de la tabla.

	J1	J2	J3	J4	J5	J6	J7	J8	J9	J10
M1	5	9	**3**	6	**4**	**3**	6	5	7	**3**
M2	8	9	7	6	6	***4***	5	**1**	**1**	**4**
M3	**1**	**1**	4	3	8	**3**	7	4	8	**1**
M4	**2**	8	3	4	**2**	3	**1**	7	**1**	6
M5	**3**	7	8	5	**2**	8	**4**	9	5	**3**

Tabla 5.4 – Ejemplo de valores seleccionados de p_{ij} por el método de reducción de tamaño 2MB. En negrita los valores seleccionados.

Hay un importante inconveniente en el método de reducción de tamaño lMB. Dado que estamos seleccionando solo los menores valores de p_{ij} de cada máquina (filas en la tabla 5.4), algunos trabajos pueden quedarse sin tener ningún valor de p_{ij} seleccionado. Esto implicaría el llegar a una solución no factible por parte del model MILP de reducción de tamaño, dado que habría algún trabajo que no se realizaría. De hecho, este es el caso del trabajo J4 en el ejemplo de la tabla 5.4. Una solución para este inconveniente es combinar los métodos previamente comentados, es decir, kJB y lMB. Obviamente, muchos de los valores seleccionados de p_{ij} serán los mismos en ambos métodos, pero no todos. Por ejemplo, en la tabla 5.4 podemos observar que el valor seleccionado $p_{26} = 4$ por el método 2MB no figura como seleccionado por el método 3JB en la tabla 5.3.

Los métodos de reducción de tamaño expuestos hasta ahora son simples, pero como podremos comprobar más tarde, producen buenos resultados. Sin

embargo, ya hemos visto como habían muchos métodos que eran muy efectivos y eficientes para el problema de $R//C_{\text{máx}}$ y nos parecía plausible hacer una selección de los menores valores de p_{ij} para la reducción de tamaño del MILP después de observar las soluciones dadas por alguna heurística efectiva. Para ello, seleccionamos la que hemos mostrado como mejor heurística para este problema, el algoritmo NVST-IG+. En lo sucesivo, nos referiremos a esta heurística como DIG, para abreviar (vecindario variable Descendente con Iterated Greedy). Básicamente, lo que hacemos es ejecutar DIG durante una cantidad muy breve de tiempo (5 segundos) y cada vez que alcance una nueva solución en esos 5 segundos, los valores de p_{ij} se seleccionan, sin repetición, para el modelo de reducción de tamaño del MILP. Podemos ver un ejemplo de este método en la tabla 5.5 donde se muestra una solución obtenida por DIG en una iteración.

	J1	J2	J3	J4	J5	J6	J7	J8	J9	J10
M1	**5**	9	3	6	4	3	6	5	7	3
M2	8	9	7	6	6	**4**	5	**1**	1	4
M3	1	**1**	4	**3**	8	3	7	4	8	1
M4	2	8	**3**	4	2	3	**1**	7	**1**	6
M5	3	7	8	5	**2**	8	4	9	5	**3**

Tabla 5.5 – Ejemplo de valores de p_{ij} seleccionados por DIG para el método de reducción de tamaño. En negrita los valores seleccionados.

También es posible mezclar DIG con los otros métodos de reducción de tamaño, como por ejemplo kJB y DIG para obtener un método híbrido con todos los valores de p_{ij} del método kJB más todos los valores de p_{ij} diferentes obtenidos por DIG. Siguiendo con el ejemplo anterior, podemos observar que el valor $p_{11} = 5$ procedente del método DIG de la tabla 5.5, no está presente en los valores seleccionados por el método 3JB de la tabla 5.3.

5.1.2. Algoritmos serie de reducción de tamaño

Los métodos de reducción de tamaño previamente expuestos solo simplifican el modelo MILP original pero no constituyen unos métodos de solución por ellos mismos. Sin embargo, uniendo a estos métodos un solver como CPLEX, ya obtendremos soluciones completas al problema. Por tanto, los algoritmos de resolución contendrán dos fases: 1) reducción de tamaño de la matriz original de p_{ij} o x_{ij} y 2) ejecutar CPLEX con esta matriz de reducción de tamaño hasta que el tiempo predefinido de CPU dado como criterio de finalización se alcance. Con este enfoque se proponen tres algoritmos de reducción de tamaño:

1. El primer algoritmo consiste en emplear el método kJB, con $k = 3$, para obtener una matriz reducida, para entonces ejecutar CPLEX. Denominaremos a este algoritmo de reducción de tamaño simplemente como 3J.

2. Para el segundo algoritmo emplearemos una combinación de los métodos de reducción de tamaño kJB y lMB con $k = 3$ y $l = 2$ para posteriormente ejecutar CPLEX. A este algoritmo lo llamaremos 3J2M.

3. La combinación del método de reducción de tamaño kJB, con $k = 3$, junto con el método de reducción de tamaño obtenido mediante DIG nos proporcionan un nuevo algoritmo, denominado 3JD. Nótese que este último algoritmo es más complicado que los anteriores al requerir de una heurística como DIG para su desarrollo.

Llegados a este punto, cabe señalar que la solución óptima de CPLEX obtenida durante la segunda fase es óptima para el problema de reducción de tamaño, lo cual no implica que sea la solución óptima del problema original. Los tres algoritmos previos pueden mejorarse de una forma significativa. Dado que CPLEX se ejecuta para un problema de reducido tamaño, es muy probable que sea capaz de encontrar una solución óptima a este problema reducido

antes que el tiempo de CPU predefinido como criterio de parada se alcance. La mejora más lógica es por tanto iterar el algoritmo de reducción de tamaño con más valores seleccionados de p_{ij}, es decir, repetir el algoritmo con una matriz mayor de x_{ij} y seguir haciéndolo mientras quede tiempo. Esto nos proporciona mayores matrices cada vez, aproximándonos al problema original. Bastante diferente es el caso que se da cuando CPLEX emplea mucho tiempo cerrando el árbol de búsqueda, probando la optimalidad pero sin realmente mejorar la solución que ya ha obtenido. En estas circunstancias, tiene poco sentido el malgastar un valioso tiempo de CPU. Parece más correcto parar el solver durante la segunda fase, una vez haya trascurrido un pequeño lapso de tiempo con respecto al total dado, y repetir el algoritmo de reducción de tamaño con una matriz mayor de tiempos de x_{ij}. Con todo esto en mente se propusieron dos nuevos algoritmos de reducción de tamaño adicionales:

4. Algoritmo de reducción de tamaño 2JDi. En el se emplea el método de reducción de tamaño kJB empezando con $k = 2$, junto con el método de reducción basado en DIG, para posteriormente ejecutar CPLEX. Cada vez que CPLEX finaliza y el tiempo dado como criterio de terminación no se ha alcanzado, k se incrementa en 1, los nuevos valores de p_{ij} se añaden y CPLEX se ejecuta otra vez. Nótese que el método de reducción de tamaño basado en DIG solo se ejecuta una vez, durante la primera iteración.

5. Algoritmo de reducción de tamaño 2JDi(90). Es similar al algoritmo 2JDi con la única diferencia de que cada vez que se ejecuta CPLEX, se le da un máximo de 90 segundos de tiempo de CPU, que una vez alcanzado, se aumenta k en 1 y tras añadir los nuevos valores de p_{ij}, CPLEX se ejecuta de nuevo.

Nótese que los 90 segundos de tiempo límite de CPU para 2JDi(90), procede de una calibración previa para 30, 60, 90 y 120 segundos. Así los valores de k y l en los algoritmos previamente propuestos se han obtenido

después de realizar una serie de experimentos donde se han hecho variar estos parámetros hasta encontrar los valores más adecuados para cada algoritmo.

5.1.3. Algoritmos paralelos de reducción de tamaño

Hoy en día, los ordenadores suelen tener más de un procesador integrado en el mismo chip, es decir, los conocidos como multi-core o multi-núcleo. En el momento de escribir estas líneas, hay en el mercado ordenadores de hasta 6 cores físicos en el mismo chip. Este incremento de la potencia disponible se usa por solvers potentes como CPLEX, el cual es capaz de funcionar en modo paralelo cuando hay varios cores o procesadores disponibles. Por ello, proponemos unos sencillos algoritmos paralelos basados en los algoritmos anteriormente expuestos de reducción de tamaño.
La más sencilla aproximación a la paralelización consiste en ejecutar tantas veces los algoritmos de reducción de tamaño como cores hayan disponibles y quedarse con la mejor solución obtenida al final, es decir, no hay comunicación entre las diferentes replicas de los algoritmos ejecutados. Esta aproximación mediante "fuerza bruta" podría ser el más simple empleo de un método multi-core. Hasta ahora, el mejor algoritmo propuesto, NVST-IG+ (DIG), se ha desarrollado en su versión serie. Aquí probaremos una versión multi-core de DIG que hemos denominado M-DIG (multicore DIG) la cual básicamente consiste en ejecutar varios algoritmos DIG, tantas veces como cores disponibles.

Para los algoritmos de reducción de tamaño, se proponen versiones para su uso de forma paralela también muy simples. Más específicamente, proponemos la versión 2JDi(90) como M-2JDi(90). El funcionamiento básico de esta versión paralelo es sencillo. Imaginemos por ejemplo un procesador con dos cores, entonces, M-2JDi(90) se lanza en el core 1 y M-3JDi(90) se lanza en el core 2, es decir, cada core esta trabajando con una matriz diferente de x_{ij} procedente del algoritmo de reducción de tamaño. Al primer core que finalice, se le asigna la siguiente matriz de reducción de tamaño, es decir M-4JDi(90).

El proceso continúa hasta que se alcanza el criterio de parada. Nótese que no se han propuesto versiones en paralelo para los algoritmos 3J, 3J2M, 3JD o 2JDi ya que 2JDi(90) domina claramente al resto como mostraremos más tarde.

5.2. Implementación de los métodos propuestos

De nuevo emplearemos el conjunto de las 1.400 instancias, repartidas en los consabidos 7 intervalos, que venimos empleando a lo largo de esta Tesis Doctoral. Como ya se ha indicado con anterioridad, el estudio del comportamiento de los algoritmos en su formato paralelo entra dentro de los resultados a considerar en esta parte de la Tesis Doctoral, por lo que el cluster de ordenadores empleado en la primera parte del trabajo, al ser mono-núcleo, no nos permitía realizar las pruebas pertinentes, así que se migró, para realizar todas las pruebas de esta parte de la Tesis Doctoral, a un cluster de 12 ordenadores PC/AT con procesadores Intel Core 2 Duo E6600, con 2.4GHz y 2 GB de memoria RAM bajo el sistema operativo Windows XP SP3. Cada procesador contiene dos cores. Llegados a este punto, debemos hacer especial mención a IBM-ILOG CPLEX. Hasta ahora habíamos empleado la versión serie 11.0 en unos ordenadores bastante más antiguos. En esta parte de la Tesis, probaremos la versión de CPLEX 11.1, tanto en su faceta serie como paralela, en unos ordenadores mucho más modernos, los arriba comentados. Nótese que es un pequeño cambio de version, pero que se anunciaba como una gran mejora para los problemas de asignación; ello unido al cambio de ordenadores, hizo que ambos cambios redundaran en un mejor comportamiento de CPLEX. En cuanto al tiempo de CPU, todos los métodos algoritmos de reducción de tamaño, el algoritmo DIG y CPLEX se paran después de un determinado lapso de tiempo de CPU. Se fijó este tiempo en 15 segundos como referencia, ya que, tal como se explicó, este es el tiempo que necesita CPLEX para resolver todas las instancias de 100 trabajos en el intervalo $U(1, 100)$. No obstante, se hicieron también pruebas para 30, 60, 120, 240 y 300 segundos. Los nuevos algoritmos

aquí propuestos se codificaron igualmente en Delphi 2007 y se compilaron con la opción de optimización activada. Todas las calibraciones de estos nuevos algoritmos se llevaron a cabo en el intervalo $U(1, 100)$ y para un subconjunto de instancias. La calibración para todos los intervalos e instancias seguramente mejoraría los resultados, pero no se hicieron posteriores calibraciones para evitar un sobre ajuste en los resultados.

5.3. Análisis computacional

El análisis computacional está dividido entre los métodos serie y paralelo. más específicamente, hemos llevado a cabo unos exhaustivos análisis computacionales y estadísticos para evaluar el comportamiento de los diferentes algoritmos. Empezaremos nuestro estudio con CPLEX (forzado a usar un solo núcleo), DIG (el mejor algoritmo propuesto hasta ahora), 3J, 3J2M, 3JD, 2JDi y 2JDi(90) como se definieron en secciones anteriores. Todos estos algoritmos son en serie y usan un solo núcleo en la CPU. Dado que todos los métodos están codificados en el mismo lenguaje de programación (excepto CPLEX), ejecutados en los mismos ordenadores, con las mismas características e instancias y con el mismo tiempo de CPU, creemos que todos los resultados son totalmente comparables. Posteriormente, presentamos los resultados de los algoritmos paralelos representados por M-DIG, M-2JDi(90) y M-CPLEX usando todos los cores disponibles (2). De nuevo, los datos vendrán dados por la desviación porcentual relativa (RPD, siglas en inglés) con respecto a la referencia, que recordemos son los resultados obtenidos por CPLEX con hasta 2 horas con cada instancia.

Después de mostrar los resultados, llevamos a cabo un análisis estadístico para probar en profundidad la significación estadística de las medias obtenidas. Usaremos un test estadístico para garantizar que las diferencias observadas en las medias son estadísticamente significativas. Llevaremos a cabo de nuevo

un Diseño de Experimentos donde estudiaremos un factor, es decir, el tipo de algoritmo a siete niveles para el caso serie y a tres niveles para el caso paralelo. La variable de respuesta es la desviación porcentual relativa. Tomamos todos los resultados anteriores en los que se considera cada caso. En el caso serie, hay 1.400 instancias y siete algoritmos, lo cual nos lleva a 9.800 observaciones. Para el caso paralelo, solo hay tres algoritmos, lo que hace un total de 4.200 observaciones. Este número de observaciones se realiza para cada uno de los tiempos de parada. Llevamos a cabo seis ejecuciones independientes de todos los métodos para los tiempos de parada de 15, 30, 60, 120, 240 y 300 segundos. Como resultado, el elevado número de observaciones para el caso serie se llega hasta 57.600 y hasta 25.200 para el caso paralelo. El resultado del DOE se analiza por medio de la técnica de factor único del Análisis de Varianza. Se comprobaron las tres principales hipótesis del ANOVA: normalidad, homocedasticidad e independencia de los residuos. Con tan elevado número de resultados, los residuos del ANOVA fácilmente satisfacen las tres hipótesis.

Se exponen aquí solo los resúmenes de los resultados o alguna tabla específica por su singular importancia. No obstante, se puede consultar todas las tablas y ANOVAS de forma más detallada en el anexo D.

5.3.1. Resultados de los algoritmos serie

La primera prueba se ha hecho con el mejor algoritmo testado hasta el momento, DIG y la nueva versión de CPLEX 11.1. Los comparamos con los primeros algoritmos propuestos, los sencillos 3J y 3J2M. La tabla 5.6 muestra los resultados obtenidos de la desviación porcentual relativa promediada para todos los intervalos excepto las máquinas correlacionadas, es decir, la media de 1.200 resultados. Los diferentes tiempos de parada se muestran como filas. No se ha incluido el intervalo de máquinas correlacionadas dado que, como se mostró en la tabla 5.2, para este intervalo la estrategia de 3J da unos pobres resultados, tal y como se esperaba. Por tanto, estos resultados se deben tomar

como una primera aproximación.

Tiempo (seg.)	CPLEX	DIG	3J	3J2M
15	*0,59*	**0,42**	0,52	0,48
30	0,41	**0,34**	*0,43*	0,37
60	**0,26**	0,29	*0,38*	0,32
120	**0,15**	0,22	*0,34*	0,28
240	**0,09**	0,17	*0,30*	0,24
300	**0,07**	0,16	*0,28*	0,22

Tabla 5.6 – Desviación porcentual relativa media para todos los intervalos excepto máquinas correlacionadas para CPLEX, DIG, 3J y 3J2M.

El primer resultado destacable es que el rendimiento de CPLEX se ha incrementado de forma significativa respecto a los resultados anteriormente expuestos. Esto es debido tanto a la nueva versión 11.1 como a que los ordenadores usados son más rápidos. No obstante, podemos observar como DIG sigue mejorando los resultados de CPLEX hasta los 30 segundos, mientras que antes, cuando consideramos CPLEX 11.0 y el antiguo grupo de ordenadores, lo superaba hasta los 60 segundos. La conclusión a la que puede llegarse es que la nueva versión de CPLEX hace un uso mucho más efectivo de los nuevos ordenadores más rápidos.
Los primeros dos algoritmos de reducción de tamaño propuestos, a pesar de su sencillez, resultan bastante competitivos. Para tiempos de CPU más largos, es de esperar que CPLEX de mejores resultados, dado que CPLEX resuelve el problema completo, pero nótese como para los tiempos más breves de CPU, es decir, para 15 y 30 segundos, 3J2M mejora los resultados de CPLEX. La tabla 5.7 muestra con más detalle los resultados de los otros tres algoritmos de reducción de tamaño propuestos: 3JD, 2JDi y 2JDi(90) junto a los resultados de DIG y CPLEX.

Los resultados de la tabla 5.7 muestran que, de media (columna más

a la derecha) todos los algoritmos propuestos producen mejores resultados que los mejores resultados conocidos y para todos los tiempos de parada considerados. Esta tabla incluso muestra valores negativos por lo que los algoritmos, para estos casos, proporcionan mejores valores que los dados como referencia (CPLEX 11.0 con hasta dos horas con cada instancia). No obstante, se puede observar como en el intervalo $U(1, 100)$, CPLEX proporciona buenos resultados para los tiempos de parada más largos, mejorando a 3JD e incluso a 2JDi pero sin alcanzar los valores obtenidos por 2JDi(90). En general, cuando el tiempo de parada excede al valor dado para la calibración del algoritmo 2JDi(90), que son 90 segundos, se puede apreciar como el algoritmo 2JDi(90) alcanza mejores valores que su versión más simple representada por el algoritmo 2JDi donde la parte del solver no se detiene. Podemos apreciar como CPLEX da pobres resultados para trabajos correlacionados y buenos resultados en el caso de máquinas correlacionadas, único caso este último en donde CPLEX puede superar al mejor algoritmo propuesto para los tiempos de parada más grandes. Sólo podemos decir que esto no puede atribuirse fácilmente a algo en concreto, debido a la estructura desconocida del algoritmo específico que está usando CPLEX para este problema de asignación. En cualquier caso, CPLEX tiene una dependencia del tipo de instancia al que se enfrenta, y se muestra menos robusto que los algoritmos propuestos.

Tiempo	Algoritmos	$U(1,100)$	$U(10,100)$	Jobcorre	Maqcorre	$U(100,200)$	$U(100,120)$	$U(1000,1100)$	**Media**
15	CPLEX	*0,82*	*0,70*	*1,33*	0,48	*0,55*	*0,09*	*0,05*	*0,58*
	DIG	1,07	0,62	0,45	*0,53*	0,29	0,04	0,02	0,43
	3JD	0,49	0,17	0,24	0,24	0,13	0,01	0,00	0,18
	2JDi	**0,44**	**0,17**	**0,24**	**0,22**	**0,13**	**0,01**	**0,00**	**0,17**
	2JDi(90)	**0,44**	**0,17**	**0,24**	**0,22**	**0,13**	**0,01**	**0,00**	**0,17**
30	CPLEX	0,56	0,47	*0,93*	0,30	*0,37*	*0,06*	*0,04*	*0,39*
	DIG	*0,93*	*0,49*	0,35	*0,50*	0,25	0,04	0,01	0,36
	3JD	0,37	0,08	0,13	0,23	0,11	0,01	0,00	0,13
	2JDi	**0,27**	**0,07**	**0,11**	**0,19**	**0,10**	**0,00**	**0,00**	**0,11**
	2JDi(90)	**0,27**	**0,07**	**0,11**	**0,19**	**0,10**	**0,00**	**0,00**	**0,11**
60	CPLEX	0,33	0,25	*0,65*	**0,16**	*0,25*	*0,04*	*0,03*	0,25
	DIG	*0,83*	*0,39*	0,25	*0,47*	0,21	0,03	0,01	*0,31*
	3JD	0,33	0,03	0,00	0,22	0,09	0,01	**0,00**	0,10
	2JDi	**0,19**	**0,01**	**-0,01**	0,17	**0,08**	**0,00**	**0,00**	**0,06**
	2JDi(90)	**0,19**	**0,01**	**-0,01**	0,17	**0,08**	**0,00**	**0,00**	**0,06**
120	CPLEX	0,17	0,10	*0,45*	**0,09**	0,13	*0,03*	*0,02*	0,14
	DIG	*0,75*	*0,24*	0,14	*0,42*	*0,17*	0,02	**0,00**	*0,25*
	3JD	0,31	0,01	-0,09	0,22	0,08	**0,00**	**0,00**	0,08
	2JDi	0,15	-0,06	**-0,11**	0,16	0,06	**0,00**	**0,00**	0,03
	2JDi(90)	**0,06**	**-0,07**	-0,10	0,15	**0,05**	**0,00**	**0,00**	**0,01**
240	CPLEX	0,08	0,05	*0,33*	**0,04**	0,06	*0,02*	*0,01*	0,09
	DIG	*0,63*	*0,17*	0,06	*0,37*	*0,14*	*0,02*	0,00	*0,20*
	3JD	0,26	-0,02	-0,13	0,22	0,07	0,00	0,00	0,06
	2JDi	0,10	-0,09	**-0,16**	0,15	0,06	0,00	0,00	0,01
	2JDi(90)	**-0,02**	**-0,14**	-0,15	0,11	**0,02**	**-0,01**	**-0,01**	**-0,03**
300	CPLEX	0,06	0,02	*0,29*	**0,03**	0,04	0,01	*0,01*	0,07
	DIG	*0,63*	*0,14*	0,03	*0,36*	*0,13*	*0,02*	0,00	*0,19*
	3JD	0,24	-0,07	-0,17	0,21	0,03	0,00	0,00	0,03
	2JDi	0,08	-0,11	**-0,19**	0,14	0,02	0,00	0,00	-0,01
	2JDi(90)	**-0,03**	**-0,14**	-0,17	0,10	**0,00**	**-0,01**	**-0,01**	**-0,04**

Tabla 5.7 – Desviación porcentual relativa media para los algoritmos serie CPLEX, DIG, 3JD, 2JDi y 2JDi(90) con diferentes tiempos de CPU como criterio de parada. En negrita (cursiva) se representan los mejores (peores) valores, respectivamente.

Coma ya hemos comentado en otras secciones, todas las tablas anteriormente expuestas representan las medias de los resultados obtenidos. Ahora lo que precisamos es llevar a cabo una prueba estadística tipo ANOVA para comprobar que las diferencias observadas en esas medias son estadísticamente significativas. La figura 5.1 muestra tanto la media como los intervalos de confianza Tukey HSD de un 95 % para todos los intervalos con 120 segundos como criterio de parada. Este tiempo permite a CPLEX alcazar unos buenos resultados a la par que le permite al algoritmo 2JDi(90) alcanzar el tiempo de calibración (90 segundos). Recordemos que cuando los intervalos se solapan indica que no existen diferencias significativas en los algoritmos que se estudian. Podemos observar como CPLEX, para esta cantidad de tiempo, es capaz de superar a DIG de una forma significativa. Por otra parte, los tres algoritmos de reducción de tamaño propuestos, pueden considerarse el nuevo estado del arte, a pesar de que hay algunos solapamientos entre ellos. 2JDi(90) es estadísticamente mejor que 3JD pero 2JDi es equivalente a los otros dos. Para más detalles y otros tiempos, puede consultarse el anexo D donde se muestra el comportamiento para todos los casos.

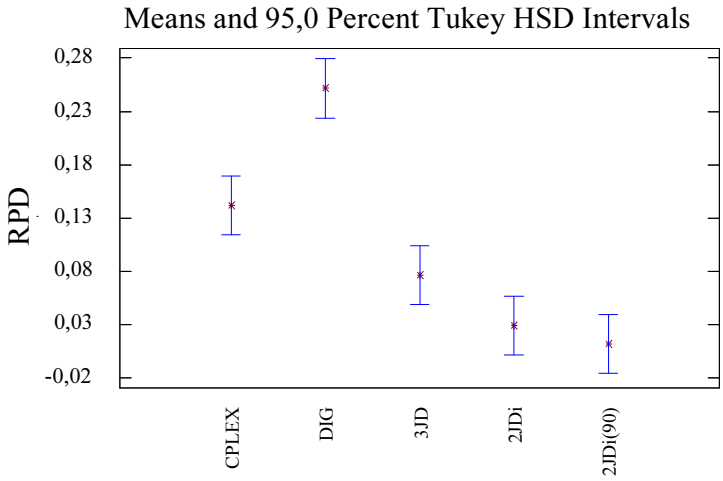

Figura 5.1 – Gráfico de los algoritmos serie para todas las instancias y 120 segundos de tiempo de parada

5.3.2. Resultado de los algoritmos paralelos

Procedemos ahora a probar el algoritmo M-2JDi(90), versión paralela del algoritmo 2JDi(90), junto con las versiones paralelas de DIG y CPLEX (que denominamos M-DIG y M-CPLEX, respectivamente). Los resultados se muestran en la tabla 5.8. Todos los algoritmos paralelos mejoran a sus versiones serie si se comparan estos resultados con los dados en la tabla 5.7. Recordemos que solo se han usado dos cores y que probablemente estos resultados mejorasen si se emplearan cuatro, seis o incluso ocho cores. Podemos ver como, de media, los algoritmos propuestos de reducción de tamaño producen mejores resultados que los otros algoritmos. M-DIG da mejores resultados que M-CPLEX para los tiempos más cortos de CPU y la situación se invierte para los tiempos más largos. De modo similar al caso serie, el algoritmo M-2JDi(90) consigue mejores resultados en todos los casos y para todos los tiempos excepto para máquinas correlacionadas, donde M-CPLEX tiene mejores resultados aunque solo para los tiempos de CPU más largos. M-CPLEX también presenta

unos pobres resultados para el intervalo de trabajos correlacionados, como en el caso serie.

Tiempo	Algoritmos	$U(1,100)$	$U(10,100)$	Jobcorre	Maqcorre	$U(100,200)$	$U(100,120)$	$U(1000,1100)$	**media**
15	M-CPLEX	0,66	*0,63*	*1,33*	0,37	*0,40*	*0,07*	*0,04*	*0,50*
	M-DIG	*0,91*	0,51	0,35	*0,48*	0,26	0,03	0,02	0,37
	M-2JDi(90)	**0,42**	**0,10**	**0,18**	**0,20**	**0,11**	**0,00**	**0,00**	**0,15**
30	M-CPLEX	0,41	0,33	*0,99*	0,20	*0,29*	*0,05*	*0,03*	*0,33*
	M-DIG	*0,86*	*0,41*	0,24	*0,45*	0,21	0,03	0,01	0,32
	M-2JDi(90)	**0,18**	**-0,01**	**0,03**	**0,16**	**0,08**	**0,00**	**0,00**	**0,06**
60	M-CPLEX	0,30	0,18	*0,74*	**0,09**	0,15	*0,03*	*0,02*	0,22
	M-DIG	*0,70*	*0,31*	0,16	*0,42*	*0,18*	0,02	0,01	*0,26*
	M-2JDi(90)	**0,10**	**-0,07**	**-0,07**	0,14	**0,06**	**0,00**	**0,00**	**0,02**
120	M-CPLEX	0,23	0,11	*0,60*	**0,04**	0,08	*0,03*	*0,01*	0,16
	M-DIG	*0,62*	*0,18*	0,07	*0,40*	*0,14*	0,02	0,00	*0,20*
	M-2JDi(90)	**-0,02**	**-0,14**	**-0,14**	0,12	**0,02**	**-0,01**	**-0,01**	**-0,03**
240	M-CPLEX	0,16	0,04	*0,47*	**0,02**	0,04	*0,01*	*0,00*	0,11
	M-DIG	*0,57*	*0,11*	-0,01	*0,34*	*0,11*	*0,01*	*0,00*	*0,16*
	M-2JDi(90)	**-0,03**	**-0,16**	**-0,19**	0,07	**0,00**	**-0,01**	**-0,01**	**-0,05**
300	M-CPLEX	0,08	-0,02	*0,35*	**0,01**	0,02	*0,01*	*0,00*	0,06
	M-DIG	*0,56*	*0,08*	-0,04	*0,34*	*0,10*	*0,01*	-0,00	*0,15*
	M-2JDi(90)	**-0,03**	**-0,17**	**-0,20**	0,05	**-0,01**	**-0,01**	**-0,01**	**-0,05**

Tabla 5.8 – Desviación porcentual relativa media para los algoritmos paralelo M-CPLEX, M-DIG y M-2JDi(90) con diferentes tiempos de CPU como criterio de parada. En negrita (cursiva) se representan los mejores (peores) valores, respectivamente.

La figura 5.2 muestra el gráfico de interacción entre el tiempo de parada y los algoritmos paralelos. Nótese que se ha usado un ANOVA de dos factores en este caso. Este gráfico de ANOVA nos permite ver como M-DIG es significativamente mejor que M-CPLEX para 15 segundos de tiempo de parada. Para 30 segundos son estadísticamente equivalentes y la situación se mantiene hasta los 300 segundos, cuando finalmente M-CPLEX mejora de forma significativa

los valores de M-DIG. En todos los casos, el algoritmo propuesto de reducción de tamaño M-2JDi(90), muestra valores estadísticamente mejores que los otros dos algoritmos, por un amplio margen.

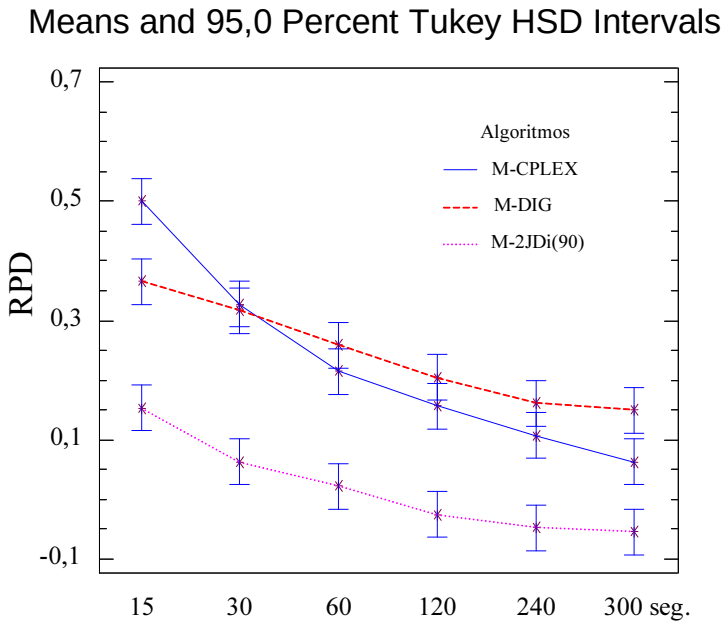

Figura 5.2 – Gráfico para la interacción entre el tiempo de parada y los algoritmos paralelos probados, para todas las instancias e intervalos.

Finalmente, presentamos en la figura 5.3 una comparativa entre los algoritmos serie y paralelo. Los intervalos de confianza se han quitado para mejorar la legibilidad del gráfico. Como se puede apreciar, las versiones paralelo muestran la mayoría del tiempo mejores resultados que sus equivalentes versiones serie. Tal y como cabría esperar, para largos periodos de tiempo de CPU, la mayoría de los algoritmos empiezan a converger y la ventaja de los algoritmos paralelos disminuye.

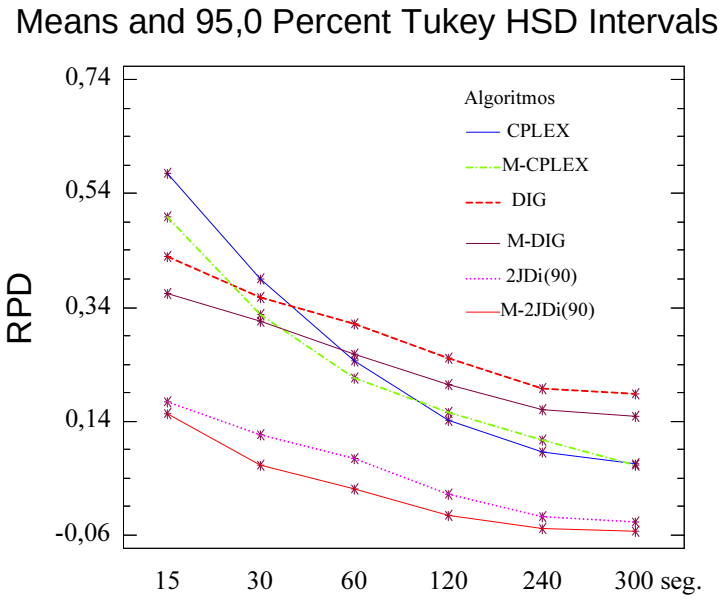

Figura 5.3 – Algoritmos paralelos frente a serie: gráfico de medias para todas las instancias e intervalos y todos los tiempos de parada.

5.4. Resumen de algoritmos de reducción de tamaño

En esta parte de la Tesis Doctoral hemos propuesto nuevos métodos y algoritmos de reducción de tamaño para el problema que nos atañe: máquinas paralelas no relacionadas con el objetivo de reducir el makespan o $R//C_{\text{máx}}$. Los métodos expuestos son muy simples y están basados en la idea de reducir la posible asignación de trabajos a las máquinas a solo las asignaciones más prometedoras. El problema reducido entonces se resuelve con la ayuda de un solver comercial, el cual, para este capítulo, ha sido IBM-ILOG CPLEX versión 11.1.

Se ha hecho una comparativa con los mejores métodos que había has-

ta ahora, identificados como IBM-ILOG CPLEX 11.1 y DIG, tanto en sus versiones serie como paralelo. Los algoritmos propuestos mejoraron a estos anteriores mejores métodos en una extensa serie de pruebas computacionales y estadísticas tanto en el caso serie como en el paralelo. Más específicamente, la versión paralela del mejor algoritmo propuesto, M-2JDi(90), obtiene, en solo 15 segundos de tiempo de CPU, una desviación porcentual media, respecto a la mejor cota inferior obtenida para cada instancia por el solver, de solo un 0,63 % entre las 1.400 instancias, que llegan a un tamaño de 1000 trabajos y 50 máquinas. Nótese que estos resultados son con respecto a la cota inferior, no contra la mejor solución conocida como referencia mostrada en la tabla 5.8 en cuyo caso es de un 0,37 %. Esto significa que, en el peor de los casos, 15 segundos son suficientes para alcanzar una desviación media inferior al 1 %. Con estos resultados, consideramos que estamos muy cerca de la resolución efectiva del problema $R//C_{\text{máx}}$ dado lo próximo que se está de alcanzar los resultados óptimos incluso para instancias de tamaño considerable.

CAPÍTULO 6

MÁQUINAS OPCIONALES Y SELECCIÓN DE TRABAJOS

Hasta ahora solo se ha tratado el problema de máquinas paralelas bajo las condiciones de que todas las máquinas están disponibles y todos los trabajos han de realizarse. A partir de este punto, abordaremos un nuevo problema. Este nuevo problema consiste en una relajación del $R//C_{\text{máx}}$ en el que no se van a usar todas las máquinas o no se van a realizar todos los trabajos, es decir, el uso de todas las m máquinas disponibles es opcional y se relaja la restricción de que todos los n trabajos deben secuenciarse, por lo que se permite la selección de los trabajos.

La justificación de este nuevo problema puede venir dada por varios motivos. Uno es la visión estratégica de la adquisición, venta o parada de una o de varias máquinas. Sabiendo los trabajos a realizar, se puede modelizar el problema con un número determinado de máquinas e indicar el límite de máquinas a emplear. Así por ejemplo, si queremos adquirir 3 máquinas nuevas y en el mercado existen 5 modelos distintos, no tenemos más que modelizar

3 máquinas de cada modelo ($5 \times 3 = 15$) e indicar que solo se emplearán 3 máquinas. La resolución del problema nos indicará qué 3 máquinas son las óptimas para realizar las tareas encomendadas. Igualmente, se puede hacer la comparativa para averiguar qué máquinas vender o parar, ya que se puede realizar el modelo de forma que nos indique que máquinas de las que actualmente disponemos, debemos dejar de usar. También nos servirá para saber qué máquinas arrendar, para los casos de necesitar más máquinas por un incremento de nuestra producción, o subarrendar en el caso de querer ceder el uso de parte de nuestras máquinas a un tercero. También se puede necesitar solucionar este tipo de problemas si nos vemos forzados a dejar, por seguridad o por reserva, algunas máquinas (léase por máquinas también puertas de acceso/salida, ordenadores de una red o clúster, procesadores disponibles, etc). En cuanto a los trabajos, podemos igualmente establecer que trabajos realizar y cuales no realizar, por motivos tan diversos como descentralizar la realización de los mismos o selección de productos más rentables.

Existen estudios que tratan temas similares, como los problemas de JIT (*Just in Time*) que tienen cierto parecido a los problemas de no realizar todos los trabajos o selección de trabajos. No obstante, existen diferencias fundamentales con estos problemas, ya que en los problemas JIT viene limitado el número de trabajos por cumplir una fecha límite, mientras que en nuestro caso, la limitación de trabajos puede estar forzada a un número determinado de trabajos fijos a realizar o a dejar de realizar. Igualmente, se podría examinar como problema ligeramente similar al de no usar todas las máquinas o máquinas opcionales aquellos de máquinas que se detienen para mantenimiento y/o por rotura, pero en nuestro caso, podemos estar marcando un número fijo de máquinas que se van a dejar de usar. Por último, han aparecido recientemente artículos como los de Chen y Li (2008), Finke et al. (2009) o Kravchenko y Werner (2009), pero en ellos se trata de máquinas idénticas o tiempos idénticos y se buscan diferentes objetivos o se tienen distintas restricciones. Por todo

ello, nos encontramos que nuestro caso apenas ha sido tratado en la literatura.

El problema a tratar es una ampliación al conocido de máquinas paralelas no relacionadas con el objetivo de minimizar el $C_{\text{máx}}$, sujeto a las restricciones de poder limitar el número de trabajos o de máquinas. El problema puede ampliarse aún más haciéndolo multiobjetivo aunque excede a lo pretendido en esta Tesis Doctoral, en la cual nos limitaremos al objetivo único de minimizar el $C_{\text{máx}}$ sujeto a limitar el número de máquinas o de trabajos. No obstante, haremos un planteamiento inicial del problema multiobjetivo para futuras investigaciones.

En primer lugar expondremos la modelización del problema, con las restricciones más habituales y las limitaciones que encierra este planteamiento. Seguidamente, continuaremos con el estudio de no emplear todas las máquinas, para finalizar adentrándonos en el campo de no realizar todos los trabajos.

6.1. Modelos matemáticos, restricciones y limitaciones

En la presente sección estudiaremos los modelos matemáticos, así como las restricciones que les afectan y las limitaciones a las que están sujetos.

6.1.1. Modelos matemáticos

Vamos a detallar como son unos posibles modelos matemáticos para los problemas de minimizar el $C_{\text{máx}}$, minimizar el número de trabajos no realizados y minimizar el número de máquinas empleadas.

Partimos del modelo de programación lineal entera mixta para el problema de partida de secuenciación en máquinas paralelas no relacionadas o $R//C_{\text{máx}}$ ya explicado en la sección 2.3:

$$\text{mín } C_{\text{máx}} \tag{6.1}$$

$$\sum_{i=1}^{m} x_{ij} = 1 \quad \forall j \in N \tag{6.2}$$

$$\sum_{j=1}^{n} p_{ij} \cdot x_{ij} \leq C_{\text{máx}} \quad \forall i \in M \tag{6.3}$$

Donde x_{ij} es una variable de naturaleza binaria que toma el valor 1 si el trabajo j se asigna a la máquina i y 0 en caso contrario. Las restricciones (6.2) aseguran que todos los trabajos quedan asignados a exactamente una única máquina. El grupo de restricciones (6.3) asigna el valor de $C_{\text{máx}}$, que no puede ser inferior a ningún C_i de las máquinas.

El modelo anterior se modifica de manera sencilla para el caso de máquinas opcionales como sigue:

$$\sum_{j=1}^{n} x_{ij} \leq n \cdot z_i \quad \forall i \in M \tag{6.4}$$

Siendo z_i una variable de naturaleza binaria que toma el valor 1 si la máquina i se usa y 0 en caso contrario. Solo restaría el añadir el número máximo de máquinas a no utilizar, número que denotaremos como Z, para tener completo el problema donde existen máquinas opcionales. Este problema lo denominaremos como "*Not all machines*" o NAM de forma abreviada.

La modificación para considerar la extensión de selección de trabajos es la siguiente:

$$\sum_{i=1}^{m} x_{ij} = h_j \quad \forall j \in N \tag{6.5}$$

Siendo h_j una variable de nuevo binaria que toma el valor 1 si el trabajo j se procesa y 0 en caso contrario. El grupo de restricciones (6.5) sustituiría al

anterior (6.2) en el caso de existir selección de trabajos. Añadiendo un número mínimo de trabajos a realizar, que denotamos por H, tendríamos la formulación completa para el problema de selección de trabajos que denominamos como "*Not all jobs*" o NAJ.

Como objetivos alternativos y/o complementarios a minimizar el $C_{\text{máx}}$, estarían los de minimizar el número de máquinas empleadas y/o el número de trabajos no realizados, tal y como se muestra en los objetivos siguientes (6.6) y (6.7). El trabajar con estos objetivos conjuntamente con el de $C_{\text{máx}}$ queda fuera del ámbito del presente trabajo, donde nos centraremos en un tanto por cien de máquinas sin usar o de trabajos que se dejan de realizar.

$$\text{mín} \sum_{i=1}^{m} z_i \tag{6.6}$$

$$\text{mín } n - \sum_{j=1}^{n} h_j \tag{6.7}$$

6.1.2. Restricciones más habituales

Procedemos a comentar alguna de las restricciones que más habitualmente se podrían dar. En cuanto al número de trabajos a realizar, se pueden restringir tanto el mínimo de trabajos a realizar como el máximo de ellos, según:

$$a \leq \sum_{j=1}^{n} h_j \leq b \tag{6.8}$$

Donde a = número mínimo de trabajos a realizar y b = número máximo de trabajos a realizar. En el caso de $a = b = n$, siendo n el número total de trabajos, se realizarían todos los trabajos.

En cuanto al número de máquinas se pueden restringir tanto el mínimo de máquinas a emplear como el máximo de ellas, según:

$$c \leq \sum_{i=1}^{m} z_i \leq d \tag{6.9}$$

Donde c = número mínimo de máquinas a usar y d = número máximo de máquinas a usar. En el caso de $c = d = m$, siendo m el número total de máquinas, se emplearían todas las máquinas.

Por último, en cuanto a limitaciones del $C_{\text{máx}}$ podríamos limitar el tiempo del $C_{\text{máx}}$ a un límite temporal fijo, haciendo:

$$C_{\text{máx}} \leq t; \tag{6.10}$$

Donde t= límite temporal. Igualmente, podemos realizar cálculos sobre una medida del $C_{\text{máx}}$ ya hallada, haciendo que la limitación temporal venga dada por:

$$t = C^*_{\text{máx}} \pm \% \, C^*_{\text{máx}}; \tag{6.11}$$

Siendo t= límite temporal. Donde $C^*_{\text{máx}}$ representa el $C_{\text{máx}}$ óptimo obtenido previamente. El signo + de la ecuación nos sirve para permitir un empeoramiento del $C_{\text{máx}}$, en el caso de utilizar menos máquinas y el signo - para buscar una mejora en el $C_{\text{máx}}$ en el caso de hacer menos trabajos.

6.1.3. Limitaciones de los modelos

Si se analizan las distintas opciones de agregar restricciones y añadir más de un objetivo, nos encontramos con una serie de limitaciones que vienen dadas por el mismo modelo, dependiendo de la combinación de restricciones y de los objetivos a buscar. Según los objetivos, las limitaciones de los modelos son las siguientes:

1. Objetivo único: minimizar el número de máquinas (min). Si no se limita nada más, no usará ninguna máquina, pero no se haría ningún trabajo.

Si, además de limitar el número de máquinas, se limita solo el $C_{\text{máx}} = t$ como máximo, tampoco hará ningún trabajo, con lo que se emplearían cero máquinas (min) y el $C_{\text{máx}}$ será cero. Si, además de limitar el número de máquinas, se limita solo el número de trabajos mínimos a hacer, los hará todos en una sola máquina, minimizando el número de máquinas a una, pero haciendo un $C_{\text{máx}}$ muy elevado. Por tanto, para conseguir una minimización en el número de máquinas, debemos dar un número de trabajos a realizar y un tiempo de $C_{\text{máx}}$ máximo.

2. Objetivo único: maximizar el número de trabajos realizados (o minimizar el número de trabajos NO hechos). Si no se limita nada más, hará todos los trabajos, pero puede hacerlos en una sola máquina o en varias máquinas, incluso en todas, y tardando un tiempo indefinido, en función de las máquinas que use. Si, además de minimizar el número de trabajos no hechos, se limita sólo el número de máquinas, pasará como el caso anterior, usará una o varias máquinas tardando un tiempo no definido. Si, además minimizar el número de trabajos no hechos, se limita sólo el $C_{\text{máx}}$, esto si fuerza a usar generalmente todas las máquinas y dará como resultado la ejecución del máximo de trabajos, usando todas las máquinas y en el $C_{\text{máx}}$ establecido. Por tanto se puede conseguir una buena maximización de los trabajos realizados, simplemente limitando el $C_{\text{máx}}$.

3. Objetivo único: Minimizar el $C_{\text{máx}}$. Si no se limita nada más, no usará ninguna máquina, ni hará ningún trabajo, consiguiendo un $C_{\text{máx}}$ de cero. Si, además de minimizar el $C_{\text{máx}}$, se limita sólo el número de máquinas, no hará ningún trabajo, dado que con eso se consigue un $C_{\text{máx}}$ igual a cero y un número de máquinas igual a cero. Si, además de minimizar el $C_{\text{máx}}$, se limita sólo el número de trabajos, esto si fuerza a que se busque una solución usando las máquinas que precise para hacer los trabajos y

conseguir un mínimo de $C_{\text{máx}}$. Por tanto, se puede conseguir una buena minimización del $C_{\text{máx}}$, simplemente dando un número de trabajos a realizar.

4. Objetivo doble: minimizar el número de máquinas y maximizar el número de trabajos hechos. A pesar de ser objetivos contrapuestos, si no se limita el tiempo o $C_{\text{máx}}$, se realizarán todos los trabajos en un sola máquina.

5. Objetivo doble: minimizar el número de máquinas y el $C_{\text{máx}}$. Estos objetivos son similares, con lo que si no se da un número de trabajos mínimos a realizar, no se hará ningún trabajo consiguiendo un $C_{\text{máx}}$ de cero y usando cero máquinas.

6. Objetivo doble: maximizar número de trabajos hechos y minimizar $C_{\text{máx}}$. Esto si dará buenos resultados, ya que para conseguir ambos objetivos contrapuestos, tratará de usar el mayor número de máquinas posibles, lo que dará el resultado óptimo del problema.

7. Objetivo triple: maximizar número de trabajos hechos, minimizar $C_{\text{máx}}$ y minimizar el número de máquinas. Dado que con los dos primeros objetivos ya se puede obtener una solución viable, el añadir el último (minimizar el número de máquinas), añadiría una nueva parte a la función objetivo a alcanzar.

Por tanto, para cualquier objetivo simple, debemos limitar uno o más factores; para los objetivos dobles, debemos limitar el otro factor salvo en el caso de ser los dos objetivos maximizar número de trabajos hechos y

minimizar $C_{\text{máx}}$, ya que entonces no haría falta indicar nada sobre el número de máquinas a emplear. Finalmente, en el objetivo triple no hace falta añadir ninguna restricción para su correcta resolución.

6.2. No todas las máquinas (NAM)

Dentro de todas las opciones que hemos estudiado, nos centraremos en resolver aquella en la que el objetivo único es minimizar el $C_{\text{máx}}$, sujeto a las restricciones de limitar el número de trabajos o de máquinas.

Estudiaremos en primer lugar el caso en el que no se emplean todas las máquinas (Not All Machines - NAM), para lo cual hay que distinguir dos partes del problema. Lo primero es decidir qué máquinas no se van a emplear y lo segundo resolver un problema de máquinas paralelas sin esas máquinas. Al ser las máquinas no relacionadas, la decisión de qué máquinas hay que dejar de usar no es para nada sencilla, dado que el tiempo de proceso de cada trabajo depende de la máquina y no existen máquinas que sean consistentemente lentas o rápidas para todos los trabajos (este sería el caso menos general de máquinas uniformes). Como resultado de lo anterior, y para encontrar la mejor posible combinación de máquinas, deberíamos resolver todas las posibles combinaciones de $m - Z$ máquinas que se seleccionan para usarse (no usando Z máquinas) de entre las m máquinas totales. Por ejemplo, en un problema con 10 máquinas, si solo se quisieran usar 7 de estas máquinas, tendríamos un número de combinaciones sin repetición de 10 elementos tomados de 3 en 3 para seleccionar las máquinas no usadas o, equivalentemente, 10 elementos tomados de 7 en 7 para seleccionar las máquinas a usar. El resultado sería nada menos que 120 posibles combinaciones. Cada combinación genera un problema $R//C_{\text{máx}}$ diferente. Evidentemente, estamos ante un problema de dimensiones importantes, dado que para un ejemplo más realista con un taller de 50 máquinas paralelas donde se desea no usar 10 de ellas tendríamos un total

de 10.272.278.170 posibles combinaciones. Dada la imposibilidad de resolver este problema de forma óptima, se propone una aproximación heurística que surge de descomponer el problema en tres fases:

1. Analizar los tiempos de proceso y hacer un ranking de máquinas más prometedoras.
2. Selección de máquinas según ranking
3. Resolver el problema $R//C_{\text{máx}}$ resultante.

Los dos últimos pasos se repiten de forma iterativa hasta que se alcanza un criterio de parada, ya que, la selección de máquinas a no emplear, sin la resolución del problema con la totalidad de las máquinas, será de carácter aproximado. Una vez resuelto el problema, hemos de tener en cuenta que las máquinas seleccionadas pueden resultar no ser las óptimas para este problema. Por tanto, habrá que seguir un cierto criterio para no sólo hacer una primera selección de máquinas, si no para hacer sucesivas selecciones de máquinas con la correspondiente resolución del problema. Por ejemplo, si tenemos un problema con 100 trabajos y 10 máquinas en las que sólo se deben emplear 8 de ellas, puede que nuestra mejor opción sea no emplear las máquinas 1 y 2. Después resolvemos el problema de máquinas paralelas no relacionadas para los 100 trabajos y las restantes 8 máquinas, pero dentro del tiempo límite dado al algoritmo, si éste no ha expirado, podemos seleccionar la segunda mejor opción de máquinas a no emplear, que puede ser el no emplear las máquinas 1 y 3, por ejemplo. Con esta nueva selección de máquinas resolveríamos de nuevo el problema. Es por ello que hemos descompuesto el problema en las fases antes mencionadas. La primera consistente en aplicar un criterio para ordenar las máquinas de forma que durante la segunda fase se puedan seleccionar aquellas que a priori parezcan las más apropiadas. Al acabar se hace una nueva selección para re-lanzar el problema, y así sucesivamente hasta agotar el tiempo disponible para resolver cada instancia.

6.2.1. Ranking de máquinas

Como primer paso para la resolución del problema de máquinas paralelas donde no se emplean todas las máquinas, debemos obtener una forma de ordenar las máquinas de forma que sepamos cuál es la máquina o máquinas más favorables. La primera opción obvia de obtener un ranking de máquinas, es resolver el problema con $m - 1$ máquinas, retirando sucesivamente todas y cada una de ellas, resolviendo cada problema y finalmente retirando la máquina que más convenga por obtener peores resultados. Seguidamente, se resuelve para $m - 2$ máquinas y se retira la máquina que nos de la peor solución, y así sucesivamente hasta retirar el total de máquinas marcado para el problema, denominado Z. Esto da unos pobres resultados, dado que no es lo mismo resolver quitando máquinas de una en una que retirar la totalidad de máquinas que se nos solicita de una sola vez. Además, esto supondría la resolución de numerosos problemas consecutivos. Por tanto, necesitamos una ordenación de máquinas más efectiva que nos proporcione las máquinas a retirar siguiendo ese orden y evitando la resolución de problemas parciales.

Para poder dar un orden de las máquinas se ha procedido a emplear los conocimientos adquiridos de los algoritmos de reducción de tamaño. Así, se ha procedido a escoger los tres menores valores de p_{ij} de cada trabajo para luego restárselos al cuarto menor valor. Esto nos da una idea de cómo son de buenos esos tres valores respecto al resto de valores no contemplados. Sumamos esas diferencias para cada máquina. En el caso de no tener ningún valor, se le asignará el correspondiente a la suma de todos sus valores originales. Esto se hace porque estas máquinas resultan ser las que peores resultados dan, al no tener reflejado ningún trabajo en el que sus tres menores valores caiga en esas máquinas. Eso nos proporciona, para cada máquina, un valor de lo buena que es realizando los trabajos comprendidos entre los tres menores valores de p_{ij} de entre todos los trabajos. Finalmente, las máquinas se ordenan en modo ascendente. Pongamos un ejemplo para ilustrar esta forma de ordenar

las máquinas:

	J1	J2	J3	J4	J5	J6	J7	J8	J9	J10	C_i
M1	4	5	5	5	5	4	5	5	4	5	47
M2	1	3	4	5	1	1	5	2	3	1	26
M3	3	3	1	1	2	2	4	4	5	4	29
M4	1	3	1	4	3	3	2	3	1	2	23
M5	1	4	1	3	2	2	4	2	1	2	24

Tabla 6.1 – Ejemplo NAM: Tabla inicial.

Partiendo de la tabla inicial 6.1 se seleccionan los tres menores valores de cada trabajo y se marca en negrita el cuarto menor valor, tal y como muestra la tabla 6.2.

	J1	J2	J3	J4	J5	J6	J7	J8	J9	J10
M1				**5**			**5**		**4**	
M2	1	3	**4**		1	1		2	3	1
M3	**3**	3	1	1	2	2	4	**4**		**4**
M4	1	3	1	4	**3**	**3**	2	3	1	2
M5	1	**4**	1	3	2	2	4	2	1	2

Tabla 6.2 – Ejemplo NAM: Tabla de tres menores valores por trabajo y cuarto menor valor en negrita.

Ahora se procede la resta de los tres menores valores respecto al cuarto, tal y como muestra la tabla 6.3. Esto se hace para comprobar lo bueno que son estos tres primeros valores respecto al resto de valores no considerados. Por ejemplo, no es lo mismo que los tres menores valores sean 1 y el cuarto sea un 5 que si el cuarto menor valor sea un 2. En el primer caso, los tres primeros valores estarán, muy probablemente, en la solución óptima, mientras que en el segundo caso, a priori, la probabilidad de estar en la solución final es mucho más pequeña.

	J1	J2	J3	J4	J5	J6	J7	J8	J9	J10	C'_i	C_o
M1											0	47
M2	-2	-1			-2	-2		-2	-1	-3	-13	-13
M3		-1	-3	-4	-1	-1	-1				-11	-11
M4	-2	-1	-3	-1			-3	-1	-3	-2	-16	-16
M5	-2		-3	-2	-1	-1	-1	-2	-3	-2	-17	-17

Tabla 6.3 – Ejemplo NAM: Tabla de diferencia de los tres menores valores con el cuarto. C_o = C'_i si C'_i<> 0 ó C_o = C_i de la tabla 6.1 si C'_i = 0

En la tabla 6.3 el valor C'_i representa la suma de los valores así obtenidos para cada máquina. C_o es igual a C'_i en el caso de que C'_i sea distinto de cero; en caso de ser C'_i igual a cero, C_o para esa máquina es igual a la suma de todos los valores p_{ij} de la tabla 6.1 correspondientes a esa máquina. Esto último se hace para poder ordenar las máquinas cuyo valor en la tabla de diferencias sea cero. Así, finalmente, alcanzamos un ranking de ordenación de máquinas según la prioridad para ser desechadas, en nuestro ejemplo dicho ranking sería: $\{M1, M3, M2, M4, M5\}$. Si se produce algún empate se ordena según el C_i de los implicados y en caso de persistir la igualdad, se haría aleatoriamente entre los mismos.

Para mayor claridad se ha expuesto en la figura 6.1 el pseudo-código que proporciona el ranking de máquinas para el método planteado.

```
procedure Ranking_máquinas_NAM
   M_ori := Matriz original de tiempos de proceso p_ij;
   d := num. de valores min. de p_ij de cada trabajo a seleccionar

   for j := 1 to n do          % selección k valores mínimos
      for k := 1 to d do
         l = argmin_{h∈M} {p_hj};
         M_red := escribir p_lj como parte de la nueva matriz reducida;
         M_ori := Marcar p_lj como no seleccionable;
      endfor k

      l = argmin_{h∈M} {p_hj};   % selección del k+1 valor mínimo
      p_0j:=p_lj;

      % Restamos el k+1 de los otros
      for i := 1 to m do
         M_red := p_ij − p_0j;   ∀p_ij > 0;  ∀p_ij ∈ M_red
      endfor i
   endfor j

% Sumamos los p_ij de cada máquina de M_red
   for i := 1 to m do

      S_i := Σ_{j=1}^{n} p_ij;   ∀p_ij ∈ M_red

      % para las máquinas sin valores en M_red
      if S_i = 0  then
         S_i := Σ_{j=1}^{n} p_ij;   ∀p_ij ∈ M_ori
      endif

   endfor i

   H_i:= Vector con las máquinas ordenadas de menor a mayor valor de S_i;
end
```

Figura 6.1 – Orden de máquinas o ranking (NAM).

6.2.2. Método de las selecciones de máquinas

Una vez obtenido el ranking de máquinas, se procede a la selección de las mismas por distintos procedimientos. Tras cada selección de máquinas se resuelve el problema resultante con las máquinas seleccionadas. Veamos como se seleccionan las máquinas.

1. Hay m máquinas disponibles y se quieren seleccionar solo $m - Z$ de éstas, donde Z es el número de máquinas que no se usarán en el taller. Se ha realizado un ranking previo de todas las máquinas.

2. La primera selección corresponde a las $m - Z$ primeras máquinas del ranking.

3. Para la segunda selección, se escoge la máquina en la posición $m-Z+1$ del ranking, que se corresponde con la primera máquina no seleccionada y se intercambia con la última máquina seleccionada, la que ocupa la posición $m - Z$.

4. Para la tercera selección, respecto a la primera selección, se intercambia de nuevo la primera máquina no seleccionada, $m-Z+1$, con la máquina en la posición $m - Z - 1$.

5. El proceso continua hasta que se hace una selección donde la primera máquina no seleccionada se intercambia con la primera máquina seleccionada (la que ocupa la posición 1 del ranking), es decir, cuando la primera máquina no seleccionada se ha intercambiado con todas las máquinas seleccionadas en la primera opción.

6. El proceso anterior se repite de nuevo pero intercambiando esta vez la segunda máquina no seleccionada, es decir, aquella que ocupa la posición $m - Z + 2$ del ranking, con todas las máquinas seleccionadas en la primera opción.

7. En total, se hacen $(m - Z) \cdot Z + 1$ selecciones o conjuntos distintos de máquinas.

8. Una vez realizadas todas las selecciones mediante este método, si el criterio de parada no se ha cumplido, entonces se hacen sucesivas selecciones de máquinas al azar hasta que éste se alcanza.

Apliquemos el procedimiento de selección de máquinas al ejemplo anterior, donde el ranking es $\{M5, M4, M2, M3, M1\}$. Pongamos que se quiere no usar el 40 % de máquinas, es decir, hay que desechar $Z = 2$ máquinas de las 5 máquinas totales, con lo que $m - Z = 5 - 2 = 3$ máquinas se usarán en el taller. Como primera opción se seleccionarían las tres primeras máquinas del ranking, es decir, $\{M5, M4, M2\}$. Nótese que se trata de las tres máquinas más prometedoras según el ranking. Una vez solucionado este problema de asignación en las tres primeras máquinas paralelas no relacionadas del ranking, se pasaría a una nueva selección. Se toma la primera máquina no seleccionada según el ranking, la máquina $M3$ y se cambiaría por la última máquina seleccionada, $M2$, quedando por tanto las tres máquinas seleccionadas a usar como $\{M5, M4, M3\}$. Tras resolver este nuevo problema se sustituiría de nuevo la primera máquina no seleccionada por la segunda seleccionada, es decir $M3$ por $M4$, quedando la selección como $\{M5, M3, M2\}$. La siguiente selección nos dejaría $\{M3, M4, M2\}$. Llegados a este punto, donde ya se ha sustituido la primera máquina no seleccionada por cada una de las máquinas seleccionadas, se procede a tomar la siguiente máquina no seleccionada $M1$. Así pues la siguiente combinación a probar sería $\{M5, M4, M1\}$. Seguiríamos por $\{M5, M1, M2\}$ y finalmente por $\{M1, M4, M2\}$. Tras acabar estas selecciones y si no se hubiera cumplido el criterio de parada, se continuaría con selecciones de máquinas aleatorias. El resultado final de cada algoritmo, cuando se usan estos métodos de ranking y selección es la mejor solución obtenida de entre todas las selecciones de máquinas probadas. La figura 6.2 muestra el pseudo-código del método para la selección de máquinas.

```
procedure Método_selección_Máquinas_NAM
    P_ori := Problema original;
    H := Lista ordenada del proceso Lista_Ordenada_máquinas_NAM;
    r := Porcentaje de máquinas de no usadas;
    Z := m · r;                % número de máquinas no usadas;

    for i := m − Z + 1 to m do
        P_red := Marcar H_i como máquina no seleccionable;
    endfor i

    P'_red := P_red;
    π = Resolución P_red ;          % por diferentes métodos;
    π_b = π;                        % π_b = mejor solución;

    % Orden de cambio de las máquinas no seleccionadas
    for i := 1 to Z do
        k = m − Z + i;
        for i' := 1 to m − Z do
            q = m − Z + 1 − i';
            P_red := P'_red;
            P_red := Marcar Z_k como máquina seleccionable;
            P_red := Marcar Z_q como máquina no seleccionable;
            if time >= criterio de parada then end
            π = Resolver P_red ;        % por diferentes métodos;
            if C_máx(π) < C_máx(π_b) then π_b := π;
        endfor i'
    endfor i

    % Cambio aleatorio de máquinas no seleccionables
    while (criterio de parada no satisfecho) do
        P_red := P_ori;
        for s := m − Z + 1 to m do
            i'' = Selección aleatoria de máquinas sin repetición;
            P_red := Marcar Z_i'' como máquina no seleccionable;
        endfor s
            π = Resolver P_red ;        % por diferentes métodos;
        if C_máx(π) < C_máx(π_b) then π_b := π;
    endwhile
end
```

Figura 6.2 – Método de selección de máquinas (NAM).

6.2.3. Algoritmos para la resolución de NAM

Como se comentó inicialmente, el primer método a considerar es la simple solución del modelo formado por los conjuntos de restricciones anteriores (6.2), (6.4), (6.3) y (6.1) mediante un moderno solver. Usamos el solver IBM-ILOG CPLEX versión 12.1, que es la última versión disponible en el momento de la escritura de este capítulo. Dado que este problema es distinto a los tratados en capítulos anteriores, estudiado en último lugar y que recientemente se había adquirido esta versión del solver comercial, quisimos emplear esta versión a efectos de compararnos y usar la versión más potente disponible. Denominaremos a este solver simplemente como CPLEX de aquí en adelante. El segundo método a considerar es usar también CPLEX, pero conjuntamente con los algoritmos de ranking y de selección de máquinas anteriormente propuestos, para ver como de buena es la propuesta de selección de máquinas respecto a la resolución del problema sólo con CPLEX. Es decir, cada vez que se hace una selección, se resuelve el modelo resultante, que al ser de dimensiones más reducidas que el original, se espera que se resuelva más rápidamente. Al empleo del ranking de máquinas unido a la selección de máquinas expuestas anteriormente las denominaremos NAM. Para mejorar el rendimiento, se le fijaba un tiempo a CPLEX para reiniciarse con una nueva combinación proporcionada por NAM caso de no haber alcanzado el óptimo antes. Esto se hace por que la primera opción dada por el ranking puede no contener las máquinas óptimas a retirar. Además, CPLEX, pasado un tiempo, se demora en ocasiones en demasía cerrando el árbol de búsqueda. Así, una reiniciación de CPLEX con una nueva combinación de máquinas y proporcionándole el mejor resultado obtenido hasta el momento como cota inferior, permiten a CPLEX, generalmente, hallar los óptimos rápidamente. Nos referimos a este segundo método como NAM+CPLEX.

La tercera propuesta tiene en cuenta los algoritmos anteriores de ranking y selección pero en vez de usar CPLEX para resolver el problema de secuen-

ciación se usa algún algoritmo que nos de una buena respuesta. En principio podría ser DIG, estado del arte, como heurística sencilla y de fácil implementación. Ahora bien, este problema requiere de algún algoritmo que resuelva rápido, ya que hay que probar distintas combinaciones de máquinas, y esto, no lo dan fácilmente las heurísticas. Para solucionar esta cuestión, se recurrió, durante una parte del tiempo total dado para resolución de cada instancia, a una heurística muy simple y rápida para seleccionar qué combinaciones son las mejores. Esta heurística no es otra que la inserción con estrategia "best" hasta óptimo local seguida del intercambio con la estrategia "first" igualmente hasta óptimo local; es decir la heurística ST, ya vista en la sección 3.1 para la base de los primeros algoritmos propuestos. Esta heurística ST resuelve de forma rápida las distintas combinaciones de máquinas que le proporciona el ranking y la selección, obteniéndose por tanto la mejor solución que este algoritmo puede hallar, lo que nos proporciona una lista de las mejores máquinas a emplear. Finalmente, una vez agotado el tiempo de la parte dada a la heurística ST, se deja el resto del tiempo a DIG para que refine la solución de la mejor opción de máquinas a emplear. Es decir, el ranking y la selección ordena las máquinas en función de su posibilidad de ser retiradas, después el problema reducido usa ST como solver y durante la última parte del tiempo total se aplica DIG para refinar la mejor solución hallada en la fase anterior por ST. Este algoritmo se denomina NAM+ST+DIG. Nótese que no requiere de ningún solver comercial para su aplicación.

Como cuarto y último método propuesto se probó a usar también la opción de ST antes de CPLEX para ver como reaccionaba con esta nueva opción, es decir, en vez de lanzar CPLEX después del ranking y de la selección, se aplica una búsqueda local rápida tipo ST para inicializar CPLEX con una primera buena solución para el problema de máquinas paralelas. Pasó a denominarse NAM+ST+CPLEX. Como en el caso de NAM+CPLEX se le fijaba un tiempo a CPLEX para que se reiniciase con una nueva combinación proporcionada

por la lista fijada por ST caso de no haber alcanzado el óptimo antes.

6.2.4. Análisis computacional

Como en anteriores ocasiones, empleamos las mismas 1.400 instancias agrupadas en los siete intervalos ya mencionados; así mismo se emplearan el mismo grupo de 12 ordenadores PC/AT.
Se hicieron pruebas para distintos porcentajes de máquinas sin usar, en concreto para 20 %, 50 % y 80 %, así como a tiempos de parada de 60 segundos y de 300 segundos. Las calibraciones para estos tiempos vienen reflejados en la tabla 6.4.

Criterio parada		60	300
NAM+CPLEX	Tiempo reinicio	10	60
NAM+ST+DIG	Tiempo ST	10	60
	Tiempo DIG	50	240
NAM+ST+CPLEX	Tiempo ST	20	100
	Tiempo CPLEX	40	200
	Tiempo reinicio	10	60

Tabla 6.4 – Tabla de los resultados de las calibraciones para los algoritmos NAM propuestos. Valores en segundos.

Procedemos primeramente a mostrar las tablas de las desviaciones porcentuales de los distintos algoritmos, aunque en esta ocasión las diferencias se han realizado respecto a la referencia de la mejor solución conocida. En este caso no se han hecho las comparativas respecto a CPLEX 2 horas con este nuevo problema, por producirse en algunos casos gap superiores al 50 %, por lo que no parece ser esta referencia la más indicada, por lo que se optó por la referencia de la mejor solución conocida. Finalmente, mostramos unos gráficos de ANOVA que representan la significación estadística de las diferencias

observadas entre los distintos algoritmos.

En el presente capítulo se expondrán sólo los resúmenes de los resultados y algún gráfico especifico por su singular importancia. No obstante, se pueden consultar todas las tablas y ANOVAS de forma más detallada en el anexo E.

Hemos querido empezar este estudio por el caso intermedio del 50 % de maquinas sin usar para poder tener una primera referencia sobre el comportamiento de los algoritmos para este novedoso problema. Posteriormente podremos ver el estudio para un menor porcentaje (20 %) y un mayor porcentaje (80 %). Igualmente, para cada uno de los porcentajes a estudiar se empezó con un tiempo de 300 segundos, lo que supone un periodo amplio para el funcionamiento de CPLEX. El estudio de cada uno de los porcentajes se completa para un tiempo de 60 segundos, más acorde con una rápida resolución de las instancias.

Empezando por un 50 % de máquinas sin usar y 300 segundos, podemos apreciar en la tabla 6.5 como los tres algoritmos propuestos superan ampliamente el uso de CPLEX. En este caso NAM+ST+CPLEX da unos resultados ligeramente mejores que el resto. Cabe destacar como para este tipo de problemas, el intervalo $U(100, 120)$ es uno de los que más dificultades entraña para su resolución por el solver CPLEX, contrariamente a lo que sucedía en el caso de máquinas paralelas no relacionadas pero sin restricción en el número de máquinas, donde CPLEX ofrecía uno de sus mejores resultados para este intervalo de tiempos de proceso. El error medio para CPLEX en este intervalo $U(100, 120)$, es del 4,4 %, el segundo más alto de la tabla, después del más del 5 % ofrecido por CPLEX en el intervalo $U(1, 100)$.

Intervalo	CPLEX	NAM+CPLEX	NAM+ST+DIG	NAM+ST+CPLEX
U(1,100)	5,11	1,05	1,68	0,34
U(10,100)	2,94	0,49	0,41	0,10
Job Corre	3,65	0,16	0,05	0,32
Maq Corre	1,25	0,01	0,64	0,06
U(100,200)	3,32	0,10	0,21	0,08
U(100,120)	4,40	0,03	0,05	0,02
U(1000,1100)	0,18	0,02	0,01	0,01
Media	2,98	0,27	0,44	0,13

Tabla 6.5 – Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para todos los intervalos y 50 % de máquinas sin usar con 300 segundos.

Cuando el tiempo de resolución desciende de 300 segundos a 60 segundos, tal como los muestra la tabla 6.6, los resultados empeoran ligeramente para los algoritmos propuestos, pero lo hace de forma más brusca para la utilización del solver CPLEX, ya que pasa a obtener valores cercanos al 9,7 % de diferencia respecto al mejor resultado obtenido. En este caso se aprecia claramente la influencia del tiempo de resolución en CPLEX. Podemos destacar que la utilización de NAM junto con CPLEX palía en gran medida este problema que presenta el solver para este tipo de problema. En esta tabla, los valores medios del error cometido llegan a superar el 14 %. Así mismo, tal y como se puede ver en el anexo E, existen grupos de instancias donde la diferencia obtenida por CPLEX frente a la mejor solución, llega a alcazar valores por encima del 60 %, como en el caso de las instancias de 1000×50 para el intervalo $U(100, 120)$.

Intervalo	CPLEX	NAM+CPLEX	NAM+ST+DIG	NAM+ST+CPLEX
U(1,100)	12,87	2,64	1,68	1,04
U(10,100)	10,29	0,98	0,57	0,52
Job Corre	11,90	0,34	0,14	0,32
Maq Corre	3,73	0,29	0,74	0,20
U(100,200)	12,88	0,33	0,27	0,26
U(100,120)	14,20	0,07	0,06	0,07
U(1000,1100)	1,94	0,04	0,02	0,03
Media	9,69	0,67	0,50	0,35

Tabla 6.6 – Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para todos los intervalos y 50 % de máquinas sin usar con 60 segundos.

En el caso de no emplear solamente el 20 % de las máquinas y 300 segundos, las diferencias entre los algoritmos propuestos y CPLEX se suavizan mucho, llegando a alcanzar a alguno de ellos e incluso superar a otro. En este último caso se encuentra el algoritmo NAM+ST+CPLEX, que lo supera CPLEX, aunque el algoritmo NAM+CPLEX sigue superando al resto, tal y como muestra la tabla 6.7. Esto nos indica que la introducción de la parte de ST como solver previo al uso de CPLEX introduce cierta cantidad de ruido que no beneficia al algoritmo en el caso de no emplear un número reducido de máquinas.

Intervalo	CPLEX	NAM+CPLEX	NAM+ST+DIG	NAM+ST+CPLEX
U(1,100)	0,54	0,69	1,51	5,00
U(10,100)	0,68	0,18	0,39	2,78
Job Corre	0,69	0,31	0,06	1,02
Maq Corre	0,13	0,03	0,48	1,63
U(100,200)	0,27	0,08	0,11	0,66
U(100,120)	0,06	0,02	0,03	0,15
U(1000,1100)	0,02	0,01	0,02	0,08
Media	0,34	0,19	0,37	1,62

Tabla 6.7 – Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para todos los intervalos y 20 % de máquinas sin usar con 300 segundos.

Para el caso del no empleo de un 20 % de máquinas y solo 60 segundos, de nuevo CPLEX vuelve a empeorar notablemente sus resultados respecto a tiempos superiores como el caso de los anteriores 300 segundos, mientras que los algoritmos NAM+CPLEX y NAM+ST+DIG conservan buenos porcentajes respecto a la mejor solución obtenida, como se aprecia en la tabla 6.8.

Intervalo	CPLEX	NAM+CPLEX	NAM+ST+DIG	NAM+ST+CPLEX
U(1,100)	2,41	1,24	1,95	5,24
U(10,100)	2,10	0,75	0,58	2,89
Job Corre	1,76	0,66	0,21	1,29
Maq Corre	0,73	0,25	0,55	1,66
U(100,200)	0,84	0,30	0,17	0,69
U(100,120)	1,63	0,07	0,04	0,16
U(1000,1100)	0,10	0,04	0,02	0,08
Media	1,37	0,47	0,50	1,72

Tabla 6.8 – Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para todos los intervalos y 20 % de máquinas sin usar con 60 segundos.

Finalmente, cuando se analiza el comportamiento de los algoritmos estudiados para el caso de un 80 % de máquinas sin utilizar, vemos en la tabla 6.9 como las diferencias entre el empleo del solver solo frente a los algoritmos propuestos se incrementa notablemente. Para el caso de 300 segundos ya se ven valores de CPLEX superiores en varias veces al resto de algoritmos. Incluso se aprecian intervalos donde CPLEX produce unas diferencias de más de un 30 %, con diferencias en algunos grupos de instancias superiores al 200 %, como el caso de las instancias de 1000×40 ó 1000×50 en el intervalo de máquinas correlacionadas, como se detalla en el anexo E.

Intervalo	CPLEX	NAM+CPLEX	NAM+ST+DIG	NAM+ST+CPLEX
U(1,100)	12,95	0,67	0,63	1,19
U(10,100)	7,90	0,47	0,29	0,70
Job Corre	2,45	0,07	0,06	0,19
Maq Corre	30,33	0,00	0,39	1,03
U(100,200)	3,95	0,13	0,08	0,32
U(100,120)	4,19	0,02	0,05	0,13
U(1000,1100)	2,76	0,02	0,02	0,05
Media	9,22	0,20	0,22	0,52

Tabla 6.9 – Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para todos los intervalos y 80 % de máquinas sin usar con 300 segundos.

En esta última tabla 6.10 relativa al 80 % de máquinas sin emplear y para 60 segundos vemos como, respecto al mismo porcentaje de máquinas sin emplear y 300 segundos, los valores obtenidos se extreman, así como sus diferencias llegando a superar la diferencia total media respecto a la mejor solución en un 17 % y obteniendo diferencias para en el caso del intervalo de máquinas correlacionadas del 56 %.

Intervalo	CPLEX	NAM+CPLEX	NAM+ST+DIG	NAM+ST+CPLEX
U(1,100)	23,17	2,04	1,47	1,94
U(10,100)	13,97	1,19	0,55	0,99
Job Corre	5,65	0,12	0,11	0,24
Maq Corre	56,64	0,04	0,44	1,33
U(100,200)	7,72	0,24	0,15	0,38
U(100,120)	7,23	0,07	0,07	0,14
U(1000,1100)	5,47	0,04	0,03	0,06
Media	17,12	0,53	0,40	0,73

Tabla 6.10 – Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para todos los intervalos y 80 % de máquinas sin usar con 60 segundos.

Por último, se ha incluido la tabla resumen 6.11 donde se puede apreciar en su conjunto el comportamiento de todos los algoritmos probados para los tiempos de 60 segundos y 300 segundos Se pone de manifiesto que los tres algoritmos propuestos superan a CPLEX. NAM+ST+CPLEX no mejora a NAM+CPLEX siendo este último más sencillo y más robusto, como se ve por el comportamiento de NAM+ST+CPLEX para 20 % de máquinas no usadas. NAM+ST+DIG da los mismos o parecidos resultados que los algoritmos que emplean CPLEX con la ventaja de no usar ningún solver comercial, obteniendo mejores resultados de media para tiempos menores y peores medias para tiempos más elevados.

Maq.	Tiempo	CPLEX	NAM+CPLEX	NAM+ST+DIG	NAM+ST+CPLEX
50 %	300 seg	2,98	0,27	0,44	0,13
50 %	60 seg	9,69	0,67	0,50	0,35
20 %	300 seg	0,34	0,19	0,37	1,62
20 %	60 seg	1,37	0,47	0,50	1,72
80 %	300seg	9,22	0,20	0,22	0,52
80 %	60seg	17,12	0,53	0,40	0,73
media total 300seg		4,18	0,22	0,34	0,76
media total 60seg		9,39	0,56	0,47	0,93

Tabla 6.11 – Tabla porcentual resumen para la media de todos los intervalos y 20 %, 50 % y 80 % de máquinas sin usar con 60 y 300 segundos.

Siguiendo la misma secuencia que en el caso de las tablas previamente mostradas, presentamos los ANOVA correspondientes que nos permitirán comparar las diferencias entre los distintos resultados obtenidos.

La figura 6.3, para un 50 % de máquinas y 300 segundos, muestra claramente que los tres algoritmos propuestos, a pesar de tener diferentes medias, no resultan estadísticamente distintos entre si, mientras que CPLEX muestra claras diferencias con el resto.

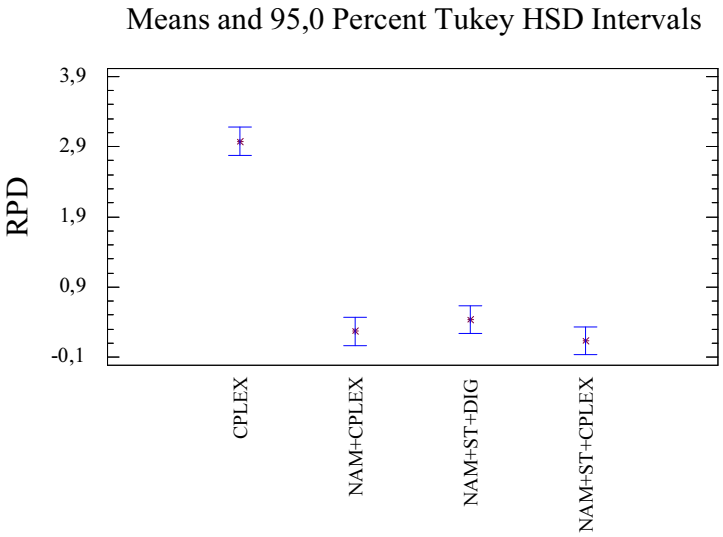

Figura 6.3 – ANOVA para todos los intervalos con 50 % máquinas sin usar y 300 segundos.

En el caso de no emplear el 50 % de máquinas y solo 60 segundos, tal y como muestra la figura 6.4, la diferencia entre los algoritmos propuestos y CPLEX se hace aun más amplia en sus valores que la precedente para 300 segundos, ya que CPLEX, como se ha venido mostrando a lo largo de todo el trabajo expuesto, necesita de una cantidad considerable de tiempo para poder dar valores competitivos y en algunos casos, como el presente, ni siquiera con bastante tiempo es capaz de dar valores competitivos.

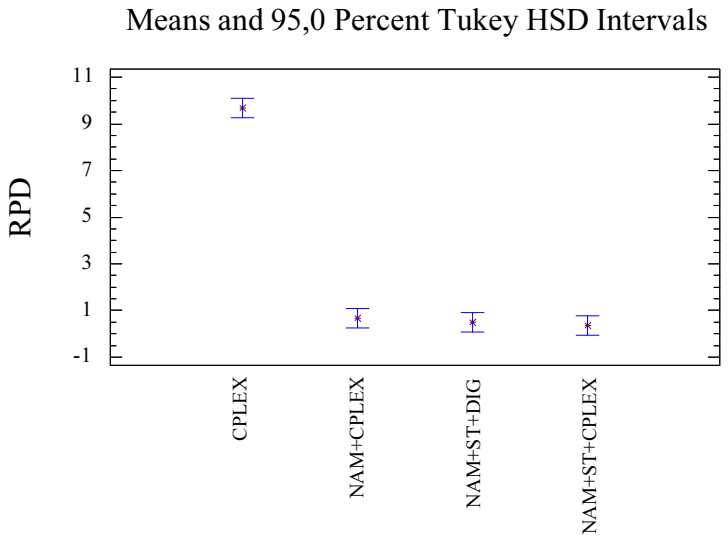

Figura 6.4 – ANOVA para todos los intervalos con 50 % máquinas sin usar y 60 segundos.

Para el caso de 20 % de máquinas sin usar y 300 segundos (figura 6.5) , tenemos que el último algoritmo probado, NAM+ST+CPLEX da pobres valores respecto al resto. Esto es debido a que el funcionamiento de CPLEX ya es lo bastante eficaz para este problema, ya que al ser retiradas un pequeña porción de máquinas, el problema se asemeja bastante al de máquinas paralelas no relacionadas, donde CPLEX ya daba buenos resultados. El resto de algoritmos apenas si muestran diferencias significativas entre si.

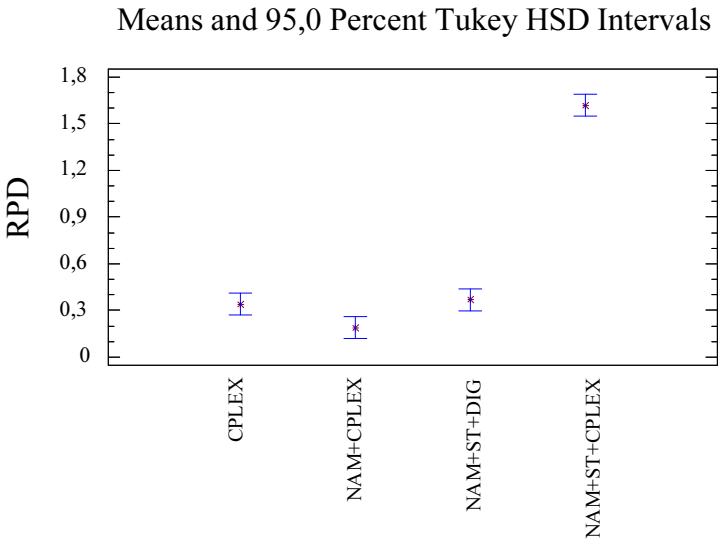

Figura 6.5 – ANOVA para todos los intervalos con 20 % máquinas sin usar y 300 segundos.

El el caso mostrado en la figura 6.6, para un 20 % de máquinas no usadas y 60 segundos, vemos como el factor tiempo fuerza a CPLEX a estar significativamente distanciado de los algoritmos NAM+CPLEX y NAM+ST+DIG, e incluso acercarse a los intervalos dados por NAM+ST+CPLEX.

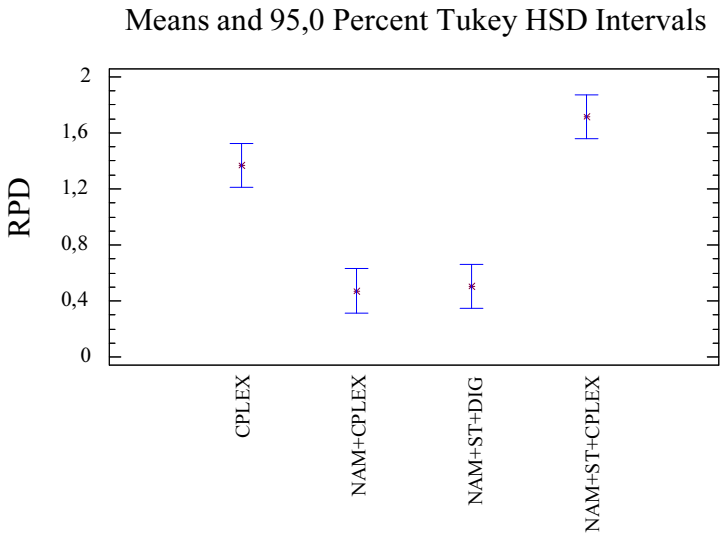

Figura 6.6 – ANOVA para todos los intervalos con 20 % máquinas sin usar y 60 segundos.

La figura 6.7 muestra los valores obtenidos para el caso del 80 % de máquinas no usadas y 300 segundos de tiempo, con una igualdad notable entre los tres algoritmos propuestos y una diferencia significativa clara sobre CPLEX. El orden de magnitud de las diferencias entre los valores de CPLEX y el resto son notables en este caso.

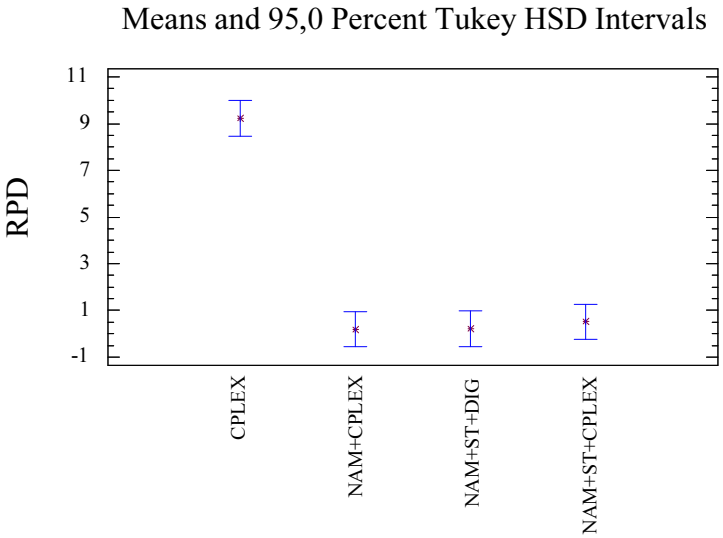

Figura 6.7 – ANOVA para todos los intervalos con 80 % máquinas sin usar y 300 segundos.

Para el último caso en que el tiempo a considerar es de 60 segundos, con un 80 % de máquinas sin usar, nos encontramos con que las diferencias entre CPLEX y el resto de los algoritmos son las más grandes que se dan dentro de este estudio, tal y como muestra la figura 6.8.

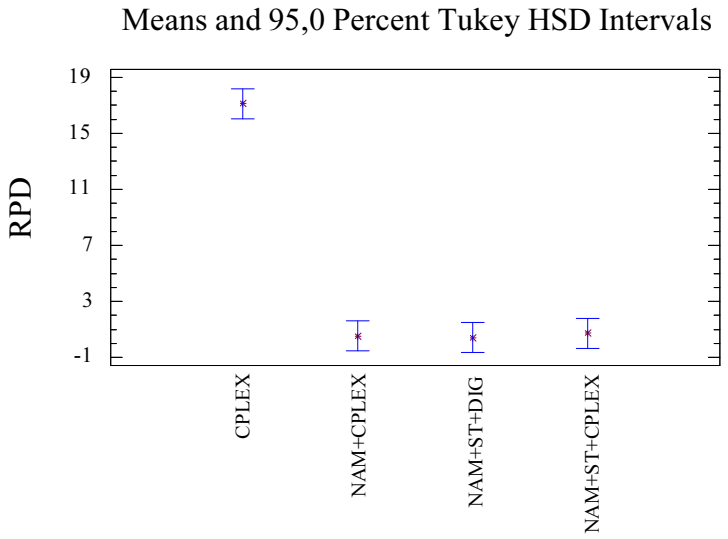

Figura 6.8 – ANOVA para todos los intervalos con 80 % máquinas sin usar y 60 segundos.

Finalmente, hemos querido añadir una evaluación conjunta de todos los porcentajes de máquinas no empleadas para poder ilustrar mejor el efecto de los distintos algoritmos. Estas comparativas vienen dadas por la figura 6.9 para 300 segundos y por la figura 6.10 para 60 segundos, en la que no se han añadido los intervalos para mayor claridad de la figura. En ambas figuras se puede apreciar el buen comportamiento de los algoritmos NAM+CPLEX así como de NAM+ST+DIG. También se puede ver como solo en el caso del 20 % de máquinas no usadas se acerca CPLEX al resto y como NAM+ST+CPLEX, a pesar de tener buenos valores, no supera a los otros dos algoritmos propuestos.

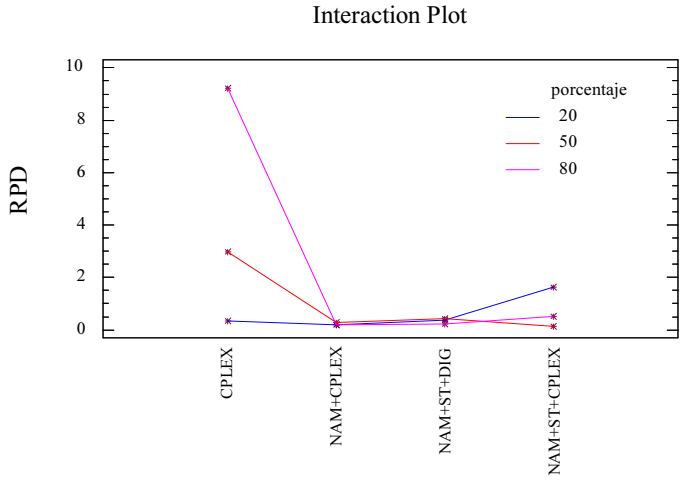

Figura 6.9 – ANOVA con la interacción por % máquinas sin usar y algoritmo para todos los intervalos y 300 segundos.

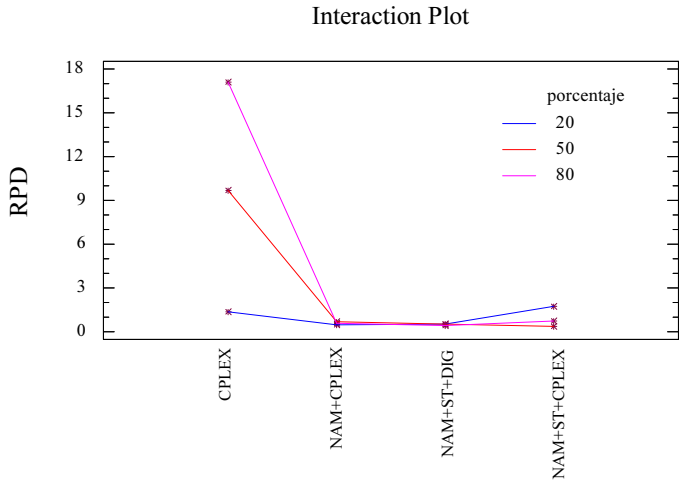

Figura 6.10 – ANOVA con la interacción por % máquinas sin usar y algoritmo para todos los intervalos y 60 segundos.

Como resumen del estudio de no todas las máquinas, podemos decir que a mayor porcentaje de máquinas sin usar, CPLEX encuentra más dificultades en conseguir buenos resultados. Pasa de valores parejos al resto de algoritmos, en el caso de pocas máquinas sin usar, a resultados muy alejados de los mismos cuando se dejan de usar un número elevado de máquinas. También se nota la influencia del tiempo en la consecución de resultados por parte de CPLEX. Esto no es de extrañar, dado que al usar un algoritmo exacto, con un tiempo indefinidamente elevado, debería alcanzar la respuesta óptima. En cuanto a los algoritmos propuestos, vemos como NAM+ST+CPLEX no da mejores resultados que NAM+CPLEX, siendo este último algoritmo más sencillo. Así mismo, los resultados obtenidos NAM+ST+DIG son similares a los obtenidos por NAM+CPLEX, siendo ambos algoritmos los de mejores resultados globales para los intervalos estudiados, por lo que tenemos dos buenos algoritmos para la resolución de este problema, uno empleando un solver comercial y otro que no emplea más que heurísticas simples.

6.3. No todos los trabajos (NAJ)

En esta sección tratamos el problema de no realizar todos los trabajos sujeto al objetivo de minimizar el $C_{\text{máx}}$. Como primer paso, se analizó el uso de CPLEX, empleando la formulación matemática anteriormente mencionada en la sección 6.1. La idea previa es comprobar como se desenvuelve un solver comercial con este tipo de problemas. Los resultados obtenidos se reflejan en la tabla 6.12. En ella se muestra, respecto a la cota inferior, los siete intervalos que venimos empleando habitualmente y para 300 segundos, el error máximo que comete CPLEX, la media total de error cometido (promediado entre las 200 instancias de cada intervalo) y el error parcial medio (promediado sólo entre las instancias con error/gap). Lo primero que podemos apreciar es que el error máximo más grande lo tiene para los trabajos correlacionados, lo cual no es sorprendente, dado que para este caso resulta mucho más difícil de decidir

que trabajos eliminar dada la inherente relación entre los trabajos. No obstante, cuando vemos la media de error que se produce al estudiar las instancias de este intervalo, esta se queda por debajo del 2 %. En el resto de intervalos estos valores caen muy debajo de los obtenidos para trabajos correlacionados. Así podemos observar como la media de error obtenida por CPLEX para las 1.400 instancias, respecto a la cota inferior, es de solo un 0,46 % e incluso haciendo la media entre solo las instancias que no han alcanzado el óptimo, este error medio no llega a alcanzar el 1 %. Todo ello respecto a la cota inferior, lo cual indica que, como máximo, ese es el error que encontraremos, aunque evidentemente, podría ser menor si se hubiera obtenido el óptimo para todas las instancias. Así que no parece muy rentable el adentrarse en buscar algoritmos no exactos que sean capaces de igualar o superar los valores tan buenos proporcionados, mediante algoritmo exacto, por parte de este solver.

	Error máximo	Media total error	Error parcial medio
U(1,100)	0,00	0,00	0,00
U(10,100)	2,26	0,17	1,28
JobCorre	12,28	1,94	2,36
MaqCorre	1,13	0,01	0,48
U(100,200)	3,28	0,11	0,49
U(100,120)	0,08	0,00	0,05
U(1000,1100)	8,34	1,01	1,87
Media	3,91	0,46	0,93

Media total error = suma de errores entre total de instancias; error parcial medio = suma de errores entre instancias con error

Tabla 6.12 – Tabla porcentual de la media de las diferencias respecto a la cota minima para todos los intervalos y 50 % de trabajos realizados con 300 segundos.

No obstante, se hizo la prueba de tratar de adaptar el algoritmo que tan buenos resultados nos había dado para no todas las máquinas y aplicarlo a no todos los trabajos. Así que lo que se hizo fue trasponer la matriz del problema

p_{ij}, de forma que los trabajos figurasen como máquinas y las máquinas como trabajos y aplicarles el algoritmo NAM adaptado a esta nueva situación, que así paso a denominarse NAJ, para después aplicarle, como en el caso anterior, la heurística ST y finalizar afinando con la heurística DIG, quedando por tanto el algoritmo como NAJ+ST+DIG. Los resultados no fueron competitivos con los obtenidos por la utilización de CPLEX, expresados en la tabla 6.12. Se muestra como ejemplo los tres ANOVA de los intervalos con peores resultados dados por CPLEX, U(10,100), trabajos correlacionados y U(100,200), donde se aprecia claramente que CPLEX es estadísticamente mejor que el algoritmo propuesto adaptado a este problema, salvo para el caso de trabajos correlacionados, donde a pesar de no ser estadísticamente diferente, CPLEX presenta mejor media aun en este caso. Estos resultados son los esperados, tal como anticipábamos, dado el escaso gap existente en las soluciones dadas por CPLEX.

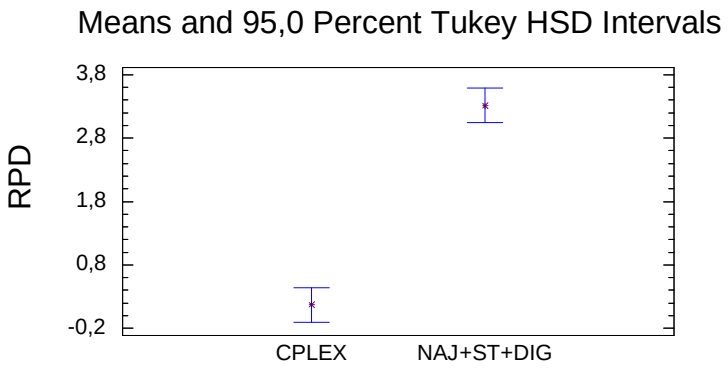

Figura 6.11 – ANOVA del intervalo U(10,100) con 50 % trabajos realizados y 300 segundos.

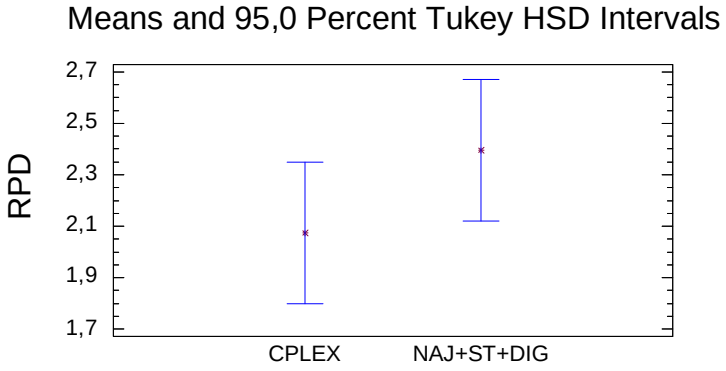

Figura 6.12 – ANOVA del intervalo Trabajos Correlacionados con 50 % trabajos realizados y 300 segundos.

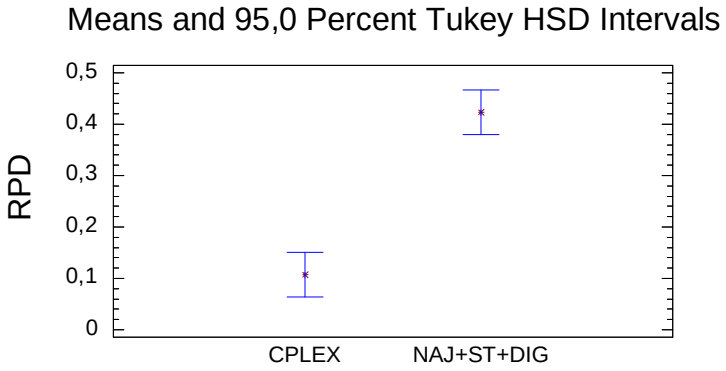

Figura 6.13 – ANOVA del intervalo U(100,200) con 50 % trabajos realizados y 300 segundos.

6.4. Resumen de máquinas opcionales y selección de trabajos

En este capítulo se estudian por primera vez, al mejor saber y conocimiento de los autores, dos generalizaciones del problema de secuenciación de trabajos

en máquinas paralelas no relacionadas con el objetivo de minimizar el $C_{\text{máx}}$. Estas generalizaciones son las que resultan de introducir más dimensiones de decisión, en concreto decidir qué máquinas son opcionales (no se usan todas las máquinas) o qué trabajos se seleccionan para producción (no se procesan todos los trabajos). Para ambos problemas se han propuesto modelizaciones matemáticas sencillas y potentes si se resuelven con CPLEX.

Para el problema con máquinas opcionales se ha propuesto un sencillo ranking que ordena las máquinas de más interesantes para la minimización del $C_{\text{máx}}$ a aquellas más adecuadas para no ser usadas. Junto con el ranking, se ha presentado un algoritmo también muy sencillo para ir seleccionando las máquinas usadas de entre las m máquinas disponibles. Con estos dos algoritmos se han propuesto tres métodos de resolución, resultado de unir el ranking y la selección de máquinas con CPLEX o con técnicas de búsqueda local o algoritmos heurísticos propuestos en capítulos anteriores. Los resultados computacionales y tests estadísticos realizados, con distintos parámetros y criterios de parada, y con un conjunto muy amplio de instancias de prueba (1.400 instancias) nos permiten concluir que la combinación de los algoritmos de ranking y de selección de máquinas, junto a los métodos propuestos, superan ampliamente a CPLEX en todos los tests llevados a cabo.

En cambio, para el problema de la selección de trabajos se ha observado que el uso de CPLEX con la modelización propuesta da resultados extremadamente buenos, óptimos en muchos casos y en cualquier caso con una desviación media con respecto a la cota inferior de menos del 0,5 %.

Como posteriores estudios en este campo, queda el adentrarse en los temas multiobjetivo, empleando dos o tres de los objetivos posibles, a saber, minimizar el $C_{\text{máx}}$, minimizar el número de máquinas y/o minimizar el número de trabajos no realizados. Así mismo quedaría por realizar una paralelización de los algoritmos.

CAPÍTULO 7

CONCLUSIONES Y LÍNEAS FUTURAS DE INVESTIGACIÓN

En esta Tesis Doctoral hemos tratado con el problema de máquinas paralelas no relacionadas, donde n trabajos independientes deben ser asignados a una de las m máquinas paralelas y los tiempos de proceso de cada trabajo difieren de una máquina a otra. Hemos trabajado sobre el objetivo de minimizar el máximo tiempo de finalización o $C_{\text{máx}}$. Este es un tipo de problema de asignación que ha sido frecuentemente estudiado en la literatura científica dadas sus múltiples aplicaciones. Por tanto, hemos realizado primero, en el capítulo 2, un estudio de la literatura relativa a este tema y hemos comprobado como el estado del arte estaba representado por los algoritmos Partial de Mokotoff y Jimeno (2002) y RBS de Ghirardi y Potts (2005), a la vez que constatamos que el solver IBM-ILOG CPLEX 11.0 es un método a tener en cuenta para la resolución de este problema de asignación. También pudimos observar que no existía un conjunto de instancias estándar para este tipo de problema, dado que cada autor realizaba las suyas propias y posteriormente no las dejaba disponibles. Por lo tanto, se generaron un conjunto de 1.400 instancias repartidas

en 7 intervalos de tiempos de proceso distintos: $U(1, 100)$, $U(10, 100)$, máquinas Correlacionadas, Trabajos Correlacionados, $U(100, 200)$, $U(100, 120)$ y $U(1000, 1100)$. Para cada uno de estos intervalos se hicieron combinaciones de $m = \{10, 20, 30, 40, 50\}$ y $n = \{100, 200, 500, 1000\}$ y dentro de cada una de estas combinaciones se generaron 10 instancias al azar. Todas estas instancias, junto con sus soluciones de referencia, están disponibles en `http://soa.iti.es/`, lo que ya representa una contribución para el problema de $R//C_{\text{máx}}$.

En una primera aproximación diseñamos en el capítulo 3 unos algoritmos basados en una búsqueda local de inserción seguida de una búsqueda local de intercambio, ambas llevadas hasta su óptimo local (esta aplicación consecutiva de varios vecindarios se conoce como VNS (*Variable Neighborhood Search*) de Hansen y Mladenovic, 2001), para posteriormente proceder a una modificación de la solución por diversos métodos como los de búsquedas locales restringidas (NSP o Vir) o bien por el Iterated Greedy (IG) desarrollado por Ruiz y Stützle (2007), y volver con esta solución modificada al VNS. Esto nos llevó a algoritmos que superaban, en el promedio de los resultados obtenidos en las instancias estudiadas, al estado del arte e incluso a CPLEX, el cual demostró igualar, si no superar, al estado del arte por su parte.
Con todo, estos algoritmos tienen un fuerte contenido estocástico, por lo que se desarrollaron en el capítulo 4 nuevos métodos que limitan en gran medida esta parte estocástica y permiten a los algoritmos superar cualquier método existente hasta la fecha para casi todos los casos estudiados. El último algoritmo desarrollado en esta parte de la Tesis, el NVST-IG o DIG, es un híbrido en el que se aúnan las mejores características de los métodos desarrollados. En él, las búsquedas locales restrictivas (NSP y Vir) se integran dentro del VNS y como modificador de la solución se empleó el IG adaptado para evitar casi completamente la aleatoriedad en la elección de los elementos a modificar.
No obstante, a pesar de los buenos resultados obtenidos, buscamos nuevos

métodos en el capítulo 4. Estos están basados en un enfoque diferente del problema a la par que exploramos la posibilidad de realizar algoritmos que pudieran ser empleados en paralelo, aspecto relevante dado que los modernos ordenadores llevan en su mayoría varios procesadores (llamados núcleos o cores) integrados en el mismo chip. Así pues, se procedió a la implementación de unos algoritmos basados en la reducción de tamaño del problema de asignación original, lo que nos proporcionó unas buenas soluciones tanto para tiempos de CPU cortos como largos. La idea consistió en considerar sólo algunas de las mejores posiciones a la hora de evaluar las posibles asignaciones de los trabajos a las máquinas, en lugar de todas. Esta aproximación, supone una idea original y sencilla que además proporciona resultados muy buenos. Finalmente, se aplicó un solver para obtener la solución definitiva. Se presentaron diversos algoritmos, desde los más simples consistentes en tomar sólo los tres menores valores de tiempos de proceso para cada trabajo, hasta una implementación apoyada en DIG con una retroalimentación truncada al solver, denominada como 2JDi(90). Además, se hicieron variaciones de los mejores algoritmos que habíamos desarrollado para su uso en modo paralelo, que hemos denominado M-DIG y M-2JDi(90). Esto nos llevó a resultados significativamente mejores, para las instancias estudiadas, que la versión de CPLEX 11.1 tanto en su uso serie como en paralelo. El resultado final fue que el último algoritmo desarrollado, en tan solo 15 segundos, tiene un "gap" con respecto a la mejor cota inferior conocida, de sólo 0,63 % de media en las 1.400 instancias estudiadas, lo que deja el problema de $R//C_{\text{máx}}$ cerca de estar cerrado.

Por último, en el capítulo 5, tratamos el problema de no usar todas las máquinas o de no realizar todos los trabajos, es decir, el problema con máquinas opcionales o con selección de trabajos. Este campo apenas se ha tocado en la literatura, a pesar de sus numerosas aplicaciones. Por ello, tuvimos primero que probar modelos matemáticos y posteriormente ejecutarlos en un solver

comercial para comprobar sus soluciones, que resultaron proporcionar gaps muy elevados en el caso de las máquinas opcionales pero gaps muy pequeños cuando se trataba de selección de trabajos. Para el problema con máquinas opcionales, se ha propuesto un sencillo ranking que ordena las máquinas de más interesantes para la minimización del $C_{\text{máx}}$ a aquellas candidatas a no ser usadas. Junto con el ranking, se ha presentado un algoritmo también muy sencillo para ir seleccionando las máquinas usadas de entre las m máquinas disponibles. Con estos dos algoritmos se han propuesto tres métodos de resolución, resultado de unir el ranking y la selección de máquinas con CPLEX 12.1 o con técnicas de búsqueda local o algoritmos metaheurísticos propuestos en capítulos anteriores. Los resultados nos permiten concluir que la combinación de los algoritmos de ranking y de selección de máquinas, junto a los métodos propuestos, supera ampliamente a CPLEX en todos los tests llevados a cabo. En cambio, para el problema de la selección de trabajos se ha observado que el uso de CPLEX con la modelización propuesta da resultados extremadamente buenos, óptimos en muchos casos y en cualquier caso con una desviación media con respecto a la cota inferior de menos del 0,5 %.

En resumen, hemos desarrollado varios algoritmos para la resolución de $R//C_{\text{máx}}$ de los cuales cabe destacar DIG, un algoritmo híbrido basado en un VNS con cuatro vecindarios seguido de un modificador de la solución basado en el IG mejorado, que proporciona buenos resultados para este problema. Este algoritmo no emplea ningún tipo de solver comercial y además produce mejores resultados a medida que se le proporciona mayor tiempo de CPU. También se ha desarrollado el algoritmo denominado 2JDi(90) basado en una reducción del tamaño del problema que ofrece mejores resultados que cualquier otro método para $R//C_{\text{máx}}$. Ambos algoritmos se han desarrollado en sus versiones serie y paralelo. Finalmente, se ha estudiado el problema de máquinas opcionales y selección de trabajos presentado unos algoritmos eficaces para la resolución de máquinas opcionales, como el NAM+CPLEX y

el NAM+ST+DIG. También se ha mostrado como CPLEX es un buen solver para el caso de selección de trabajos.
Es de destacar que el algoritmo DIG no solo se ha empleado en la resolución directa del problema $R//C_{\text{máx}}$, si no que se emplea como parte integrante de los algoritmos 2JDi(90) y NAM+ST+DIG, mostrando su versatilidad. Así mismo la metodología de reducción de tamaño podría ser empleada en otro tipo de problemas.

Como trabajos futuros, nos proponemos comprobar el resultado de los métodos propuestos para otros tipos de problemas como $Q//C_{\text{máx}}$, así como profundizar los problemas introducidos en este trabajo en cuanto al empleo de máquinas opcionales y selección de trabajos, ya que estos permiten abrir múltiples líneas de actuación. Por ejemplo, es de interés el estudio combinado de máquinas opcionales y selección de trabajos. Igualmente, es necesario hacer un estudio en el campo de la optimización multi-objetivo donde el $C_{\text{máx}}$ se combina con diferentes valores de máquinas a usar y/o trabajos a realizar. Por otra parte, y en especial en lo que se refiere al problema de las máquinas opcionales, estamos ante un problema totalmente paralelizable, dado que cada combinación de máquinas se puede resolver de manera independiente, por lo que las técnicas de computación paralela y técnicas metaheurísticas cooperativas pueden resultar de interés, sobre todo en estos tiempos donde encontramos cada vez más núcleos de cómputo disponibles en los ordenadores.

Como resultado final, se han publicado dos artículos en revistas de muy alto impacto, se han presentado dos ponencias en congresos nacionales y tres ponencias en congresos internacionales. Los detalles de todo esto pueden verse en el anexo F.

REFERENCIAS

Adamopoulos, G. I. y Pappis, C. P. (1998). Scheduling under a common due-date on parallel unrelated machines. *European Journal of Operational Research*, 105(3):494–501.

Allahverdi, A. y Mittenthal, J. (1994). Scheduling on m parallel machines subject to random breakdowns to minimize expected mean flow time. *Naval Research Logistics*, 41(5):677–682.

Armacost, R. y Salem, A. (2001). Unrelated parallel-machines scheduling with setup times and machine eligibility restrictions. *International Journal of Production Economics*, 70(3):215–226.

Baker, K. R. (1974). *Intoduction to Sequencing and Scheduling*. John Wiley & Sons, New York.

Blazewicz, J., Ecker, K., Pesch, E., Schmidt, G., y Weglarz, J. (1996). *Scheduling Computer and Manufacturing Processes*. Springer-Verlag, Berlin.

Blazewicz, J., Lenstra, J., y Kan, A. R. (1983). Scheduling subject to constraints: Classification and complexity. *Discrete Applied Mathematics*, 5(1):11–24.

Brucker, P. (1995). *Scheduling Algorithms*. Springer-Verlag, Berlin.

Brucker, P. (2005). *Complex Scheduling*. Springer-Verlag, Berlin.

Chen, Z. L. y Li, C. L. (2008). Scheduling with subcontracting options. IIE *Transactions*, 40(12):1171–1184.

Cheng, T. C. E. y Sin, C. C. S. (1990). A state-of-the-art review of parallel-machine scheduling research. *European Journal of Operational Research*, 47(3):271–292.

Chrétienne, P., Coffman, E. G., Lenstra, J. K., y Liu, Z. (1995). *Scheduling Theory and its Applications*. John Wiley & Sons, New York.

Coffman, E. G., Garey, M. R., y Johnson, D. S. (1978). An application of bin-packing to multiprocessor scheduling. *SIAM Journal on Computing*, 7(1):1–17.

Conway, R., Maxwell, W., y Miller, L. (1967). *Theory of Scheduling*. Addison-Wesley, Reading, Massachusetts.

Croce, F., Ghirardi, M., y Tadei, R. (2004). Recovering beam search: Enhancing the beam search approach for combinatorial optimization problems. *Journal of Heuristics*, 10(1):89–104.

Davis, E. y Jaffe, J. M. (1981). Algorithms for scheduling tasks on unrelated processors. *Journal of the ACM*, 28(4):721–736.

De, P. y Morton, T. E. (1980). Scheduling to minimise makespan on unequal parallel processors. *Management Science*, 11(4):586–602.

Dell'Amico, M., Iori, M., Martello, S., y Monaci, M. (2008). Heuristic and exact algorithms for the identical parallel machine scheduling problem. *INFORMS Journal on Computing*, 20(3):333–344.

Dell'Amico, M. y Martello, S. (1995). Optimal scheduling of tasks on identical parallel processors. *ORSA Journal on Computing*, 7(2):191–200.

Dell'Amico, M. y Martello, S. (2005). A note on exact algorithms for the identical parallel machine scheduling problem. *European Journal of Operational Research*, 160:576–578.

Dillenberger, C., Escudero, L. F., Wollensak, A., y Wu, Z. (1994). On practical resource-allocation for production planning and scheduling with period overlapping setups. *European Journal of Operational Research*, 75(2):275–286.

Dosa, G. y Epstein, L. (2008). Preemptive scheduling on a small number of hierarchical machines. *Information and Computation*, 206(5):602–619.

Epstein, L. y Favrholdt, L. M. (2005). Optimal non-preemptive semi-online scheduling on two related machines. *ACM Journal of Algorithms*, 57(1):49–73.

Fanjul-Peyro, L. y Ruiz, R. (2010a). Iterated greedy local search methods for unrelated parallel machine scheduling. *European Journal of Operational Research.*, 207(1):55–69.

Fanjul-Peyro, L. y Ruiz, R. (2010b). Size-reduction heuristics for the unrelated parallel machines scheduling problem. *Computers & Operations Research. In press*.

Finke, G., Lemaire, P., Proth, J. M., y Queyranne, M. (2009). Minimizing the number of machines for minimum length schedules. *European Journal of Operational Research*, 199(3):702–705.

Fishkin, A. V., Jansen, K., y Mastrolilli, M. (2008). Grouping techniques for scheduling problems: Simpler and faster. *Algorithmica*, 51(2):183–199.

French, S. (1982). *Sequencing and Scheduling: An Introduction to the Mathematics of the Job-Shop*. Ellis Horwood Limited, Chichester.

Friesen, D. K. (1984). Tighter bounds for multifit processor scheduling algorithm. *SIAM Journal on Computing*, 13(1):170–181.

Friesen, D. K. (1987). Tighter bounds for LPT scheduling on uniform processors. *SIAM Journal on Computing*, 16(3):554–560.

Friesen, D. K. y Langston, M. A. (1983). Bounds for multifit scheduling on uniform machines. *SIAM Journal on Computing*, 12(1):60–70.

Gairing, M., Monien, B., y Woclaw, A. (2007). A faster combinatorial approximation algorithm for scheduling unrelated parallel machines. *Theoretical Computer Science*, 380(1-2):87–99.

Gantt, H. (1910). *Work, Wages and Profit. Their influence on the cost of living*. The Engineering Magazine Co, New York.

Gantt, H. (1919). *Organizing for Work*. Harcourt, Brace and Howe, New York.

Garey, M. R. y Johnson, D. S. (1979). *Computers and Intractability:A Guide to the Theory of $\mathcal{NP}$-Completeness*. Freeman. San Francisco.

Ghirardi, M. y Potts, C. N. (2005). Makespan minimization for scheduling unrelated parallel machines: A recovering beam search approach. *European Journal of Operational Research*, 165(2):457–467.

Glass, C. A., Potts, C. N., y Shade, P. (1994). Unrelated parallel machine scheduling using local search. *Mathematical and Computer Modelling*, 20(2):41–52.

Gonzalez, T., Ibarra, O. H., y Sahni, S. (1977). Bounds for lpt schedules on uniform processors. *SIAM Journal on Computing*, 6(1):155–166.

Graham, R. L. (1966). Bounds for certain multiprocessor anomalies. *Bell System Technical Journal*, 45(1):1563–1581.

Graham, R. L. (1969). Bounds on multiprocessing timing anomalies. *SIAM Journal on Applied Mathematics*, 17(2):416–429.

Graham, R. L., Lawler, E. L., Lenstra, J. K., y Rinnooy Kan, A. H. G. (1979). Optimization and Approximation in Deterministic Sequencing and Scheduling: A Survey. *Annals of Discrete Mathematics*, 5(1):287–326.

Hansen, P. y Mladenovic, N. (2001). Variable neighborhood search: Principles and applications. *European Journal of Operational Research*, 130(3):449–467.

Hariri, A. M. A. y Potts, C. N. (1991). Heuristics for scheduling unrelated parallel machines. *Computers & Operations Research*, 18(3):323–331.

Herrmann, J., Proth, J. M., y Sauer, N. (1997). Heuristics for unrelated machine scheduling with precedence constraints. *European Journal of Operational Research*, 102(3):528–537.

Hochbaum, D. S. (1997). *Approximation Algorithms for NP-hard Problems*. PWS Publishing Co., Boston.

Hochbaum, D. S. y Shmoys, D. (1987). Using dual approximation algorithms for scheduling problems: Theoretical and practical results. *Journal of the ACM*, 34(1):144–162.

Hochbaum, D. S. y Shmoys, D. (1988). A polynomial approximation scheme for scheduling on uniform processors: using the dual approximation approach. *SIAM Journal on Computing*, 17(3):539–551.

Horowitz, E. y Sahni, S. (1976). Exact and approximate algorithms for scheduling nonidentical processors. *Journal of the ACM*, 23(2):317–327.

Ibarra, O. H. y Kim, C. E. (1977). Heuristic algorithms for scheduling independent tasks on nonidentical processors. *Journal of the ACM*, 24(2):280–289.

Jackson, J. R. (1956). An extension of johnson's results on job lot scheduling. *Naval Research Logistics Quarterly*, 3:201–203.

Jansen, K. y Porkolab, L. (1999). Linear time approximation schemes for scheduling malleable parallel tasks. *In Proceedings of the 10th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA'99)*, página 490–498.

Johnson, S. M. (1954). Optimal two- and three-stage production schedules with setup times included. *Naval Research Logistics Quarterly*, 1(1):61–68.

Karp, R. M. (1975). Reducibility among combinatorial problems. *The Journal of Symbolic Logic*, 40(4):618–619.

Kravchenko, S. A. y Werner, F. (2009). Minimizing the number of machines for scheduling jobs with equal processing times. *European Journal of Operational Research*, 199(2):595–600.

Lancia, G. (2000). Scheduling jobs with release dates and tails on two unrelated parallel machines to minimize the makespan. *European Journal of Operational Research*, 120(2):277–288.

Langston, M. A. (1981). *Processor scheduling with improved heuristic algorithms*. Tesis doctoral, Texas A&M University, College Station, Texas.

Lawler, E., Lenstra, J., y Kan, A. R. (1993). Sequencing and scheduling: Algorithms amd complexity. *Handbooks in Operations Research and Management Science*, 4(1):445–522.

Lenstra, J. K., Rinnooy Kan, A. H. G., y Brucker, P. (1977). Complexity of machine scheduling problems. *Annals of Discrete Mathematics*, 1(1):343–362.

Lenstra, J. K., Shmoys, D. B., y Tardos, E. (1990). Approximation algorithms for scheduling unrelated parallel machines. *Mathematical Programming*, 46(3):259–271.

Li, C. (2006). Scheduling unit-length jobs with machine eligibility restrictions. *European Journal of Operational Research*, 174(2):1325–1328.

Lin, Y. y Li, W. (2004). Parallel machine scheduling of machine-dependent jobs with unit-length. *European Journal of Operational Research*, 156(1):261–266.

Liu, J. W. S. y Liu, C. L. (1974). Bounds on scheduling algorithms for heterogeneous computing systems. *Information Processing, J. L. Rosenfeld, ed., North-Holland, Amsterdam*, 74(1):349–353.

Mansini, R. y Speranza, M. G. (1999). Heuristic algorithms for the portfolio selection problem with minimum transaction lots. *European Journal of Operational Research*, 114(2):219–233.

Martello, S., Soumis, F., y Toth, P. (1997). Exact and approximation algorithms for makespan minimization on unrelated parallel machines. *Discrete Applied Mathematics*, 75(2):169–188.

McNaughton, R. (1959). Scheduling with deadlines and loss functions. *Management Science*, 6(1):1–12.

Mokotoff, E. (2001). Parallel machine scheduling problems: A survey. *Asia-Pacific Journal of Operational Research*, 18(2):193–242.

Mokotoff, E. (2004). An exact algorithm for the identical parallel machine scheduling problem. *European Journal of Operational Research*, 152:758–769.

Mokotoff, E. y Chretienne, P. (2002). A cutting plane algorithm for the unrelated parallel machine scheduling problem. *European Journal of Operational Research*, 141(3):515–525.

Mokotoff, E. y Jimeno, J. L. (2002). Heuristics based on partial enumeration for the unrelated parallel processor scheduling problem. *Annals of Operations Research*, 117(1-4):133–150.

Montgomery, D. C. (2009). *Design and Analysis of Experiments*. Wiley, New York, septima edición.

Ovacik, I. M. y Uzsoy, R. (1996). Decomposition methods for scheduling semiconductor testing facilities. *International Journal of Flexible Manufacturing Systems*, 8(4):357–388.

Ow, P. S. y Morton, T. E. (1988). Filtered beam search in scheduling. *International Journal of Production Research*, 26(1):35–62.

Piersma, N. y van Dijk, W. (1996). A local search heuristic for unrelated parallel machine scheduling with efficient neighborhood search. *Mathematical and Computer Modelling*, 24(9):11–19.

Pinedo, M. (2008). *Scheduling: Theory, Algorithms and Systems*. Prentice Hall, New Jersey, tercera edición.

Pinedo, M. (2009). *Planning and Scheduling in Manufacturing and Services*. Springer Series in Operations Research and Financial Engineering, segunda edición.

Portmann, M. (1997). Scheduling methology: optimization and compusearch approches I. *The Planning ans Shedulig of Production Systems. Methodologies and Applications, London. Chapman & Hall*.

Potts, C. N. (1985). Analysis of a linear-programming heuristic for scheduling unrelated parallel machines. *Discrete Applied Mathematics*, 10(2):155–164.

Raghavan, P. y Thompson, C. D. (1987). Randomized rounding: a technique for provably good algorithms and algorithmic proofs. *Combinatorica*, 7(4):365–374.

Rinnooy Kan, A. H. G. (1976). *Machine Scheduling Problems: Classification, Complexity and Computations*. Martinus Nijhoff, The Hague.

Ruiz, R. y Stützle, T. (2007). A simple and effective iterated greedy algorithm for the permutation flowshop scheduling problem. *European Journal of Operational Research*, 177(3):2033–2049.

Sahni, S. (1976). Algorithms for scheduling independent tasks. *Journal of the ACM*, 23(1):116–127.

Schuurman, P. y Woeginger, G. (1999). Polynomial time approximation algorithms for machine scheduling: Ten open problems. *Journal of Scheduling*, 2(5):203–213.

Seiden, S., Sgall, J., y Woeginger, G. J. (2000). Semi-online scheduling with decreasing job sizes. *Operations Research Letters*, 27(5):215–221.

Shchepin, E. V. y Vakhania, N. (2005). An optimal rounding gives a better approximation for scheduling unrelated machines. *Operations Research Letters*, 33(2):127–133.

Shmoys, D. B. y Tardos, E. (1993). An approximation algorithm for the generalized assignment problem. *Mathematical Programming*, 62(3):461–474.

Smith, W. E. (1956). Various optimizers for single-stage production. *Naval Research Logistics Quarterly*, 3(1):59–66.

Sourd, F. (2001). Scheduling tasks on unrelated machines: Large neighborhood improvement procedures. *Journal of Heuristics*, 7(6):519–531.

Srivastava, B. (1998). An effective heuristic for minimising makespan on unrelated parallel machines. *Journal of the Operational Research Society*, 49(8):886–894.

Tsang, E. y Voudouris, C. (1997). Fast local search and guided local search and their application to british telecom's workforce scheduling problem. *Operations Research Letters*, 20(3):119–127.

Unlu, Y. y Mason, S. J. (2010). Evaluation of mixed integer programming formulations for non-preemptive parallel machine scheduling problems. *Computers & Industrial Engineering*, 58(4):785–800.

Vallada, E. y Ruiz, R. (2010). Genetic algorithms for the unrelated parallel machine scheduling problem with sequence dependent setup times. *European Journal of Operational Research. In review.*

Van De Velde, S. L. (1993). Duality based algorithms for scheduling unrelated parallel machines. *ORSA Journal on Computing*, 5(5):192–205.

Woclaw, A. (2006). *Scheduling Unrelated Parallel Machines Algorithms, Complexity, and Performance*. Tesis doctoral, Fakultat fur Elektrotechnik, Informatik und Mathematik der Universitat Paderborn, Deutschland.

Yu, L., Shih, H. M., Pfund, M., Carlyle, W. M., y Fowler, J. W. (2002). Scheduling of unrelated parallel machines: an application to PWB manufacturing. *IIE Transactions*, 34(11):921–931.

ANEXO A

TABLAS DE LOS ALGORITMOS INICIALES

En el presente anexo se detallan las tablas completas de todos los intervalos de tiempos de proceso objeto de estudio. En ellas se reflejan todas las instancias, una a una. Se designará con n el número de trabajos, con m el número de máquinas y con i la instancia que corresponda.
Las tablas muestras los resultados de CPLEX 11.0 y la media de las 5 replicas de los algoritmos iniciales propuestos tras 15 segundos, mientras que estos 15 segundos los emplea Partial en su segunda fase (lo que le lleva en ocasiones a tiempos totales de más de 60 segundos) y RBS utiliza el tiempo que precise hasta su finalización (llegando a tiempos superiores, en ocasiones, a los 100 segundos). Así mismo, en la última columna, se refleja el tiempo necesario para alcanzar el óptimo por CPLEX 11.0 dentro del margen de dos horas, o en caso de no alcanzar el óptimo en esas dos horas, el *Gap*, es decir, la diferencia, en tanto por cien, entre el $C_{\text{máx}}$ hallado, y la cota inferior, o *LB*, que vendrán reflejados en negrita y se han calculado según la formula:

$$Gap(\,\%) = \frac{C_{\text{máx}} - \mathit{LB}}{C_{\text{máx}}} \cdot 100 \tag{A.1}$$

A.1. Tabla completa para el intervalo U(1,100)

n	m	i	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir	$C_{\text{máx}}$	t. o **Gap**
100	10	1	0,00	0,00	1,71	0,00	0,00	0,17	0,00	117	5,89 seg
100	10	2	0,00	0,00	3,45	0,00	0,00	0,00	0,00	87	0,41 seg
100	10	3	0,00	1,02	2,04	0,00	0,00	0,20	0,00	98	7,16 seg
100	10	4	0,00	0,87	1,74	0,87	0,87	0,35	0,35	115	7,42 seg
100	10	5	0,00	0,96	1,92	0,58	0,96	0,58	0,96	104	8,84 seg
100	10	6	0,00	0,99	2,97	0,00	0,00	0,20	0,00	101	3,44 seg
100	10	7	0,00	0,00	3,51	0,00	0,00	0,00	0,00	114	10,7 seg
100	10	8	0,00	0,00	1,94	0,00	0,00	0,00	0,00	103	3,11 seg
100	10	9	0,00	0,00	2,63	0,00	0,00	0,53	0,18	114	1,84 seg
100	10	10	0,00	0,00	2,04	0,00	0,20	0,20	0,00	98	4,56 seg
100	20	1	0,00	0,00	3,33	0,00	0,67	0,00	1,33	30	4,58 seg
100	20	2	0,00	0,00	2,78	0,00	0,00	1,11	0,56	36	6,52 seg
100	20	3	0,00	0,00	0,00	0,00	0,00	0,00	0,00	32	4,61 seg
100	20	4	0,00	0,00	3,03	0,00	0,61	0,00	0,00	33	6,3 seg
100	20	5	0,00	0,00	3,23	0,65	0,00	0,65	1,29	31	3,5 seg
100	20	6	0,00	3,45	3,45	2,07	3,45	3,45	3,45	29	4,55 seg
100	20	7	0,00	2,86	2,86	0,57	1,14	1,71	1,14	35	11,58 seg
100	20	8	0,00	0,00	3,33	1,33	3,33	1,33	2,67	30	1,58 seg
100	20	9	0,00	0,00	3,33	0,00	0,00	0,00	0,67	30	2,2 seg
100	20	10	0,00	0,00	3,45	0,00	0,69	2,07	0,69	29	0,59 seg
100	30	1	0,00	5,26	0,00	0,00	0,00	0,00	0,00	19	0,34 seg
100	30	2	0,00	0,00	0,00	0,00	0,00	0,00	0,00	23	0,41 seg
100	30	3	0,00	0,00	6,25	6,25	6,25	6,25	6,25	16	1,17 seg
100	30	4	0,00	0,00	0,00	0,00	0,00	2,22	0,00	18	0,34 seg
100	30	5	0,00	6,25	0,00	0,00	2,50	0,00	0,00	16	1,33 seg
100	30	6	0,00	0,00	0,00	0,00	0,00	0,00	0,00	24	0,36 seg
100	30	7	0,00	0,00	5,88	0,00	5,88	0,00	2,35	17	2,11 seg

n	m	i	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir	$C_{\text{máx}}$	t. o **Gap**
100	30	8	0,00	0,00	0,00	0,00	0,00	0,00	0,00	19	0,48 seg
100	30	9	0,00	12,50	0,00	0,00	0,00	0,00	0,00	16	0,42 seg
100	30	10	0,00	5,88	0,00	0,00	0,00	0,00	0,00	17	0,38 seg
100	40	1	0,00	10,00	10,00	4,00	14,00	10,00	10,00	10	0,36 seg
100	40	2	0,00	8,33	0,00	0,00	3,33	0,00	0,00	12	0,45 seg
100	40	3	0,00	16,67	8,33	1,67	16,67	8,33	8,33	12	0,34 seg
100	40	4	0,00	30,00	0,00	0,00	0,00	0,00	0,00	10	0,41 seg
100	40	5	0,00	9,09	9,09	9,09	9,09	9,09	9,09	11	0,51 seg
100	40	6	0,00	0,00	0,00	0,00	0,00	0,00	0,00	12	0,34 seg
100	40	7	0,00	0,00	0,00	0,00	0,00	0,00	0,00	14	0,39 seg
100	40	8	0,00	0,00	0,00	0,00	0,00	0,00	0,00	13	0,41 seg
100	40	9	0,00	0,00	0,00	0,00	0,00	0,00	0,00	23	0,34 seg
100	40	10	0,00	0,00	0,00	0,00	0,00	0,00	0,00	14	0,39 seg
100	50	1	0,00	12,50	0,00	0,00	5,00	0,00	0,00	8	0,31 seg
100	50	2	0,00	0,00	0,00	0,00	0,00	0,00	0,00	13	0,31 seg
100	50	3	0,00	0,00	0,00	0,00	0,00	0,00	0,00	10	0,39 seg
100	50	4	0,00	0,00	0,00	0,00	0,00	0,00	0,00	11	0,34 seg
100	50	5	0,00	0,00	0,00	0,00	0,00	0,00	0,00	9	0,33 seg
100	50	6	0,00	10,00	0,00	0,00	0,00	0,00	0,00	10	0,28 seg
100	50	7	0,00	12,50	0,00	12,50	12,50	12,50	12,50	8	0,34 seg
100	50	8	0,00	0,00	0,00	0,00	0,00	0,00	0,00	11	0,38 seg
100	50	9	0,00	11,11	0,00	0,00	11,11	0,00	0,00	9	0,39 seg
100	50	10	0,00	33,33	0,00	0,00	0,00	0,00	0,00	9	0,42 seg
200	10	1	0,49	0,00	0,98	0,10	0,20	0,39	0,29	204	1023,5 seg
200	10	2	0,00	0,00	2,45	0,29	0,59	0,39	0,49	204	28,63 seg
200	10	3	0,00	1,02	1,02	0,51	0,61	0,71	0,61	197	11,66 seg
200	10	4	0,00	0,00	1,03	0,82	0,92	0,92	1,03	195	9,66 seg
200	10	5	0,00	0,92	1,83	0,64	0,46	0,83	0,46	218	28,19 seg
200	10	6	0,55	0,00	0,55	0,22	0,00	0,22	0,00	183	177,38 seg

n	m	i	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir	$C_{\text{máx}}$	t. o **Gap**
200	10	7	0,00	0,00	1,06	0,00	0,00	0,32	0,11	189	5,88 seg
200	10	8	0,00	0,51	1,01	0,10	0,10	0,00	0,20	198	305,55 seg
200	10	9	0,51	0,51	1,52	0,41	0,30	0,41	0,30	197	199,63 seg
200	10	10	0,49	0,49	1,94	0,58	0,68	0,78	0,68	206	58,92 seg
200	20	1	1,82	0,00	0,00	0,00	0,36	0,00	0,73	55	236,03 seg
200	20	2	1,72	1,72	3,45	1,38	1,72	1,03	1,38	58	90,34 seg
200	20	3	0,00	0,00	1,89	1,13	1,13	0,75	1,51	53	4,27 seg
200	20	4	1,67	1,67	3,33	3,00	3,33	3,33	3,00	60	78,25 seg
200	20	5	1,82	0,00	0,00	1,45	1,09	1,82	0,36	55	503,16 seg
200	20	6	1,69	3,39	1,69	2,37	1,69	2,37	1,69	59	102,91 seg
200	20	7	3,45	1,72	3,45	3,45	3,10	3,45	3,45	58	409,78 seg
200	20	8	1,79	1,79	3,57	1,79	1,79	1,79	1,79	56	30,86 seg
200	20	9	1,72	1,72	1,72	0,34	1,38	1,03	1,03	58	40,92 seg
200	20	10	1,85	1,85	3,70	1,85	1,85	2,59	1,85	54	58,88 seg
200	30	1	3,70	3,70	3,70	3,70	4,44	4,44	3,70	27	17,03 seg
200	30	2	3,70	0,00	3,70	2,96	3,70	2,96	3,70	27	27,42 seg
200	30	3	0,00	3,57	3,57	1,43	1,43	2,86	2,86	28	62,67 seg
200	30	4	0,00	0,00	0,00	2,22	0,74	2,22	0,00	27	28,48 seg
200	30	5	0,00	4,17	4,17	2,50	4,17	4,17	3,33	24	8,76 seg
200	30	6	0,00	0,00	3,23	0,65	1,94	1,94	0,65	31	22,52 seg
200	30	7	0,00	0,00	3,70	0,00	1,48	0,00	0,00	27	45,75 seg
200	30	8	3,57	0,00	3,57	3,57	3,57	3,57	3,57	28	23,27 seg
200	30	9	6,90	3,45	6,90	3,45	3,45	4,14	4,14	29	35,25 seg
200	30	10	3,33	3,33	3,33	3,33	3,33	3,33	3,33	30	55,31 seg
200	40	1	5,88	0,00	0,00	1,18	5,88	1,18	3,53	17	101,41 seg
200	40	2	5,88	5,88	5,88	5,88	5,88	8,24	5,88	17	50,3 seg
200	40	3	0,00	0,00	6,25	2,50	6,25	2,50	1,25	16	8,94 seg
200	40	4	0,00	0,00	0,00	0,00	3,53	4,71	0,00	17	5,55 seg
200	40	5	0,00	0,00	0,00	0,00	0,00	0,00	0,00	19	0,84 seg

n	m	i	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir	$C_{\text{máx}}$	t. o **Gap**
200	40	6	0,00	0,00	0,00	0,00	2,22	0,00	0,00	18	11,16 seg
200	40	7	0,00	0,00	5,56	5,56	5,56	5,56	5,56	18	27,69 seg
200	40	8	5,88	5,88	5,88	5,88	7,06	7,06	5,88	17	34,8 seg
200	40	9	7,14	0,00	7,14	7,14	7,14	7,14	7,14	14	17,17 seg
200	40	10	5,56	5,56	11,11	5,56	5,56	5,56	5,56	18	27,61 seg
200	50	1	0,00	0,00	7,69	0,00	4,62	0,00	4,62	13	5,83 seg
200	50	2	0,00	0,00	0,00	0,00	0,00	0,00	0,00	16	1,45 seg
200	50	3	0,00	16,67	8,33	8,33	8,33	8,33	8,33	12	9,73 seg
200	50	4	0,00	8,33	8,33	6,67	6,67	8,33	6,67	12	9,27 seg
200	50	5	0,00	7,69	0,00	7,69	7,69	7,69	7,69	13	12,95 seg
200	50	6	0,00	0,00	0,00	0,00	3,33	1,67	0,00	12	21,91 seg
200	50	7	0,00	7,69	0,00	0,00	0,00	1,54	0,00	13	5,81 seg
200	50	8	0,00	16,67	8,33	8,33	8,33	8,33	8,33	12	11,22 seg
200	50	9	7,69	7,69	7,69	7,69	12,31	7,69	9,23	13	17,92 seg
200	50	10	8,33	8,33	8,33	8,33	8,33	8,33	8,33	12	32,81 seg
500	10	1	0,20	0,40	1,00	0,72	0,72	0,84	0,72	501	117,92 seg
500	10	2	0,41	0,21	1,03	0,66	0,62	0,54	0,45	484	27,78 seg
500	10	3	0,00	0,40	0,40	0,40	0,40	0,52	0,44	500	120,86 seg
500	10	4	0,21	0,21	0,43	0,51	0,51	0,38	0,47	469	1249,16 seg
500	10	5	0,40	0,20	0,40	0,52	0,44	0,56	0,36	503	265,88 seg
500	10	6	0,20	0,40	1,81	0,60	0,68	0,64	0,64	497	139,45 seg
500	10	7	0,44	0,66	1,10	0,57	0,48	0,57	0,40	455	217,17 seg
500	10	8	0,43	0,21	1,71	0,77	0,73	0,64	0,86	467	254,11 seg
500	10	9	0,41	0,20	0,41	0,29	0,37	0,37	0,33	490	**0,2 %**
500	10	10	0,21	0,21	1,45	0,70	0,66	0,66	0,62	483	321,47 seg
500	20	1	0,74	0,74	1,48	2,37	2,37	3,11	1,93	135	138,55 seg
500	20	2	1,54	0,77	1,54	2,46	2,77	2,77	2,00	130	171,59 seg
500	20	3	1,46	1,46	1,46	2,92	2,77	2,63	2,19	137	3641,05 seg
500	20	4	2,17	1,45	1,45	2,46	2,61	2,46	2,17	138	6097,81 seg

n	m	i	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir	$C_{\text{máx}}$	t. o **Gap**
500	20	5	1,36	0,68	1,36	2,04	2,04	2,04	1,50	147	**0,68 %**
500	20	6	1,49	1,49	1,49	2,24	2,39	1,94	1,49	134	3142,56 seg
500	20	7	0,00	0,76	1,52	2,27	2,27	2,27	2,12	132	13,75 seg
500	20	8	0,74	0,00	1,47	2,06	1,47	2,06	1,47	136	**0,74 %**
500	20	9	1,46	1,46	1,46	2,19	2,48	2,48	2,19	137	386,7 seg
500	20	10	1,61	0,81	1,61	1,77	1,77	1,77	1,29	124	**0,81 %**
500	30	1	3,23	3,23	1,61	3,23	3,23	3,55	3,23	62	52,48 seg
500	30	2	4,41	2,94	2,94	4,41	4,12	4,12	2,94	68	382,5 seg
500	30	3	1,49	1,49	1,49	3,58	3,28	3,88	2,99	67	1647,09 seg
500	30	4	6,06	6,06	3,03	5,76	5,15	5,15	3,94	66	4540,97 seg
500	30	5	0,00	0,00	0,00	2,99	2,09	2,69	1,49	67	**1,49 %**
500	30	6	3,13	1,56	1,56	2,81	2,81	3,13	1,88	64	**1,56 %**
500	30	7	4,92	3,28	1,64	4,59	4,92	4,92	3,61	61	4455,2 seg
500	30	8	3,03	1,52	3,03	4,55	3,94	4,85	3,03	66	874,26 seg
500	30	9	4,76	1,59	3,17	4,76	4,76	5,08	3,81	63	1795,39 seg
500	30	10	2,94	1,47	1,47	3,82	3,53	4,71	2,65	68	33,11 seg
500	40	1	2,56	5,13	2,56	3,59	3,59	5,13	3,08	39	103,73 seg
500	40	2	2,56	5,13	2,56	4,62	5,13	6,15	4,10	39	1129,17 seg
500	40	3	2,56	0,00	2,56	4,10	3,59	4,62	2,56	39	1109,48 seg
500	40	4	5,13	2,56	2,56	5,13	4,62	5,13	4,10	39	57,75 seg
500	40	5	7,69	5,13	2,56	5,13	5,13	5,13	5,13	39	672,38 seg
500	40	6	5,26	5,26	2,63	5,26	5,26	6,84	4,74	38	2097,22 seg
500	40	7	5,13	5,13	0,00	3,59	2,56	3,59	2,56	39	**2,56 %**
500	40	8	2,50	0,00	0,00	2,50	2,50	3,50	1,50	40	**2,5 %**
500	40	9	0,00	0,00	0,00	2,56	2,56	2,56	2,56	39	**2,56 %**
500	40	10	8,11	8,11	2,70	5,41	4,86	5,95	4,86	37	100,14 seg
500	50	1	3,85	3,85	0,00	4,62	5,38	6,92	3,85	26	380,31 seg
500	50	2	3,70	3,70	0,00	6,67	4,44	7,41	3,70	27	37,38 seg
500	50	3	3,85	3,85	0,00	3,85	3,85	6,15	3,85	26	28,25 seg

n	m	i	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir	$C_{\text{máx}}$	t. o **Gap**
500	50	4	7,41	11,11	3,70	7,41	7,41	7,41	3,70	27	96,84 seg
500	50	5	7,69	7,69	3,85	6,15	4,62	7,69	3,85	26	540,67 seg
500	50	6	7,41	11,11	3,70	7,41	7,41	9,63	7,41	27	1302,88 seg
500	50	7	3,57	3,57	0,00	3,57	3,57	3,57	2,86	28	**3,57 %**
500	50	8	7,69	7,69	3,85	8,46	7,69	9,23	7,69	26	210,52 seg
500	50	9	3,70	3,70	0,00	5,19	3,70	5,19	3,70	27	**3,7 %**
500	50	10	7,41	7,41	3,70	7,41	7,41	8,89	7,41	27	1195,7 seg
1000	10	1	0,10	0,10	1,43	0,49	0,53	0,57	0,51	977	1547,67 seg
1000	10	2	0,11	0,11	0,55	0,46	0,49	0,46	0,44	905	253,53 seg
1000	10	3	0,20	0,10	0,20	0,39	0,39	0,41	0,37	978	6374,25 seg
1000	10	4	0,10	0,21	0,21	0,35	0,40	0,40	0,29	960	1149,09 seg
1000	10	5	0,10	0,00	1,04	0,35	0,33	0,37	0,33	964	18,05 seg
1000	10	6	0,00	0,10	1,33	0,45	0,37	0,37	0,31	974	**0,1 %**
1000	10	7	0,32	3,40	0,42	0,38	0,36	0,36	0,36	942	295,28 seg
1000	10	8	0,40	0,81	0,81	0,48	0,51	0,46	0,42	990	1840,55 seg
1000	10	9	0,53	0,32	0,53	0,45	0,43	0,38	0,45	937	1662,3 seg
1000	10	10	0,00	0,20	0,81	0,34	0,34	0,32	0,28	986	**0,1 %**
1000	20	1	1,16	0,78	0,78	1,16	1,24	1,16	1,16	258	127,17 seg
1000	20	2	1,12	1,12	0,75	1,12	0,90	1,27	0,97	267	**0,37 %**
1000	20	3	1,58	1,58	0,40	1,11	1,11	1,19	0,87	253	**0,4 %**
1000	20	4	1,88	0,75	1,13	1,58	1,65	1,58	1,35	266	595,28 seg
1000	20	5	1,45	1,09	1,09	0,94	1,09	1,09	1,01	276	**0,36 %**
1000	20	6	1,12	0,75	1,12	1,95	1,87	1,95	1,65	267	2495,17 seg
1000	20	7	1,43	0,72	0,72	1,29	1,36	1,15	1,15	279	**0,36 %**
1000	20	8	0,74	0,00	1,47	1,54	1,76	1,91	1,62	272	77,2 seg
1000	20	9	1,50	1,50	1,12	1,42	1,42	1,27	1,05	267	**0,37 %**
1000	20	10	1,54	0,77	2,31	2,23	2,15	2,08	2,08	260	1028,27 seg
1000	30	1	1,55	0,78	0,78	2,64	2,79	3,10	2,17	129	**0,78 %**
1000	30	2	2,19	2,19	1,46	2,63	2,48	2,77	2,19	137	**0,73 %**

n	m	i	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir	$C_{\text{máx}}$	t. o **Gap**
1000	30	3	4,07	1,63	1,63	2,60	2,60	2,76	2,44	123	678,8 seg
1000	30	4	2,38	0,79	2,38	3,65	3,65	4,13	3,02	126	2894 seg
1000	30	5	1,56	2,34	1,56	2,97	2,97	3,13	2,34	128	**0,78 %**
1000	30	6	3,20	2,40	2,40	3,68	3,20	3,52	3,20	125	538,53 seg
1000	30	7	1,56	3,13	0,78	3,91	3,28	3,91	2,50	128	66,34 seg
1000	30	8	2,38	1,59	1,59	3,17	3,33	3,49	2,86	126	4969,34 seg
1000	30	9	1,56	3,13	1,56	3,44	2,97	3,44	3,13	128	3109,19 seg
1000	30	10	3,15	2,36	1,57	3,78	3,62	3,94	3,15	127	100,09 seg
1000	40	1	6,76	5,41	2,70	5,95	5,41	5,95	5,14	74	2443,94 seg
1000	40	2	4,11	4,11	1,37	4,11	4,11	4,93	3,29	73	1828,64 seg
1000	40	3	5,19	3,90	2,60	4,94	4,42	5,19	4,42	77	603,69 seg
1000	40	4	4,00	5,33	1,33	4,80	4,53	5,33	3,73	75	3375,47 seg
1000	40	5	2,60	3,90	2,60	4,94	4,94	4,68	3,90	77	130,86 seg
1000	40	6	5,33	4,00	2,67	5,33	5,33	5,33	4,27	75	265,11 seg
1000	40	7	0,00	3,90	0,00	2,60	2,60	2,86	2,08	77	**1,3 %**
1000	40	8	5,33	4,00	2,67	5,87	5,07	5,87	5,07	75	1839,72 seg
1000	40	9	5,26	3,95	1,32	4,47	4,47	5,00	3,68	76	428,06 seg
1000	40	10	4,05	2,70	2,70	5,14	5,41	4,86	4,05	74	7100,34 seg
1000	50	1	5,88	5,88	3,92	6,67	6,67	7,45	5,88	51	602,16 seg
1000	50	2	8,16	10,20	2,04	6,12	6,53	7,35	5,71	49	2833,36 seg
1000	50	3	3,92	1,96	1,96	5,88	5,88	6,67	4,31	51	986,56 seg
1000	50	4	5,88	5,88	1,96	6,67	5,88	7,84	5,49	51	4656,75 seg
1000	50	5	3,92	0,00	1,96	5,88	5,49	5,88	3,92	51	328,14 seg
1000	50	6	5,88	0,00	0,00	3,92	3,92	3,92	1,96	51	**1,96 %**
1000	50	7	8,00	6,00	2,00	6,00	6,00	6,00	4,80	50	3050,95 seg
1000	50	8	11,32	5,66	0,00	4,53	4,53	5,28	3,02	53	**1,89 %**
1000	50	9	5,56	5,56	1,85	5,56	5,56	6,30	4,81	54	1499,34 seg
1000	50	10	1,92	3,85	0,00	5,38	4,62	5,38	3,85	52	194,44 seg

Tabla A.1 – Tabla completa para el intervalo U(1,100).

A.2. Tabla completa para el intervalo U(10,100)

n	m	i	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir	$C_{\text{máx}}$	t. o **Gap**
100	10	1	0,00	1,01	2,02	0,51	0,40	0,61	0,30	198	8,98 seg
100	10	2	0,00	0,00	0,58	0,00	0,00	0,00	0,00	173	0,53 seg
100	10	3	0,00	0,00	2,19	0,00	0,00	0,00	0,11	183	16,44 seg
100	10	4	0,58	0,58	1,16	0,69	0,58	0,46	0,35	173	40,92 seg
100	10	5	0,00	0,00	1,67	1,11	0,89	1,11	1,11	180	3,78 seg
100	10	6	0,00	0,57	2,29	0,80	0,57	1,37	0,69	175	5,59 seg
100	10	7	0,00	0,00	0,00	0,00	0,00	0,11	0,00	175	1482,23 seg
100	10	8	0,00	0,56	0,56	0,67	0,56	0,78	0,45	179	17,72 seg
100	10	9	0,00	0,58	1,17	0,58	0,58	0,58	0,58	171	3,3 seg
100	10	10	0,00	0,57	2,84	0,57	0,45	0,68	0,11	176	3,41 seg
100	20	1	0,00	0,00	1,41	1,41	1,69	1,41	1,41	71	4,08 seg
100	20	2	0,00	0,00	2,70	1,35	1,35	1,35	1,35	74	189,3 seg
100	20	3	2,70	1,35	2,70	0,00	0,54	1,08	0,00	74	**1,35 %**
100	20	4	0,00	0,00	2,70	0,81	1,62	1,89	1,08	74	1747,55 seg
100	20	5	0,00	1,41	2,82	0,00	0,00	0,00	0,00	71	**1,41 %**
100	20	6	1,27	1,27	3,80	1,27	1,27	1,27	1,27	79	**1,27 %**
100	20	7	1,37	0,00	1,37	1,37	1,37	1,10	0,82	73	**1,37 %**
100	20	8	1,33	0,00	4,00	1,87	1,87	1,60	1,33	75	**1,33 %**
100	20	9	0,00	0,00	2,70	0,00	0,00	0,00	0,00	74	2945,86 seg
100	20	10	1,27	1,27	2,53	1,77	2,78	2,53	1,27	79	24,3 seg
100	30	1	0,00	6,38	4,26	1,70	3,40	1,28	1,70	47	8,72 seg
100	30	2	4,26	0,00	4,26	0,00	2,98	1,70	0,00	47	**2,13 %**
100	30	3	8,51	0,00	4,26	2,13	2,13	2,13	1,28	47	**4,26 %**
100	30	4	0,00	0,00	6,25	2,08	2,92	1,67	1,67	48	**2,08 %**
100	30	5	0,00	2,22	6,67	1,33	3,56	1,33	1,33	45	11,2 seg
100	30	6	0,00	2,04	4,08	2,04	2,45	2,04	2,04	49	9,11 seg
100	30	7	4,26	0,00	2,13	0,43	2,13	0,85	0,00	47	**4,26 %**

n	m	i	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir	$C_{\text{máx}}$	t. o **Gap**
100	30	8	2,22	2,22	4,44	4,44	4,44	4,00	4,00	45	210,17 seg
100	30	9	2,17	0,00	4,35	2,17	2,61	2,17	2,17	46	**2,17 %**
100	30	10	0,00	0,00	4,17	0,42	4,17	1,67	0,42	48	326,98 seg
100	40	1	0,00	0,00	5,88	2,94	8,24	3,53	2,94	34	7,36 seg
100	40	2	0,00	0,00	0,00	0,57	4,57	0,57	1,14	35	25,33 seg
100	40	3	0,00	0,00	2,94	2,94	5,88	3,53	2,94	34	9,47 seg
100	40	4	0,00	0,00	5,88	3,53	6,47	5,29	3,53	34	8,38 seg
100	40	5	0,00	2,94	5,88	5,29	5,88	5,88	5,29	34	9,88 seg
100	40	6	0,00	0,00	2,78	1,11	3,33	1,67	0,00	36	**2,78 %**
100	40	7	0,00	2,94	5,88	4,12	5,88	4,12	3,53	34	7,83 seg
100	40	8	0,00	2,94	8,82	3,53	5,88	4,71	2,94	34	8,14 seg
100	40	9	2,94	0,00	2,94	1,18	4,12	2,94	1,76	34	68,8 seg
100	40	10	0,00	5,88	8,82	2,94	5,88	2,94	2,94	34	9,97 seg
100	50	1	0,00	0,00	10,71	0,00	0,00	0,00	0,00	28	6,33 seg
100	50	2	0,00	0,00	14,81	1,48	8,15	5,19	2,22	27	3,41 seg
100	50	3	0,00	0,00	7,14	0,00	0,00	0,00	0,00	28	4 seg
100	50	4	3,33	0,00	3,33	0,67	3,33	2,67	0,67	30	16,23 seg
100	50	5	0,00	0,00	3,33	0,00	0,67	0,00	0,00	30	3,59 seg
100	50	6	0,00	0,00	6,67	0,00	3,33	0,00	0,00	30	3,72 seg
100	50	7	0,00	0,00	3,45	0,00	3,45	0,00	0,00	29	3,88 seg
100	50	8	0,00	0,00	3,33	0,00	2,67	0,00	0,00	30	**3,33 %**
100	50	9	0,00	6,67	6,67	3,33	4,67	3,33	3,33	30	5,52 seg
100	50	10	0,00	0,00	7,14	0,00	7,14	3,57	0,00	28	3,58 seg
200	10	1	0,00	0,00	0,56	0,45	0,39	0,34	0,17	358	**0,28 %**
200	10	2	0,54	0,54	1,09	0,71	0,60	0,54	0,49	368	897,08 seg
200	10	3	0,28	0,28	0,55	0,66	0,44	0,55	0,39	361	62,38 seg
200	10	4	0,27	0,27	0,54	0,54	0,59	0,59	0,43	372	**0,27 %**
200	10	5	0,28	0,00	0,84	0,50	0,45	0,39	0,22	359	**0,28 %**
200	10	6	0,53	0,53	0,53	0,59	0,48	0,48	0,32	376	**0,27 %**

n	m	i	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir	$C_{\text{máx}}$	t. o **Gap**
200	10	7	0,00	0,00	0,56	0,17	0,00	0,23	0,06	355	**0,28 %**
200	10	8	0,00	0,56	0,56	0,33	0,33	0,22	0,11	359	**0,28 %**
200	10	9	0,00	0,55	1,66	0,66	0,61	0,44	0,44	361	**0,28 %**
200	10	10	0,57	0,00	1,15	0,63	0,52	0,52	0,34	349	3915,67 seg
200	20	1	0,69	1,38	1,38	1,93	1,93	1,38	1,24	145	903,45 seg
200	20	2	0,69	0,69	2,08	1,81	2,22	1,39	1,39	144	**0,69 %**
200	20	3	2,08	0,69	0,69	2,22	2,36	1,67	1,67	144	**0,69 %**
200	20	4	2,04	0,68	1,36	2,04	1,63	1,09	0,95	147	**0,68 %**
200	20	5	2,01	0,67	1,34	1,21	0,67	0,54	0,27	149	**2,01 %**
200	20	6	0,69	0,69	1,39	1,53	1,53	0,97	0,97	144	**0,69 %**
200	20	7	0,69	2,08	1,39	1,53	1,67	1,11	0,83	144	**0,69 %**
200	20	8	4,26	1,42	2,13	1,13	1,70	0,99	0,71	141	**0,71 %**
200	20	9	1,37	0,68	1,37	1,51	1,37	1,37	0,96	146	**0,68 %**
200	20	10	2,04	0,00	2,04	1,77	1,36	0,95	0,41	147	**1,36 %**
200	30	1	4,60	1,15	3,45	2,76	3,45	2,53	1,84	87	**2,3 %**
200	30	2	3,45	4,60	1,15	1,84	2,76	1,38	1,15	87	**2,3 %**
200	30	3	2,30	5,75	2,30	2,76	2,99	2,07	1,84	87	**2,3 %**
200	30	4	3,37	2,25	2,25	2,47	3,37	2,02	2,92	89	**1,12 %**
200	30	5	3,33	2,22	3,33	3,33	4,00	2,44	2,67	90	**1,11 %**
200	30	6	3,45	3,45	2,30	2,53	2,76	1,38	1,84	87	**2,3 %**
200	30	7	2,30	0,00	1,15	2,07	2,30	1,15	1,38	87	**3,45 %**
200	30	8	2,27	1,14	1,14	2,27	2,50	1,59	1,36	88	**2,27 %**
200	30	9	1,14	2,27	2,27	2,27	2,73	1,59	1,36	88	**2,27 %**
200	30	10	3,41	1,14	2,27	2,05	3,18	1,82	1,82	88	**2,27 %**
200	40	1	4,62	1,54	3,08	3,69	4,31	3,08	3,38	65	**3,08 %**
200	40	2	6,35	1,59	3,17	2,54	4,13	2,22	1,90	63	**3,17 %**
200	40	3	6,35	3,17	4,76	4,76	5,71	3,49	3,49	63	**3,17 %**
200	40	4	4,69	3,13	0,00	2,50	3,13	1,56	2,19	64	**4,69 %**
200	40	5	3,17	3,17	1,59	2,22	2,86	1,90	2,22	63	**1,59 %**

n	m	i	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir	$C_{\text{máx}}$	t. o **Gap**
200	40	6	3,08	4,62	3,08	2,46	4,62	2,77	2,77	65	**3,08 %**
200	40	7	4,69	0,00	0,00	2,50	3,75	1,88	1,88	64	**3,13 %**
200	40	8	6,25	0,00	3,13	2,81	4,06	2,19	2,50	64	**3,13 %**
200	40	9	6,06	0,00	3,03	1,52	2,73	1,21	1,21	66	**4,55 %**
200	40	10	4,92	3,28	4,92	4,26	5,90	4,26	4,26	61	2948,94 seg
200	50	1	3,92	1,96	1,96	3,14	4,71	1,96	2,35	51	**5,88 %**
200	50	2	4,00	4,00	2,00	3,20	4,40	2,40	2,80	50	**4 %**
200	50	3	4,00	2,00	2,00	3,60	5,60	2,40	4,00	50	**4 %**
200	50	4	3,92	0,00	1,96	0,00	1,96	0,00	0,78	51	**7,84 %**
200	50	5	4,00	2,00	2,00	2,40	3,20	1,60	1,60	50	**6 %**
200	50	6	3,92	0,00	0,00	0,78	2,35	0,78	0,39	51	**7,84 %**
200	50	7	5,88	1,96	1,96	3,92	5,49	2,75	2,75	51	**5,88 %**
200	50	8	2,04	4,08	4,08	3,67	6,12	3,27	3,67	49	17,22 seg
200	50	9	5,88	0,00	0,00	1,57	3,92	1,96	1,57	51	**5,88 %**
200	50	10	6,00	2,00	2,00	2,00	4,00	2,00	2,80	50	**6 %**
500	10	1	0,58	0,23	0,23	0,32	0,37	0,26	0,35	862	**0,12 %**
500	10	2	0,23	0,23	0,80	0,62	0,64	0,64	0,59	876	3471,83 seg
500	10	3	0,23	0,34	0,34	0,48	0,39	0,46	0,32	879	**0,11 %**
500	10	4	0,34	0,23	0,57	0,54	0,52	0,48	0,43	884	6587,78 seg
500	10	5	0,23	0,35	0,81	0,49	0,44	0,32	0,37	864	206,78 seg
500	10	6	0,11	0,33	0,22	0,33	0,40	0,24	0,22	901	**0,22 %**
500	10	7	0,45	0,00	0,45	0,34	0,29	0,25	0,25	895	**0,22 %**
500	10	8	0,11	0,22	0,43	0,46	0,46	0,37	0,37	922	**0,11 %**
500	10	9	0,22	0,11	0,76	0,41	0,44	0,41	0,37	916	**0,11 %**
500	10	10	0,22	0,22	0,56	0,54	0,52	0,47	0,49	893	318,25 seg
500	20	1	0,85	1,13	0,56	1,41	1,36	1,07	1,07	354	**0,28 %**
500	20	2	0,58	0,87	1,45	1,50	1,50	1,16	1,16	346	**0,29 %**
500	20	3	0,56	0,56	0,84	1,17	1,23	0,95	0,89	359	**0,28 %**
500	20	4	1,14	1,14	1,14	1,71	1,65	1,25	1,20	351	3376,27 seg

n	m	i	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir	$C_{\text{máx}}$	t. o **Gap**
500	20	5	1,44	0,57	1,72	1,72	1,32	1,38	1,32	348	**0,29 %**
500	20	6	1,13	0,56	1,13	1,36	1,36	1,13	1,02	354	**0,28 %**
500	20	7	1,14	0,57	1,43	1,26	1,26	1,14	1,03	350	2201,09 seg
500	20	8	1,14	0,86	0,57	1,31	1,14	0,91	0,91	350	**0,29 %**
500	20	9	0,56	0,84	0,56	1,24	1,12	0,84	0,79	356	**0,56 %**
500	20	10	1,13	1,13	0,85	1,35	1,13	1,18	1,01	355	**0,28 %**
500	30	1	1,89	1,89	1,42	2,26	2,36	1,70	1,70	212	**0,47 %**
500	30	2	2,39	1,44	0,96	2,49	2,68	2,11	2,20	209	**0,48 %**
500	30	3	2,35	2,35	0,94	2,63	2,72	1,97	1,97	213	**0,47 %**
500	30	4	2,88	1,92	0,96	2,60	2,40	1,92	1,92	208	**0,48 %**
500	30	5	0,96	0,96	0,96	1,72	1,91	1,34	1,34	209	**0,96 %**
500	30	6	0,47	0,47	0,47	1,70	1,98	1,42	1,42	212	**0,94 %**
500	30	7	1,88	0,94	1,41	2,16	2,25	1,97	1,78	213	**0,94 %**
500	30	8	1,87	1,40	0,93	1,96	2,15	2,06	1,87	214	**0,47 %**
500	30	9	1,90	0,47	1,42	2,18	2,18	1,90	1,61	211	**0,95 %**
500	30	10	1,42	2,37	0,47	1,99	2,09	1,80	1,61	211	**0,95 %**
500	40	1	2,68	4,03	1,34	2,82	3,09	2,28	2,68	149	**1,34 %**
500	40	2	2,00	1,33	2,00	3,33	3,33	2,80	2,80	150	**1,33 %**
500	40	3	3,33	3,33	2,00	3,47	3,87	3,07	3,33	150	**0,67 %**
500	40	4	2,01	2,68	1,34	2,95	3,22	2,82	2,82	149	**1,34 %**
500	40	5	2,65	1,99	1,32	3,05	2,78	2,25	2,25	151	**0,66 %**
500	40	6	2,67	2,00	1,33	2,67	2,67	2,13	2,53	150	**1,33 %**
500	40	7	2,63	3,29	1,32	2,89	3,42	2,37	2,63	152	**1,32 %**
500	40	8	2,61	3,27	0,65	2,88	2,88	2,48	2,48	153	**1,31 %**
500	40	9	2,70	2,03	1,35	2,84	3,24	2,43	2,43	148	**1,35 %**
500	40	10	1,99	2,65	0,66	3,18	3,58	2,52	2,52	151	**1,32 %**
500	50	1	5,17	5,17	1,72	4,31	4,83	3,97	3,97	116	**0,86 %**
500	50	2	3,39	2,54	1,69	3,90	4,24	2,88	3,56	118	**0,85 %**
500	50	3	2,56	4,27	1,71	3,93	4,27	2,91	3,42	117	**1,71 %**

n	m	i	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir	$C_{\text{máx}}$	t. o **Gap**
500	50	4	0,85	2,54	0,85	3,05	3,22	2,54	2,71	118	**1,69 %**
500	50	5	4,27	3,42	1,71	3,42	3,42	2,74	3,08	117	**1,71 %**
500	50	6	3,42	2,56	0,85	3,42	3,93	2,74	2,91	117	**1,71 %**
500	50	7	1,68	2,52	0,84	2,52	2,86	2,02	1,68	119	**2,52 %**
500	50	8	3,42	2,56	0,85	3,93	4,27	3,08	3,59	117	**1,71 %**
500	50	9	3,45	4,31	0,86	3,45	3,62	2,59	2,93	116	**1,72 %**
500	50	10	2,59	2,59	1,72	3,28	4,14	2,93	3,10	116	**1,72 %**
1000	10	1	0,28	0,34	0,28	0,22	0,19	0,20	0,17	1762	5544,78 seg
1000	10	2	0,34	1,14	0,29	0,30	0,30	0,27	0,26	1754	3011,72 seg
1000	10	3	0,22	0,00	0,39	0,24	0,20	0,19	0,20	1782	**0,06 %**
1000	10	4	0,23	0,17	0,45	0,32	0,31	0,27	0,27	1761	**0,06 %**
1000	10	5	0,11	0,44	0,44	0,21	0,19	0,17	0,15	1814	**0,06 %**
1000	10	6	0,06	0,06	0,28	0,28	0,26	0,26	0,23	1765	**0,06 %**
1000	10	7	0,17	0,33	0,17	0,29	0,23	0,22	0,24	1804	**0,06 %**
1000	10	8	0,17	1,91	0,68	0,27	0,24	0,21	0,21	1776	**0,06 %**
1000	10	9	0,29	2,24	0,57	0,28	0,25	0,26	0,24	1742	**0,06 %**
1000	10	10	0,06	0,69	0,34	0,24	0,24	0,23	0,19	1745	**0,06 %**
1000	20	1	0,99	0,57	0,43	0,74	0,68	0,63	0,54	704	**0,14 %**
1000	20	2	0,86	0,29	0,57	0,86	0,80	0,75	0,75	697	**0,14 %**
1000	20	3	1,01	0,43	0,72	0,80	0,80	0,72	0,72	696	**0,14 %**
1000	20	4	0,58	0,58	0,58	0,89	0,92	0,78	0,78	695	**0,14 %**
1000	20	5	0,58	0,87	0,43	0,78	0,70	0,72	0,64	690	**0,14 %**
1000	20	6	0,72	1,16	0,14	0,90	0,93	0,75	0,75	690	**0,14 %**
1000	20	7	0,86	0,58	0,43	0,86	0,75	0,78	0,66	695	**0,14 %**
1000	20	8	0,72	0,72	0,87	0,87	0,92	0,84	0,78	692	**0,14 %**
1000	20	9	1,01	0,86	0,14	0,86	0,84	0,84	0,81	694	**0,14 %**
1000	20	10	0,99	0,57	0,43	0,82	0,91	0,68	0,68	704	**0,14 %**
1000	30	1	1,42	0,94	0,71	1,46	1,56	1,18	1,23	424	**0,24 %**
1000	30	2	1,44	1,67	0,72	1,53	1,53	1,39	1,24	418	**0,24 %**

n	m	i	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir	$C_{\text{máx}}$	t. o **Gap**
1000	30	3	1,67	0,72	0,48	1,15	1,29	1,05	1,15	419	**0,48 %**
1000	30	4	1,43	0,24	0,95	1,43	1,52	1,24	1,28	421	**0,24 %**
1000	30	5	1,44	1,20	0,96	1,49	1,54	1,35	1,39	416	**0,24 %**
1000	30	6	0,94	1,42	0,47	1,51	1,37	1,32	1,32	424	**0,24 %**
1000	30	7	1,42	0,95	0,24	1,28	1,37	1,14	1,09	422	**0,47 %**
1000	30	8	1,43	0,48	0,71	1,57	1,33	1,29	1,05	420	**0,24 %**
1000	30	9	1,44	0,72	0,72	1,48	1,53	1,15	1,24	418	**0,24 %**
1000	30	10	1,19	1,43	0,48	1,48	1,62	1,38	1,43	420	**0,24 %**
1000	40	1	1,69	1,69	0,68	1,96	1,89	1,62	1,69	296	**0,34 %**
1000	40	2	2,03	2,03	1,02	2,17	2,10	1,69	1,97	295	**0,34 %**
1000	40	3	1,36	1,02	0,68	1,76	1,83	1,49	1,69	295	**0,68 %**
1000	40	4	2,02	1,01	0,67	1,89	1,95	1,82	1,68	297	**0,67 %**
1000	40	5	1,33	1,00	0,67	1,87	1,93	1,60	1,73	300	**0,67 %**
1000	40	6	1,33	2,67	0,67	1,87	2,00	1,73	1,87	300	**0,33 %**
1000	40	7	2,03	2,03	1,01	1,96	1,96	1,62	1,69	296	**0,34 %**
1000	40	8	1,70	2,04	1,02	1,97	2,24	1,77	1,70	294	**0,34 %**
1000	40	9	2,03	1,69	1,02	2,31	2,37	1,97	1,90	295	**0,34 %**
1000	40	10	2,00	2,00	0,67	2,27	2,13	1,80	1,80	300	**0,33 %**
1000	50	1	2,20	1,76	0,88	2,64	2,82	2,29	2,47	227	**0,88 %**
1000	50	2	2,19	1,75	0,88	2,63	2,72	2,46	2,37	228	**0,88 %**
1000	50	3	2,18	1,75	0,87	2,62	2,79	2,01	2,10	229	**0,87 %**
1000	50	4	1,74	1,74	0,87	2,17	2,17	2,09	2,09	230	**0,43 %**
1000	50	5	2,17	1,74	0,87	2,35	2,43	2,09	2,26	230	**0,87 %**
1000	50	6	2,62	3,06	1,31	2,71	3,14	2,36	2,45	229	**0,44 %**
1000	50	7	2,17	2,61	0,87	2,52	2,78	2,35	2,43	230	**0,43 %**
1000	50	8	1,74	2,17	0,43	2,17	2,35	2,00	1,83	230	**0,87 %**
1000	50	9	1,74	1,74	0,43	2,26	2,52	1,91	1,83	230	**0,87 %**
1000	50	10	1,76	2,20	0,88	2,73	3,00	2,20	2,47	227	**0,44 %**

Tabla A.2 – Tabla completa para el intervalo U(10,100).

A.3. Tabla completa para el intervalo U(100,200)

n	m	i	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir	$C_{\text{máx}}$	t. o **Gap**
100	10	1	0,27	0,18	0,54	0,43	0,41	0,40	0,25	1111	286,5 seg
100	10	2	0,00	0,09	0,83	0,48	0,37	0,52	0,24	1083	204,69 seg
100	10	3	0,00	0,09	1,10	0,29	0,13	0,29	0,13	1094	400,48 seg
100	10	4	0,09	0,18	0,72	0,09	0,09	0,14	0,09	1111	138,42 seg
100	10	5	0,00	0,09	0,45	0,42	0,45	0,42	0,35	1100	148,44 seg
100	10	6	0,00	0,09	0,82	0,33	0,22	0,24	0,18	1095	26,26 seg
100	10	7	0,00	0,09	2,34	0,27	0,16	0,27	0,18	1109	299,42 seg
100	10	8	0,09	0,00	2,26	0,63	0,33	0,43	0,24	1106	3607,81 seg
100	10	9	0,00	0,00	0,99	0,05	0,09	0,05	0,00	1108	99,45 seg
100	10	10	0,27	0,09	2,10	0,27	0,29	0,24	0,20	1095	3042,19 seg
100	20	1	0,38	0,57	1,13	0,45	0,38	0,72	0,42	529	**0,19 %**
100	20	2	0,19	0,74	1,67	0,19	0,19	0,19	0,19	540	152,59 seg
100	20	3	0,19	0,38	1,70	0,79	0,60	0,49	0,53	529	2224,48 seg
100	20	4	0,38	0,56	1,32	0,68	0,38	0,83	0,71	532	712,91 seg
100	20	5	0,00	0,00	0,00	0,00	0,00	0,00	0,00	541	3,94 seg
100	20	6	0,00	0,00	0,37	0,00	0,00	0,00	0,00	537	4,64 seg
100	20	7	0,56	0,75	0,75	0,64	0,53	0,83	0,64	533	386,02 seg
100	20	8	0,76	0,95	1,51	0,57	0,42	0,30	0,45	529	**0,19 %**
100	20	9	0,00	0,19	0,75	0,11	0,07	0,11	0,00	536	16,61 seg
100	20	10	1,32	0,57	1,51	0,49	0,38	0,26	0,30	530	1049,31 seg
100	30	1	2,22	2,46	0,99	0,39	0,25	0,30	0,05	406	**6,65 %**
100	30	2	1,96	2,94	0,98	0,10	0,05	0,15	0,00	408	**6,86 %**
100	30	3	2,71	3,20	0,74	0,25	0,00	0,05	0,00	406	**6,9 %**
100	30	4	1,23	4,66	0,74	0,29	0,25	0,25	0,25	408	**6,37 %**
100	30	5	2,70	2,95	1,23	0,00	0,00	-0,20	-0,20	407	**7,13 %**
100	30	6	0,74	1,23	1,23	0,10	0,05	0,30	0,05	406	**6,73 %**
100	30	7	2,70	2,95	0,74	0,15	0,10	0,20	-0,15	407	**6,88 %**

n	m	i	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir	$C_{\text{máx}}$	t. o **Gap**
100	30	8	0,50	2,23	1,49	0,30	0,25	0,15	0,05	404	**1,03 %**
100	30	9	1,23	2,72	0,99	0,25	0,25	0,20	0,20	405	**6,01 %**
100	30	10	2,97	2,72	0,99	0,25	0,20	0,25	0,10	404	**6,93 %**
100	40	1	2,63	1,32	1,32	0,66	0,72	0,66	0,46	304	**1,64 %**
100	40	2	2,62	4,26	0,98	0,52	0,66	0,52	0,26	305	1013,84 seg
100	40	3	0,98	3,61	0,98	0,66	0,66	0,52	0,39	305	**2,49 %**
100	40	4	0,66	1,97	0,99	0,39	0,66	0,33	0,33	304	**1,59 %**
100	40	5	1,32	2,96	0,99	0,92	0,99	0,86	0,79	304	**2,63 %**
100	40	6	0,98	1,31	0,66	0,20	0,33	0,13	0,20	305	**3,01 %**
100	40	7	0,66	1,97	1,31	0,39	0,66	0,33	0,33	305	**1,83 %**
100	40	8	2,29	0,98	0,65	0,07	0,20	0,07	0,00	306	**4,25 %**
100	40	9	0,98	2,62	0,66	0,52	0,66	0,39	0,26	305	**3,18 %**
100	40	10	0,66	3,30	1,32	0,40	0,59	0,40	0,33	303	**1,32 %**
100	50	1	0,00	0,00	0,48	0,00	0,48	0,38	0,67	208	11,25 seg
100	50	2	0,47	0,00	2,83	0,00	0,09	0,00	0,19	212	**1,42 %**
100	50	3	1,42	0,47	2,36	0,00	0,19	0,09	0,09	212	**0,94 %**
100	50	4	0,00	0,00	0,00	0,00	0,10	0,00	0,00	210	**0,48 %**
100	50	5	0,00	0,00	0,48	0,00	0,10	0,10	0,10	210	**0,95 %**
100	50	6	0,00	0,00	3,81	0,10	0,57	0,10	0,38	210	**0,95 %**
100	50	7	0,00	0,48	1,44	0,57	0,86	0,57	0,86	209	**0,48 %**
100	50	8	0,00	0,00	1,90	0,28	0,47	0,38	0,28	211	**0,47 %**
100	50	9	0,00	0,47	1,90	0,00	0,19	0,09	0,28	211	**0,65 %**
100	50	10	0,96	0,00	1,44	0,48	1,34	0,48	0,67	209	229,01 seg
200	10	1	2,10	0,87	1,64	0,42	0,33	0,29	0,39	2195	**0,36 %**
200	10	2	0,46	0,14	0,91	0,49	0,47	0,28	0,30	2189	**0,14 %**
200	10	3	0,27	0,18	0,69	0,31	0,16	0,33	0,32	2183	**0,05 %**
200	10	4	0,51	0,14	0,64	0,31	0,26	0,17	0,22	2177	384,3 seg
200	10	5	0,18	0,68	1,23	0,49	0,36	0,27	0,32	2200	2470,98 seg
200	10	6	0,60	0,28	0,78	0,27	0,16	0,07	0,15	2169	**0,14 %**

n	m	i	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir	$C_{\text{máx}}$	t. o **Gap**
200	10	7	0,41	0,23	0,87	0,41	0,30	0,23	0,24	2174	**0,05 %**
200	10	8	0,14	0,23	1,51	0,36	0,30	0,22	0,20	2181	325,25 seg
200	10	9	0,73	0,14	0,87	0,47	0,32	0,24	0,25	2180	**0,05 %**
200	10	10	1,50	1,41	1,46	0,47	0,33	0,19	0,47	2193	**0,23 %**
200	20	1	0,38	0,38	0,86	0,72	0,57	0,53	0,70	1050	**0,19 %**
200	20	2	0,28	0,38	1,71	0,78	0,72	0,42	0,66	1053	**0,19 %**
200	20	3	0,48	0,48	1,05	0,73	0,59	0,55	0,71	1046	**0,1 %**
200	20	4	0,28	1,70	1,33	0,59	0,45	0,34	0,55	1056	**0,19 %**
200	20	5	0,48	0,76	1,53	0,90	0,72	0,59	0,82	1049	**0,1 %**
200	20	6	0,47	0,28	1,23	0,87	0,72	0,57	0,65	1054	**0,19 %**
200	20	7	1,24	1,81	1,81	0,91	0,74	0,59	0,70	1052	**0,1 %**
200	20	8	0,28	0,28	1,33	0,40	0,27	0,13	0,34	1054	72,53 seg
200	20	9	0,57	1,61	1,99	0,83	0,76	0,59	0,74	1054	**0,19 %**
200	20	10	0,57	0,57	1,34	0,65	0,50	0,36	0,48	1047	**0,19 %**
200	30	1	0,00	1,65	-1,10	-0,85	-0,96	-1,21	-1,07	728	**3,02 %**
200	30	2	2,65	3,49	0,70	0,78	0,56	0,45	0,53	716	**1,26 %**
200	30	3	1,54	4,47	0,56	0,81	0,61	0,31	0,59	716	**1,4 %**
200	30	4	0,00	0,82	-1,23	-1,07	-1,23	-1,45	-1,29	729	**3,29 %**
200	30	5	1,40	3,65	0,70	0,70	0,53	0,36	0,53	713	**1,26 %**
200	30	6	0,00	0,96	-0,68	-0,60	-0,74	-1,04	-0,79	731	**3,01 %**
200	30	7	0,00	0,82	-1,23	-1,48	-1,53	-1,81	-1,72	731	**3,42 %**
200	30	8	1,12	4,05	0,84	0,98	0,81	0,50	0,67	716	**1,12 %**
200	30	9	2,65	3,91	0,84	0,92	0,75	0,59	0,73	716	**1,12 %**
200	30	10	2,09	3,49	0,70	0,95	0,84	0,36	0,59	717	**1,39 %**
200	40	1	0,97	0,39	0,97	0,85	0,74	0,50	0,85	515	**0,39 %**
200	40	2	0,58	-0,19	-0,19	0,42	0,42	0,12	0,46	518	**0,19 %**
200	40	3	0,78	0,97	0,78	1,09	0,89	0,62	0,97	514	**0,39 %**
200	40	4	0,78	0,39	1,17	1,13	0,93	0,70	1,09	514	**0,19 %**
200	40	5	1,16	0,39	2,70	1,00	0,69	0,39	0,81	519	**0,19 %**

n	m	i	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir	$C_{\text{máx}}$	t. o **Gap**
200	40	6	0,97	0,39	0,77	0,74	0,70	0,46	0,85	517	**0,39 %**
200	40	7	1,75	1,75	0,97	1,13	0,93	0,74	1,17	515	**0,39 %**
200	40	8	0,95	5,53	0,38	0,00	0,04	0,00	0,11	524	111,38 seg
200	40	9	0,59	0,00	0,59	0,90	0,66	0,55	0,98	512	**0,59 %**
200	40	10	1,55	0,97	0,77	1,01	1,16	0,74	1,04	517	863,59 seg
200	50	1	0,97	0,49	0,73	1,22	1,17	0,92	1,36	411	**0,49 %**
200	50	2	0,24	0,24	1,20	0,58	0,48	0,10	0,48	417	461,67 seg
200	50	3	0,97	0,49	1,70	1,22	1,17	0,83	1,17	411	**0,49 %**
200	50	4	1,70	0,24	1,46	1,02	0,78	0,49	0,97	411	**0,24 %**
200	50	5	0,73	0,97	1,45	0,97	1,02	0,63	1,07	413	**0,24 %**
200	50	6	0,97	0,24	0,97	0,97	0,97	0,68	1,22	411	**0,24 %**
200	50	7	0,73	0,24	0,48	0,63	0,73	0,29	0,53	413	**0,26 %**
200	50	8	0,73	2,19	0,73	1,17	0,88	0,63	0,97	411	**0,49 %**
200	50	9	0,48	0,73	1,21	0,87	0,68	0,48	0,92	413	**0,73 %**
200	50	10	0,97	0,97	1,21	0,83	0,73	0,49	0,92	412	**0,49 %**
500	10	1	0,16	0,22	0,48	0,29	0,18	0,12	0,14	5456	**0,04 %**
500	10	2	0,07	0,44	0,55	0,23	0,17	0,10	0,20	5439	**0,06 %**
500	10	3	0,75	0,15	0,55	0,23	0,22	0,14	0,20	5455	**0,07 %**
500	10	4	0,13	0,04	0,20	0,16	0,13	0,09	0,10	5423	**0,04 %**
500	10	5	0,20	0,07	0,62	0,23	0,18	0,14	0,21	5456	**0,05 %**
500	10	6	0,46	0,33	0,57	0,24	0,19	0,12	0,15	5453	**0,07 %**
500	10	7	0,39	0,18	0,33	0,31	0,22	0,16	0,21	5414	**0,09 %**
500	10	8	0,20	0,11	0,42	0,30	0,21	0,15	0,18	5421	**0,06 %**
500	10	9	0,09	0,46	0,55	0,23	0,19	0,10	0,15	5448	**0,11 %**
500	10	10	0,42	0,26	0,63	0,36	0,25	0,16	0,25	5437	**0,06 %**
500	20	1	1,38	0,11	0,84	0,65	0,52	0,44	0,53	2613	**0,08 %**
500	20	2	1,49	2,18	1,42	1,06	1,07	1,16	1,09	2612	**0,19 %**
500	20	3	1,87	1,41	1,76	0,92	0,72	0,80	0,84	2618	**0,15 %**
500	20	4	1,34	0,34	1,11	0,70	0,61	0,38	0,54	2618	**0,08 %**

n	m	i	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir	$C_{\text{máx}}$	t. o **Gap**
500	20	5	0,23	0,15	0,53	0,57	0,52	0,42	0,54	2624	**0,08 %**
500	20	6	0,92	0,50	0,92	0,70	0,59	0,47	0,67	2613	**0,08 %**
500	20	7	1,07	0,65	0,88	0,77	0,67	0,48	0,58	2610	**0,04 %**
500	20	8	0,88	0,19	0,77	0,59	0,51	0,39	0,57	2613	**0,04 %**
500	20	9	0,99	0,57	1,34	0,99	0,71	0,83	0,89	2618	**0,08 %**
500	20	10	0,35	0,46	0,85	0,58	0,48	0,42	0,50	2603	**0,08 %**
500	30	1	1,04	1,67	0,52	0,74	0,69	0,60	0,59	1738	**1,15 %**
500	30	2	0,86	1,66	0,52	0,73	0,71	0,51	0,68	1742	**1,15 %**
500	30	3	0,86	1,09	0,46	0,70	0,67	0,45	0,57	1744	**1,26 %**
500	30	4	0,92	1,84	0,52	0,81	0,75	0,58	0,66	1739	**1,09 %**
500	30	5	0,92	1,21	0,52	0,69	0,71	0,58	0,61	1738	**0,98 %**
500	30	6	1,04	1,32	0,58	0,88	0,85	0,66	0,74	1736	**1,04 %**
500	30	7	0,69	1,50	0,52	0,68	0,66	0,52	0,59	1736	**1,15 %**
500	30	8	0,98	0,81	0,63	0,87	0,74	0,56	0,70	1738	**0,98 %**
500	30	9	1,04	1,73	0,81	0,98	0,93	0,69	0,82	1734	**0,92 %**
500	30	10	0,80	1,84	0,63	0,74	0,69	0,52	0,68	1740	**1,03 %**
500	40	1	0,23	2,50	0,23	0,32	0,23	0,18	0,21	1322	**2,87 %**
500	40	2	0,08	2,19	0,08	0,29	0,20	0,06	0,17	1323	**2,95 %**
500	40	3	0,84	2,66	0,61	0,76	0,73	0,56	0,71	1316	**2,43 %**
500	40	4	0,61	1,90	0,46	0,71	0,65	0,49	0,56	1318	**2,5 %**
500	40	5	0,45	2,12	0,30	0,53	0,47	0,39	0,45	1320	**2,65 %**
500	40	6	0,53	2,66	0,38	0,65	0,61	0,47	0,53	1318	**2,5 %**
500	40	7	0,61	0,91	0,53	0,79	0,71	0,61	0,64	1316	**2,74 %**
500	40	8	0,38	1,44	0,08	0,26	0,21	0,03	0,21	1324	**2,95 %**
500	40	9	0,68	2,50	0,30	0,58	0,62	0,38	0,46	1318	**2,73 %**
500	40	10	0,38	2,66	0,23	0,50	0,49	0,32	0,39	1318	**2,58 %**
500	50	1	0,59	0,98	0,69	0,79	0,75	0,47	0,75	1019	**0,29 %**
500	50	2	0,98	0,78	0,78	0,92	0,98	0,74	0,90	1021	**0,39 %**
500	50	3	0,49	0,88	0,49	0,80	0,77	0,55	0,77	1019	**0,29 %**

n	m	i	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir	$C_{\text{máx}}$	t. o **Gap**
500	50	4	0,78	1,18	0,78	0,94	0,94	0,61	0,96	1020	**0,29 %**
500	50	5	0,00	0,39	0,39	0,45	0,43	0,21	0,39	1024	**0,78 %**
500	50	6	0,39	0,69	0,49	0,78	0,78	0,61	0,72	1021	**0,29 %**
500	50	7	1,27	1,37	1,08	0,86	0,88	0,55	0,78	1021	**0,39 %**
500	50	8	0,98	1,37	0,59	0,88	0,94	0,63	0,88	1020	**0,29 %**
500	50	9	0,79	1,28	0,98	1,00	1,02	0,71	0,94	1019	**0,2 %**
500	50	10	0,59	0,88	0,59	0,96	0,92	0,69	0,92	1020	**0,2 %**
1000	10	1	0,11	0,11	0,21	0,14	0,14	0,10	0,12	10880	**0,02 %**
1000	10	2	0,36	0,45	0,20	0,17	0,13	0,10	0,14	10805	**0,02 %**
1000	10	3	0,23	0,47	0,23	0,16	0,14	0,08	0,14	10880	**0,03 %**
1000	10	4	0,17	0,16	0,23	0,11	0,09	0,06	0,12	10861	**0,01 %**
1000	10	5	0,19	0,26	0,33	0,12	0,14	0,10	0,11	10867	**0,02 %**
1000	10	6	0,17	0,40	0,10	0,13	0,11	0,08	0,10	10873	**0,01 %**
1000	10	7	0,31	0,14	0,15	0,11	0,11	0,08	0,10	10845	**0,02 %**
1000	10	8	0,26	0,68	0,28	0,11	0,08	0,06	0,08	10894	**0,03 %**
1000	10	9	0,23	0,03	0,19	0,14	0,13	0,09	0,10	10841	**0,03 %**
1000	10	10	0,06	0,41	0,28	0,15	0,13	0,11	0,12	10888	**0,02 %**
1000	20	1	0,81	0,94	0,63	0,57	0,53	0,45	0,49	5215	**0,13 %**
1000	20	2	0,94	0,88	0,69	0,54	0,52	0,38	0,52	5222	**0,1 %**
1000	20	3	0,81	0,52	0,52	0,60	0,53	0,49	0,61	5210	**0,15 %**
1000	20	4	0,84	0,75	0,69	0,42	0,34	0,26	0,33	5221	**0,1 %**
1000	20	5	0,65	0,63	0,63	0,43	0,42	0,36	0,44	5229	**0,06 %**
1000	20	6	0,69	0,59	0,59	0,38	0,33	0,29	0,33	5222	**0,1 %**
1000	20	7	0,84	0,69	0,61	0,55	0,49	0,42	0,52	5236	**0,11 %**
1000	20	8	0,84	0,40	0,44	0,51	0,51	0,42	0,49	5229	**0,1 %**
1000	20	9	0,98	1,03	0,77	0,53	0,52	0,36	0,46	5223	**0,13 %**
1000	20	10	0,65	0,23	0,82	0,46	0,39	0,38	0,40	5214	**0,04 %**
1000	30	1	0,90	0,66	0,58	0,62	0,65	0,54	0,65	3460	**0,84 %**
1000	30	2	0,89	1,01	0,72	0,70	0,69	0,57	0,63	3467	**0,84 %**

n	m	i	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir	$C_{\text{máx}}$	t. o **Gap**
1000	30	3	0,66	0,61	0,40	0,53	0,55	0,46	0,51	3462	**1,07 %**
1000	30	4	0,90	1,07	0,67	0,70	0,74	0,67	0,71	3457	**0,84 %**
1000	30	5	0,75	1,62	0,52	0,63	0,62	0,55	0,56	3462	**0,95 %**
1000	30	6	0,75	0,78	0,46	0,58	0,66	0,51	0,58	3462	**1,01 %**
1000	30	7	0,58	0,58	0,43	0,58	0,61	0,50	0,55	3459	**0,81 %**
1000	30	8	0,90	1,19	0,55	0,78	0,80	0,70	0,77	3454	**0,75 %**
1000	30	9	1,21	0,84	0,61	0,72	0,76	0,69	0,72	3460	**0,87 %**
1000	30	10	0,84	0,93	0,69	0,75	0,80	0,73	0,76	3456	**0,81 %**
1000	40	1	0,78	3,17	0,94	0,74	0,74	0,62	0,68	2552	**0,08 %**
1000	40	2	0,82	1,02	0,67	0,58	0,68	0,50	0,59	2551	**0,12 %**
1000	40	3	1,17	0,90	0,74	0,74	0,67	0,54	0,69	2555	**0,08 %**
1000	40	4	0,90	0,86	0,67	0,67	0,61	0,57	0,64	2553	**0,08 %**
1000	40	5	0,82	0,70	0,59	0,68	0,71	0,55	0,60	2555	**0,08 %**
1000	40	6	0,86	0,94	0,71	0,76	0,78	0,67	0,75	2552	**0,04 %**
1000	40	7	1,14	1,84	0,59	0,69	0,70	0,58	0,74	2555	**0,12 %**
1000	40	8	1,06	0,71	0,71	0,72	0,74	0,62	0,67	2553	**0,08 %**
1000	40	9	1,14	0,98	0,70	0,74	0,69	0,60	0,64	2554	**0,08 %**
1000	40	10	1,14	0,67	0,67	0,62	0,64	0,55	0,63	2552	**0,08 %**
1000	50	1	0,98	0,83	0,74	0,72	0,80	0,56	0,64	2036	**0,2 %**
1000	50	2	0,89	0,84	0,69	0,74	0,76	0,65	0,71	2033	**0,15 %**
1000	50	3	0,89	1,08	0,79	0,76	0,85	0,65	0,74	2033	**0,1 %**
1000	50	4	0,59	0,54	0,44	0,69	0,81	0,53	0,68	2034	**0,1 %**
1000	50	5	0,59	0,69	0,54	0,65	0,74	0,50	0,62	2034	**0,15 %**
1000	50	6	0,59	0,74	0,44	0,56	0,63	0,47	0,60	2034	**0,2 %**
1000	50	7	0,54	0,59	0,39	0,66	0,67	0,58	0,61	2035	**0,25 %**
1000	50	8	0,00	-0,05	-0,34	-0,19	-0,20	-0,35	-0,25	2054	**1,02 %**
1000	50	9	1,03	0,93	0,88	0,73	0,78	0,67	0,82	2037	**0,15 %**
1000	50	10	1,03	0,69	0,79	0,73	0,70	0,54	0,61	2035	**0,15 %**

Tabla A.3 – Tabla completa para el intervalo U(100,200).

A.4. Tabla completa para el intervalo U(100,120)

n	m	i	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir	$C_{\text{máx}}$	t. o **Gap**
100	10	1	0,00	0,30	0,20	0,12	0,08	0,10	0,08	1014	6,36 seg
100	10	2	0,20	0,10	0,39	0,14	0,16	0,10	0,08	1018	**0,1 %**
100	10	3	0,00	0,20	0,20	0,12	0,10	0,08	0,06	1016	32,34 seg
100	10	4	0,10	0,10	0,20	0,02	0,02	0,00	0,00	1018	**0,1 %**
100	10	5	0,00	0,00	0,39	0,06	0,00	0,00	0,00	1019	0,72 seg
100	10	6	0,10	0,79	0,20	0,16	0,12	0,10	0,10	1015	20,8 seg
100	10	7	0,00	0,10	0,10	0,06	0,00	0,00	0,00	1016	32,33 seg
100	10	8	0,20	0,29	0,20	0,16	0,14	0,10	0,10	1017	42 seg
100	10	9	0,00	0,00	0,29	0,12	0,10	0,02	0,08	1018	4,33 seg
100	10	10	0,00	0,00	0,20	0,10	0,10	0,10	0,10	1015	2,98 seg
100	20	1	0,00	0,20	0,20	0,20	0,20	0,12	0,20	506	12,05 seg
100	20	2	0,40	0,20	0,20	0,20	0,20	0,00	0,16	505	**0,2 %**
100	20	3	0,20	0,20	0,20	0,20	0,20	0,20	0,20	505	846,97 seg
100	20	4	0,00	0,79	0,00	0,00	0,00	0,04	0,00	505	**0,2 %**
100	20	5	0,20	0,20	0,20	0,16	0,20	0,00	0,12	506	149,33 seg
100	20	6	0,00	0,20	0,59	0,00	0,00	0,00	0,00	508	12 seg
100	20	7	0,40	0,20	0,40	0,24	0,28	0,20	0,20	505	446,74 seg
100	20	8	0,59	0,79	0,00	0,00	0,00	0,00	0,00	507	**0,39 %**
100	20	9	0,20	0,00	0,40	0,12	0,12	0,00	0,16	505	77,63 seg
100	20	10	0,00	0,00	0,00	0,00	0,00	0,00	0,00	508	4,08 seg
100	30	1	0,00	0,25	0,25	0,10	0,25	0,00	0,00	400	13,48 seg
100	30	2	0,00	0,00	0,25	0,25	0,25	0,15	0,25	400	15,3 seg
100	30	3	0,25	0,50	0,25	0,15	0,25	0,00	0,00	400	**4,02 %**
100	30	4	0,00	0,25	0,00	0,00	0,00	0,00	0,00	400	15,45 seg
100	30	5	0,00	1,00	0,25	0,00	0,00	0,00	0,00	400	11,83 seg
100	30	6	0,00	0,25	0,25	0,25	0,25	0,00	0,15	400	9,86 seg
100	30	7	0,25	0,25	0,00	0,00	0,00	0,00	0,00	400	**4,41 %**

n	m	i	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir	$C_{\text{máx}}$	t. o **Gap**
100	30	8	0,50	0,75	0,25	0,25	0,25	0,15	0,25	400	**5,09 %**
100	30	9	0,25	0,00	0,00	0,25	0,25	0,00	0,10	400	**3,34 %**
100	30	10	0,25	0,25	0,25	0,20	0,25	0,05	0,15	400	**4,01 %**
100	40	1	0,00	0,67	0,00	0,00	0,00	0,00	0,00	300	21,86 seg
100	40	2	0,00	0,67	0,00	0,00	0,00	0,00	0,00	300	21,59 seg
100	40	3	0,00	0,00	0,00	0,00	0,00	0,00	0,00	300	25,03 seg
100	40	4	0,00	0,33	0,33	0,27	0,33	0,00	0,00	300	23,84 seg
100	40	5	0,00	0,00	0,00	0,33	0,33	0,00	0,13	300	29,06 seg
100	40	6	0,00	0,33	0,00	0,00	0,00	0,00	0,00	300	24,88 seg
100	40	7	0,00	0,33	0,00	0,00	0,00	0,00	0,00	300	31,84 seg
100	40	8	0,00	0,67	0,00	0,27	0,33	0,00	0,07	300	18,64 seg
100	40	9	0,33	0,67	0,33	0,33	0,33	0,00	0,20	300	**1,76 %**
100	40	10	0,00	0,33	0,33	0,27	0,33	0,00	0,00	300	17,11 seg
100	50	1	0,00	1,00	1,49	0,40	0,40	0,00	0,50	201	0,3 seg
100	50	2	0,00	0,99	0,00	0,00	0,00	0,00	0,00	202	32,84 seg
100	50	3	0,00	1,99	0,00	0,30	0,30	0,00	0,40	201	0,27 seg
100	50	4	0,00	0,50	0,00	0,00	0,00	0,00	0,00	202	36,81 seg
100	50	5	0,00	0,99	0,50	0,00	0,00	0,00	0,00	202	22,27 seg
100	50	6	0,00	0,50	0,50	0,00	0,00	0,00	0,00	202	44,41 seg
100	50	7	0,00	0,50	0,00	0,00	0,00	0,00	0,00	202	28,83 seg
100	50	8	0,00	0,50	0,00	0,00	0,00	0,00	0,00	202	24,67 seg
100	50	9	0,00	1,49	0,00	0,00	0,00	0,00	0,00	202	26,91 seg
100	50	10	0,50	1,00	0,50	0,50	0,50	0,30	0,50	201	26,22 seg
200	10	1	0,05	0,05	0,15	0,05	0,05	0,04	0,04	2031	60,22 seg
200	10	2	0,20	0,34	0,10	0,07	0,04	0,03	0,04	2030	**0,05 %**
200	10	3	0,05	0,25	0,20	0,08	0,06	0,05	0,05	2036	184,2 seg
200	10	4	0,05	0,05	0,15	0,13	0,10	0,06	0,07	2028	126,22 seg
200	10	5	0,05	0,39	0,20	0,06	0,05	0,02	0,03	2037	**0,05 %**
200	10	6	0,00	0,15	0,15	0,06	0,05	0,00	0,03	2028	**0,05 %**

n	m	i	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir	$C_{\text{máx}}$	t. o **Gap**
200	10	7	0,20	0,05	0,15	0,10	0,09	0,08	0,09	2031	524,98 seg
200	10	8	0,05	0,05	0,30	0,12	0,10	0,05	0,06	2030	405,75 seg
200	10	9	0,29	0,00	0,34	0,10	0,08	0,02	0,05	2034	**0,05 %**
200	10	10	0,05	0,00	0,15	0,05	0,06	0,04	0,05	2029	59,23 seg
200	20	1	0,20	0,20	0,20	0,12	0,10	0,10	0,10	1007	294,67 seg
200	20	2	0,30	0,30	0,30	0,18	0,18	0,10	0,12	1008	**0,1 %**
200	20	3	0,10	0,20	0,10	0,10	0,10	0,10	0,10	1008	**0,1 %**
200	20	4	0,30	0,00	0,30	0,12	0,14	0,10	0,12	1007	**0,1 %**
200	20	5	0,40	0,30	0,20	0,10	0,10	0,00	0,10	1008	**0,1 %**
200	20	6	0,30	0,10	0,20	0,20	0,20	0,12	0,20	1007	204,59 seg
200	20	7	0,40	0,10	0,30	0,18	0,18	0,10	0,18	1009	**0,1 %**
200	20	8	0,10	0,40	0,10	0,12	0,12	0,02	0,10	1007	**0,1 %**
200	20	9	0,20	0,79	0,30	0,20	0,18	0,10	0,18	1008	176,74 seg
200	20	10	0,10	0,20	0,20	0,18	0,16	0,08	0,14	1007	2303,34 seg
200	30	1	0,57	0,71	0,14	0,14	0,14	0,11	0,14	701	**0,57 %**
200	30	2	0,43	0,28	0,00	0,14	0,14	0,03	0,11	702	**0,71 %**
200	30	3	0,43	0,85	0,00	0,03	0,03	0,00	0,00	702	**0,85 %**
200	30	4	0,57	0,43	0,14	0,17	0,20	0,14	0,14	701	**0,57 %**
200	30	5	1,00	0,85	0,00	0,03	0,00	0,00	0,00	702	**0,71 %**
200	30	6	0,14	0,28	0,00	0,00	0,03	0,00	0,00	702	**0,71 %**
200	30	7	0,43	0,71	0,00	0,09	0,14	0,00	0,00	702	**0,85 %**
200	30	8	0,28	1,28	0,00	0,14	0,11	0,00	0,06	702	**0,71 %**
200	30	9	0,43	0,57	0,29	0,29	0,29	0,14	0,20	701	**0,71 %**
200	30	10	0,14	0,85	-0,14	0,00	-0,03	-0,14	0,00	703	**1 %**
200	40	1	0,00	0,00	0,00	0,00	0,04	0,00	0,00	503	595,14 seg
200	40	2	0,00	0,00	0,00	0,00	0,00	0,00	0,00	504	1688,64 seg
200	40	3	0,00	0,00	0,00	0,00	0,08	0,00	0,00	503	892,19 seg
200	40	4	0,00	0,20	0,00	0,20	0,20	0,20	0,20	502	13,55 seg
200	40	5	0,20	0,40	0,20	0,20	0,32	0,20	0,20	502	22,48 seg

n	m	i	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir	$C_{\text{máx}}$	t. o **Gap**
200	40	6	0,00	0,00	0,20	0,00	0,00	0,00	0,04	503	28,27 seg
200	40	7	0,20	0,20	0,00	0,00	0,00	0,00	0,00	503	18,31 seg
200	40	8	0,40	0,80	0,20	0,20	0,40	0,20	0,24	502	752,25 seg
200	40	9	0,20	0,20	0,00	0,20	0,20	0,04	0,16	502	514,56 seg
200	40	10	0,20	0,00	0,00	0,00	0,08	0,00	0,00	503	1462,52 seg
200	50	1	0,25	0,50	0,00	0,25	0,25	0,25	0,25	401	92,95 seg
200	50	2	0,00	0,50	0,25	0,05	0,05	0,00	0,15	402	400,56 seg
200	50	3	0,00	0,50	0,00	0,00	0,00	0,00	0,00	403	1998,61 seg
200	50	4	0,00	0,25	0,00	0,05	0,10	0,00	0,05	402	397,33 seg
200	50	5	0,00	0,00	0,25	0,05	0,10	0,00	0,10	402	362,28 seg
200	50	6	0,00	0,25	0,25	0,20	0,20	0,00	0,25	402	261,05 seg
200	50	7	0,00	0,25	0,00	0,00	0,00	0,00	0,00	402	712,61 seg
200	50	8	0,00	0,50	0,00	0,00	0,00	0,00	0,00	402	459,83 seg
200	50	9	0,25	0,25	0,00	0,00	0,00	0,00	0,00	403	**0,5 %**
200	50	10	0,00	0,25	0,00	0,00	0,00	0,00	0,00	402	294,16 seg
500	10	1	0,04	0,22	0,37	0,08	0,06	0,03	0,08	5076	**0,02 %**
500	10	2	0,14	0,41	0,18	0,08	0,08	0,05	0,06	5074	1960,34 seg
500	10	3	0,10	0,16	0,28	0,07	0,08	0,05	0,06	5080	29,22 seg
500	10	4	0,22	0,47	0,18	0,05	0,04	0,03	0,04	5080	44,34 seg
500	10	5	0,39	0,71	0,39	0,10	0,07	0,04	0,07	5081	**0,02 %**
500	10	6	0,18	0,12	0,20	0,08	0,07	0,05	0,06	5074	142,42 seg
500	10	7	0,12	0,16	0,16	0,06	0,04	0,04	0,04	5072	56,77 seg
500	10	8	0,18	0,43	0,20	0,07	0,07	0,06	0,09	5072	24,84 seg
500	10	9	0,04	0,12	0,16	0,07	0,06	0,04	0,06	5074	38,67 seg
500	10	10	0,12	0,08	0,20	0,05	0,04	0,03	0,04	5072	57,55 seg
500	20	1	0,20	0,08	0,28	0,10	0,10	0,05	0,08	2517	**0,04 %**
500	20	2	0,12	0,16	0,12	0,10	0,10	0,08	0,09	2513	4306,58 seg
500	20	3	0,20	0,16	0,20	0,12	0,11	0,10	0,12	2516	**0,04 %**
500	20	4	0,16	0,36	0,12	0,08	0,09	0,04	0,08	2515	**0,04 %**

n	m	i	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir	$C_{\text{máx}}$	t. o **Gap**
500	20	5	0,08	0,20	0,20	0,10	0,10	0,06	0,10	2516	91,23 seg
500	20	6	0,12	0,20	0,12	0,07	0,08	0,06	0,08	2516	**0,04 %**
500	20	7	0,28	0,32	0,16	0,12	0,10	0,08	0,08	2516	**0,04 %**
500	20	8	0,20	0,12	0,28	0,12	0,12	0,08	0,11	2514	172,17 seg
500	20	9	0,16	0,20	0,16	0,09	0,09	0,05	0,07	2516	**0,04 %**
500	20	10	0,12	0,48	0,20	0,11	0,10	0,07	0,10	2517	**0,04 %**
500	30	1	1,18	0,35	0,12	0,19	0,20	0,16	0,18	1702	**0,41 %**
500	30	2	0,18	0,29	0,12	0,18	0,18	0,14	0,18	1702	**0,53 %**
500	30	3	0,41	0,41	0,18	0,19	0,19	0,15	0,18	1702	**0,53 %**
500	30	4	0,18	0,35	0,06	0,22	0,24	0,18	0,18	1702	**0,53 %**
500	30	5	0,41	0,29	0,18	0,24	0,24	0,18	0,21	1702	**0,53 %**
500	30	6	0,35	0,53	0,12	0,21	0,20	0,16	0,18	1702	**0,47 %**
500	30	7	0,35	1,00	0,06	0,12	0,12	0,07	0,12	1703	**0,41 %**
500	30	8	0,23	0,35	0,06	0,07	0,08	0,05	0,06	1704	**0,59 %**
500	30	9	0,47	0,35	0,12	0,18	0,21	0,18	0,18	1702	**0,47 %**
500	30	10	0,18	0,29	0,06	0,07	0,08	0,04	0,06	1705	**0,65 %**
500	40	1	0,00	1,00	-0,08	0,00	0,00	-0,06	0,00	1302	**1,61 %**
500	40	2	0,00	0,00	-0,15	0,00	0,00	-0,05	-0,02	1302	**1,61 %**
500	40	3	0,08	1,31	0,15	0,15	0,15	0,11	0,15	1300	98,38 seg
500	40	4	0,00	1,31	0,08	0,15	0,15	0,09	0,12	1300	84,23 seg
500	40	5	0,31	1,92	0,15	0,22	0,23	0,15	0,15	1300	**1,62 %**
500	40	6	0,08	0,31	0,08	0,15	0,15	0,14	0,15	1300	**1,38 %**
500	40	7	0,15	0,31	-0,15	-0,08	-0,08	-0,11	-0,08	1303	**1,69 %**
500	40	8	0,08	0,38	-0,08	0,08	0,08	0,02	0,08	1301	**1,61 %**
500	40	9	0,00	0,54	-0,08	0,00	0,00	-0,08	-0,03	1302	**1,92 %**
500	40	10	0,00	1,53	-0,15	-0,08	-0,08	-0,08	-0,08	1303	**1,3 %**
500	50	1	0,10	0,10	0,00	0,10	0,10	0,04	0,10	1003	**0,1 %**
500	50	2	0,20	0,10	0,10	0,10	0,10	0,04	0,10	1003	**0,1 %**
500	50	3	0,00	0,00	0,00	0,00	0,04	0,00	0,00	1003	**0,1 %**

n	m	i	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir	$C_{\text{máx}}$	t. o **Gap**
500	50	4	0,10	0,10	0,00	0,14	0,16	0,10	0,14	1002	3210,11 seg
500	50	5	0,20	0,30	0,00	0,08	0,08	0,00	0,10	1003	**0,1 %**
500	50	6	0,00	0,10	0,00	0,10	0,12	0,08	0,10	1002	6042,52 seg
500	50	7	0,10	0,10	0,10	0,14	0,16	0,10	0,12	1002	949,95 seg
500	50	8	0,20	0,10	0,10	0,10	0,10	0,04	0,08	1003	**0,1 %**
500	50	9	0,10	0,00	0,00	0,02	0,02	0,00	0,00	1003	**0,2 %**
500	50	10	0,10	0,10	0,10	0,10	0,14	0,10	0,10	1002	111,83 seg
1000	10	1	0,07	0,07	0,13	0,04	0,03	0,02	0,03	10141	128,28 seg
1000	10	2	0,13	0,77	0,16	0,06	0,06	0,04	0,06	10141	832,89 seg
1000	10	3	0,12	0,12	0,05	0,03	0,02	0,01	0,02	10151	474,92 seg
1000	10	4	0,13	0,11	0,17	0,05	0,04	0,03	0,04	10151	226,98 seg
1000	10	5	0,09	0,18	0,07	0,04	0,03	0,03	0,03	10150	1022,72 seg
1000	10	6	0,16	0,23	0,11	0,05	0,04	0,04	0,05	10146	76,09 seg
1000	10	7	0,19	0,41	0,35	0,06	0,06	0,04	0,06	10142	3555,94 seg
1000	10	8	0,22	0,15	0,15	0,04	0,04	0,02	0,03	10150	76,13 seg
1000	10	9	0,36	0,59	0,12	0,06	0,06	0,04	0,05	10148	233,75 seg
1000	10	10	0,16	0,22	0,05	0,03	0,03	0,02	0,03	10152	181,39 seg
1000	20	1	0,22	0,18	0,16	0,09	0,08	0,08	0,08	5030	415,81 seg
1000	20	2	0,14	0,32	0,16	0,08	0,07	0,05	0,07	5032	544,28 seg
1000	20	3	0,14	0,18	0,12	0,08	0,08	0,06	0,09	5028	**0,02 %**
1000	20	4	0,18	0,16	0,16	0,07	0,08	0,06	0,08	5030	**0,02 %**
1000	20	5	0,22	0,14	0,06	0,10	0,06	0,05	0,08	5032	**0,02 %**
1000	20	6	0,16	0,22	0,10	0,05	0,05	0,03	0,04	5028	**0,02 %**
1000	20	7	0,44	0,48	0,22	0,08	0,08	0,06	0,08	5031	**0,02 %**
1000	20	8	0,36	0,22	0,14	0,09	0,07	0,08	0,08	5031	**0,02 %**
1000	20	9	0,26	0,16	0,14	0,08	0,07	0,06	0,06	5031	278,86 seg
1000	20	10	0,42	0,52	0,36	0,09	0,09	0,06	0,09	5029	**0,02 %**
1000	30	1	0,12	0,06	-0,09	0,02	0,04	0,00	0,01	3407	**1,58 %**
1000	30	2	0,06	0,12	-0,15	0,03	0,03	0,01	0,02	3407	**1,79 %**

n	m	i	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir	$C_{\text{máx}}$	t. o **Gap**
1000	30	3	0,18	0,32	0,03	0,18	0,17	0,15	0,16	3402	**1,56 %**
1000	30	4	0,15	0,00	-0,15	0,02	0,02	-0,01	0,02	3408	**1,64 %**
1000	30	5	0,06	0,06	-0,09	0,01	0,04	0,01	0,02	3406	**1,67 %**
1000	30	6	0,06	0,12	-0,09	0,01	0,01	-0,03	-0,01	3407	**1,7 %**
1000	30	7	0,26	0,29	0,03	0,21	0,22	0,19	0,19	3400	**1,5 %**
1000	30	8	0,06	0,12	0,00	0,08	0,10	0,08	0,08	3404	**1,53 %**
1000	30	9	0,06	0,09	-0,06	0,03	0,03	0,00	0,02	3407	**1,7 %**
1000	30	10	0,18	0,26	0,09	0,16	0,17	0,12	0,14	3401	**1,59 %**
1000	40	1	0,32	0,20	0,08	0,07	0,10	0,04	0,06	2506	**0,04 %**
1000	40	2	0,12	0,16	0,04	0,06	0,08	0,05	0,06	2506	**0,04 %**
1000	40	3	0,08	0,16	0,08	0,07	0,11	0,06	0,07	2505	**0,04 %**
1000	40	4	0,20	0,08	0,12	0,07	0,10	0,05	0,06	2505	**0,04 %**
1000	40	5	0,12	0,16	0,12	0,08	0,13	0,08	0,08	2505	**0,04 %**
1000	40	6	0,24	0,20	0,16	0,06	0,10	0,05	0,04	2506	**0,08 %**
1000	40	7	0,08	0,12	0,08	0,08	0,12	0,06	0,08	2505	**0,04 %**
1000	40	8	0,16	0,08	0,08	0,07	0,11	0,06	0,08	2505	**0,04 %**
1000	40	9	0,12	0,08	0,08	0,08	0,10	0,08	0,08	2505	**0,04 %**
1000	40	10	0,16	0,20	0,20	0,06	0,08	0,03	0,04	2506	**0,08 %**
1000	50	1	0,05	0,10	-0,05	-0,01	0,01	-0,04	0,00	2005	**0,1 %**
1000	50	2	0,05	0,10	0,05	0,01	0,05	0,02	0,02	2004	**0,1 %**
1000	50	3	0,05	0,05	0,00	0,01	0,04	-0,01	0,00	2004	**0,1 %**
1000	50	4	0,15	0,25	0,10	0,09	0,12	0,08	0,07	2003	**0,05 %**
1000	50	5	0,05	0,15	0,00	0,02	0,03	0,00	0,01	2004	**0,1 %**
1000	50	6	0,05	0,10	0,00	0,03	0,06	0,04	0,05	2004	**0,1 %**
1000	50	7	0,15	0,15	0,15	0,06	0,09	0,05	0,05	2003	**0,05 %**
1000	50	8	0,10	0,05	0,05	0,05	0,08	0,04	0,05	2003	**0,05 %**
1000	50	9	0,00	0,00	0,00	-0,03	0,01	-0,04	-0,03	2005	**0,15 %**
1000	50	10	0,05	0,05	0,00	0,03	0,05	0,03	0,05	2004	**0,1 %**

Tabla A.4 – Tabla completa para el intervalo U(100,120).

A.5. Tabla completa para el intervalo U(1000,1100)

n	m	i	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir	$C_{\text{máx}}$	t. o **Gap**
100	10	1	0,03	0,04	0,08	0,02	0,02	0,02	0,01	10092	**0,03 %**
100	10	2	0,02	0,00	0,08	0,04	0,03	0,05	0,04	10078	1113,42 seg
100	10	3	0,11	0,11	0,31	0,05	0,05	0,05	0,02	10108	**0,04 %**
100	10	4	0,10	0,05	0,20	0,06	0,04	0,04	0,03	10112	**0,03 %**
100	10	5	0,08	0,02	0,14	0,06	0,06	0,05	0,04	10105	1100,2 seg
100	10	6	0,06	0,08	0,15	0,04	0,01	0,04	0,01	10102	**0,01 %**
100	10	7	0,05	0,06	0,18	0,03	0,02	0,04	0,02	10121	**0,05 %**
100	10	8	0,08	0,00	0,05	0,01	0,01	0,02	-0,01	10110	**0,07 %**
100	10	9	0,02	0,00	0,16	0,00	0,00	0,00	0,00	10097	34,33 seg
100	10	10	0,09	0,00	0,09	0,05	0,04	0,03	0,03	10097	**0,05 %**
100	20	1	0,04	0,02	0,20	0,04	0,02	0,02	0,03	5028	**0,02 %**
100	20	2	0,00	0,00	0,16	0,01	0,00	0,02	0,00	5038	**0,02 %**
100	20	3	0,10	0,06	0,10	0,01	0,00	0,02	0,00	5036	225,24 seg
100	20	4	0,00	0,00	0,06	0,00	0,00	0,01	0,00	5037	6,42 seg
100	20	5	0,02	0,18	0,18	0,03	0,02	0,06	0,02	5037	1197,56 seg
100	20	6	0,00	0,02	0,40	0,02	0,01	0,02	0,02	5035	**0,12 %**
100	20	7	0,04	0,14	0,08	0,06	0,04	0,04	0,03	5035	108,16 seg
100	20	8	0,10	0,00	0,24	0,04	0,02	0,03	0,01	5036	**0,04 %**
100	20	9	0,04	0,14	0,22	0,06	0,05	0,05	0,05	5035	**0,04 %**
100	20	10	0,08	0,14	0,18	0,07	0,05	0,05	0,05	5033	**0,02 %**
100	30	1	0,20	0,30	0,27	0,00	-0,01	0,00	-0,03	4008	**5,72 %**
100	30	2	0,22	0,30	0,32	0,07	0,05	0,04	0,02	4006	**3,27 %**
100	30	3	0,07	0,47	0,22	0,02	0,01	0,01	0,00	4006	**4,33 %**
100	30	4	0,17	0,30	0,20	0,03	0,01	0,03	0,01	4007	**2,77 %**
100	30	5	0,12	0,22	0,20	0,00	0,00	0,00	0,00	4006	**5,34 %**
100	30	6	0,10	0,75	0,30	0,00	-0,01	0,00	-0,01	4007	**5,24 %**
100	30	7	0,27	0,22	0,15	0,00	-0,01	-0,01	-0,03	4007	**7,14 %**

n	m	i	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir	$C_{\text{máx}}$	t. o **Gap**
100	30	8	0,27	0,37	0,25	0,01	0,00	0,01	-0,03	4010	**6,41 %**
100	30	9	0,15	0,30	0,20	0,00	0,00	0,00	0,00	4008	**3,86 %**
100	30	10	0,12	0,37	0,20	0,00	0,00	0,01	0,00	4007	**5,69 %**
100	40	1	0,07	0,33	0,17	0,07	0,07	0,05	0,03	3005	**1,26 %**
100	40	2	0,17	0,37	0,27	0,02	0,04	0,03	0,00	3006	**2,95 %**
100	40	3	0,17	0,30	0,27	0,06	0,08	0,05	0,03	3005	**1,96 %**
100	40	4	0,07	0,23	0,30	0,07	0,07	0,06	0,03	3005	**1,61 %**
100	40	5	0,13	0,33	0,17	0,00	0,00	0,00	0,00	3006	**3,24 %**
100	40	6	0,20	0,23	0,47	0,03	0,03	0,02	0,00	3005	**2,8 %**
100	40	7	0,10	0,33	0,37	0,03	0,07	0,03	0,00	3004	**2,19 %**
100	40	8	0,13	0,50	0,33	0,07	0,09	0,03	0,02	3005	**4,04 %**
100	40	9	0,10	0,40	0,57	0,05	0,07	0,05	0,03	3005	**2,28 %**
100	40	10	0,13	0,43	0,43	0,03	0,03	0,01	0,00	3005	**3,4 %**
100	50	1	0,00	0,00	1,09	0,00	0,00	0,00	0,00	2013	0,61 seg
100	50	2	0,00	0,00	0,80	0,00	0,00	0,00	0,00	2010	8,66 seg
100	50	3	0,00	0,00	1,69	0,00	0,01	0,00	0,00	2012	12,2 seg
100	50	4	0,00	0,00	0,00	0,00	0,00	0,00	0,00	2026	11,92 seg
100	50	5	0,05	0,00	0,65	0,02	0,03	0,01	0,03	2010	59,36 seg
100	50	6	0,00	0,00	0,05	0,00	0,00	0,00	0,00	2014	9,59 seg
100	50	7	0,00	0,05	0,15	0,05	0,09	0,07	0,07	2009	14,3 seg
100	50	8	0,05	0,05	1,29	0,01	0,04	0,02	0,05	2011	15,86 seg
100	50	9	0,00	0,00	0,10	0,00	0,00	0,00	0,00	2015	11,8 seg
100	50	10	0,00	0,00	1,89	0,00	0,00	0,00	0,00	2012	8,47 seg
200	10	1	0,01	0,08	0,17	0,07	0,05	0,04	0,06	20193	**0,02 %**
200	10	2	0,02	0,02	0,11	0,04	0,03	0,02	0,02	20182	**0,01 %**
200	10	3	0,02	0,14	0,12	0,05	0,04	0,02	0,03	20177	**0,01 %**
200	10	4	0,04	0,09	0,06	0,03	0,02	0,02	0,02	20188	**0,02 %**
200	10	5	0,01	0,09	0,12	0,04	0,03	0,02	0,02	20182	**0,03 %**
200	10	6	0,08	0,08	0,12	0,06	0,04	0,03	0,04	20197	**0,01 %**

n	m	i	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir	$C_{\text{máx}}$	t. o **Gap**
200	10	7	0,03	0,05	0,27	0,03	0,03	0,03	0,02	20191	**0,02 %**
200	10	8	0,08	0,04	0,11	0,05	0,04	0,03	0,03	20195	**0,02 %**
200	10	9	0,04	0,01	0,12	0,05	0,04	0,02	0,03	20177	**0,01 %**
200	10	10	0,08	0,03	0,11	0,04	0,02	0,02	0,01	20179	**0,04 %**
200	20	1	0,06	0,07	0,22	0,09	0,07	0,05	0,09	10056	**0,05 %**
200	20	2	0,08	0,02	0,07	0,06	0,04	0,03	0,05	10050	**0,03 %**
200	20	3	0,14	0,02	0,20	0,06	0,03	0,03	0,05	10050	**0,05 %**
200	20	4	0,06	0,13	0,27	0,08	0,06	0,05	0,06	10055	**0,04 %**
200	20	5	0,12	0,14	0,14	0,09	0,06	0,05	0,07	10052	**0,05 %**
200	20	6	0,15	0,03	0,12	0,06	0,05	0,03	0,05	10054	**0,06 %**
200	20	7	0,02	0,06	0,12	0,06	0,04	0,02	0,05	10058	**0,07 %**
200	20	8	0,06	0,16	0,13	0,09	0,06	0,05	0,08	10051	**0,05 %**
200	20	9	0,07	0,08	0,25	0,09	0,07	0,05	0,08	10055	**0,05 %**
200	20	10	0,10	0,08	0,21	0,06	0,06	0,04	0,04	10048	**0,05 %**
200	30	1	0,00	0,73	-0,10	-0,15	-0,18	-0,21	-0,16	7034	**0,8 %**
200	30	2	0,13	0,56	0,06	0,00	-0,02	-0,04	-0,03	7025	**0,91 %**
200	30	3	0,27	0,94	0,04	0,00	0,00	-0,04	-0,01	7020	**0,63 %**
200	30	4	0,00	0,58	-0,03	-0,11	-0,13	-0,15	-0,14	7032	**0,78 %**
200	30	5	0,00	0,53	-0,21	-0,22	-0,24	-0,28	-0,24	7039	**1,05 %**
200	30	6	0,10	0,77	0,07	0,04	0,02	0,01	0,02	7019	**0,56 %**
200	30	7	0,00	0,85	-0,10	-0,11	-0,13	-0,15	-0,13	7031	**0,75 %**
200	30	8	0,21	1,08	0,00	0,03	0,00	-0,03	-0,01	7020	**0,87 %**
200	30	9	0,00	0,38	-0,07	-0,21	-0,23	-0,26	-0,22	7038	**0,91 %**
200	30	10	0,21	0,81	0,06	0,06	0,04	0,01	0,03	7018	**0,57 %**
200	40	1	0,10	0,06	0,24	0,06	0,06	0,02	0,05	5016	**0,1 %**
200	40	2	0,06	0,14	0,22	0,05	0,04	0,02	0,04	5019	**0,14 %**
200	40	3	0,08	0,12	0,40	0,06	0,07	0,05	0,07	5018	**0,06 %**
200	40	4	0,22	-0,08	0,48	-0,09	-0,08	-0,10	-0,09	5033	**0,26 %**
200	40	5	0,08	0,08	0,16	0,10	0,10	0,06	0,10	5014	**0,06 %**

n	m	i	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir	$C_{\text{máx}}$	t. o **Gap**
200	40	6	0,02	0,10	0,34	0,04	0,05	0,01	0,04	5020	**0,16 %**
200	40	7	0,10	0,34	0,26	0,00	0,00	0,01	0,02	5028	171,34 seg
200	40	8	0,08	0,02	0,34	0,08	0,10	0,05	0,09	5018	**0,1 %**
200	40	9	0,14	0,04	0,18	0,08	0,08	0,06	0,09	5016	**0,06 %**
200	40	10	0,12	0,06	0,32	0,10	0,11	0,06	0,08	5017	38,7 seg
200	50	1	0,05	0,10	0,17	0,07	0,07	0,04	0,10	4012	**0,02 %**
200	50	2	0,07	0,07	1,05	0,02	0,03	0,00	0,04	4015	**0,07 %**
200	50	3	0,07	0,10	0,22	0,04	0,04	0,01	0,05	4015	**0,15 %**
200	50	4	0,05	0,20	0,62	0,08	0,08	0,04	0,09	4012	**0,07 %**
200	50	5	0,05	0,05	0,40	0,02	0,03	0,00	0,02	4014	**0,1 %**
200	50	6	0,05	0,05	0,22	0,05	0,06	0,04	0,07	4013	**0,02 %**
200	50	7	0,10	0,30	0,70	0,07	0,07	0,02	0,07	4012	**0,07 %**
200	50	8	0,07	0,05	0,37	0,09	0,08	0,06	0,09	4012	**0,1 %**
200	50	9	0,12	0,12	0,87	0,08	0,05	0,04	0,08	4013	**0,12 %**
200	50	10	0,05	0,02	0,60	0,03	0,01	0,00	0,02	4015	**0,15 %**
500	10	1	0,06	0,03	0,13	0,04	0,04	0,02	0,03	50431	**0,01 %**
500	10	2	0,02	0,02	0,05	0,02	0,02	0,01	0,02	50438	**0,01 %**
500	10	3	0,08	0,05	0,18	0,04	0,04	0,02	0,04	50440	**0,01 %**
500	10	4	0,05	0,03	0,11	0,03	0,02	0,02	0,02	50466	**0 %**
500	10	5	0,04	0,05	0,08	0,03	0,02	0,01	0,02	50415	**0,01 %**
500	10	6	0,06	0,02	0,06	0,04	0,04	0,02	0,04	50459	**0,02 %**
500	10	7	0,07	0,03	0,12	0,03	0,03	0,02	0,03	50420	**0,02 %**
500	10	8	0,07	0,03	0,06	0,03	0,02	0,01	0,02	50407	**0,01 %**
500	10	9	0,07	0,04	0,08	0,04	0,03	0,01	0,02	50437	**0,01 %**
500	10	10	0,03	0,05	0,04	0,03	0,02	0,01	0,02	50442	**0,01 %**
500	20	1	0,14	0,06	0,08	0,06	0,04	0,03	0,05	25116	**0,04 %**
500	20	2	0,10	0,03	0,10	0,04	0,03	0,03	0,03	25113	**0,03 %**
500	20	3	0,04	0,02	0,03	0,04	0,03	0,03	0,04	25116	**0,02 %**
500	20	4	0,12	0,10	0,12	0,05	0,05	0,02	0,05	25126	**0,03 %**

n	m	i	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir	$C_{\text{máx}}$	t. o **Gap**
500	20	5	0,04	0,02	0,09	0,05	0,04	0,03	0,04	25115	**0,03 %**
500	20	6	0,13	0,04	0,12	0,06	0,05	0,04	0,05	25121	**0,03 %**
500	20	7	0,14	0,03	0,10	0,05	0,04	0,04	0,04	25115	**0,02 %**
500	20	8	0,11	0,03	0,06	0,05	0,04	0,03	0,05	25120	**0,03 %**
500	20	9	0,04	0,04	0,09	0,05	0,05	0,03	0,05	25121	**0,03 %**
500	20	10	0,13	0,04	0,11	0,06	0,06	0,05	0,05	25120	**0,02 %**
500	30	1	0,10	0,94	0,07	0,07	0,06	0,05	0,06	17041	**0,98 %**
500	30	2	0,01	0,83	0,06	0,06	0,06	0,04	0,05	17044	**0,94 %**
500	30	3	0,00	1,16	0,04	0,04	0,04	0,02	0,03	17043	**0,99 %**
500	30	4	0,02	0,82	0,05	0,04	0,04	0,02	0,03	17046	**0,94 %**
500	30	5	0,02	0,93	0,01	0,03	0,02	0,00	0,02	17046	**0,81 %**
500	30	6	0,01	0,91	0,01	0,01	0,01	-0,01	0,00	17051	**0,91 %**
500	30	7	0,01	0,87	0,03	0,03	0,03	0,01	0,03	17047	**0,91 %**
500	30	8	0,01	1,06	0,01	0,02	0,01	0,00	0,00	17049	**0,88 %**
500	30	9	0,00	0,97	0,03	0,04	0,03	0,02	0,03	17044	**0,88 %**
500	30	10	0,01	0,85	0,01	0,01	0,00	0,00	0,00	17050	**0,96 %**
500	40	1	0,04	1,53	0,05	0,06	0,06	0,04	0,05	13018	**2,99 %**
500	40	2	0,01	1,49	-0,01	0,01	0,01	-0,01	0,00	13026	**2,44 %**
500	40	3	0,05	0,10	0,01	0,03	0,03	0,01	0,01	13024	**2,8 %**
500	40	4	0,02	1,47	0,00	0,01	0,01	-0,02	-0,01	13026	**2,25 %**
500	40	5	0,00	1,50	0,01	0,01	0,02	0,00	0,00	13024	**2,33 %**
500	40	6	0,09	1,63	0,08	0,07	0,07	0,05	0,06	13018	**3,05 %**
500	40	7	0,02	1,18	0,05	0,06	0,05	0,03	0,04	13022	**2,18 %**
500	40	8	0,04	1,56	0,06	0,07	0,06	0,04	0,06	13018	**3,25 %**
500	40	9	0,00	1,40	0,03	0,04	0,04	0,02	0,03	13020	**2,38 %**
500	40	10	0,02	0,05	0,05	0,04	0,04	0,03	0,04	13021	**2,35 %**
500	50	1	0,07	0,07	0,42	0,06	0,06	0,05	0,06	10024	**0,07 %**
500	50	2	0,11	0,08	0,13	0,05	0,05	0,03	0,05	10025	**0,04 %**
500	50	3	0,12	0,07	0,25	0,06	0,05	0,03	0,05	10024	**0,08 %**

n	m	i	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir	$C_{\text{máx}}$	t. o **Gap**
500	50	4	0,08	0,05	0,17	0,07	0,08	0,05	0,07	10020	**0,01 %**
500	50	5	0,07	0,09	0,29	0,07	0,07	0,06	0,07	10020	**0,05 %**
500	50	6	0,07	0,10	0,13	0,08	0,08	0,06	0,06	10022	**0,05 %**
500	50	7	0,10	0,10	0,18	0,08	0,08	0,07	0,07	10023	**0,06 %**
500	50	8	0,06	0,11	0,21	0,07	0,07	0,04	0,07	10022	**0,06 %**
500	50	9	0,07	0,17	0,30	0,07	0,08	0,05	0,08	10021	**0,05 %**
500	50	10	0,09	0,10	0,18	0,07	0,08	0,05	0,07	10025	**0,07 %**
1000	10	1	0,05	0,08	0,03	0,01	0,01	0,01	0,01	100820	**0 %**
1000	10	2	0,04	0,05	0,07	0,03	0,03	0,02	0,03	100876	**0 %**
1000	10	3	0,03	0,02	0,04	0,02	0,02	0,01	0,02	100922	**0 %**
1000	10	4	0,19	0,18	0,11	0,04	0,04	0,04	0,04	100908	**0 %**
1000	10	5	0,09	0,04	0,06	0,01	0,01	0,01	0,01	100897	**0,01 %**
1000	10	6	0,09	0,08	0,08	0,02	0,02	0,01	0,02	100881	**0,01 %**
1000	10	7	0,04	0,04	0,04	0,02	0,02	0,01	0,02	100867	**0 %**
1000	10	8	0,06	0,08	0,08	0,03	0,03	0,02	0,03	100913	**0 %**
1000	10	9	0,14	0,11	0,13	0,03	0,03	0,02	0,03	100921	**0 %**
1000	10	10	0,07	0,13	0,05	0,01	0,01	0,01	0,01	100894	**0 %**
1000	20	1	0,04	0,05	0,06	0,05	0,05	0,04	0,05	50231	**0,02 %**
1000	20	2	0,06	0,06	0,09	0,03	0,02	0,02	0,03	50229	**0,02 %**
1000	20	3	0,09	0,06	0,07	0,05	0,04	0,05	0,05	50245	**0,02 %**
1000	20	4	0,06	0,09	0,06	0,04	0,04	0,04	0,04	50211	**0,01 %**
1000	20	5	0,08	0,04	0,07	0,05	0,05	0,04	0,05	50224	**0,02 %**
1000	20	6	0,06	0,04	0,04	0,03	0,04	0,03	0,03	50238	**0,03 %**
1000	20	7	0,18	0,07	0,11	0,03	0,03	0,03	0,03	50225	**0,02 %**
1000	20	8	0,08	0,10	0,08	0,04	0,03	0,03	0,03	50236	**0,04 %**
1000	20	9	0,13	0,04	0,07	0,04	0,04	0,03	0,04	50225	**0,02 %**
1000	20	10	0,18	0,14	0,13	0,05	0,05	0,04	0,05	50224	**0,02 %**
1000	30	1	0,13	0,08	0,09	0,10	0,09	0,09	0,09	34060	**1,6 %**
1000	30	2	0,13	0,11	0,07	0,08	0,08	0,06	0,07	34056	**1,48 %**

n	m	i	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir	$C_{\text{máx}}$	t. o **Gap**
1000	30	3	0,05	0,03	0,06	0,06	0,06	0,05	0,06	34061	**1,6 %**
1000	30	4	0,12	0,08	0,07	0,06	0,06	0,05	0,06	34059	**1,64 %**
1000	30	5	0,11	0,04	0,06	0,06	0,06	0,05	0,05	34068	**1,59 %**
1000	30	6	0,10	0,10	0,09	0,09	0,09	0,09	0,08	34052	**1,58 %**
1000	30	7	0,11	0,08	0,07	0,07	0,07	0,06	0,07	34060	**1,58 %**
1000	30	8	0,12	0,08	0,09	0,08	0,08	0,08	0,07	34059	**1,58 %**
1000	30	9	0,09	0,04	0,05	0,03	0,04	0,03	0,03	34066	**1,56 %**
1000	30	10	0,06	0,05	0,06	0,05	0,05	0,04	0,05	34062	**1,61 %**
1000	40	1	0,09	0,10	0,07	0,05	0,05	0,05	0,05	25060	**0,03 %**
1000	40	2	0,10	0,10	0,07	0,05	0,06	0,05	0,05	25058	**0,03 %**
1000	40	3	0,08	0,06	0,14	0,06	0,06	0,05	0,05	25058	**0,03 %**
1000	40	4	0,06	0,07	0,08	0,05	0,04	0,04	0,05	25058	**0,04 %**
1000	40	5	0,06	0,06	0,10	0,05	0,06	0,04	0,05	25056	**0,03 %**
1000	40	6	0,09	0,08	0,14	0,05	0,06	0,04	0,06	25058	**0,03 %**
1000	40	7	0,06	0,04	0,10	0,05	0,06	0,04	0,05	25060	**0,03 %**
1000	40	8	0,12	0,10	0,08	0,04	0,06	0,04	0,05	25060	**0,03 %**
1000	40	9	0,08	0,10	0,08	0,03	0,04	0,03	0,03	25067	**0,05 %**
1000	40	10	0,08	0,05	0,09	0,05	0,05	0,04	0,04	25059	**0,03 %**
1000	50	1	0,08	0,04	0,14	0,07	0,07	0,05	0,07	20040	**0,04 %**
1000	50	2	0,05	0,05	0,16	0,06	0,06	0,04	0,05	20035	**0,03 %**
1000	50	3	0,05	0,05	0,13	0,05	0,05	0,04	0,04	20039	**0,04 %**
1000	50	4	0,06	0,07	0,10	0,06	0,06	0,05	0,06	20038	**0,03 %**
1000	50	5	0,05	0,05	0,19	0,05	0,05	0,04	0,05	20038	**0,03 %**
1000	50	6	0,03	0,01	0,10	0,02	0,03	0,01	0,02	20044	**0,06 %**
1000	50	7	0,06	0,04	0,05	0,05	0,07	0,05	0,06	20035	**0,02 %**
1000	50	8	0,08	0,10	0,21	0,06	0,06	0,05	0,06	20040	**0,04 %**
1000	50	9	0,03	0,04	0,11	0,03	0,04	0,02	0,03	20041	**0,06 %**
1000	50	10	0,10	0,10	0,11	0,07	0,07	0,07	0,06	20037	**0,03 %**

Tabla A.5 – Tabla completa para el intervalo U(1000,1100).

A.6. Tabla completa para trabajos correlacionados

n	m	i	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir	$C_{\text{máx}}$	t. o **Gap**
100	10	1	0,40	0,20	0,79	0,28	0,36	0,20	0,12	505	**0,4 %**
100	10	2	0,58	0,58	0,58	0,54	0,58	0,31	0,31	516	**0,39 %**
100	10	3	0,54	0,18	0,54	1,52	1,52	1,01	0,90	554	**0,36 %**
100	10	4	0,90	0,36	0,72	0,22	0,25	0,04	0,04	553	**0,54 %**
100	10	5	0,00	0,36	0,89	0,50	0,50	0,32	0,36	562	**0,36 %**
100	10	6	0,39	0,20	0,98	1,26	1,42	0,98	0,94	508	**0,2 %**
100	10	7	0,40	1,19	0,79	0,28	0,36	0,16	0,08	506	**0,4 %**
100	10	8	0,98	0,59	0,98	1,81	1,81	1,02	1,34	508	**0,2 %**
100	10	9	0,38	0,38	0,76	0,49	0,53	0,38	0,34	529	**0,38 %**
100	10	10	0,41	1,22	0,82	0,65	0,69	0,41	0,45	490	**0,2 %**
100	20	1	0,78	2,71	1,16	2,17	2,56	0,78	1,78	258	**1,16 %**
100	20	2	0,78	2,71	1,55	0,70	1,09	0,08	0,31	258	**1,55 %**
100	20	3	1,47	2,21	1,10	1,18	1,18	0,44	0,81	272	**1,47 %**
100	20	4	1,37	2,05	1,03	1,51	1,71	0,68	1,37	292	**1,37 %**
100	20	5	1,05	1,05	0,70	0,56	0,84	0,21	0,28	287	**1,74 %**
100	20	6	2,66	2,66	1,52	2,28	2,81	1,29	2,05	263	**1,14 %**
100	20	7	0,74	2,57	1,10	0,88	1,10	0,29	0,51	272	**1,47 %**
100	20	8	1,05	0,70	1,75	1,19	1,40	0,63	0,70	286	**1,4 %**
100	20	9	1,10	2,21	1,84	1,69	1,91	1,32	1,54	272	**1,47 %**
100	20	10	0,85	2,99	0,85	0,85	1,11	0,34	0,60	234	**1,71 %**
100	30	1	3,91	6,70	0,00	1,12	2,01	-0,11	0,22	179	**3,35 %**
100	30	2	1,70	4,55	1,70	1,82	2,95	0,68	1,25	176	**3,41 %**
100	30	3	3,80	6,52	1,09	1,74	2,72	0,76	0,87	184	**2,72 %**
100	30	4	1,14	4,00	0,00	1,60	2,86	0,23	0,57	175	**3,43 %**
100	30	5	2,35	2,94	1,76	1,76	2,35	0,59	0,94	170	**2,94 %**
100	30	6	2,42	1,82	1,82	1,94	3,03	0,97	1,45	165	**3,03 %**
100	30	7	2,31	5,20	1,16	2,31	2,89	0,58	1,50	173	**2,89 %**

n	m	i	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir	$C_{\text{máx}}$	t. o **Gap**
100	30	8	1,75	4,68	1,17	2,57	3,86	1,05	1,40	171	**2,92 %**
100	30	9	2,14	3,74	0,53	1,28	2,03	0,11	0,86	187	**3,21 %**
100	30	10	1,65	7,14	0,55	2,09	2,75	0,11	1,32	182	**3,3 %**
100	40	1	2,72	7,48	0,00	2,59	4,63	0,27	0,41	147	**5,44 %**
100	40	2	3,03	9,85	1,52	5,15	10,30	3,48	3,03	132	**5,3 %**
100	40	3	2,61	6,54	1,31	5,62	8,76	2,61	4,18	153	**3,92 %**
100	40	4	2,82	11,97	1,41	3,80	6,90	1,55	2,11	142	**4,23 %**
100	40	5	2,82	9,15	0,70	3,24	5,77	2,39	1,69	142	**4,23 %**
100	40	6	2,78	8,33	2,78	5,97	10,97	3,06	3,33	144	**5,56 %**
100	40	7	1,59	1,59	0,00	3,02	6,67	0,16	0,79	126	**5,56 %**
100	40	8	2,86	7,14	1,43	3,29	6,29	0,86	1,43	140	**5 %**
100	40	9	0,78	7,81	2,34	4,06	5,94	2,81	1,72	128	**5,47 %**
100	40	10	2,22	2,22	0,74	3,56	6,81	2,07	1,63	135	**4,44 %**
100	50	1	1,92	6,73	0,00	1,35	5,00	0,19	1,73	104	**2,88 %**
100	50	2	5,31	13,27	4,42	7,61	10,44	5,31	4,96	113	**5,31 %**
100	50	3	4,24	11,02	2,54	4,58	10,68	2,54	1,86	118	**8,47 %**
100	50	4	1,72	6,90	1,72	7,24	11,38	5,00	4,48	116	**8,62 %**
100	50	5	2,86	4,76	1,90	4,38	9,33	3,05	2,67	105	**4,76 %**
100	50	6	4,35	8,70	6,09	7,48	13,22	6,43	4,87	115	**6,09 %**
100	50	7	2,36	7,09	2,36	5,20	9,29	5,51	3,46	127	**8,66 %**
100	50	8	3,45	7,76	2,59	5,17	9,66	6,38	3,79	116	**8,62 %**
100	50	9	1,83	5,50	1,83	3,49	6,79	1,47	2,20	109	**7,34 %**
100	50	10	3,77	6,60	1,89	4,15	8,30	4,34	3,21	106	**4,72 %**
200	10	1	0,45	0,18	0,27	0,45	0,48	0,34	0,41	1121	**0,18 %**
200	10	2	0,47	0,28	0,38	1,07	1,13	0,81	0,96	1063	**0,09 %**
200	10	3	0,18	0,27	0,55	0,26	0,24	0,16	0,22	1095	**0,18 %**
200	10	4	0,18	0,36	0,45	0,45	0,51	0,27	0,36	1100	**0,09 %**
200	10	5	0,10	0,30	0,39	0,71	0,71	0,45	0,51	1014	**0,1 %**
200	10	6	0,28	0,19	0,37	0,26	0,28	0,24	0,21	1071	**0,19 %**

n	m	i	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir	$C_{\text{máx}}$	t. o **Gap**
200	10	7	0,52	0,21	0,21	0,31	0,37	0,23	0,27	970	**0,1 %**
200	10	8	0,18	0,00	0,36	0,53	0,60	0,42	0,42	1104	**0,18 %**
200	10	9	0,19	0,19	0,29	0,31	0,29	0,23	0,23	1039	**0,19 %**
200	10	10	0,37	0,64	0,18	0,59	0,68	0,46	0,59	1087	**0,18 %**
200	20	1	1,36	1,36	0,78	0,47	0,58	0,27	0,39	515	**0,58 %**
200	20	2	1,54	0,39	0,58	0,50	0,69	0,31	0,35	518	**0,58 %**
200	20	3	1,25	1,25	0,18	0,61	0,71	0,25	0,50	560	**0,89 %**
200	20	4	1,76	1,76	0,88	0,70	0,67	0,32	0,49	569	**0,53 %**
200	20	5	1,79	1,25	0,71	0,43	0,46	0,18	0,36	560	**0,71 %**
200	20	6	1,87	2,06	0,75	0,75	0,75	0,52	0,56	534	**0,37 %**
200	20	7	1,37	0,98	0,39	0,67	0,71	0,43	0,51	510	**0,59 %**
200	20	8	2,10	0,57	0,38	0,99	1,07	0,80	0,88	525	**0,57 %**
200	20	9	1,79	1,20	0,40	0,44	0,60	0,32	0,36	502	**0,6 %**
200	20	10	1,53	0,95	0,57	0,73	0,80	0,38	0,57	524	**0,57 %**
200	30	1	3,68	1,70	0,57	1,19	1,30	0,57	0,96	353	**1,42 %**
200	30	2	3,94	4,23	1,13	2,20	2,59	1,46	1,86	355	**1,13 %**
200	30	3	2,97	2,97	0,59	1,13	1,13	0,42	0,53	337	**1,19 %**
200	30	4	1,96	1,68	0,56	0,56	0,90	0,06	0,34	357	**1,4 %**
200	30	5	2,50	2,50	0,31	1,00	1,13	0,44	0,56	320	**1,56 %**
200	30	6	2,59	2,01	0,57	1,44	1,38	0,52	1,26	348	**1,15 %**
200	30	7	2,22	3,06	0,56	1,50	1,83	0,83	1,22	360	**1,67 %**
200	30	8	2,23	1,96	0,28	1,01	1,23	0,45	0,89	358	**1,12 %**
200	30	9	2,87	2,01	0,00	0,57	1,09	0,00	0,34	348	**1,44 %**
200	30	10	3,23	2,93	1,17	2,05	2,11	1,35	1,64	341	**0,88 %**
200	40	1	3,19	4,78	0,40	1,51	2,31	0,24	0,80	251	**1,99 %**
200	40	2	3,05	4,58	0,38	2,37	2,21	0,76	1,76	262	**2,29 %**
200	40	3	5,04	2,33	0,39	1,47	1,94	0,54	0,70	258	**1,94 %**
200	40	4	3,75	4,12	0,37	1,87	2,62	0,52	1,12	267	**1,87 %**
200	40	5	3,79	4,55	0,76	2,27	2,58	1,14	2,12	264	**1,89 %**

n	m	i	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir	$C_{\text{máx}}$	t. o **Gap**
200	40	6	3,76	4,89	-0,38	0,68	1,13	-0,38	0,23	266	**2,63 %**
200	40	7	1,84	2,57	-1,47	0,00	0,51	-1,03	-0,44	272	**3,31 %**
200	40	8	1,20	1,99	-1,99	-0,24	0,16	-1,43	-0,88	251	**3,98 %**
200	40	9	2,32	3,47	0,00	2,32	2,63	0,69	1,39	259	**2,7 %**
200	40	10	3,31	1,65	-0,83	1,16	1,40	0,08	1,07	242	**2,48 %**
200	50	1	5,33	5,78	0,00	2,40	4,44	0,80	1,42	225	**2,67 %**
200	50	2	4,23	4,69	-0,47	1,97	3,76	0,47	1,22	213	**3,29 %**
200	50	3	4,13	6,88	0,00	2,66	3,67	0,73	2,39	218	**2,75 %**
200	50	4	6,07	7,48	-0,93	3,46	4,39	1,68	2,52	214	**3,27 %**
200	50	5	4,93	3,94	-0,99	1,67	2,56	0,30	0,59	203	**3,45 %**
200	50	6	3,09	6,19	0,00	2,16	3,51	0,62	1,44	194	**3,09 %**
200	50	7	6,22	4,44	-0,44	2,40	3,38	0,53	1,24	225	**3,11 %**
200	50	8	5,29	7,69	-0,48	2,12	3,37	0,38	1,63	208	**3,37 %**
200	50	9	7,32	2,93	-0,98	3,41	3,41	1,46	2,34	205	**2,93 %**
200	50	10	3,07	0,88	-2,19	0,53	1,32	-1,58	-0,61	228	**4,82 %**
500	10	1	0,31	0,23	0,08	0,59	0,66	0,71	0,72	2602	**0,08 %**
500	10	2	0,30	0,15	0,15	0,13	0,12	0,12	0,11	2643	**0,04 %**
500	10	3	0,30	0,22	0,19	0,12	0,13	0,10	0,13	2680	**0,04 %**
500	10	4	0,25	0,15	0,11	0,15	0,15	0,13	0,12	2753	**0,04 %**
500	10	5	0,19	0,12	0,12	0,22	0,22	0,25	0,22	2597	**0,04 %**
500	10	6	0,19	0,27	0,08	0,55	0,53	0,47	0,44	2612	**0,04 %**
500	10	7	0,12	0,28	0,16	0,16	0,16	0,17	0,17	2522	**0,04 %**
500	10	8	0,30	0,08	0,11	0,16	0,16	0,14	0,15	2639	**0,04 %**
500	10	9	0,26	0,11	0,15	0,40	0,42	0,38	0,40	2711	**0,04 %**
500	10	10	0,30	0,04	0,15	0,28	0,25	0,31	0,30	2631	**0,04 %**
500	20	1	0,78	0,78	0,24	0,75	0,93	0,78	0,78	1274	**0,16 %**
500	20	2	0,54	0,54	0,23	0,17	0,25	0,19	0,20	1296	**0,23 %**
500	20	3	0,98	1,21	0,23	0,32	0,30	0,23	0,27	1323	**0,23 %**
500	20	4	0,52	0,44	0,00	0,49	0,52	0,34	0,44	1358	**0,37 %**

n	m	i	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir	$C_{\text{máx}}$	t. o **Gap**
500	20	5	0,53	0,90	0,15	0,17	0,21	0,17	0,21	1330	**0,23 %**
500	20	6	0,70	0,47	0,00	0,30	0,25	0,23	0,23	1286	**0,23 %**
500	20	7	0,45	0,90	0,15	0,42	0,45	0,34	0,43	1336	**0,22 %**
500	20	8	0,62	0,70	0,08	0,31	0,29	0,28	0,25	1293	**0,23 %**
500	20	9	0,86	1,10	0,16	0,58	0,72	0,53	0,61	1278	**0,16 %**
500	20	10	0,93	0,70	0,08	0,17	0,19	0,17	0,17	1284	**0,23 %**
500	30	1	2,02	0,79	0,00	0,31	0,27	0,22	0,27	889	**0,45 %**
500	30	2	1,55	0,95	0,24	0,57	0,55	0,45	0,53	838	**0,48 %**
500	30	3	1,73	1,85	0,12	0,72	0,72	0,42	0,58	867	**0,58 %**
500	30	4	2,56	1,16	0,12	0,14	0,19	0,00	0,09	860	**0,58 %**
500	30	5	2,00	0,89	-0,22	0,18	0,20	0,00	0,13	898	**0,67 %**
500	30	6	1,81	1,02	-0,11	0,02	0,11	-0,11	0,05	885	**0,68 %**
500	30	7	2,25	1,07	0,24	0,62	0,64	0,43	0,52	844	**0,36 %**
500	30	8	2,15	0,95	0,00	0,86	0,88	0,72	0,83	839	**0,6 %**
500	30	9	2,54	1,73	0,35	0,46	0,44	0,25	0,39	865	**0,35 %**
500	30	10	3,45	1,15	0,00	0,53	0,46	0,30	0,37	869	**0,46 %**
500	40	1	4,24	2,27	0,00	0,61	0,51	0,15	0,45	661	**0,91 %**
500	40	2	2,58	1,06	0,15	0,30	0,30	-0,03	0,18	660	**0,91 %**
500	40	3	4,51	2,95	-0,31	0,00	0,09	-0,22	-0,09	643	**1,09 %**
500	40	4	4,58	1,07	-0,46	-0,09	-0,06	-0,43	-0,27	655	**1,22 %**
500	40	5	2,50	1,91	0,00	0,50	0,62	0,27	0,44	679	**0,88 %**
500	40	6	5,06	1,69	0,15	0,74	0,83	0,34	0,71	652	**0,77 %**
500	40	7	4,03	1,34	0,00	0,30	0,24	0,00	0,15	670	**0,9 %**
500	40	8	4,37	2,34	0,16	0,44	0,41	0,09	0,34	641	**0,78 %**
500	40	9	4,29	1,91	-0,16	0,45	0,54	0,16	0,35	629	**0,95 %**
500	40	10	3,37	0,96	-0,16	0,29	0,35	0,03	0,29	623	**0,96 %**
500	50	1	4,31	2,54	-0,39	0,63	0,70	0,08	0,43	511	**1,37 %**
500	50	2	4,13	3,75	-0,56	0,08	0,26	-0,49	-0,11	533	**1,69 %**
500	50	3	5,12	2,85	-0,76	0,15	0,34	-0,27	0,00	527	**1,52 %**

n	m	i	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir	$C_{\text{máx}}$	t. o **Gap**
500	50	4	6,35	1,35	-0,38	0,38	0,42	-0,15	0,15	520	**1,54 %**
500	50	5	5,00	4,63	-0,37	0,07	0,26	-0,41	-0,26	540	**1,67 %**
500	50	6	4,36	3,03	-0,57	0,23	0,30	-0,34	0,08	528	**1,52 %**
500	50	7	5,02	3,47	-0,77	0,23	0,31	-0,42	-0,04	518	**1,54 %**
500	50	8	3,54	2,99	-0,19	0,93	0,82	0,26	0,60	536	**1,31 %**
500	50	9	4,51	1,88	-0,19	0,79	0,94	0,41	0,56	532	**1,13 %**
500	50	10	5,95	6,91	0,00	0,65	0,88	0,35	0,42	521	**0,96 %**
1000	10	1	0,20	0,13	0,13	0,07	0,09	0,09	0,08	5485	3441,58 seg
1000	10	2	0,23	0,09	0,08	0,11	0,13	0,14	0,12	5291	2543,42 seg
1000	10	3	0,10	0,08	0,10	0,22	0,25	0,23	0,23	5223	**0,02 %**
1000	10	4	0,11	0,09	0,13	0,09	0,09	0,10	0,10	5401	1943,83 seg
1000	10	5	0,15	0,19	0,08	0,09	0,09	0,08	0,08	5257	3988,52 seg
1000	10	6	0,21	0,09	0,09	0,07	0,08	0,08	0,07	5339	236,19 seg
1000	10	7	0,12	0,08	0,08	0,07	0,09	0,10	0,10	5208	**0,02 %**
1000	10	8	0,15	0,02	0,09	0,25	0,27	0,34	0,33	5336	**0,02 %**
1000	10	9	0,08	0,08	0,04	0,06	0,07	0,06	0,07	5237	**0,02 %**
1000	10	10	0,14	0,06	0,10	0,08	0,09	0,08	0,08	5146	5515,55 seg
1000	20	1	0,88	1,11	0,04	0,20	0,25	0,20	0,22	2604	**0,12 %**
1000	20	2	0,81	0,54	0,04	0,50	0,50	0,45	0,48	2581	**0,12 %**
1000	20	3	0,79	0,28	0,04	0,16	0,14	0,13	0,13	2523	**0,08 %**
1000	20	4	0,76	0,15	0,11	0,11	0,12	0,10	0,09	2622	**0,11 %**
1000	20	5	0,63	0,31	0,08	0,18	0,19	0,19	0,19	2555	**0,08 %**
1000	20	6	0,84	0,73	0,00	0,28	0,27	0,26	0,29	2608	**0,15 %**
1000	20	7	0,45	0,15	0,08	0,63	0,62	0,56	0,59	2659	**0,11 %**
1000	20	8	0,66	0,31	0,04	0,12	0,12	0,09	0,11	2577	**0,12 %**
1000	20	9	0,54	0,73	0,08	0,15	0,15	0,11	0,15	2615	**0,11 %**
1000	20	10	0,63	0,43	0,04	0,40	0,43	0,39	0,41	2535	**0,12 %**
1000	30	1	1,73	1,04	0,06	0,25	0,29	0,22	0,23	1734	**0,23 %**
1000	30	2	1,72	1,55	0,06	0,52	0,44	0,33	0,46	1744	**0,23 %**

n	m	i	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir	$C_{\text{máx}}$	t. o **Gap**
1000	30	3	1,47	0,65	-0,12	0,05	0,02	0,01	0,02	1695	**0,29 %**
1000	30	4	1,15	1,21	-0,06	0,12	0,14	0,10	0,09	1735	**0,29 %**
1000	30	5	1,58	0,58	0,00	0,28	0,29	0,27	0,33	1712	**0,29 %**
1000	30	6	1,41	1,93	-0,12	0,00	0,01	-0,02	-0,01	1707	**0,29 %**
1000	30	7	1,53	2,24	0,00	0,18	0,16	0,09	0,09	1697	**0,24 %**
1000	30	8	1,59	1,25	0,06	0,26	0,25	0,23	0,20	1761	**0,28 %**
1000	30	9	1,86	1,05	-0,06	0,13	0,10	0,08	0,09	1717	**0,29 %**
1000	30	10	1,57	0,99	0,00	0,34	0,36	0,26	0,31	1715	**0,29 %**
1000	40	1	2,40	1,65	-0,08	0,05	0,00	-0,08	-0,06	1332	**0,45 %**
1000	40	2	3,14	1,87	-0,07	0,06	0,06	-0,04	0,04	1336	**0,45 %**
1000	40	3	2,38	2,38	-0,16	0,22	0,21	0,14	0,17	1261	**0,48 %**
1000	40	4	2,40	1,65	-0,15	0,01	0,04	-0,03	-0,01	1334	**0,45 %**
1000	40	5	2,68	1,50	-0,16	0,00	0,00	-0,06	0,00	1269	**0,47 %**
1000	40	6	2,24	1,68	-0,16	0,24	0,24	0,18	0,22	1248	**0,4 %**
1000	40	7	2,43	1,67	-0,08	0,12	0,09	0,00	0,09	1316	**0,46 %**
1000	40	8	2,43	0,91	-0,15	0,23	0,23	0,11	0,23	1318	**0,46 %**
1000	40	9	2,82	2,59	0,00	0,12	0,15	0,06	0,11	1314	**0,38 %**
1000	40	10	2,95	1,79	-0,08	0,12	0,11	0,02	0,05	1288	**0,39 %**
1000	50	1	2,73	2,15	-0,39	-0,04	-0,12	-0,12	-0,02	1024	**0,78 %**
1000	50	2	4,07	1,84	-0,19	0,60	0,60	0,48	0,56	1032	**0,58 %**
1000	50	3	4,84	2,91	-0,29	-0,04	-0,02	-0,21	-0,14	1032	**0,68 %**
1000	50	4	5,69	3,99	-0,30	0,04	-0,02	-0,14	-0,04	1002	**0,6 %**
1000	50	5	5,79	2,85	-0,39	-0,06	-0,02	-0,18	-0,08	1019	**0,69 %**
1000	50	6	4,20	3,51	-0,20	0,18	0,16	0,00	0,08	1025	**0,68 %**
1000	50	7	4,99	2,26	-0,28	0,38	0,43	0,17	0,32	1063	**0,66 %**
1000	50	8	3,60	1,46	-0,29	0,02	0,00	-0,10	-0,08	1029	**0,58 %**
1000	50	9	4,29	4,67	-0,38	-0,17	-0,10	-0,34	-0,17	1050	**0,86 %**
1000	50	10	5,53	3,10	-0,10	0,48	0,45	0,21	0,35	1031	**0,58 %**

Tabla A.6 – Tabla completa de trabajos correlacionados.

A.7. Tabla completa para máquinas correlacionadas

n	m	i	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir	$C_{\text{máx}}$	t. o **Gap**
100	10	1	0,00	0,00	2,26	0,23	0,27	0,23	0,23	443	12,61 seg
100	10	2	0,00	0,24	1,65	0,19	0,14	0,33	0,00	425	3,53 seg
100	10	3	0,00	0,00	2,67	0,00	0,00	0,20	0,00	300	4,55 seg
100	10	4	0,00	0,00	0,28	0,28	0,22	0,28	0,22	356	4,13 seg
100	10	5	0,00	0,00	2,15	0,31	0,31	0,31	0,25	326	5,17 seg
100	10	6	0,00	0,00	2,44	0,12	0,00	0,37	0,00	164	2,3 seg
100	10	7	0,00	0,19	1,56	0,19	0,08	0,00	0,00	514	41,47 seg
100	10	8	0,00	0,00	1,50	0,75	1,13	0,38	0,23	266	2,16 seg
100	10	9	0,00	0,28	1,39	0,28	0,28	0,22	0,22	359	3,58 seg
100	10	10	0,00	0,32	0,64	0,45	0,32	0,38	0,25	314	3,73 seg
100	20	1	0,00	0,00	4,72	0,94	0,94	0,94	0,94	106	4,89 seg
100	20	2	0,00	0,70	2,80	0,70	0,70	0,42	0,42	143	6,27 seg
100	20	3	0,00	0,00	6,67	0,67	0,83	0,67	0,50	120	1,08 seg
100	20	4	0,00	0,00	5,88	1,18	1,18	1,18	1,18	85	7,27 seg
100	20	5	0,00	0,00	2,27	0,57	0,57	0,23	0,34	176	4 seg
100	20	6	0,00	0,00	3,92	0,00	0,00	0,00	0,00	102	3,89 seg
100	20	7	0,00	0,00	0,90	0,09	0,00	0,00	0,00	221	8,22 seg
100	20	8	0,00	0,00	4,46	0,64	0,64	0,25	0,51	157	8,63 seg
100	20	9	0,82	0,00	0,82	0,00	0,00	0,00	0,00	122	**1,64 %**
100	20	10	0,00	0,00	1,92	0,29	0,10	0,00	0,00	208	4,42 seg
100	30	1	0,00	0,00	8,97	1,28	1,28	1,28	1,28	78	5,39 seg
100	30	2	0,00	0,00	3,06	0,20	0,82	0,41	0,00	98	26,81 seg
100	30	3	0,00	0,00	0,99	0,00	0,00	0,00	0,00	101	11,19 seg
100	30	4	0,00	0,00	6,06	0,76	0,00	0,00	0,15	132	7,06 seg
100	30	5	0,00	0,00	5,26	1,05	2,63	1,05	0,79	76	4,33 seg
100	30	6	0,00	0,00	3,03	0,00	0,00	0,00	0,00	132	**0,76 %**
100	30	7	0,00	0,00	2,82	0,00	0,00	0,00	0,00	142	6,72 seg

n	m	i	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir	$C_{\text{máx}}$	t. o **Gap**
100	30	8	0,00	0,82	4,10	0,82	0,82	0,82	0,82	122	7,06 seg
100	30	9	0,00	0,00	1,79	0,00	0,00	0,00	0,00	112	12,75 seg
100	30	10	0,00	0,00	1,64	0,33	0,16	0,16	0,00	122	7,83 seg
100	40	1	0,00	0,00	5,97	1,49	1,49	1,49	1,49	67	2,67 seg
100	40	2	0,00	0,00	5,48	0,55	0,82	0,55	0,00	73	5,42 seg
100	40	3	0,00	0,00	4,00	0,00	0,00	0,00	0,00	75	2,64 seg
100	40	4	0,00	0,00	6,31	0,00	0,18	0,00	0,00	111	8,17 seg
100	40	5	0,00	0,00	5,26	0,00	0,00	0,00	0,00	95	16,81 seg
100	40	6	0,00	0,00	3,41	0,00	0,00	0,23	0,00	88	1,42 seg
100	40	7	0,00	0,00	2,74	0,00	0,00	0,00	0,00	73	5,33 seg
100	40	8	0,00	0,00	4,17	0,00	0,00	0,00	0,00	72	3,36 seg
100	40	9	0,00	0,00	1,02	0,20	0,61	0,00	0,00	98	5,97 seg
100	40	10	1,15	0,00	3,45	0,00	0,00	0,00	0,00	87	**1,15 %**
100	50	1	0,00	0,00	5,19	0,00	0,00	0,00	0,00	77	14,81 seg
100	50	2	0,00	0,00	6,06	0,00	0,00	0,00	0,00	66	6,5 seg
100	50	3	0,00	0,00	1,61	0,00	0,32	0,00	0,00	62	1,08 seg
100	50	4	0,00	0,00	2,53	0,00	0,00	0,00	0,00	79	7,13 seg
100	50	5	0,00	0,00	2,70	0,00	0,00	0,00	0,00	74	9,45 seg
100	50	6	1,52	0,00	10,61	1,52	1,52	1,82	1,52	66	23,22 seg
100	50	7	1,35	0,00	5,41	0,27	0,00	0,00	0,00	74	39,16 seg
100	50	8	0,00	0,00	5,13	0,00	0,00	0,00	0,00	78	5,06 seg
100	50	9	0,00	0,00	5,80	1,16	1,45	1,45	0,29	69	1,5 seg
100	50	10	0,00	0,00	8,62	1,72	1,72	2,07	1,72	58	4,39 seg
200	10	1	0,39	0,39	1,75	1,05	1,13	1,05	0,97	515	**0,19 %**
200	10	2	0,00	0,21	2,11	0,55	0,51	0,47	0,42	473	10,03 seg
200	10	3	0,12	0,12	0,97	0,36	0,32	0,27	0,24	822	**0,12 %**
200	10	4	1,02	0,13	1,27	0,48	0,51	0,41	0,38	785	3225,13 seg
200	10	5	0,17	0,34	2,39	0,78	0,68	0,48	0,61	587	4298,13 seg
200	10	6	0,00	0,00	0,62	0,21	0,21	0,05	0,11	1138	11,64 seg

n	m	i	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir	$C_{\text{máx}}$	t. o **Gap**
200	10	7	1,21	1,04	1,04	0,69	0,59	0,52	0,62	578	**0,35 %**
200	10	8	0,84	0,12	1,45	0,65	0,58	0,39	0,53	830	**0,12 %**
200	10	9	0,00	0,93	1,12	0,19	0,19	0,04	0,15	535	**0,93 %**
200	10	10	0,76	0,46	1,99	0,67	0,61	0,55	0,61	654	**0,31 %**
200	20	1	0,24	0,24	0,71	0,05	0,00	0,00	0,00	422	19,28 seg
200	20	2	1,40	0,70	0,47	0,37	0,33	0,14	0,23	428	**0,7 %**
200	20	3	1,12	0,00	2,81	1,12	1,24	1,12	1,12	178	32,91 seg
200	20	4	0,79	0,26	1,06	0,26	0,26	0,00	0,11	378	**0,53 %**
200	20	5	1,42	0,85	0,85	0,00	0,00	0,00	0,00	352	**1,14 %**
200	20	6	1,13	1,13	1,97	0,68	0,51	0,34	0,51	355	**0,56 %**
200	20	7	1,64	1,31	0,98	0,72	0,72	0,46	0,52	305	**0,66 %**
200	20	8	0,00	0,53	0,53	0,11	0,21	0,00	0,00	376	14,99 seg
200	20	9	1,03	0,26	0,26	0,26	0,26	0,26	0,26	387	**1,29 %**
200	20	10	0,28	0,00	1,93	0,61	0,61	0,44	0,50	363	30,25 seg
200	30	1	0,83	0,00	3,75	0,42	0,50	0,00	0,17	240	**0,83 %**
200	30	2	1,25	1,88	3,13	1,38	1,00	1,25	1,25	160	**1,25 %**
200	30	3	0,47	0,00	1,88	0,47	0,47	0,28	0,28	213	17,08 seg
200	30	4	1,91	1,44	1,44	0,57	0,77	0,48	0,48	209	**1,44 %**
200	30	5	1,79	0,36	0,72	0,00	0,00	0,00	0,00	279	41,09 seg
200	30	6	1,54	2,05	3,08	1,54	1,13	1,23	0,92	195	4248,7 seg
200	30	7	0,00	2,30	0,57	0,46	0,46	0,34	0,34	348	25,67 seg
200	30	8	1,89	1,89	3,14	1,51	1,64	1,13	1,13	159	30,7 seg
200	30	9	1,88	0,00	3,75	0,63	0,63	0,38	0,63	160	**0,63 %**
200	30	10	1,22	1,63	1,22	0,65	0,65	0,16	0,57	246	**0,81 %**
200	40	1	2,33	1,55	13,95	1,40	1,24	1,40	0,93	129	43,83 seg
200	40	2	1,14	1,70	0,57	0,23	0,57	0,00	0,00	176	64,63 seg
200	40	3	2,00	0,00	8,00	1,20	1,33	1,07	1,33	150	45,58 seg
200	40	4	2,40	2,40	1,60	0,80	0,80	0,80	0,80	125	28,92 seg
200	40	5	1,27	0,00	3,80	0,51	0,25	0,00	0,13	158	**0,63 %**

n	m	i	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir	$C_{\text{máx}}$	t. o **Gap**
200	40	6	1,18	0,59	2,35	0,59	0,59	0,35	0,47	170	48,56 seg
200	40	7	1,51	0,00	1,01	0,60	0,50	0,50	0,50	199	67,88 seg
200	40	8	1,42	1,42	2,13	0,57	0,43	0,14	0,28	141	**1,42 %**
200	40	9	1,31	0,65	3,27	0,13	0,00	0,13	0,13	153	**1,31 %**
200	40	10	1,27	1,27	7,01	0,64	0,64	0,64	0,64	157	88,64 seg
200	50	1	2,03	0,00	5,41	0,41	0,68	0,41	0,27	148	109,5 seg
200	50	2	1,61	0,00	4,84	0,65	0,81	0,65	0,81	124	101,92 seg
200	50	3	3,05	1,22	1,22	0,00	0,00	0,00	0,00	164	125,91 seg
200	50	4	1,33	0,00	10,67	1,33	0,27	0,27	0,27	75	**1,33 %**
200	50	5	1,61	1,61	1,61	1,13	1,29	0,81	0,97	124	**0,81 %**
200	50	6	0,84	0,00	5,88	0,84	0,84	0,67	0,67	119	27,19 seg
200	50	7	1,67	0,00	1,67	0,00	0,00	0,00	0,00	120	64,81 seg
200	50	8	2,59	1,72	4,31	1,38	1,72	0,86	1,03	116	49,23 seg
200	50	9	1,39	1,39	1,39	0,56	0,56	0,14	0,56	144	**1,39 %**
200	50	10	2,94	0,98	5,88	1,96	1,57	1,76	1,96	102	83,09 seg
500	10	1	0,16	0,08	1,17	0,89	0,91	0,80	0,84	1279	**0,23 %**
500	10	2	0,38	0,11	0,65	0,36	0,34	0,35	0,33	2610	**0,11 %**
500	10	3	0,21	0,10	0,73	0,71	0,68	0,69	0,65	1912	53,81 seg
500	10	4	0,24	0,16	0,72	0,92	0,86	0,89	0,84	1255	**0,08 %**
500	10	5	0,13	0,17	0,35	0,35	0,37	0,31	0,34	2298	**0,09 %**
500	10	6	0,33	0,28	0,71	0,96	0,93	0,85	0,93	2116	**0,09 %**
500	10	7	0,34	0,13	0,63	0,49	0,45	0,35	0,38	2374	231,58 seg
500	10	8	0,65	0,12	0,77	0,25	0,22	0,19	0,21	2603	**0,08 %**
500	10	9	0,09	0,17	0,96	0,44	0,47	0,35	0,40	1146	**0,17 %**
500	10	10	0,24	0,16	0,64	0,64	0,54	0,71	0,46	1248	**0,16 %**
500	20	1	0,76	0,76	0,91	0,94	1,00	0,82	0,91	661	**0,45 %**
500	20	2	0,68	0,68	1,09	0,90	0,93	0,90	0,82	734	**0,41 %**
500	20	3	0,97	0,12	1,09	0,51	0,51	0,48	0,53	829	**0,36 %**
500	20	4	0,82	0,49	1,15	0,82	0,79	0,69	0,79	611	**0,16 %**

n	m	i	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir	$C_{\text{máx}}$	t. o **Gap**
500	20	5	0,61	0,35	0,79	0,49	0,47	0,39	0,42	1140	**0,26 %**
500	20	6	1,02	0,16	0,47	0,19	0,24	0,09	0,16	1275	**0,86 %**
500	20	7	1,13	1,25	1,13	0,82	0,86	0,86	0,84	882	**0,11 %**
500	20	8	0,61	0,87	0,78	0,26	0,24	0,21	0,23	1150	**0,78 %**
500	20	9	0,88	0,66	6,39	1,32	1,23	1,28	1,19	454	**0,44 %**
500	20	10	1,15	0,23	2,75	0,92	0,78	0,92	0,83	436	155 seg
500	30	1	0,96	0,96	0,72	0,72	0,77	0,67	0,67	418	**0,72 %**
500	30	2	1,31	1,12	0,75	0,82	1,12	0,60	0,67	534	130,48 seg
500	30	3	0,41	1,44	0,21	1,15	1,03	1,03	1,03	486	**0,41 %**
500	30	4	0,70	1,40	0,93	1,26	1,17	1,17	1,17	429	768,67 seg
500	30	5	1,15	0,92	1,61	0,46	0,46	0,51	0,37	434	**0,46 %**
500	30	6	1,64	1,64	1,37	0,77	0,77	0,55	0,71	366	**0,55 %**
500	30	7	1,52	0,95	1,52	0,95	0,98	0,91	0,91	528	**0,38 %**
500	30	8	0,78	1,56	0,93	0,87	0,97	0,90	0,90	642	211,23 seg
500	30	9	1,52	0,00	0,00	0,39	0,26	0,22	0,22	462	**0,87 %**
500	30	10	0,94	1,72	0,16	0,53	0,47	0,38	0,47	638	**0,63 %**
500	40	1	1,62	1,62	2,92	1,30	1,30	1,10	1,17	308	**0,97 %**
500	40	2	1,27	2,22	1,27	1,27	1,27	1,27	1,27	316	**0,32 %**
500	40	3	1,62	2,70	2,70	0,54	0,59	0,54	0,54	370	**0,54 %**
500	40	4	0,96	0,96	7,69	0,71	0,71	0,38	0,58	312	**0,96 %**
500	40	5	1,96	1,31	1,31	0,65	0,72	0,78	0,65	306	**0,65 %**
500	40	6	2,54	0,51	3,05	0,56	0,61	0,51	0,56	394	2412,98 seg
500	40	7	2,12	1,89	1,89	0,42	0,47	0,47	0,28	424	**0,47 %**
500	40	8	2,15	3,08	0,92	1,11	1,11	0,92	0,98	325	**0,62 %**
500	40	9	1,09	1,74	2,17	0,30	0,35	0,17	0,09	460	**0,65 %**
500	40	10	1,41	2,54	1,41	0,73	0,68	0,68	0,73	355	**0,56 %**
500	50	1	4,44	3,33	2,22	2,44	2,44	2,33	2,33	180	2036,47 seg
500	50	2	1,36	2,04	2,04	0,75	0,82	0,68	0,68	294	**0,68 %**
500	50	3	2,49	4,27	4,98	1,35	1,42	1,35	1,35	281	121,83 seg

n	m	i	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir	$C_{\text{máx}}$	t. o **Gap**
500	50	4	2,45	2,86	2,86	0,82	0,90	0,82	0,90	245	93,88 seg
500	50	5	4,69	3,13	2,81	0,88	0,81	0,69	0,63	320	**0,31 %**
500	50	6	2,60	0,52	4,69	0,42	0,42	0,52	0,52	192	307,52 seg
500	50	7	1,73	5,19	0,58	0,35	0,29	0,29	0,29	347	**0,29 %**
500	50	8	1,69	3,73	3,73	0,68	0,81	0,61	0,41	295	437,14 seg
500	50	9	2,95	1,27	1,27	1,18	1,27	1,27	1,10	237	**0,84 %**
500	50	10	2,78	2,38	4,76	1,11	1,19	1,11	1,19	252	**0,79 %**
1000	10	1	0,12	0,08	0,24	0,76	0,73	0,71	0,71	2546	**0,04 %**
1000	10	2	0,40	0,08	0,45	0,63	0,61	0,59	0,63	3537	**0,06 %**
1000	10	3	0,25	0,15	0,20	0,82	0,84	0,83	0,85	2040	**0,1 %**
1000	10	4	0,12	0,37	0,16	0,62	0,64	0,55	0,58	4313	**0,14 %**
1000	10	5	0,26	0,15	0,30	0,76	0,74	0,67	0,64	2649	**0,15 %**
1000	10	6	0,13	0,37	0,37	0,22	0,20	0,18	0,17	6286	**0,05 %**
1000	10	7	0,36	0,11	0,36	0,88	0,85	0,80	0,76	3582	**0,2 %**
1000	10	8	0,43	0,35	0,64	0,61	0,60	0,64	0,58	2822	**0,07 %**
1000	10	9	0,23	0,13	0,33	0,77	0,74	0,74	0,65	3037	**0,03 %**
1000	10	10	0,39	0,23	0,32	0,32	0,29	0,29	0,30	5672	**0,04 %**
1000	20	1	0,57	0,38	0,77	0,88	0,88	0,86	0,92	1045	**0,38 %**
1000	20	2	0,51	0,41	0,81	1,73	1,73	1,69	1,66	985	**0,1 %**
1000	20	3	0,68	0,54	0,49	0,60	0,55	0,52	0,59	2044	**0,39 %**
1000	20	4	0,38	0,29	0,38	0,91	1,05	0,88	0,82	1051	**0,19 %**
1000	20	5	0,54	0,59	0,36	0,54	0,58	0,50	0,46	1682	**0,59 %**
1000	20	6	0,55	0,71	0,24	0,66	0,63	0,66	0,69	1274	**0,39 %**
1000	20	7	0,82	0,47	0,93	1,35	1,47	1,38	1,42	858	**0,23 %**
1000	20	8	0,00	-0,08	0,40	0,74	0,78	0,84	0,78	1237	**0,49 %**
1000	20	9	0,35	0,39	0,35	0,40	0,39	0,33	0,38	2582	**0,43 %**
1000	20	10	0,40	0,24	0,40	0,92	0,84	0,79	0,78	1235	**0,08 %**
1000	30	1	0,40	0,87	0,40	0,65	0,63	0,60	0,58	1265	**0,16 %**
1000	30	2	0,55	1,11	0,78	1,35	1,47	1,44	1,40	901	**0,33 %**

n	m	i	CPLEX	Partial	RBS	IGS	IGD	NSP	Vir	$C_{\text{máx}}$	t. o **Gap**
1000	30	3	0,66	1,06	4,39	0,69	0,69	0,69	0,66	752	**0,13 %**
1000	30	4	0,92	1,17	0,58	0,62	0,67	0,58	0,57	1197	**0,58 %**
1000	30	5	0,70	0,70	0,50	0,94	1,00	0,96	0,96	1004	**0,3 %**
1000	30	6	1,11	3,61	0,69	1,22	1,33	1,25	1,28	721	**0,14 %**
1000	30	7	1,48	0,74	1,19	1,34	1,34	1,25	1,28	674	**0,3 %**
1000	30	8	1,47	1,47	1,16	0,76	0,84	0,90	0,90	952	**0,32 %**
1000	30	9	0,94	0,57	1,32	1,70	1,66	1,70	1,66	530	498,17 seg
1000	30	10	1,40	0,64	1,02	0,97	0,92	0,89	0,89	786	**0,25 %**
1000	40	1	1,18	1,84	1,97	0,92	1,00	0,79	0,92	760	**0,26 %**
1000	40	2	2,89	1,93	0,84	0,51	0,70	0,60	0,51	830	**0,36 %**
1000	40	3	2,05	2,61	2,39	0,66	0,75	0,66	0,59	880	896,92 seg
1000	40	4	1,46	3,73	4,38	0,97	0,88	0,91	0,94	616	**0,16 %**
1000	40	5	2,27	0,30	1,97	0,36	0,42	0,39	0,36	660	3056,44 seg
1000	40	6	4,28	3,74	0,36	1,03	1,07	1,03	1,03	561	**0,18 %**
1000	40	7	0,47	1,75	1,17	0,56	0,63	0,61	0,58	855	**0,47 %**
1000	40	8	1,34	1,18	0,84	0,94	0,94	0,87	0,97	595	**0,34 %**
1000	40	9	4,13	1,59	0,74	0,42	0,44	0,42	0,42	945	**0,32 %**
1000	40	10	5,17	4,14	3,10	1,14	1,17	1,14	1,07	580	1184,86 seg
1000	50	1	2,98	2,86	0,83	0,74	0,81	0,81	0,83	840	305,42 seg
1000	50	2	5,60	2,99	2,99	0,93	1,16	1,01	1,12	536	**0,19 %**
1000	50	3	9,92	7,98	5,84	0,97	1,09	0,97	0,86	514	**0,19 %**
1000	50	4	2,78	2,23	2,23	0,14	0,14	0,14	0,14	719	**0,42 %**
1000	50	5	5,19	5,97	4,16	1,51	1,40	1,40	1,40	385	3119,19 seg
1000	50	6	5,60	5,60	3,05	1,12	1,12	1,17	1,22	393	172,56 seg
1000	50	7	5,53	5,26	1,32	1,32	1,47	1,26	1,32	380	**0,26 %**
1000	50	8	3,62	1,70	2,98	0,81	0,98	1,11	0,98	470	3874,78 seg
1000	50	9	4,54	3,55	1,97	0,95	0,95	0,91	0,91	507	**0,2 %**
1000	50	10	6,14	4,21	2,11	0,84	0,74	0,84	0,70	570	172,34 seg

Tabla A.7 – Tabla completa para máquinas correlacionadas.

ANEXO B

GRÁFICOS Y TABLAS DE INTERACCIÓN DE LOS ALGORITMOS INICIALES

En el este anexo vemos dos gráficos por cada intervalo de tiempos de proceso con la interacción de los resultados obtenidos de la desviación porcentual con el número de trabajos y el número de máquinas respectivamente. En estas gráficos solo se representan las medias, y no los intervalos del 95 % de Tukey HSD, por claridad de las figuras, ya que en muchos casos se produce un gran solapamiento de las mismos. No obstante, los intervalos de confianza se adjuntaran en unas tablas donde se reflejan, además, los algoritmos, el número de instancias evaluadas de cada uno (recordemos que para los algoritmos propuestos algoritmos se han realizado 5 replicas por cada instancia), la media, el error standard y los límites inferior y superior del intervalo de confianza del 95 % de cada media.

B.1. Interacciones en el intervalo U(1,100)

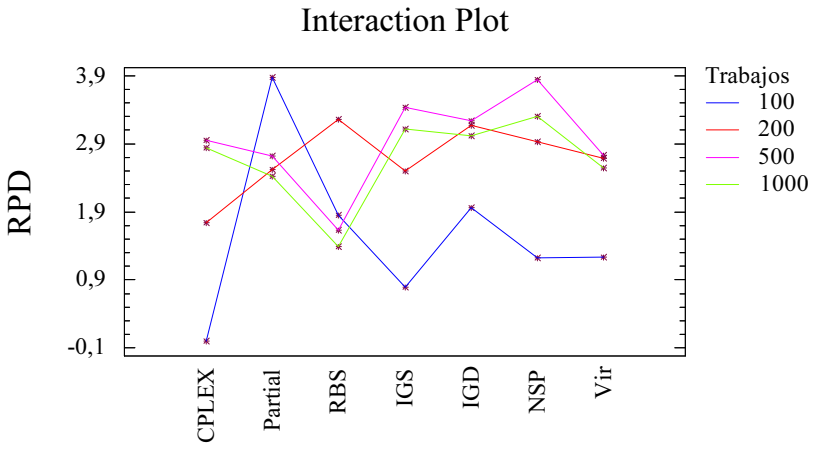

Figura B.1 – Gráfico de interacción de algoritmos frente número de trabajos, para la distribución de p_{ij} U(1,100).

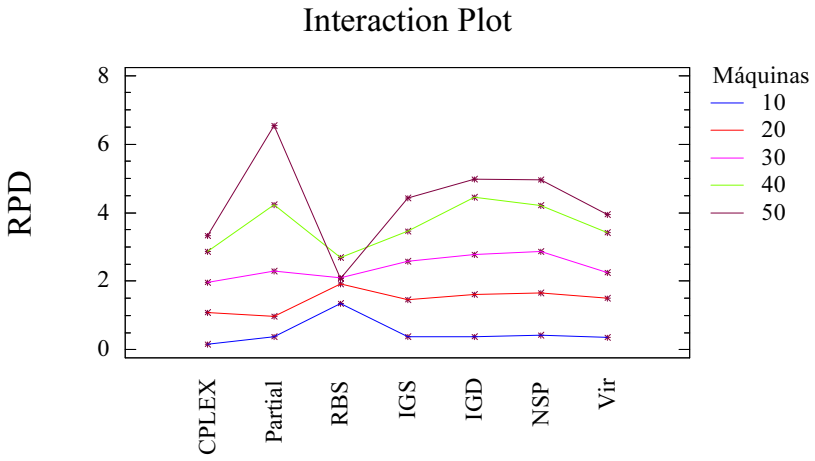

Figura B.2 – Gráfico de interacción de algoritmos frente número de máquinas, para la distribución de p_{ij} U(1,100).

Algoritmo	Trab.	Núm.	Media	Error Stnd.	Lim. Inf.	Lim. Sup.
CPLEX	100	50	0,000	0,314	-0,610	0,615
CPLEX	200	50	1,743	0,314	1,127	2,358
CPLEX	500	50	2,945	0,314	2,329	3,560
CPLEX	1000	50	2,842	0,314	2,227	3,457
Partial	100	50	3,872	0,314	3,256	4,486
Partial	200	50	2,518	0,314	1,903	3,133
Partial	500	50	2,720	0,314	2,104	3,335
Partial	1000	50	2,418	0,314	1,803	3,033
RBS	100	50	1,846	0,314	1,231	2,461
RBS	200	50	3,252	0,314	2,637	3,867
RBS	500	50	1,630	0,314	1,014	2,245
RBS	1000	50	1,392	0,314	0,776	2,007
IGS	100	250	0,791	0,140	0,516	1,066
IGS	200	250	2,500	0,140	2,225	2,775
IGS	500	250	3,433	0,140	3,157	3,708
IGS	1000	250	3,114	0,140	2,839	3,389
IGD	100	250	1,965	0,140	1,690	2,240
IGD	200	250	3,165	0,140	2,890	3,440
IGD	500	250	3,234	0,140	2,958	3,509
IGD	1000	250	3,019	0,140	2,744	3,294
NSP	100	250	1,219	0,140	0,943	1,494
NSP	200	250	2,932	0,140	2,657	3,207
NSP	500	250	3,840	0,140	3,565	4,115
NSP	1000	250	3,301	0,140	3,025	3,576
Vir	100	250	1,236	0,140	0,961	1,511
Vir	200	250	2,685	0,140	2,410	2,960
Vir	500	250	2,728	0,140	2,453	3,003
Vir	1000	250	2,541	0,140	2,266	2,816

Núm.= número de instancias evaluadas ($\times 5$ replicas para nuestros algoritmos), Lim. Inf. y Sup. del intervalo del 95 %.

Tabla B.1 – Tabla de datos estadísticos agrupados por trabajos para el intervalo U(1,100).

Algoritmo	Máq.	Núm.	Media	Error Stnd.	Lim. Inf.	Lim. Sup.
CPLEX	10	40	0,170	0,351	-0,510	0,858
CPLEX	20	40	1,091	0,351	0,403	1,778
CPLEX	30	40	1,970	0,351	1,282	2,657
CPLEX	40	40	2,863	0,351	2,175	3,550
CPLEX	50	40	3,319	0,351	2,631	4,006
Partial	10	40	0,393	0,351	-0,290	1,081
Partial	20	40	0,971	0,351	0,283	1,658
Partial	30	40	2,290	0,351	1,602	2,977
Partial	40	40	4,226	0,351	3,538	4,914
Partial	50	40	6,530	0,351	5,842	7,217
RBS	10	40	1,361	0,351	0,673	2,048
RBS	20	40	1,933	0,351	1,245	2,620
RBS	30	40	2,092	0,351	1,404	2,779
RBS	40	40	2,684	0,351	1,996	3,371
RBS	50	40	2,080	0,351	1,393	2,768
IGS	10	200	0,376	0,157	0,068	0,683
IGS	20	200	1,463	0,157	1,155	1,771
IGS	30	200	2,576	0,157	2,268	2,883
IGS	40	200	3,462	0,157	3,154	3,769
IGS	50	200	4,422	0,157	4,114	4,730
IGD	10	200	0,391	0,157	0,083	0,699
IGD	20	200	1,621	0,157	1,314	1,929
IGD	30	200	2,790	0,157	2,482	3,098
IGD	40	200	4,456	0,157	4,148	4,764
IGD	50	200	4,969	0,157	4,662	5,277
NSP	10	200	0,426	0,157	0,118	0,733
NSP	20	200	1,667	0,157	1,359	1,974
NSP	30	200	2,859	0,157	2,551	3,166
NSP	40	200	4,199	0,157	3,891	4,506
NSP	50	200	4,965	0,157	4,657	5,272
Vir	10	200	0,368	0,157	0,060	0,675
Vir	20	200	1,496	0,157	1,188	1,804
Vir	30	200	2,261	0,157	1,953	2,568
Vir	40	200	3,426	0,157	3,118	3,734
Vir	50	200	3,937	0,157	3,629	4,244

Tabla B.2 – Tabla de datos estadísticos agrupados por máquinas para el intervalo U(1,100).

B.2. Interacciones en el intervalo U(10,100)

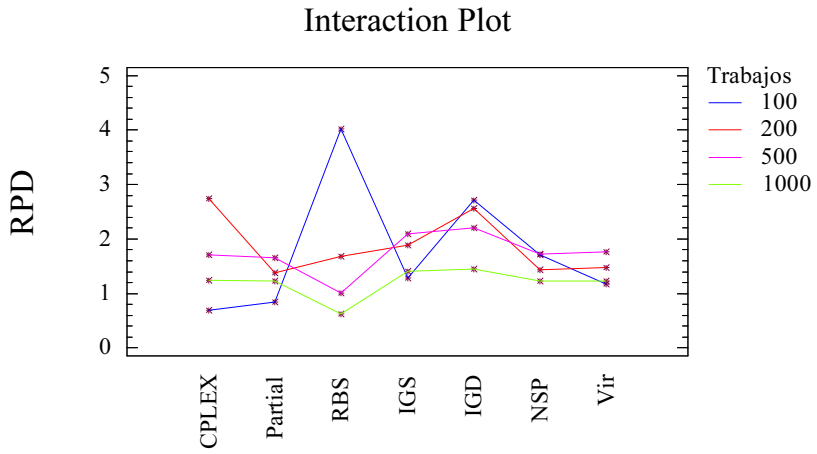

Figura B.3 – Gráfico de interacción de algoritmos frente número de trabajos, para la distribución de p_{ij} U(10,100).

Figura B.4 – Gráfico de interacción de algoritmos frente número de máquinas, para la distribución de p_{ij} U(10,100).

Algoritmo	Trab.	Núm.	Media	Error Stnd.	Lim. Inf.	Lim. Sup.
CPLEX	100	50	0,724	0,134	0,462	0,986
CPLEX	200	50	2,848	0,134	2,586	3,110
CPLEX	500	50	1,729	0,134	1,467	1,991
CPLEX	1000	50	1,242	0,134	0,980	1,504
Partial	100	50	0,868	0,134	0,606	1,130
Partial	200	50	1,484	0,134	1,222	1,746
Partial	500	50	1,676	0,134	1,414	1,938
Partial	1000	50	1,228	0,134	0,966	1,490
RBS	100	50	4,050	0,134	3,787	4,311
RBS	200	50	1,791	0,134	1,528	2,052
RBS	500	50	1,030	0,134	0,768	1,292
RBS	1000	50	0,629	0,134	0,367	0,891
IGS	100	250	1,303	0,060	1,186	1,420
IGS	200	250	1,996	0,060	1,879	2,113
IGS	500	250	2,111	0,060	1,993	2,228
IGS	1000	250	1,405	0,060	1,287	1,522
IGD	100	250	2,737	0,060	2,620	2,854
IGD	200	250	2,677	0,060	2,559	2,794
IGD	500	250	2,222	0,060	2,105	2,339
IGD	1000	250	1,450	0,060	1,332	1,567
NSP	100	250	1,734	0,060	1,617	1,851
NSP	200	250	1,548	0,060	1,431	1,665
NSP	500	250	1,732	0,060	1,615	1,850
NSP	1000	250	1,222	0,060	1,105	1,339
Vir	100	250	1,202	0,060	1,084	1,319
Vir	200	250	1,582	0,060	1,464	1,699
Vir	500	250	1,780	0,060	1,663	1,897
Vir	1000	250	1,234	0,060	1,117	1,351

Núm.= número de instancias evaluadas ($\times 5$ replicas para nuestros algoritmos), Lim. Inf. y Sup. del intervalo del 95 %.

Tabla B.3 – Tabla de datos estadísticos agrupados por trabajos para el intervalo U(10,100).

Algoritmo	Máq.	Núm.	Media	Error Stnd.	Lim. Inf.	Lim. Sup.
CPLEX	10	40	0,192	0,149	-0,100	0,485
CPLEX	20	40	1,063	0,149	0,770	1,355
CPLEX	30	40	2,071	0,149	1,778	2,364
CPLEX	40	40	2,398	0,149	2,105	2,691
CPLEX	50	40	2,456	0,149	2,163	2,748
Partial	10	40	0,405	0,149	0,111	0,697
Partial	20	40	0,729	0,149	0,435	1,021
Partial	30	40	1,520	0,149	1,227	1,812
Partial	40	40	1,974	0,149	1,681	2,267
Partial	50	40	1,942	0,149	1,649	2,235
RBS	10	40	0,789	0,149	0,496	1,082
RBS	20	40	1,423	0,149	1,129	1,715
RBS	30	40	2,071	0,149	1,778	2,363
RBS	40	40	2,450	0,149	2,157	2,743
RBS	50	40	2,642	0,149	2,349	2,935
IGS	10	200	0,433	0,067	0,302	0,564
IGS	20	200	1,224	0,067	1,092	1,355
IGS	30	200	1,929	0,067	1,798	2,060
IGS	40	200	2,688	0,067	2,556	2,819
IGS	50	200	2,245	0,067	2,113	2,375
IGD	10	200	0,383	0,067	0,251	0,514
IGD	20	200	1,256	0,067	1,125	1,387
IGD	30	200	2,455	0,067	2,324	2,586
IGD	40	200	3,746	0,067	3,614	3,876
IGD	50	200	3,517	0,067	3,386	3,648
NSP	10	200	0,405	0,067	0,274	0,536
NSP	20	200	1,055	0,067	0,923	1,185
NSP	30	200	1,687	0,067	1,555	1,817
NSP	40	200	2,550	0,067	2,419	2,681
NSP	50	200	2,100	0,067	1,969	2,231
Vir	10	200	0,315	0,067	0,184	0,446
Vir	20	200	0,886	0,067	0,754	1,017
Vir	30	200	1,566	0,067	1,435	1,697
Vir	40	200	2,426	0,067	2,295	2,557
Vir	50	200	2,054	0,067	1,923	2,185

Tabla B.4 – Tabla de datos estadísticos agrupados por máquinas para el intervalo U(10,100).

B.3. Interacciones en el intervalo U(100,200)

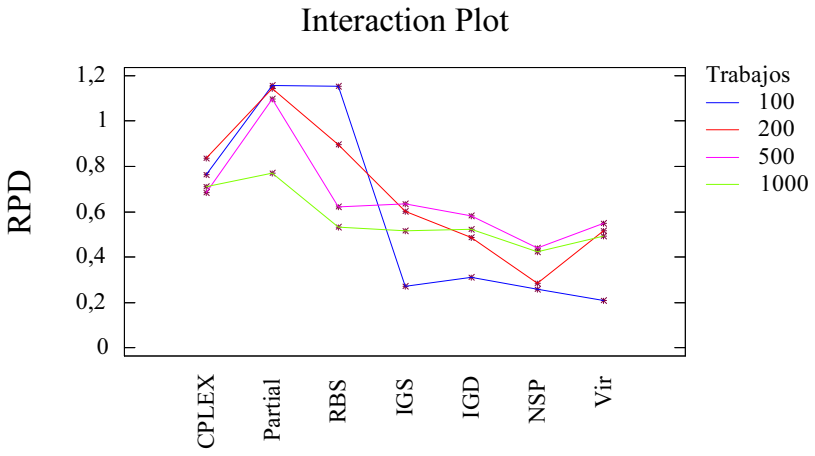

Figura B.5 – Gráfico de interacción de algoritmos frente número de trabajos, para la distribución de p_{ij} U(100,200).

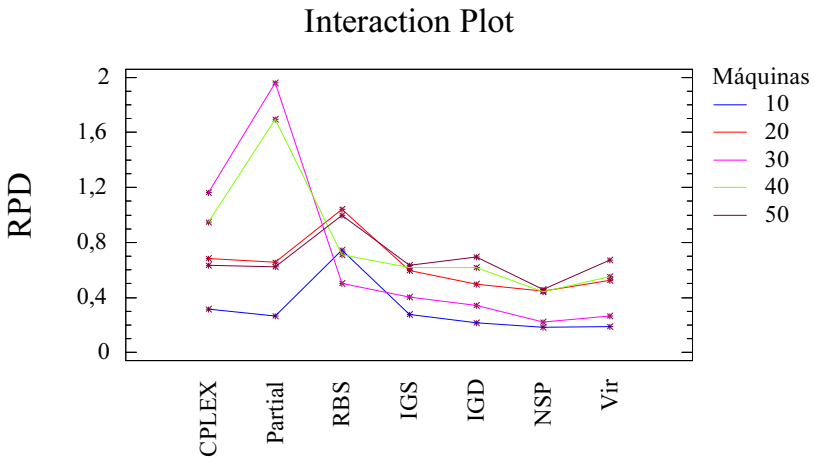

Figura B.6 – Gráfico de interacción de algoritmos frente número de máquinas, para la distribución de p_{ij} U(100,200).

Algoritmo	Trab.	Núm.	Media	Error Stnd.	Lim. Inf.	Lim. Sup.
CCPLEX	100	50	0,802	0,051	0,702	0,901
CPLEX	200	50	0,839	0,051	0,739	0,939
CPLEX	500	50	0,684	0,051	0,584	0,784
CPLEX	1000	50	0,709	0,051	0,610	0,809
Partial	100	50	1,188	0,051	1,088	1,288
Partial	200	50	1,145	0,051	1,045	1,245
Partial	500	50	1,097	0,051	0,997	1,197
Partial	1000	50	0,754	0,051	0,654	0,854
RBS	100	50	1,189	0,051	1,089	1,289
RBS	200	50	0,899	0,051	0,799	0,998
RBS	500	50	0,621	0,051	0,521	0,721
RBS	1000	50	0,531	0,051	0,431	0,631
IGS	100	250	0,308	0,023	0,264	0,353
IGS	200	250	0,605	0,023	0,560	0,650
IGS	500	250	0,634	0,023	0,590	0,679
IGS	1000	250	0,518	0,023	0,473	0,562
IGD	100	250	0,347	0,023	0,303	0,392
IGD	200	250	0,489	0,023	0,444	0,534
IGD	500	250	0,581	0,023	0,536	0,626
IGD	1000	250	0,523	0,023	0,478	0,567
NSP	100	250	0,295	0,023	0,250	0,340
NSP	200	250	0,288	0,023	0,243	0,332
NSP	500	250	0,439	0,023	0,395	0,484
NSP	1000	250	0,423	0,023	0,379	0,468
Vir	100	250	0,246	0,023	0,202	0,291
Vir	200	250	0,519	0,023	0,474	0,563
Vir	500	250	0,550	0,023	0,506	0,595
Vir	1000	250	0,491	0,023	0,447	0,536

Núm.= número de instancias evaluadas ($\times 5$ replicas para nuestros algoritmos), Lim. Inf. y Sup. del intervalo del 95 %.

Tabla B.5 – Tabla de datos estadísticos agrupados por trabajos para el intervalo U(100,200).

Algoritmo	Máq.	Núm.	Media	Error Stnd.	Lim. Inf.	Lim. Sup.
CPLEX	10	40	0,315	0,057	0,203	0,427
CPLEX	20	40	0,684	0,057	0,573	0,796
CPLEX	30	40	1,199	0,057	1,087	1,310
CPLEX	40	40	0,961	0,057	0,850	1,073
CPLEX	50	40	0,633	0,057	0,522	0,745
Partial	10	40	0,264	0,057	0,152	0,376
Partial	20	40	0,655	0,057	0,544	0,767
Partial	30	40	1,983	0,057	1,871	2,094
Partial	40	40	1,705	0,057	1,593	1,817
Partial	50	40	0,623	0,057	0,511	0,734
RBS	10	40	0,747	0,057	0,635	0,859
RBS	20	40	1,042	0,057	0,930	1,154
RBS	30	40	0,538	0,057	0,426	0,650
RBS	40	40	0,723	0,057	0,611	0,835
RBS	50	40	1,000	0,057	0,888	1,112
IGS	10	200	0,280	0,025	0,230	0,330
IGS	20	200	0,595	0,025	0,545	0,645
IGS	30	200	0,440	0,025	0,390	0,490
IGS	40	200	0,633	0,025	0,583	0,683
IGS	50	200	0,633	0,025	0,583	0,683
IGD	10	200	0,217	0,025	0,167	0,267
IGD	20	200	0,499	0,025	0,449	0,549
IGD	30	200	0,383	0,025	0,333	0,433
IGD	40	200	0,629	0,025	0,579	0,679
IGD	50	200	0,698	0,025	0,648	0,748
NSP	10	200	0,186	0,025	0,136	0,236
NSP	20	200	0,450	0,025	0,400	0,500
NSP	30	200	0,257	0,025	0,207	0,307
NSP	40	200	0,457	0,025	0,407	0,507
NSP	50	200	0,457	0,025	0,408	0,507
Vir	10	200	0,191	0,025	0,141	0,241
Vir	20	200	0,523	0,025	0,473	0,573
Vir	30	200	0,304	0,025	0,254	0,354
Vir	40	200	0,567	0,025	0,517	0,617
Vir	50	200	0,673	0,025	0,623	0,723

Tabla B.6 – Tabla de datos estadísticos agrupados por máquinas para el intervalo U(100,200).

B.4. Interacciones en el intervalo U(100,120)

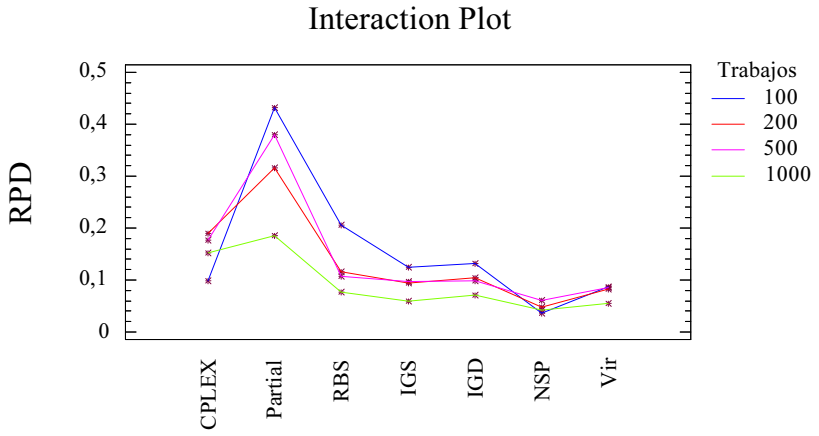

Figura B.7 – Gráfico de interacción de algoritmos frente número de trabajos, para la distribución de p_{ij} U(100,120).

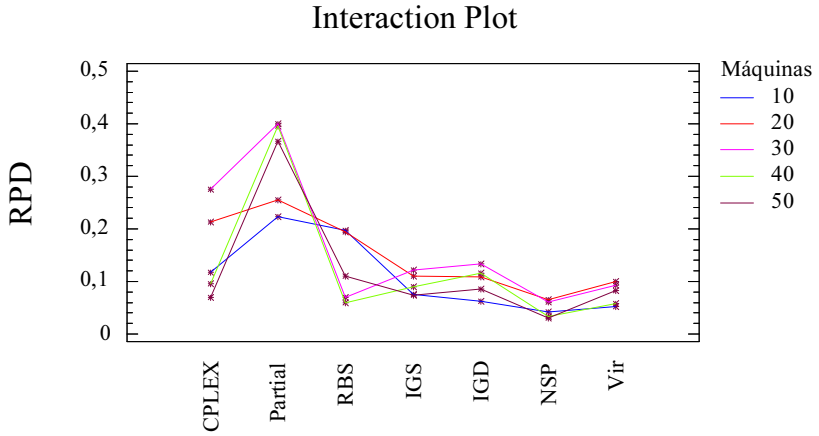

Figura B.8 – Gráfico de interacción de algoritmos frente número de máquinas, para la distribución de p_{ij} U(100,120).

Algoritmo	Trab.	Núm.	Media	Error Stnd.	Lim. Inf.	Lim. Sup.
CPLEX	100	50	0,098	0,016	0,067	0,129
CPLEX	200	50	0,190	0,016	0,158	0,221
CPLEX	500	50	0,177	0,016	0,146	0,209
CPLEX	1000	50	0,152	0,016	0,121	0,183
Partial	100	50	0,431	0,016	0,400	0,462
Partial	200	50	0,315	0,016	0,284	0,347
Partial	500	50	0,380	0,016	0,348	0,411
Partial	1000	50	0,186	0,016	0,154	0,217
RBS	100	50	0,205	0,016	0,174	0,236
RBS	200	50	0,117	0,016	0,085	0,148
RBS	500	50	0,107	0,016	0,076	0,138
RBS	1000	50	0,076	0,016	0,045	0,108
IGS	100	250	0,125	0,007	0,111	0,139
IGS	200	250	0,094	0,007	0,080	0,108
IGS	500	250	0,097	0,007	0,084	0,111
IGS	1000	250	0,059	0,007	0,045	0,073
IGD	100	250	0,132	0,007	0,118	0,146
IGD	200	250	0,104	0,007	0,090	0,118
IGD	500	250	0,099	0,007	0,085	0,113
IGD	1000	250	0,070	0,007	0,056	0,084
NSP	100	250	0,036	0,007	0,022	0,050
NSP	200	250	0,047	0,007	0,033	0,061
NSP	500	250	0,060	0,007	0,046	0,074
NSP	1000	250	0,042	0,007	0,028	0,056
Vir	100	250	0,086	0,007	0,072	0,100
Vir	200	250	0,082	0,007	0,068	0,096
Vir	500	250	0,086	0,007	0,072	0,100
Vir	1000	250	0,054	0,007	0,040	0,068

Núm.= número de instancias evaluadas ($\times 5$ replicas para nuestros algoritmos), Lim. Inf. y Sup. del intervalo del 95 %.

Tabla B.7 – Tabla de datos estadísticos agrupados por trabajos para el intervalo U(100,120).

Algoritmo	Máq.	Núm.	Media	Error Stnd.	Lim. Inf.	Lim. Sup.
CPLEX	10	40	0,118	0,018	0,083	0,153
CPLEX	20	40	0,213	0,018	0,178	0,248
CPLEX	30	40	0,276	0,018	0,241	0,311
CPLEX	40	40	0,095	0,018	0,061	0,130
CPLEX	50	40	0,070	0,018	0,035	0,105
Partial	10	40	0,223	0,018	0,188	0,258
Partial	20	40	0,254	0,018	0,220	0,289
Partial	30	40	0,400	0,018	0,365	0,435
Partial	40	40	0,396	0,018	0,361	0,431
Partial	50	40	0,366	0,018	0,331	0,401
RBS	10	40	0,197	0,018	0,162	0,232
RBS	20	40	0,195	0,018	0,160	0,230
RBS	30	40	0,069	0,018	0,034	0,104
RBS	40	40	0,060	0,018	0,025	0,095
RBS	50	40	0,111	0,018	0,076	0,146
IGS	10	200	0,076	0,008	0,060	0,091
IGS	20	200	0,110	0,008	0,095	0,126
IGS	30	200	0,122	0,008	0,106	0,138
IGS	40	200	0,089	0,008	0,074	0,105
IGS	50	200	0,073	0,008	0,058	0,089
IGD	10	200	0,062	0,008	0,047	0,078
IGD	20	200	0,108	0,008	0,093	0,124
IGD	30	200	0,134	0,008	0,118	0,149
IGD	40	200	0,116	0,008	0,100	0,131
IGD	50	200	0,086	0,008	0,071	0,102
NSP	10	200	0,042	0,008	0,026	0,057
NSP	20	200	0,066	0,008	0,050	0,081
NSP	30	200	0,062	0,008	0,046	0,077
NSP	40	200	0,033	0,008	0,018	0,049
NSP	50	200	0,030	0,008	0,015	0,046
Vir	10	200	0,052	0,008	0,036	0,067
Vir	20	200	0,101	0,008	0,085	0,116
Vir	30	200	0,093	0,008	0,077	0,109
Vir	40	200	0,059	0,008	0,043	0,074
Vir	50	200	0,082	0,008	0,067	0,098

Tabla B.8 – Tabla de datos estadísticos agrupados por máquinas para el intervalo U(100,120).

B.5. Interacciones en el intervalo U(1000,1100)

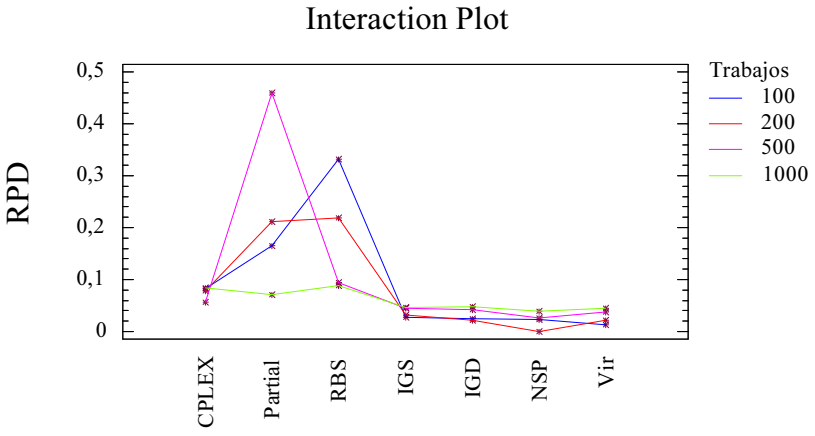

Figura B.9 – Gráfico de interacción de algoritmos frente número de trabajos, para la distribución de p_{ij} U(1000,1100).

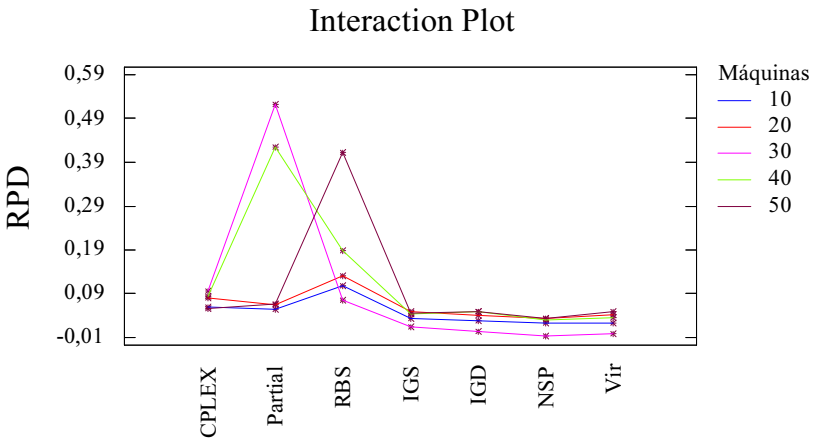

Figura B.10 – Gráfico de interacción de algoritmos frente número de máquinas, para la distribución de p_{ij} U(1000,1100).

Algoritmo	Trab.	Núm.	Media	Error Stnd.	Lim. Inf.	Lim. Sup.
CPLEX	100	50	0,083	0,011	0,062	0,104
CPLEX	200	50	0,078	0,011	0,057	0,099
CPLEX	500	50	0,057	0,011	0,036	0,078
CPLEX	1000	50	0,085	0,011	0,064	0,105
Partial	100	50	0,165	0,011	0,144	0,185
Partial	200	50	0,212	0,011	0,191	0,233
Partial	500	50	0,459	0,011	0,438	0,480
Partial	1000	50	0,071	0,011	0,050	0,092
RBS	100	50	0,332	0,011	0,311	0,353
RBS	200	50	0,218	0,011	0,197	0,239
RBS	500	50	0,094	0,011	0,073	0,115
RBS	1000	50	0,089	0,011	0,068	0,110
IGS	100	250	0,027	0,005	0,018	0,036
IGS	200	250	0,032	0,005	0,022	0,041
IGS	500	250	0,046	0,005	0,036	0,055
IGS	1000	250	0,046	0,005	0,037	0,056
IGD	100	250	0,025	0,005	0,016	0,034
IGD	200	250	0,022	0,005	0,013	0,031
IGD	500	250	0,042	0,005	0,033	0,051
IGD	1000	250	0,048	0,005	0,038	0,057
NSP	100	250	0,024	0,005	0,014	0,033
NSP	200	250	0,001	0,005	-0,008	0,010
NSP	500	250	0,026	0,005	0,017	0,035
NSP	1000	250	0,039	0,005	0,030	0,049
Vir	100	250	0,013	0,005	0,004	0,022
Vir	200	250	0,022	0,005	0,013	0,032
Vir	500	250	0,038	0,005	0,028	0,047
Vir	1000	250	0,045	0,005	0,035	0,054

Núm.= número de instancias evaluadas ($\times 5$ replicas para nuestros algoritmos), Lim. Inf. y Sup. del intervalo del 95 %.

Tabla B.9 – Tabla de datos estadísticos agrupados por trabajos para el intervalo U(1000,1100).

Algoritmo	Máq.	Núm.	Media	Error Stnd.	Lim. Inf.	Lim. Sup.
CPLEX	10	40	0,060	0,012	0,037	0,084
CPLEX	20	40	0,081	0,012	0,058	0,104
CPLEX	30	40	0,096	0,012	0,073	0,120
CPLEX	40	40	0,084	0,012	0,061	0,107
CPLEX	50	40	0,057	0,012	0,033	0,080
Partial	10	40	0,054	0,012	0,031	0,077
Partial	20	40	0,064	0,012	0,041	0,087
Partial	30	40	0,523	0,012	0,500	0,546
Partial	40	40	0,425	0,012	0,401	0,448
Partial	50	40	0,067	0,012	0,044	0,090
RBS	10	40	0,108	0,012	0,085	0,132
RBS	20	40	0,130	0,012	0,107	0,153
RBS	30	40	0,076	0,012	0,053	0,099
RBS	40	40	0,188	0,012	0,165	0,211
RBS	50	40	0,413	0,012	0,390	0,436
IGS	10	200	0,034	0,005	0,024	0,045
IGS	20	200	0,050	0,005	0,039	0,060
IGS	30	200	0,014	0,005	0,003	0,024
IGS	40	200	0,045	0,005	0,034	0,055
IGS	50	200	0,046	0,005	0,036	0,057
IGD	10	200	0,028	0,005	0,018	0,039
IGD	20	200	0,040	0,005	0,029	0,050
IGD	30	200	0,004	0,005	-0,006	0,014
IGD	40	200	0,049	0,005	0,039	0,060
IGD	50	200	0,049	0,005	0,039	0,059
NSP	10	200	0,023	0,005	0,012	0,033
NSP	20	200	0,034	0,005	0,024	0,045
NSP	30	200	-0,007	0,005	-0,017	0,003
NSP	40	200	0,029	0,005	0,019	0,040
NSP	50	200	0,033	0,005	0,022	0,043
Vir	10	200	0,024	0,005	0,013	0,034
Vir	20	200	0,042	0,005	0,031	0,052
Vir	30	200	-0,002	0,005	-0,012	0,008
Vir	40	200	0,035	0,005	0,024	0,045
Vir	50	200	0,049	0,005	0,039	0,060

Tabla B.10 – Tabla de datos estadísticos agrupados por máquinas para el intervalo U(1000,1100).

B.6. Interacciones en el intervalo de trabajos correlacionados

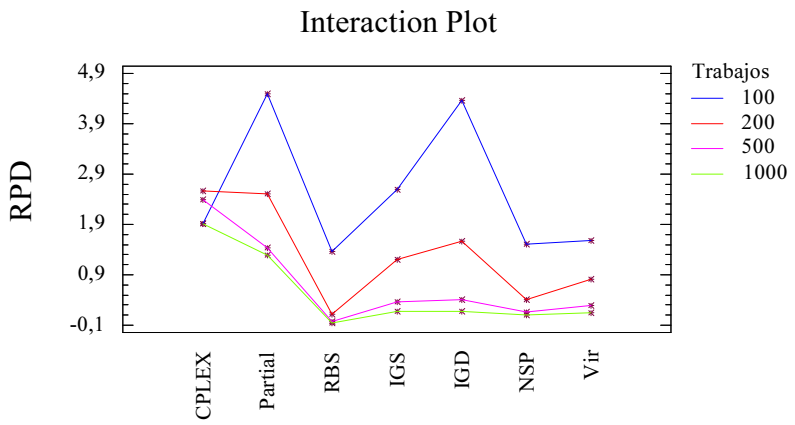

Figura B.11 – Gráfico de interacción de algoritmos frente número de trabajos, para tiempos de p_{ij} de trab. correlacionados.

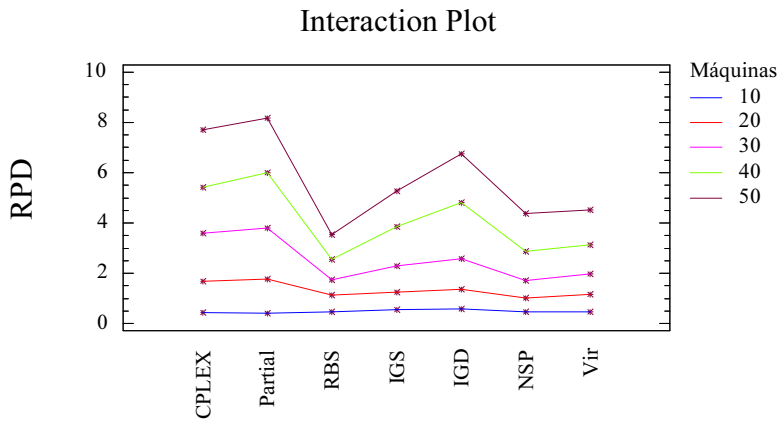

Figura B.12 – Gráfico de interacción de algoritmos frente número de máquinas, para tiempos de p_{ij} de trab. correlacionados.

Algoritmo	Trab.	Núm.	Media	Error Stnd.	Lim. Inf.	Lim. Sup.
CPLEX	100	50	1,921	0,128	1,670	2,172
CPLEX	200	50	2,568	0,128	2,316	8,000
CPLEX	500	50	2,387	0,128	2,135	2,638
CPLEX	1000	50	1,914	0,128	1,663	8,000
Partial	100	50	4,497	0,128	4,246	4,748
Partial	200	50	2,505	0,128	2,254	2,756
Partial	500	50	1,437	0,128	1,186	1,688
Partial	1000	50	1,291	0,128	1,040	1,542
RBS	100	50	1,357	0,128	1,105	1,608
RBS	200	50	0,119	0,128	-0,131	0,371
RBS	500	50	-0,030	0,128	-0,281	0,221
RBS	1000	50	-0,053	0,128	-0,303	0,199
IGS	100	250	2,594	0,057	2,482	2,707
IGS	200	250	1,201	0,057	1,089	1,313
IGS	500	250	0,370	0,057	0,258	0,482
IGS	1000	250	0,171	0,057	0,058	0,283
IGD	100	250	4,366	0,057	4,254	4,478
IGD	200	250	1,566	0,057	1,454	1,679
IGD	500	250	0,409	0,057	0,296	0,521
IGD	1000	250	0,172	0,057	0,060	0,284
NSP	100	250	1,507	0,057	1,395	3,000
NSP	200	250	0,401	0,057	0,288	0,513
NSP	500	250	0,162	0,057	0,050	0,274
NSP	1000	250	0,109	0,057	-0,003	0,221
Vir	100	250	1,576	0,057	1,464	1,688
Vir	200	250	0,816	0,057	0,704	0,929
Vir	500	250	0,290	0,057	0,178	0,403
Vir	1000	250	0,148	0,057	0,035	0,260

Núm.= número de instancias evaluadas ($\times 5$ replicas para nuestros algoritmos), Lim. Inf. y Sup. del intervalo del 95 %.

Tabla B.11 – Tabla de datos estadísticos agrupados por trabajos para trabajos correlacionados

Algoritmo	Máq.	Núm.	Media	Error Stnd.	Lim. Inf.	Lim. Sup.
CPLEX	10	40	0,298	0,143	0,017	0,578
CPLEX	20	40	1,053	0,143	0,772	1,334
CPLEX	30	40	2,227	0,143	1,946	2,508
CPLEX	40	40	3,021	0,143	2,741	3,302
CPLEX	50	40	4,387	0,143	4,107	7,000
Partial	10	40	0,260	0,143	-0,020	0,541
Partial	20	40	1,153	0,143	0,872	1,434
Partial	30	40	2,410	0,143	2,129	2,691
Partial	40	40	3,556	0,143	3,275	3,836
Partial	50	40	4,784	0,143	4,503	5,065
RBS	10	40	0,338	0,143	0,057	0,618
RBS	20	40	0,502	0,143	0,221	0,783
RBS	30	40	0,402	0,143	0,121	0,682
RBS	40	40	0,204	0,143	-0,077	0,485
RBS	50	40	0,297	0,143	0,016	0,578
IGS	10	200	0,409	0,064	0,283	0,535
IGS	20	200	0,642	0,064	0,516	0,768
IGS	30	200	0,935	0,064	0,810	1,061
IGS	40	200	1,460	0,064	1,334	1,586
IGS	50	200	1,974	0,064	1,848	2,100
IGD	10	200	0,434	0,064	0,308	0,559
IGD	20	200	0,741	0,064	0,615	0,866
IGD	30	200	1,217	0,064	1,091	1,342
IGD	40	200	2,387	0,064	2,262	2,513
IGD	50	200	3,363	0,064	3,237	3,488
NSP	10	200	0,313	0,064	0,188	0,439
NSP	20	200	0,390	0,064	0,264	0,516
NSP	30	200	0,382	0,064	0,257	0,508
NSP	40	200	0,527	0,064	0,401	0,652
NSP	50	200	1,111	0,064	0,985	1,236
Vir	10	200	0,327	0,064	0,202	0,453
Vir	20	200	0,530	0,064	0,404	0,656
Vir	30	200	0,640	0,064	0,514	0,766
Vir	40	200	0,790	0,064	0,664	0,915
Vir	50	200	1,251	0,064	1,126	1,377

Tabla B.12 – Tabla de datos estadísticos agrupados por máquinas para trabajos correlacionados

B.7. Interacciones en el intervalo de máquinas correlacionadas

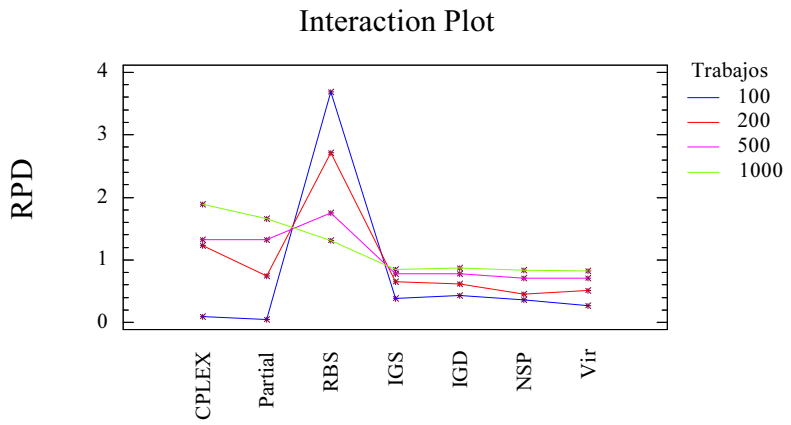

Figura B.13 – Gráfico de interacción de algoritmos frente número de trabajos, para tiempos de p_{ij} de máq. correlacionadas.

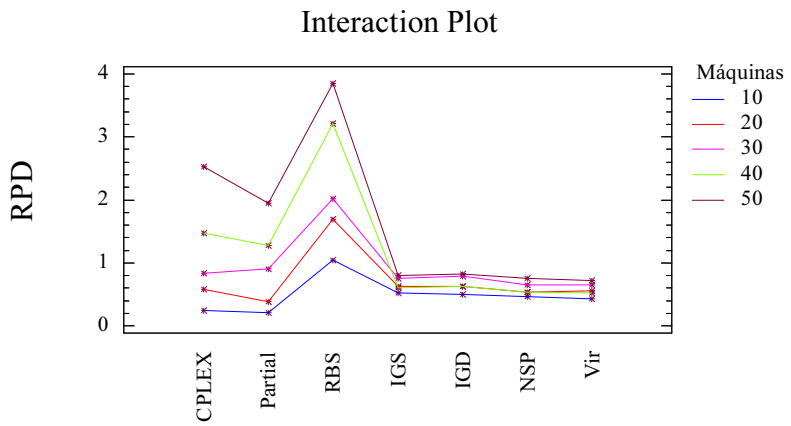

Figura B.14 – Gráfico de interacción de algoritmos frente número de máquinas, para tiempos de p_{ij} de máq. correlacionadas.

Algoritmo	Trab.	Núm.	Media	Error Stnd.	Lim. Inf.	Lim. Sup.
CPLEX	100	50	0,097	0,088	-0,074	0,268
CPLEX	200	50	1,224	0,088	1,053	1,396
CPLEX	500	50	1,325	0,088	1,153	1,497
CPLEX	1000	50	1,885	0,088	1,713	2,057
Partial	100	50	0,051	0,088	-0,120	0,223
Partial	200	50	0,741	0,088	0,570	0,913
Partial	500	50	1,321	0,088	1,149	1,492
Partial	1000	50	1,661	0,088	1,489	1,833
RBS	100	50	3,682	0,088	3,510	3,853
RBS	200	50	2,710	0,088	2,538	2,882
RBS	500	50	1,747	0,088	1,575	1,918
RBS	1000	50	1,315	0,088	1,143	1,486
IGS	100	250	0,384	0,039	0,308	0,461
IGS	200	250	0,647	0,039	0,570	0,724
IGS	500	250	0,773	0,039	0,696	0,850
IGS	1000	250	0,844	0,039	0,767	0,921
IGD	100	250	0,431	0,039	0,354	0,507
IGD	200	250	0,615	0,039	0,538	0,692
IGD	500	250	0,779	0,039	0,702	0,856
IGD	1000	250	0,870	0,039	0,793	0,947
NSP	100	250	0,354	0,039	0,277	0,431
NSP	200	250	0,456	0,039	0,380	0,533
NSP	500	250	0,711	0,039	0,634	0,788
NSP	1000	250	0,835	0,039	0,759	0,912
Vir	100	250	0,267	0,039	0,190	0,344
Vir	200	250	0,508	0,039	0,432	0,585
Vir	500	250	0,709	0,039	0,632	0,786
Vir	1000	250	0,829	0,039	0,752	0,906

Núm.= número de instancias evaluadas ($\times 5$ replicas para nuestros algoritmos), Lim. Inf. y Sup. del intervalo del 95 %.

Tabla B.13 – Tabla de datos estadísticos agrupados por trabajos para máquinas correlacionadas

Algoritmo	Máq.	Núm.	Media	Error Stnd.	Lim. Inf.	Lim. Sup.
CPLEX	10	40	0,249	0,098	0,057	0,441
CPLEX	20	40	0,582	0,098	0,391	0,774
CPLEX	30	40	0,833	0,098	0,642	1,025
CPLEX	40	40	1,474	0,098	1,282	1,666
CPLEX	50	40	2,525	0,098	2,333	2,717
Partial	10	40	0,207	0,098	0,015	0,399
Partial	20	40	0,387	0,098	0,195	0,579
Partial	30	40	0,900	0,098	0,708	1,092
Partial	40	40	1,274	0,098	1,082	1,466
Partial	50	40	1,949	0,098	1,758	2,141
RBS	10	40	1,048	0,098	0,856	1,240
RBS	20	40	1,690	0,098	1,498	1,882
RBS	30	40	2,015	0,098	1,823	2,207
RBS	40	40	3,214	0,098	3,023	3,406
RBS	50	40	3,848	0,098	3,656	4,040
IGS	10	200	0,521	0,044	0,435	0,607
IGS	20	200	0,628	0,044	0,543	0,714
IGS	30	200	0,755	0,044	0,669	0,841
IGS	40	200	0,600	0,044	0,515	0,686
IGS	50	200	0,805	0,044	0,719	0,891
IGD	10	200	0,501	0,044	0,415	0,587
IGD	20	200	0,625	0,044	0,540	0,711
IGD	30	200	0,787	0,044	0,701	0,873
IGD	40	200	0,631	0,044	0,546	0,717
IGD	50	200	0,824	0,044	0,738	0,910
NSP	10	200	0,460	0,044	0,374	0,546
NSP	20	200	0,538	0,044	0,452	0,624
NSP	30	200	0,654	0,044	0,568	0,740
NSP	40	200	0,539	0,044	0,453	0,625
NSP	50	200	0,755	0,044	0,669	0,840
Vir	10	200	0,433	0,044	0,347	0,519
Vir	20	200	0,558	0,044	0,473	0,644
Vir	30	200	0,653	0,044	0,567	0,739
Vir	40	200	0,524	0,044	0,438	0,610
Vir	50	200	0,723	0,044	0,637	0,809

Tabla B.14 – Tabla de datos estadísticos agrupados por máquinas para máquinas correlacionadas

ANEXO C

RESULTADO DE CALIBRACIÓN DE LOS ALGORITMOS

En este anexo veremos como se han calibrado los distintos algoritmos. Se realizaron más pruebas de las mostradas, pero se han mostrado las más importantes. Para todas las pruebas se han efectuado 5 réplicas.
Se ha seguido la notación $k + l$, donde k representa el número de trabajos que se modifican de la máquina con el $C_{\text{máx}}$, usándose la palabra *"all"* para indicar que se modifican todos los trabajos de esa máquina y l representa el número de trabajos que se modifican del resto de máquinas seleccionadas al azar.

C.1. Calibración de IGS

En la calibración de este algoritmo se ha tratado de destruir desde la totalidad de los trabajos asignados a la máquina con el $C_{\text{máx}}$, hasta solo 5. Así mismo, se hizo variar el número de trabajos a destruir del resto de máquinas seleccionadas al azar, desde 50 trabajos hasta 5. Con ello se trató de ver el

comportamiento del algoritmo ante la variación de destrucciones y delimitar una buena calibración para su funcionamiento.

trabajos	all+10	all+20	all+30	all+50
100	1,15	0,94	1,04	1,22
200	3,22	2,99	2,99	3,89
500	3,68	3,57	3,81	4,37
1000	3,10	3,06	3,23	3,45
media	2,79	2,64	2,77	3,24

Tabla C.1 – Calibración del algoritmo IGS: primera parte.

trabajos	5+5	10+10	20+20	30+30	5+20	10+20	10+30
100	1,12	0,97	0,77	0,71	0,70	0,77	0,71
200	3,16	2,56	2,44	2,77	2,77	2,55	2,77
500	3,62	3,44	3,47	3,71	3,53	3,46	3,71
1000	3,24	3,00	3,05	3,18	3,11	3,08	3,18
media	2,78	2,49	2,43	2,59	2,53	2,46	2,59

Tabla C.2 – Calibración del algoritmo IGS: segunda parte.

Como resultado de estas pruebas se determinó que una destrucción de 20 trabajos de la máquina del $C_{\text{máx}}$ y 20 trabajos del resto de las máquinas, escogidas al azar, proporcionaba los mejores resultados.

C.2. Calibración de IGD

En la calibración de este algoritmo solo tiene significado el componente k de la notación de la calibración, ya que siempre se destruye un trabajo de la

máquina que, después de la destrucción de cada trabajo, detenta el $C_{\text{máx}}$, tal y como se explico en la sección 3.2.2.

trabajos	10+0	20+0	30+0	40+0
100	2,73	2,12	2,16	2,39
200	4,60	3,77	3,55	3,95
500	3,28	3,19	3,38	3,57
1000	2,99	3,00	3,11	3,08
media	3,40	3,02	3,05	3,24

Tabla C.3 – Calibración del algoritmo IGD.

Con los resultados de esta tabla, se determinó que una buena elección seria la destrucción de un total de 20 trabajos.

C.3. Calibración de NSP

En la presente calibración se fue aumentando el número de trabajos seleccionados, tanto de la máquina con el $C_{\text{máx}}$, como del resto de máquinas escogidas al azar, para, finalmente hacer combinaciones de ambas.

trabajos	1+1	2+2	3+3	5+5	10+10	20+20	2+5	5+2	0+2	2+0
100	1,44	1,17	1,18	0,82	1,32	1,54	1,24	0,92	1,89	9,15
200	2,84	2,68	3,21	3,91	4,29	5,18	4,05	3,48	3,57	9,46
500	3,84	4,25	4,40	4,83	5,13	5,42	4,53	4,34	4,32	4,03
1000	3,24	3,44	3,39	3,71	3,70	3,95	3,54	3,47	3,41	3,14
media	2,84	2,89	3,05	3,32	3,61	4,02	3,34	3,05	3,30	6,45

Tabla C.4 – Calibración del algoritmo NSP.

De los resultados de la tabla se desprende que la opción de seleccionar 1 solo trabajo de la máquina que detenta el $C_{\text{máx}}$ y 1 trabajo del resto de las máquinas, es la que mejores resultados proporciona. Nótese que en esta mejor opción, se seleccionan en total 2 trabajos (1+1), pero que si se seleccionan 2 trabajos solo de la máquina del $C_{\text{máx}}$ (2+0), o solo de máquinas al azar (0+2), el resultado es mucho peor.

C.4. Calibración de Virtual

En esta calibración se procedió a ir aumentando a la par el número de trabajos seleccionados para modificar, tanto de la máquina con el $C_{\text{máx}}$ como del resto de máquinas. También se realizaron combinaciones de seleccionar un número distinto de trabajos a modificar.

trabajos	1+1	0+2	2+2	3+3	5+5	7+7	0+10	5+10	10+5	10+10
100	3,14	2,61	2,35	1,68	1,38	1,27	1,47	1,22	1,59	1,35
200	3,31	4,96	3,18	2,86	2,54	2,70	3,23	2,78	2,87	2,77
500	2,93	3,67	2,73	2,71	2,71	2,74	3,12	2,86	2,70	2,82
1000	2,80	3,01	2,60	2,63	2,53	2,51	2,72	2,62	2,54	2,59
media	3,05	3,56	2,72	2,47	2,29	2,30	2,64	2,37	2,42	2,38

Tabla C.5 – Calibración del algoritmo Vir: Primera parte.

trabajos	5+20	20+5	0+25	10+20	30+30	5+50	50+5	0+55	all+5	all+20
100	1,21	1,74	1,19	1,15	1,26	1,47	1,15	1,68	1,48	1,34
200	3,08	2,98	3,47	3,07	3,51	4,31	3,07	4,32	2,99	3,16
500	3,06	2,72	3,37	3,00	3,38	4,00	3,00	3,99	2,89	3,20
1000	2,78	2,54	2,81	2,79	2,89	3,05	2,79	3,15	2,56	2,78
media	2,53	2,50	2,71	2,50	2,76	3,21	2,50	3,28	2,48	2,62

Tabla C.6 – Calibración del algoritmo Vir: Segunda parte.

Tras las pruebas realizadas, se decidió que un número de trabajos a modificar de 5 dentro de la máquina que detenta el $C_{\text{máx}}$ y 5 del resto de las máquinas, escogidas al azar, era la mejor opción.

ANEXO D

TABLAS Y ANOVAS DE LOS ALGORITMOS DE REDUCCIÓN DE TAMAÑO

En el presente anexo mostramos las tablas de desviaciones porcentuales medias respecto a la referencia a la referencia obtenida por CPLEX 11.0 y hasta 2 horas con cada instancia, para los algoritmos de reducción de tamaño. Se desglosan por intervalos y, dentro de ellos, por cada uno de los tiempos de parada sujetos a estudio, a saber, 15, 30, 60, 120, 240 y 300 segundos. Posteriormente se presentaran los ANOVAS correspondientes a estas tablas. Para el caso de los ANOVAS del intervalo máquinas correlacionadas, dada la enorme diferencia del algoritmo 3J con respecto al resto, se ha suprimido este, así como el algoritmo 3J2M, para una mayor claridad de las diferencias.

D.1. Tablas de algoritmos de reducción de tamaño

n	m	CPLEX	DIG	3J	3J2M	3JD	2JDi	2JDi(90)	M-CPLEX	M-DIG	M-2JDi(90)
100	10	0,00	0,18	0,27	0,20	0,10	0,00	0,10	0,00	0,18	0,09
	20	0,00	0,62	0,34	0,00	0,62	0,00	0,34	0,00	0,34	0,00
	30	0,00	0,00	0,63	0,63	0,00	0,00	0,00	0,00	0,00	0,00
	40	0,00	0,00	2,74	1,74	0,00	0,00	0,00	0,00	0,00	0,00
	50	0,00	0,00	1,25	1,25	0,00	0,00	0,00	0,00	0,00	0,00
Media 100		0,00	0,16	1,05	0,76	0,14	0,00	0,09	0,00	0,11	0,02
200	10	0,05	0,20	0,29	0,35	0,15	0,15	0,15	0,10	0,20	0,15
	20	0,69	0,69	1,06	0,86	0,87	0,53	0,69	0,68	0,69	1,23
	30	0,69	1,44	0,36	1,07	0,00	0,34	0,68	0,73	1,07	1,06
	40	1,14	2,45	1,14	1,14	1,18	0,59	1,18	0,59	1,73	0,59
	50	0,83	2,44	1,67	0,83	1,60	0,83	0,77	0,00	2,44	1,60
Media 200		0,68	1,44	0,90	0,85	0,76	0,49	0,69	0,42	1,23	0,93
500	10	0,14	0,39	0,14	0,25	0,13	0,12	0,08	0,23	0,35	0,08
	20	0,81	1,11	0,52	0,59	0,59	0,66	0,37	0,67	1,04	0,45
	30	1,54	1,69	0,92	1,22	0,92	0,62	0,77	1,56	1,38	0,62
	40	1,30	1,30	0,78	1,30	0,26	1,05	0,79	1,05	1,05	0,28
	50	2,64	2,25	1,50	1,88	1,50	1,13	1,50	2,25	1,88	1,88
Media 500		1,29	1,35	0,77	1,05	0,68	0,72	0,70	1,15	1,14	0,66
1000	10	0,08	0,27	0,07	0,09	0,03	0,03	0,03	0,10	0,26	0,03
	20	0,34	0,83	0,19	0,30	0,08	0,07	0,15	0,71	0,64	0,11
	30	1,42	1,57	0,48	0,56	0,39	0,40	0,39	1,02	1,42	0,40
	40	2,67	1,87	0,54	1,33	0,54	0,41	0,41	2,14	1,74	0,54
	50	1,97	2,15	0,98	1,18	0,80	0,80	0,80	1,38	1,76	0,98
Media 1000		1,30	1,34	0,45	0,69	0,37	0,34	0,36	1,07	1,16	0,41
Media general		0,82	1,07	0,79	0,84	0,49	0,39	0,46	0,66	0,91	0,50

Tabla D.1 – Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo U(1,100) de los algoritmos de reducción de tamaño para 15 segundos como tiempo de parada.

n	m	CPLEX	DIG	3J	3J2M	3JD	2JDi	2JDi(90)	M-CPLEX	M-DIG	M-2JDi(90)
100	10	0,00	0,10	0,19	0,10	0,10	0,00	0,10	0,00	0,10	0,09
	20	0,00	0,34	0,34	0,00	0,62	0,00	0,34	0,00	0,34	0,00
	30	0,00	0,00	0,63	0,63	0,00	0,00	0,00	0,00	0,00	0,00
	40	0,00	0,00	2,74	1,74	0,00	0,00	0,00	0,00	0,00	0,00
	50	0,00	0,00	1,25	1,25	0,00	0,00	0,00	0,00	0,00	0,00
Media 100		0,00	0,09	1,03	0,74	0,14	0,00	0,09	0,00	0,09	0,02
200	10	0,05	0,20	0,25	0,20	0,10	0,10	0,10	0,10	0,20	0,15
	20	0,34	0,51	0,69	0,51	0,70	0,53	0,53	0,68	0,51	1,06
	30	0,36	1,07	0,36	0,34	0,00	0,34	0,68	0,37	1,07	0,00
	40	0,00	2,45	1,14	1,14	0,59	0,59	0,59	0,00	1,73	0,00
	50	0,00	2,44	1,67	0,83	1,60	0,00	0,00	0,00	2,44	0,00
Media 200		0,15	1,33	0,82	0,61	0,60	0,31	0,38	0,23	1,19	0,24
500	10	0,14	0,33	0,12	0,10	0,11	0,10	0,06	0,14	0,29	0,04
	20	0,66	0,96	0,37	0,45	0,45	0,59	0,37	0,59	0,81	0,45
	30	1,23	1,38	0,77	0,92	0,62	0,47	0,63	0,93	1,38	0,62
	40	0,79	1,05	0,53	1,05	0,01	0,03	0,52	0,53	1,05	-0,51
	50	1,88	1,88	1,50	1,52	1,14	1,13	1,13	1,50	1,88	0,77
Media 500		0,94	1,12	0,66	0,81	0,46	0,46	0,54	0,74	1,08	0,27
1000	10	0,05	0,26	0,04	0,09	0,02	0,03	0,03	0,08	0,25	0,01
	20	0,15	0,64	0,04	0,11	0,04	0,04	0,15	0,30	0,49	0,07
	30	1,34	1,49	0,48	0,40	0,39	0,40	0,39	0,63	1,26	0,32
	40	2,41	1,74	0,54	1,33	0,41	0,41	0,41	1,35	1,74	0,27
	50	1,78	1,76	0,80	1,18	0,60	0,39	0,41	0,99	1,76	0,40
Media 1000		1,15	1,18	0,38	0,62	0,29	0,25	0,28	0,67	1,10	0,22
Media general		0,56	0,93	0,72	0,70	0,37	0,26	0,32	0,41	0,86	0,19

Tabla D.2 – Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo U(1,100) de los algoritmos de reducción de tamaño para 30 segundos como tiempo de parada.

n	m	CPLEX	DIG	3J	3J2M	3JD	2JDi	2JDi(90)	M-CPLEX	M-DIG	M-2JDi(90)
100	10	0,00	0,10	0,19	0,10	0,10	0,00	0,10	0,00	0,10	0,09
	20	0,00	0,00	0,34	0,00	0,62	0,00	0,00	0,00	0,00	0,00
	30	0,00	0,00	0,63	0,63	0,00	0,00	0,00	0,00	0,00	0,00
	40	0,00	0,00	2,74	1,74	0,00	0,00	0,00	0,00	0,00	0,00
	50	0,00	0,00	1,25	1,25	0,00	0,00	0,00	0,00	0,00	0,00
Media 100		0,00	0,02	1,03	0,74	0,14	0,00	0,02	0,00	0,02	0,02
200	10	0,05	0,20	0,15	0,15	0,05	0,10	0,05	0,10	0,20	0,05
	20	0,17	0,51	0,51	0,51	0,70	0,53	0,35	0,51	0,51	0,51
	30	0,00	1,07	0,36	0,34	0,00	0,34	0,68	0,37	0,72	0,00
	40	0,00	2,45	1,14	1,14	0,59	0,59	0,59	0,00	1,73	0,00
	50	0,00	1,60	1,67	0,83	1,60	0,00	0,00	0,00	0,77	0,00
Media 200		0,04	1,17	0,76	0,60	0,59	0,31	0,33	0,20	0,78	0,11
500	10	0,08	0,29	0,08	0,10	0,04	0,10	0,02	0,10	0,25	0,02
	20	0,44	0,96	0,37	0,22	0,45	0,51	0,30	0,37	0,81	0,30
	30	0,92	1,38	0,62	0,92	0,62	0,32	0,48	0,78	1,23	0,31
	40	0,28	0,79	-0,25	0,52	0,01	0,03	0,27	0,53	0,79	-0,51
	50	1,88	1,88	0,77	1,14	0,77	0,40	0,40	1,14	1,88	0,38
Media 500		0,72	1,06	0,32	0,58	0,38	0,27	0,29	0,59	0,99	0,10
1000	10	0,03	0,24	0,04	0,07	0,01	0,03	0,02	0,06	0,23	0,01
	20	0,04	0,49	0,00	0,08	0,04	0,04	0,04	0,15	0,42	-0,07
	30	0,64	1,25	0,40	0,32	0,32	0,40	0,39	0,48	1,10	0,16
	40	0,94	1,74	0,54	1,33	0,41	0,41	0,41	0,54	1,74	0,27
	50	1,19	1,76	0,40	0,79	0,21	-0,19	0,20	0,80	1,57	0,01
Media 1000		0,57	1,10	0,28	0,52	0,20	0,14	0,21	0,40	1,01	0,07
Media general		0,33	0,83	0,60	0,61	0,33	0,18	0,21	0,30	0,70	0,08

Tabla D.3 – Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo U(1,100) de los algoritmos de reducción de tamaño para 60 segundos como tiempo de parada.

n	m	CPLEX	DIG	3J	3J2M	3JD	2JDi	2JDi(90)	M-CPLEX	M-DIG	M-2JDi(90)
100	10	0,00	0,10	0,19	0,10	0,10	0,00	0,00	0,00	0,10	0,00
	20	0,00	0,00	0,34	0,00	0,62	0,00	0,00	0,00	0,00	0,00
	30	0,00	0,00	0,63	0,63	0,00	0,00	0,00	0,00	0,00	0,00
	40	0,00	0,00	2,74	1,74	0,00	0,00	0,00	0,00	0,00	0,00
	50	0,00	0,00	1,25	1,25	0,00	0,00	0,00	0,00	0,00	0,00
Media 100		0,00	0,02	1,03	0,74	0,14	0,00	0,00	0,00	0,02	0,00
200	10	0,05	0,15	0,15	0,15	0,05	0,10	0,00	0,05	0,15	0,00
	20	0,17	0,51	0,51	0,51	0,70	0,53	0,17	0,34	0,51	0,17
	30	0,00	0,72	0,36	0,34	0,00	0,34	0,00	0,00	0,37	0,00
	40	0,00	2,45	1,14	1,14	0,59	0,59	0,00	0,00	1,73	0,00
	50	0,00	1,60	1,67	0,83	1,60	0,00	0,00	0,00	0,77	0,00
Media 200		0,04	1,08	0,76	0,60	0,59	0,31	0,03	0,08	0,71	0,03
500	10	0,06	0,27	0,06	0,10	0,00	0,08	0,00	0,08	0,23	-0,02
	20	0,37	0,52	0,37	0,22	0,45	0,44	0,30	0,22	0,52	0,22
	30	0,77	1,38	0,62	0,77	0,47	0,32	0,48	0,47	1,23	0,16
	40	-0,76	0,79	-0,76	0,01	0,01	0,03	-0,51	0,53	0,79	-0,76
	50	0,77	1,51	0,40	0,03	0,77	0,03	0,03	1,14	1,51	-0,36
Media 500		0,24	0,89	0,14	0,23	0,34	0,18	0,06	0,49	0,86	-0,15
1000	10	0,01	0,21	0,03	0,05	0,00	0,02	0,02	0,05	0,19	0,01
	20	-0,04	0,37	0,00	0,04	-0,04	0,04	0,00	0,08	0,34	-0,07
	30	0,40	1,10	0,17	0,32	0,32	0,32	0,24	0,40	1,10	0,16
	40	0,67	1,74	0,27	0,93	0,41	0,41	0,41	0,54	1,74	0,27
	50	1,00	1,57	-0,19	0,59	0,21	-0,19	0,01	0,80	1,17	-0,19
Media 1000		0,41	1,00	0,06	0,39	0,18	0,12	0,13	0,37	0,91	0,04
Media general		0,17	0,75	0,50	0,49	0,31	0,15	0,06	0,23	0,62	-0,02

Tabla D.4 – Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo U(1,100) de los algoritmos de reducción de tamaño para 120 segundos como tiempo de parada.

n	m	CPLEX	DIG	3J	3J2M	3JD	2JDi	2JDi(90)	M-CPLEX	M-DIG	M-2JDi(90)
100	10	0,00	0,10	0,19	0,10	0,10	0,00	0,00	0,00	0,10	0,00
	20	0,00	0,00	0,34	0,00	0,62	0,00	0,00	0,00	0,00	0,00
	30	0,00	0,00	0,63	0,63	0,00	0,00	0,00	0,00	0,00	0,00
	40	0,00	0,00	2,74	1,74	0,00	0,00	0,00	0,00	0,00	0,00
	50	0,00	0,00	1,25	1,25	0,00	0,00	0,00	0,00	0,00	0,00
Media 100		0,00	0,02	1,03	0,74	0,14	0,00	0,00	0,00	0,02	0,00
200	10	0,05	0,15	0,15	0,15	0,05	0,00	0,00	0,05	0,15	0,00
	20	0,17	0,51	0,51	0,51	0,51	0,53	0,00	0,34	0,51	0,00
	30	0,00	0,72	0,36	0,34	0,00	0,34	0,00	0,00	0,37	0,00
	40	0,00	2,45	1,14	1,14	0,59	0,00	0,00	0,00	1,73	0,00
	50	0,00	0,77	1,67	0,83	1,60	0,00	0,00	0,00	0,77	0,00
Media 200		0,04	0,92	0,76	0,60	0,55	0,17	0,00	0,08	0,71	0,00
500	10	0,02	0,21	0,04	0,08	0,00	0,02	-0,02	0,04	0,16	-0,02
	20	0,30	0,52	0,30	0,22	0,37	0,30	0,30	0,22	0,52	0,22
	30	0,63	1,08	0,32	0,32	0,32	0,32	0,17	0,47	0,92	0,16
	40	-0,76	0,79	-0,76	-0,51	-0,25	0,03	-0,76	0,28	0,79	-0,76
	50	0,40	1,13	-0,36	0,03	0,77	0,03	-0,36	0,75	1,13	-0,36
Media 500		0,12	0,74	-0,09	0,03	0,24	0,14	-0,13	0,35	0,70	-0,15
1000	10	0,00	0,21	0,02	0,04	0,00	0,01	0,01	0,04	0,19	0,01
	20	-0,04	0,34	0,00	0,00	-0,04	-0,04	0,00	0,04	0,34	-0,07
	30	0,40	0,86	0,01	0,08	0,32	0,24	0,24	0,24	0,86	0,16
	40	0,14	1,60	0,14	0,53	0,41	0,41	0,27	0,27	1,60	0,27
	50	0,40	1,17	-0,19	0,39	-0,18	-0,19	-0,19	0,40	1,17	-0,19
Media 1000		0,18	0,84	0,00	0,21	0,10	0,09	0,07	0,20	0,83	0,04
Media general		0,08	0,63	0,42	0,39	0,26	0,10	-0,02	0,16	0,57	-0,03

Tabla D.5 – Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo U(1,100) de los algoritmos de reducción de tamaño para 240 segundos como tiempo de parada.

n	m	CPLEX	DIG	3J	3J2M	3JD	2JDi	2JDi(90)	M-CPLEX	M-DIG	M-2JDi(90)
100	10	0,00	0,10	0,19	0,10	0,10	0,00	0,00	0,00	0,10	0,00
	20	0,00	0,00	0,34	0,00	0,62	0,00	0,00	0,00	0,00	0,00
	30	0,00	0,00	0,63	0,63	0,00	0,00	0,00	0,00	0,00	0,00
	40	0,00	0,00	2,74	1,74	0,00	0,00	0,00	0,00	0,00	0,00
	50	0,00	0,00	1,25	1,25	0,00	0,00	0,00	0,00	0,00	0,00
Media 100		0,00	0,02	1,03	0,74	0,14	0,00	0,00	0,00	0,02	0,00
200	10	0,05	0,15	0,15	0,10	0,05	0,00	0,00	0,05	0,15	0,00
	20	0,17	0,51	0,51	0,51	0,51	0,53	0,00	0,17	0,51	0,00
	30	0,00	0,72	0,36	0,34	0,00	0,00	0,00	0,00	0,37	0,00
	40	0,00	2,45	1,14	1,14	0,59	0,00	0,00	0,00	1,73	0,00
	50	0,00	0,77	1,67	0,83	1,60	0,00	0,00	0,00	0,77	0,00
Media 200		0,04	0,92	0,76	0,59	0,55	0,11	0,00	0,04	0,71	0,00
500	10	-0,02	0,21	0,04	0,08	0,00	0,02	-0,02	0,02	0,16	-0,02
	20	0,30	0,52	0,22	0,22	0,30	0,23	0,30	0,15	0,52	0,22
	30	0,47	1,08	0,32	0,32	0,32	0,32	0,17	0,31	0,92	0,16
	40	-0,76	0,79	-0,76	-0,51	-0,50	0,03	-0,76	0,03	0,79	-0,76
	50	0,40	1,13	-0,36	0,03	0,77	0,03	-0,36	0,38	1,13	-0,36
Media 500		0,08	0,74	-0,11	0,03	0,18	0,13	-0,13	0,18	0,70	-0,15
1000	10	0,00	0,19	0,02	0,04	0,00	0,01	0,01	0,02	0,18	0,01
	20	-0,04	0,34	-0,04	0,00	-0,04	-0,04	-0,04	-0,04	0,34	-0,07
	30	0,32	0,86	0,01	0,08	0,32	0,24	0,24	0,08	0,86	0,16
	40	0,14	1,60	0,00	0,53	0,41	0,41	0,00	0,14	1,46	0,27
	50	0,20	1,17	-0,19	0,19	-0,18	-0,19	-0,19	0,20	1,17	-0,19
Media 1000		0,12	0,83	-0,04	0,17	0,10	0,09	0,00	0,08	0,80	0,04
Media general		0,06	0,63	0,41	0,38	0,24	0,08	-0,03	0,08	0,56	-0,03

Tabla D.6 – Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo U(1,100) de los algoritmos de reducción de tamaño para 300 segundos como tiempo de parada.

n	m	CPLEX	DIG	3J	3J2M	3JD	2JDi	2JDi(90)	M-CPLEX	M-DIG	M-2JDi(90)
100	10	0,06	0,29	0,22	0,11	0,11	0,28	0,17	0,17	0,17	0,11
	20	0,26	0,51	0,26	0,39	0,39	0,25	0,27	0,13	0,25	0,13
	30	-0,20	0,43	0,43	0,43	-0,21	0,00	-0,20	-0,43	0,22	-0,43
	40	0,00	0,88	1,47	0,88	0,00	0,00	0,00	0,00	0,59	0,29
	50	0,00	0,00	3,22	0,00	0,00	0,00	0,00	0,00	0,00	0,00
Media 100		0,02	0,42	1,12	0,36	0,06	0,11	0,05	-0,02	0,25	0,02
200	10	0,14	0,19	0,14	0,19	0,08	0,08	0,11	0,14	0,11	0,05
	20	0,69	0,62	0,34	0,62	0,55	0,55	0,62	0,62	0,49	0,41
	30	1,71	0,57	1,36	0,57	0,34	0,57	0,68	1,70	0,34	0,34
	40	0,32	0,48	0,47	0,33	0,01	-0,15	-0,16	0,16	0,33	-0,31
	50	-0,39	0,01	-0,19	0,00	-0,39	-0,39	-0,78	-0,58	-0,19	-0,59
Media 200		0,49	0,38	0,42	0,34	0,12	0,13	0,09	0,41	0,21	-0,02
500	10	0,09	0,25	0,08	0,09	0,05	0,05	0,03	0,14	0,19	0,03
	20	0,48	0,68	0,26	0,37	0,20	0,17	0,28	0,45	0,54	0,14
	30	1,33	0,85	0,38	0,28	0,33	0,24	0,19	0,66	0,80	0,14
	40	2,20	1,26	0,53	0,33	0,40	0,26	0,60	2,13	1,26	0,33
	50	2,48	1,37	0,34	0,51	0,52	0,34	0,68	2,65	1,11	0,26
Media 500		1,31	0,88	0,32	0,32	0,30	0,21	0,36	1,21	0,78	0,18
1000	10	0,06	0,16	0,06	0,07	0,03	0,02	0,03	0,07	0,15	0,01
	20	0,60	0,49	0,26	0,29	0,10	0,07	0,09	0,79	0,45	0,07
	30	0,95	0,88	0,62	0,74	0,19	0,19	0,17	0,69	0,83	0,14
	40	1,52	1,21	1,18	1,08	0,24	0,24	0,24	1,48	1,15	0,13
	50	1,70	1,53	1,27	1,40	0,57	0,22	0,39	1,66	1,40	0,39
Media 1000		0,97	0,85	0,68	0,71	0,23	0,15	0,18	0,94	0,79	0,15
Media general		0,70	0,63	0,63	0,43	0,17	0,15	0,17	0,63	0,51	0,08

Tabla D.7 – Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo U(10,100) de los algoritmos de reducción de tamaño para 15 segundos como tiempo de parada.

n	m	CPLEX	DIG	3J	3J2M	3JD	2JDi	2JDi(90)	M-CPLEX	M-DIG	M-2JDi(90)
100	10	0,00	0,17	0,11	0,06	0,11	0,05	0,12	0,06	0,12	0,05
	20	0,26	0,25	0,13	0,25	0,25	0,25	0,27	0,13	0,13	0,00
	30	-0,20	0,22	0,43	0,22	-0,21	0,00	-0,20	-0,43	0,22	-0,43
	40	0,00	0,88	1,47	0,88	0,00	0,00	0,00	0,00	0,59	0,29
	50	0,00	0,00	3,22	0,00	0,00	0,00	0,00	0,00	0,00	0,00
Media 100		0,01	0,31	1,07	0,28	0,03	0,06	0,04	-0,05	0,21	-0,02
200	10	0,11	0,14	0,11	0,19	0,08	0,05	0,05	0,11	0,08	0,05
	20	0,62	0,35	0,28	0,41	0,41	0,48	0,49	0,55	0,35	0,34
	30	0,80	0,34	1,13	0,57	0,34	0,45	0,68	0,91	0,22	0,22
	40	-0,30	0,01	-0,16	0,01	-0,31	-0,62	-0,62	-0,16	0,01	-0,62
	50	-0,78	-0,19	-0,19	-0,20	-0,39	-0,39	-0,78	-0,78	-0,19	-0,59
Media 200		0,09	0,13	0,23	0,20	0,03	0,00	-0,04	0,13	0,10	-0,12
500	10	0,08	0,18	0,06	0,03	0,03	0,02	0,03	0,09	0,17	0,02
	20	0,28	0,51	0,26	0,31	0,14	0,11	0,17	0,29	0,45	0,09
	30	0,66	0,71	0,14	0,09	0,28	0,19	0,14	0,52	0,62	0,14
	40	1,86	1,13	0,33	0,26	0,20	0,07	0,60	1,13	1,06	0,07
	50	2,31	1,03	0,00	0,17	0,09	0,00	0,26	1,20	0,86	-0,08
Media 500		1,04	0,71	0,16	0,18	0,15	0,08	0,24	0,64	0,63	0,05
1000	10	0,03	0,14	0,03	0,03	0,02	0,02	0,02	0,06	0,13	0,01
	20	0,26	0,45	0,10	0,14	0,04	0,06	0,07	0,43	0,39	0,03
	30	0,86	0,81	0,19	0,12	0,07	0,14	0,07	0,33	0,74	0,10
	40	1,18	1,04	0,37	0,30	0,10	0,03	0,10	0,88	0,94	0,00
	50	1,44	1,44	0,39	0,39	0,35	0,09	0,09	1,31	1,31	0,09
Media 1000		0,75	0,78	0,22	0,20	0,12	0,07	0,07	0,60	0,70	0,04
Media general		0,47	0,48	0,42	0,21	0,08	0,05	0,08	0,33	0,41	-0,01

Tabla D.8 – Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo U(10,100) de los algoritmos de reducción de tamaño para 30 segundos como tiempo de parada.

n	m	CPLEX	DIG	3J	3J2M	3JD	2JDi	2JDi(90)	M-CPLEX	M-DIG	M-2JDi(90)
100	10	0,00	0,17	0,11	0,06	0,11	0,05	0,12	0,06	0,12	0,05
	20	0,26	0,25	0,13	0,25	0,25	0,00	0,14	0,13	0,13	0,00
	30	-0,20	0,22	0,43	0,22	-0,21	0,00	-0,20	-0,43	0,22	-0,43
	40	0,00	0,59	1,47	0,88	0,00	0,00	0,00	0,00	0,59	0,29
	50	0,00	0,00	3,22	0,00	0,00	0,00	0,00	0,00	0,00	0,00
Media 100		0,01	0,25	1,07	0,28	0,03	0,01	0,01	-0,05	0,21	-0,02
200	10	0,06	0,14	0,08	0,11	0,03	0,03	0,03	0,09	0,05	0,05
	20	0,41	0,21	0,21	0,35	0,41	0,41	0,49	0,41	0,21	0,14
	30	0,57	0,22	1,02	0,11	0,34	0,23	0,68	0,91	0,22	0,11
	40	-0,61	-0,14	-0,77	-0,14	-0,46	-0,62	-0,62	-0,47	-0,30	-0,78
	50	-0,78	-0,39	-0,38	-0,20	-0,59	-0,39	-0,78	-0,78	-0,59	-0,59
Media 200		-0,07	0,01	0,03	0,05	-0,05	-0,07	-0,04	0,03	-0,08	-0,21
500	10	0,03	0,16	0,02	0,02	0,03	0,00	0,02	0,07	0,15	0,02
	20	0,20	0,46	0,26	0,20	0,09	0,09	0,11	0,26	0,34	0,03
	30	0,24	0,57	0,09	0,09	0,24	0,05	0,05	0,14	0,52	-0,05
	40	0,93	1,00	0,20	0,13	0,00	0,07	0,47	0,60	0,80	0,00
	50	1,19	0,94	0,00	0,09	0,09	-0,17	0,17	0,86	0,86	-0,25
Media 500		0,52	0,62	0,11	0,11	0,09	0,01	0,17	0,38	0,53	-0,05
1000	10	0,01	0,11	0,01	0,01	0,01	0,02	0,01	0,02	0,11	0,01
	20	0,12	0,40	0,07	0,10	0,04	0,04	0,04	0,13	0,36	0,00
	30	0,40	0,69	0,14	0,10	0,05	0,05	0,02	0,33	0,57	0,00
	40	1,01	0,98	0,24	0,14	0,03	0,00	0,03	0,54	0,81	0,00
	50	1,22	1,18	0,13	0,17	0,22	0,00	0,04	0,65	1,09	0,09
Media 1000		0,55	0,67	0,12	0,10	0,07	0,02	0,03	0,34	0,59	0,02
Media general		0,25	0,39	0,33	0,13	0,03	-0,01	0,04	0,18	0,31	-0,06

Tabla D.9 – Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo U(10,100) de los algoritmos de reducción de tamaño para 60 segundos como tiempo de parada.

n	m	CPLEX	DIG	3J	3J2M	3JD	2JDi	2JDi(90)	M-CPLEX	M-DIG	M-2JDi(90)
100	10	0,00	0,12	0,11	0,06	0,11	0,00	0,00	0,06	0,12	0,00
	20	0,26	0,13	0,13	0,25	0,25	0,00	0,14	0,13	0,13	0,00
	30	-0,20	0,00	0,43	0,22	-0,21	0,00	-0,20	-0,43	-0,21	-0,43
	40	0,00	0,29	1,47	0,88	0,00	0,00	0,00	0,00	0,29	0,00
	50	0,00	0,00	3,22	0,00	0,00	0,00	0,00	0,00	0,00	0,00
Media 100		0,01	0,11	1,07	0,28	0,03	0,00	-0,01	-0,05	0,06	-0,09
200	10	0,03	0,08	0,05	0,08	0,03	0,03	0,00	0,06	0,03	0,00
	20	0,21	-0,07	0,14	0,35	0,34	0,28	0,07	0,41	-0,07	0,00
	30	0,46	0,22	0,90	0,11	0,34	0,23	0,11	0,80	0,11	-0,12
	40	-0,61	-0,46	-0,77	-0,45	-0,62	-0,78	-0,78	-0,47	-0,46	-1,09
	50	-0,78	-0,39	-0,38	-0,20	-0,59	-0,59	-0,78	-0,78	-0,59	-0,78
Media 200		-0,14	-0,12	-0,01	-0,02	-0,10	-0,17	-0,28	0,00	-0,20	-0,40
500	10	0,01	0,16	0,01	0,01	0,02	0,00	0,01	0,06	0,14	0,02
	20	0,14	0,40	0,17	0,11	0,09	0,09	0,09	0,23	0,31	0,03
	30	0,19	0,43	0,05	0,09	0,19	0,00	-0,05	0,10	0,33	-0,05
	40	0,40	0,66	0,20	0,00	0,00	-0,07	0,00	0,27	0,53	-0,13
	50	0,60	0,60	0,00	-0,08	0,09	-0,43	-0,08	0,51	0,43	-0,25
Media 500		0,27	0,45	0,09	0,03	0,08	-0,08	-0,01	0,23	0,35	-0,08
1000	10	0,01	0,11	0,01	0,00	0,01	0,02	0,01	0,02	0,11	0,01
	20	0,03	0,34	0,06	0,06	0,04	0,03	0,03	0,09	0,29	0,00
	30	0,24	0,60	0,10	0,10	0,02	0,02	0,02	0,24	0,52	0,00
	40	0,40	0,77	0,17	0,10	0,00	-0,07	-0,03	0,30	0,67	-0,03
	50	0,66	1,05	0,09	0,13	0,18	0,00	0,04	0,52	0,96	0,00
Media 1000		0,27	0,57	0,08	0,08	0,05	0,00	0,01	0,23	0,51	0,00
Media general		0,10	0,25	0,31	0,09	0,01	-0,06	-0,07	0,11	0,18	-0,14

Tabla D.10 – Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo U(10,100) de los algoritmos de reducción de tamaño para 120 segundos como tiempo de parada.

n	m	CPLEX	DIG	3J	3J2M	3JD	2JDi	2JDi(90)	M-CPLEX	M-DIG	M-2JDi(90)
100	10	0,00	0,12	0,11	0,06	0,06	0,00	0,00	0,06	0,12	0,00
	20	0,26	0,13	0,13	0,25	0,25	0,00	0,00	0,13	0,13	0,00
	30	-0,20	-0,21	0,43	0,22	-0,21	0,00	-0,43	-0,43	-0,21	-0,43
	40	0,00	0,00	1,47	0,88	0,00	0,00	0,00	0,00	0,00	0,00
	50	0,00	0,00	3,22	0,00	0,00	0,00	0,00	0,00	0,00	0,00
Media 100		0,01	0,01	1,07	0,28	0,02	0,00	-0,09	-0,05	0,01	-0,09
200	10	0,03	0,05	0,03	0,06	0,03	0,03	0,00	0,03	0,03	0,00
	20	0,21	-0,07	0,14	0,35	0,27	0,21	-0,07	0,34	-0,07	-0,14
	30	0,34	0,22	0,90	0,00	0,34	0,00	-0,34	0,68	0,00	-0,12
	40	-0,61	-0,46	-0,77	-0,45	-0,62	-0,78	-0,78	-0,47	-0,46	-1,09
	50	-0,78	-0,58	-0,38	-0,39	-0,59	-0,59	-0,78	-0,78	-0,78	-0,78
Media 200		-0,16	-0,17	-0,02	-0,09	-0,11	-0,23	-0,39	-0,04	-0,26	-0,42
500	10	0,00	0,11	0,00	0,01	0,01	0,00	-0,01	0,06	0,09	0,02
	20	0,11	0,31	0,11	0,11	0,06	0,06	0,09	0,17	0,28	0,03
	30	0,09	0,33	0,05	0,05	0,09	0,00	-0,05	0,05	0,19	-0,09
	40	0,33	0,53	0,00	0,00	-0,07	-0,13	-0,13	0,20	0,46	-0,13
	50	0,26	0,51	-0,25	-0,25	-0,08	-0,51	-0,17	0,09	0,34	-0,34
Media 500		0,16	0,36	-0,02	-0,02	0,00	-0,12	-0,06	0,11	0,27	-0,10
1000	10	0,01	0,10	0,01	0,00	0,01	0,01	0,01	0,02	0,10	0,00
	20	0,03	0,27	0,00	0,01	0,04	0,03	0,03	0,06	0,26	0,00
	30	0,14	0,45	0,05	0,05	0,00	0,02	0,02	0,05	0,40	-0,02
	40	0,37	0,57	0,10	0,03	-0,07	-0,07	-0,07	0,24	0,51	-0,10
	50	0,35	0,83	-0,04	0,09	0,13	-0,09	-0,04	0,35	0,74	-0,09
Media 1000		0,18	0,45	0,02	0,04	0,02	-0,02	-0,01	0,14	0,40	-0,04
Media general		0,05	0,16	0,26	0,05	-0,02	-0,09	-0,14	0,04	0,11	-0,16

Tabla D.11 – Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo U(10,100) de los algoritmos de reducción de tamaño para 240 segundos como tiempo de parada.

n	m	CPLEX	DIG	3J	3J2M	3JD	2JDi	2JDi(90)	M-CPLEX	M-DIG	M-2JDi(90)
100	10	0,00	0,12	0,11	0,06	0,06	0,00	0,00	0,00	0,12	0,00
	20	0,13	0,00	0,13	0,25	0,25	0,00	0,00	0,00	0,00	0,00
	30	-0,43	-0,21	0,43	0,22	-0,21	0,00	-0,43	-0,43	-0,21	-0,43
	40	0,00	0,00	1,47	0,88	0,00	0,00	0,00	0,00	0,00	0,00
	50	0,00	0,00	3,22	0,00	0,00	0,00	0,00	0,00	0,00	0,00
Media 100		-0,06	-0,02	1,07	0,28	0,02	0,00	-0,09	-0,09	-0,02	-0,09
200	10	0,03	0,03	0,03	0,06	0,03	0,00	0,00	0,00	0,03	0,00
	20	0,21	-0,07	0,00	0,28	0,07	0,14	-0,14	0,21	-0,07	-0,14
	30	0,34	0,22	0,22	-0,12	0,22	-0,23	-0,34	0,45	0,00	-0,23
	40	-0,61	-0,46	-0,77	-0,77	-0,78	-0,78	-0,78	-0,77	-0,46	-1,09
	50	-0,78	-0,58	-0,38	-0,59	-0,59	-0,59	-0,78	-0,78	-0,78	-0,78
Media 200		-0,16	-0,17	-0,18	-0,23	-0,21	-0,29	-0,41	-0,18	-0,26	-0,45
500	10	0,00	0,10	0,00	0,01	0,00	0,00	-0,01	0,05	0,08	0,01
	20	0,11	0,28	0,09	0,11	0,06	0,06	0,09	0,11	0,26	0,03
	30	0,09	0,33	-0,05	0,05	0,00	0,00	-0,05	0,05	0,19	-0,09
	40	0,33	0,53	0,00	0,00	-0,07	-0,20	-0,13	0,07	0,46	-0,13
	50	0,17	0,51	-0,25	-0,25	-0,34	-0,51	-0,26	0,09	0,17	-0,43
Media 500		0,14	0,35	-0,04	-0,02	-0,07	-0,13	-0,07	0,07	0,23	-0,12
1000	10	0,01	0,08	0,01	0,00	0,01	0,01	0,01	0,01	0,07	0,00
	20	0,03	0,26	0,00	0,01	0,01	0,03	0,03	0,04	0,22	0,00
	30	0,10	0,40	0,00	0,00	0,00	0,02	0,02	0,05	0,38	-0,02
	40	0,30	0,54	0,07	0,03	-0,07	-0,07	-0,07	0,17	0,51	-0,10
	50	0,35	0,70	-0,09	0,09	-0,09	-0,09	-0,04	0,31	0,66	-0,09
Media 1000		0,16	0,40	0,00	0,03	-0,03	-0,02	-0,01	0,12	0,37	-0,04
Media general		0,02	0,14	0,21	0,02	-0,07	-0,11	-0,14	-0,02	0,08	-0,17

Tabla D.12 – Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo U(10,100) de los algoritmos de reducción de tamaño para 300 segundos como tiempo de parada.

n	m	CPLEX	DIG	3J	3J2M	3JD	2JDi	2JDi(90)	M-CPLEX	M-DIG	M-2JDi(90)
100	10	0,23	0,35	0,29	0,25	0,25	0,27	0,25	0,41	0,29	0,16
	20	0,85	0,71	0,63	0,58	0,77	0,74	0,74	1,42	0,63	0,64
	30	1,71	0,85	1,66	1,37	0,51	0,51	0,80	1,65	0,63	0,40
	40	1,92	1,91	4,51	3,78	1,03	0,76	1,14	2,02	1,91	0,64
	50	2,72	1,90	10,75	10,67	0,65	0,45	0,05	2,37	1,65	-0,11
Media 100		1,49	1,14	3,57	3,33	0,64	0,55	0,60	1,57	1,02	0,35
200	10	0,15	0,24	0,10	0,11	0,10	0,10	0,09	0,21	0,18	0,05
	20	1,09	0,38	0,60	0,83	0,42	0,40	0,36	1,07	0,34	0,28
	30	2,45	0,60	0,86	0,69	0,46	0,58	0,49	1,69	0,52	0,58
	40	2,27	0,31	0,92	0,66	-0,12	0,26	0,07	2,20	0,19	-0,20
	50	2,83	0,59	1,59	1,82	0,24	0,19	0,39	2,51	0,34	0,20
Media 200		1,76	0,42	0,82	0,82	0,22	0,31	0,28	1,54	0,31	0,18
500	10	0,14	0,12	0,07	0,08	0,05	0,04	0,04	0,13	0,11	0,04
	20	0,58	0,20	0,28	0,46	0,18	0,20	0,19	0,43	0,18	0,20
	30	1,02	0,17	0,42	0,36	0,19	0,21	0,19	0,81	0,17	0,16
	40	1,44	-0,06	0,34	0,40	-0,02	-0,06	-0,09	1,88	-0,11	-0,05
	50	2,17	-0,15	0,17	0,08	-0,21	-0,19	-0,23	3,38	-0,19	-0,25
Media 500		1,07	0,05	0,26	0,27	0,04	0,04	0,02	1,32	0,03	0,02
1000	10	0,13	0,08	0,04	0,05	0,01	0,01	0,02	0,04	0,07	0,02
	20	0,35	0,18	0,15	0,17	0,04	0,05	0,04	0,30	0,17	0,04
	30	0,76	0,10	0,23	0,35	0,03	0,02	0,03	0,67	0,06	0,04
	40	1,57	-0,02	0,43	0,40	0,05	0,04	0,04	1,41	-0,03	0,02
	50	2,27	-0,10	0,35	0,51	0,10	0,09	0,10	2,00	-0,15	0,09
Media 1000		1,02	0,05	0,24	0,30	0,05	0,04	0,04	0,88	0,02	0,04
Media general		1,33	0,42	1,22	1,18	0,24	0,23	0,23	1,33	0,35	0,15

Tabla D.13 – Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo trabajos correlacionados de los algoritmos de reducción de tamaño para 15 segundos como tiempo de parada.

n	m	CPLEX	DIG	3J	3J2M	3JD	2JDi	2JDi(90)	M-CPLEX	M-DIG	M-2JDi(90)
100	10	0,18	0,25	0,24	0,20	0,21	0,25	0,21	0,31	0,23	0,14
	20	0,64	0,59	0,48	0,44	0,77	0,64	0,67	1,15	0,52	0,38
	30	0,98	0,62	1,37	1,15	0,46	0,12	0,52	1,42	0,40	0,18
	40	1,34	1,48	4,30	3,71	0,27	0,34	0,61	1,67	1,28	-0,02
	50	2,27	1,48	10,75	10,67	0,21	-0,18	-0,56	2,11	1,30	-0,63
Media 100		1,08	0,88	3,43	3,23	0,39	0,23	0,29	1,33	0,75	0,01
200	10	0,08	0,18	0,08	0,09	0,08	0,09	0,07	0,19	0,15	0,02
	20	0,58	0,32	0,53	0,65	0,30	0,30	0,34	0,58	0,28	0,17
	30	1,90	0,49	0,66	0,52	0,43	0,52	0,49	1,55	0,40	0,46
	40	1,69	0,19	0,85	0,39	-0,20	0,15	-0,08	2,01	0,04	-0,23
	50	1,79	0,45	1,31	1,39	0,15	-0,09	0,07	2,13	0,25	-0,09
Media 200		1,21	0,33	0,68	0,61	0,15	0,19	0,18	1,29	0,22	0,06
500	10	0,08	0,12	0,04	0,04	0,02	0,02	0,02	0,09	0,11	0,02
	20	0,42	0,15	0,24	0,33	0,17	0,19	0,15	0,43	0,15	0,14
	30	0,81	0,11	0,36	0,33	0,16	0,16	0,16	0,61	0,08	0,15
	40	1,00	-0,11	0,31	0,20	-0,05	-0,09	-0,11	0,97	-0,12	-0,11
	50	1,48	-0,27	0,10	-0,15	-0,36	-0,42	-0,29	1,73	-0,29	-0,34
Media 500		0,76	0,00	0,21	0,15	-0,01	-0,03	-0,01	0,77	-0,01	-0,03
1000	10	0,05	0,07	0,02	0,02	0,01	0,01	0,01	0,04	0,07	0,02
	20	0,18	0,17	0,08	0,09	0,02	0,02	0,02	0,18	0,15	0,02
	30	0,52	0,07	0,06	0,13	0,01	-0,01	0,01	0,31	0,05	0,01
	40	1,05	-0,04	0,12	0,16	0,02	0,00	-0,01	0,87	-0,05	0,01
	50	1,61	-0,18	0,11	0,19	-0,03	0,00	0,00	1,40	-0,19	0,00
Media 1000		0,68	0,02	0,08	0,12	0,01	0,01	0,01	0,56	0,01	0,01
Media general		0,93	0,31	1,10	1,03	0,13	0,10	0,12	0,99	0,24	0,01

Tabla D.14 – Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo trabajos correlacionados de los algoritmos de reducción de tamaño para 30 segundos como tiempo de parada.

n	m	CPLEX	DIG	3J	3J2M	3JD	2JDi	2JDi(90)	M-CPLEX	M-DIG	M-2JDi(90)
100	10	0,16	0,21	0,22	0,12	0,17	0,21	0,16	0,25	0,21	0,14
	20	0,52	0,52	0,41	0,26	0,67	0,45	0,56	0,97	0,52	0,26
	30	0,57	0,45	1,08	0,98	0,18	-0,10	0,17	1,15	0,23	0,01
	40	0,87	1,27	4,23	3,71	-0,15	-0,46	-0,03	1,45	0,97	-0,16
	50	1,04	1,31	10,75	10,67	-0,99	-1,01	-1,00	1,69	0,97	-1,16
Media 100		0,63	0,75	3,34	3,15	-0,02	-0,18	-0,03	1,10	0,58	-0,18
200	10	0,05	0,16	0,05	0,05	0,06	0,06	0,06	0,14	0,14	0,02
	20	0,36	0,26	0,45	0,46	0,29	0,17	0,30	0,47	0,24	0,15
	30	1,46	0,35	0,57	0,52	0,43	0,41	0,49	1,07	0,29	0,37
	40	1,65	0,04	0,70	0,20	-0,27	-0,01	-0,19	1,65	-0,08	-0,23
	50	1,47	0,25	1,07	1,06	-0,17	-0,14	-0,13	1,69	0,11	-0,27
Media 200		1,00	0,21	0,57	0,46	0,07	0,10	0,10	1,00	0,14	0,01
500	10	0,05	0,11	0,02	0,03	0,02	0,00	0,01	0,05	0,10	0,01
	20	0,34	0,14	0,15	0,18	0,14	0,15	0,12	0,36	0,13	0,11
	30	0,56	0,11	0,26	0,31	0,12	0,13	0,13	0,61	0,06	0,14
	40	0,75	-0,14	0,18	0,11	-0,05	-0,09	-0,11	0,81	-0,15	-0,14
	50	0,99	-0,30	0,00	-0,30	-0,40	-0,46	-0,38	0,99	-0,38	-0,38
Media 500		0,54	-0,02	0,12	0,06	-0,04	-0,05	-0,05	0,57	-0,05	-0,05
1000	10	0,03	0,07	0,02	0,02	0,01	0,01	0,01	0,03	0,06	0,01
	20	0,09	0,15	0,05	0,04	0,01	0,02	0,00	0,15	0,14	0,01
	30	0,26	0,05	0,02	0,06	-0,02	-0,03	-0,01	0,29	0,02	-0,01
	40	0,64	-0,07	0,04	0,08	0,00	-0,01	-0,04	0,45	-0,08	-0,02
	50	1,18	-0,19	0,05	0,06	-0,13	-0,11	-0,09	0,62	-0,20	-0,13
Media 1000		0,44	0,00	0,04	0,05	-0,02	-0,03	-0,02	0,31	-0,01	-0,03
Media general		0,65	0,24	1,02	0,93	0,00	-0,04	0,00	0,74	0,16	-0,06

Tabla D.15 – Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo trabajos correlacionados de los algoritmos de reducción de tamaño para 60 segundos como tiempo de parada.

n	m	CPLEX	DIG	3J	3J2M	3JD	2JDi	2JDi(90)	M-CPLEX	M-DIG	M-2JDi(90)
100	10	0,12	0,15	0,14	0,10	0,14	0,15	0,10	0,19	0,15	0,06
	20	0,34	0,30	0,33	0,11	0,49	0,45	0,27	0,86	0,26	0,15
	30	0,46	0,22	0,92	0,75	0,01	-0,22	0,00	0,75	0,00	-0,05
	40	0,65	0,84	4,01	3,71	-0,58	-0,66	-0,53	1,01	0,62	-0,52
	50	0,70	0,96	10,75	10,67	-1,44	-1,52	-1,35	1,24	0,51	-1,53
Media 100		0,45	0,50	3,23	3,07	-0,28	-0,36	-0,30	0,81	0,31	-0,38
200	10	0,05	0,14	0,02	0,04	0,03	0,06	0,03	0,08	0,12	-0,01
	20	0,21	0,23	0,40	0,38	0,23	0,15	0,26	0,40	0,19	0,13
	30	0,83	0,35	0,43	0,46	0,41	0,29	0,43	0,77	0,29	0,32
	40	1,24	-0,12	0,31	0,12	-0,31	-0,20	-0,27	1,42	-0,23	-0,31
	50	1,24	0,11	0,92	0,96	-0,27	-0,23	-0,37	1,50	-0,03	-0,37
Media 200		0,71	0,14	0,42	0,39	0,02	0,01	0,02	0,83	0,07	-0,05
500	10	0,04	0,10	0,02	0,02	0,02	0,00	0,01	0,02	0,09	0,00
	20	0,16	0,12	0,11	0,09	0,09	0,14	0,07	0,16	0,09	0,06
	30	0,50	0,07	0,21	0,24	0,09	0,13	0,12	0,58	0,05	0,13
	40	0,65	-0,17	0,17	0,06	-0,05	-0,09	-0,11	0,81	-0,20	-0,15
	50	0,57	-0,36	-0,04	-0,42	-0,46	-0,46	-0,38	0,93	-0,40	-0,46
Media 500		0,38	-0,05	0,09	0,00	-0,06	-0,06	-0,06	0,50	-0,07	-0,08
1000	10	0,02	0,05	0,01	0,01	0,01	0,01	0,01	0,02	0,05	0,01
	20	0,05	0,15	0,03	0,03	0,00	0,00	0,00	0,10	0,12	0,00
	30	0,14	0,03	-0,01	0,01	-0,04	-0,04	-0,02	0,21	0,01	-0,03
	40	0,41	-0,11	-0,02	0,02	-0,03	-0,02	-0,05	0,34	-0,11	-0,04
	50	0,68	-0,20	-0,04	-0,06	-0,15	-0,13	-0,13	0,60	-0,23	-0,16
Media 1000		0,26	-0,02	0,00	0,00	-0,04	-0,04	-0,04	0,26	-0,03	-0,04
Media general		0,45	0,14	0,93	0,87	-0,09	-0,11	-0,10	0,60	0,07	-0,14

Tabla D.16 – Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo trabajos correlacionados de los algoritmos de reducción de tamaño para 120 segundos como tiempo de parada.

n	m	CPLEX	DIG	3J	3J2M	3JD	2JDi	2JDi(90)	M-CPLEX	M-DIG	M-2JDi(90)
100	10	0,10	0,13	0,08	0,06	0,14	0,15	0,08	0,15	0,12	0,00
	20	0,26	0,26	0,15	0,11	0,41	0,30	0,19	0,60	0,19	0,01
	30	0,40	0,06	0,75	0,41	-0,16	-0,33	-0,17	0,69	-0,17	-0,22
	40	0,50	0,34	4,01	3,71	-0,71	-1,09	-0,67	0,79	0,26	-0,60
	50	0,35	0,71	10,75	10,67	-1,63	-1,52	-1,80	0,99	0,25	-1,88
Media 100		0,32	0,30	3,15	2,99	-0,39	-0,50	-0,47	0,65	0,13	-0,54
200	10	0,04	0,13	0,02	0,04	0,03	0,05	0,02	0,04	0,09	-0,02
	20	0,17	0,19	0,27	0,30	0,19	0,15	0,21	0,32	0,13	0,10
	30	0,51	0,23	0,32	0,46	0,38	0,20	0,38	0,49	0,20	0,20
	40	0,74	-0,23	0,27	0,04	-0,39	-0,20	-0,31	1,27	-0,31	-0,35
	50	1,19	-0,08	0,56	0,77	-0,27	-0,27	-0,46	1,37	-0,22	-0,42
Media 200		0,53	0,05	0,29	0,32	-0,01	-0,01	-0,03	0,70	-0,02	-0,10
500	10	0,03	0,08	0,01	0,01	0,01	0,00	0,00	0,02	0,08	0,00
	20	0,07	0,10	0,07	0,06	0,08	0,12	0,05	0,11	0,07	0,06
	30	0,40	0,04	0,13	0,17	0,07	0,09	0,10	0,43	0,04	0,13
	40	0,60	-0,18	0,08	0,03	-0,05	-0,09	-0,12	0,57	-0,23	-0,15
	50	0,49	-0,42	-0,08	-0,46	-0,48	-0,46	-0,40	0,78	-0,49	-0,46
Media 500		0,32	-0,08	0,04	-0,04	-0,07	-0,07	-0,07	0,38	-0,11	-0,08
1000	10	0,02	0,05	0,01	0,01	0,01	0,01	0,01	0,01	0,04	0,01
	20	0,03	0,12	0,03	0,02	-0,01	-0,01	0,00	0,03	0,11	0,00
	30	0,09	0,02	-0,02	-0,01	-0,04	-0,04	-0,02	0,14	0,00	-0,04
	40	0,24	-0,11	-0,02	-0,05	-0,04	-0,05	-0,05	0,22	-0,12	-0,04
	50	0,41	-0,22	-0,11	-0,10	-0,15	-0,14	-0,14	0,37	-0,25	-0,16
Media 1000		0,16	-0,03	-0,02	-0,03	-0,04	-0,05	-0,04	0,15	-0,04	-0,05
Media general		0,33	0,06	0,86	0,81	-0,13	-0,16	-0,15	0,47	-0,01	-0,19

Tabla D.17 – Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo trabajos correlacionados de los algoritmos de reducción de tamaño para 240 segundos como tiempo de parada.

n	m	CPLEX	DIG	3J	3J2M	3JD	2JDi	2JDi(90)	M-CPLEX	M-DIG	M-2JDi(90)
100	10	0,08	0,13	0,02	0,06	0,08	0,04	0,06	0,10	0,12	0,00
	20	0,22	0,23	0,11	0,04	0,23	0,16	0,16	0,40	0,15	0,01
	30	0,34	0,06	0,75	0,41	-0,33	-0,33	-0,28	0,58	-0,23	-0,33
	40	0,43	0,06	4,01	3,64	-0,87	-1,17	-0,82	0,51	-0,02	-0,67
	50	0,35	0,71	10,75	10,67	-1,63	-1,61	-1,80	0,72	0,16	-1,88
Media 100		0,28	0,24	3,13	2,96	-0,51	-0,58	-0,53	0,46	0,04	-0,57
200	10	0,04	0,12	0,02	0,02	0,03	0,03	0,02	0,04	0,09	-0,02
	20	0,15	0,17	0,25	0,29	0,15	0,13	0,17	0,30	0,11	0,10
	30	0,40	0,17	0,32	0,46	0,35	0,20	0,38	0,41	0,17	0,18
	40	0,61	-0,27	0,23	0,04	-0,39	-0,28	-0,31	1,04	-0,39	-0,35
	50	1,09	-0,08	0,51	0,64	-0,31	-0,37	-0,51	1,07	-0,22	-0,42
Media 200		0,46	0,02	0,27	0,29	-0,03	-0,06	-0,05	0,57	-0,05	-0,10
500	10	0,03	0,08	0,01	0,01	0,00	0,00	0,00	0,01	0,07	0,00
	20	0,05	0,10	0,06	0,05	0,08	0,06	0,05	0,06	0,07	0,05
	30	0,28	0,04	0,12	0,15	0,06	0,08	0,10	0,19	0,04	0,12
	40	0,55	-0,18	0,08	0,02	-0,05	-0,09	-0,12	0,46	-0,23	-0,15
	50	0,48	-0,46	-0,08	-0,46	-0,48	-0,46	-0,40	0,53	-0,51	-0,46
Media 500		0,28	-0,09	0,04	-0,05	-0,08	-0,08	-0,07	0,25	-0,11	-0,09
1000	10	0,02	0,05	0,01	0,01	0,01	0,00	0,00	0,01	0,04	0,01
	20	0,02	0,10	0,01	0,01	-0,01	-0,01	0,00	0,02	0,08	0,00
	30	0,09	0,02	-0,02	-0,02	-0,04	-0,04	-0,02	0,10	0,00	-0,04
	40	0,20	-0,12	-0,06	-0,05	-0,05	-0,05	-0,05	0,20	-0,12	-0,04
	50	0,36	-0,22	-0,11	-0,10	-0,16	-0,14	-0,14	0,34	-0,25	-0,16
Media 1000		0,14	-0,04	-0,03	-0,03	-0,05	-0,05	-0,04	0,13	-0,05	-0,05
Media general		0,29	0,03	0,85	0,79	-0,17	-0,19	-0,17	0,35	-0,04	-0,20

Tabla D.18 – Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo trabajos correlacionados de los algoritmos de reducción de tamaño para 300 segundos como tiempo de parada.

n	m	CPLEX	DIG	3J	3J2M	3JD	2JDi	2JDi(90)	M-CPLEX	M-DIG	M-2JDi(90)
100	10	0,00	0,24	54,42	1,15	0,06	0,00	0,00	0,00	0,24	0,03
	20	0,00	0,42	74,42	2,39	0,09	0,00	0,00	0,00	0,33	0,12
	30	0,00	0,42	94,09	2,78	0,00	0,00	0,00	0,00	0,29	0,00
	40	0,00	0,15	108,18	2,60	0,00	0,00	0,00	0,00	0,15	0,00
	50	0,00	0,32	91,07	4,77	0,00	0,00	0,00	0,00	0,32	0,00
Media 100		0,00	0,31	84,44	2,74	0,03	0,00	0,00	0,00	0,27	0,03
200	10	0,27	0,55	62,96	1,24	0,31	0,25	0,22	0,04	0,48	0,17
	20	0,09	0,30	84,80	1,30	0,17	0,00	-0,02	0,03	0,25	-0,02
	30	0,43	0,46	101,70	1,61	0,06	0,06	0,06	0,17	0,40	0,00
	40	0,46	0,42	113,92	2,37	0,15	0,08	0,00	0,12	0,34	0,00
	50	0,61	0,26	122,88	3,49	-0,13	-0,05	-0,04	0,43	0,26	-0,13
Media 200		0,37	0,40	97,25	2,00	0,11	0,07	0,05	0,16	0,35	0,00
500	10	0,19	0,65	65,37	1,08	0,43	0,38	0,44	0,13	0,62	0,41
	20	0,68	0,70	108,46	2,19	0,40	0,36	0,42	0,40	0,66	0,40
	30	0,95	0,61	118,53	2,44	0,25	0,30	0,26	0,86	0,52	0,27
	40	1,20	0,65	117,72	2,42	0,27	0,18	0,21	1,00	0,55	0,18
	50	1,27	0,68	128,72	3,47	0,32	0,19	0,21	1,46	0,64	0,25
Media 500		0,86	0,66	107,76	2,32	0,33	0,28	0,31	0,77	0,60	0,30
1000	10	0,21	0,58	55,77	0,85	0,41	0,39	0,43	0,11	0,57	0,38
	20	0,39	0,83	91,28	2,10	0,68	0,67	0,68	0,40	0,80	0,65
	30	0,56	0,90	101,62	2,77	0,63	0,59	0,61	0,51	0,87	0,70
	40	0,86	0,66	135,92	2,70	0,34	0,40	0,43	0,53	0,65	0,41
	50	1,48	0,77	133,24	3,92	0,43	0,45	0,47	1,15	0,73	0,44
Media 1000		0,70	0,75	103,56	2,47	0,50	0,50	0,52	0,54	0,72	0,51
Media general		0,48	0,53	98,25	2,38	0,24	0,21	0,22	0,37	0,48	0,21

Tabla D.19 – Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo máquinas correlacionadas de los algoritmos de reducción de tamaño para 15 segundos como tiempo de parada.

n	m	CPLEX	DIG	3J	3J2M	3JD	2JDi	2JDi(90)	M-CPLEX	M-DIG	M-2JDi(90)
100	10	0,00	0,24	54,42	1,15	0,06	0,00	0,00	0,00	0,20	0,03
	20	0,00	0,42	74,42	2,39	0,09	0,00	0,00	0,00	0,27	0,00
	30	0,00	0,34	94,09	2,78	0,00	0,00	0,00	0,00	0,21	0,00
	40	0,00	0,15	108,18	2,60	0,00	0,00	0,00	0,00	0,15	0,00
	50	0,00	0,32	91,07	4,77	0,00	0,00	0,00	0,00	0,32	0,00
Media 100		0,00	0,29	84,44	2,74	0,03	0,00	0,00	0,00	0,23	0,01
200	10	0,14	0,51	62,96	1,22	0,31	0,19	0,20	0,02	0,45	0,14
	20	-0,02	0,28	84,80	1,30	0,17	0,00	-0,02	-0,02	0,22	-0,02
	30	0,20	0,28	101,70	1,61	0,06	0,06	0,06	0,03	0,28	0,00
	40	0,13	0,42	113,92	2,37	0,15	0,00	0,00	0,00	0,34	0,00
	50	-0,04	0,26	122,88	3,49	-0,13	-0,05	-0,04	0,05	0,26	-0,13
Media 200		0,08	0,35	97,25	2,00	0,11	0,04	0,04	0,01	0,31	0,00
500	10	0,11	0,61	65,37	1,05	0,41	0,33	0,43	0,11	0,59	0,32
	20	0,39	0,64	108,46	2,17	0,40	0,32	0,35	0,32	0,59	0,36
	30	0,76	0,55	118,53	2,44	0,25	0,24	0,22	0,40	0,47	0,20
	40	0,95	0,63	117,72	2,35	0,20	0,12	0,15	0,58	0,55	0,12
	50	1,08	0,64	128,72	3,47	0,32	0,16	0,18	0,77	0,64	0,04
Media 500		0,66	0,61	107,76	2,30	0,32	0,24	0,27	0,43	0,57	0,21
1000	10	0,07	0,55	55,77	0,81	0,40	0,36	0,43	0,04	0,54	0,35
	20	0,32	0,81	91,27	2,04	0,65	0,61	0,64	0,26	0,77	0,57
	30	0,47	0,88	101,61	2,69	0,59	0,54	0,53	0,33	0,84	0,64
	40	0,59	0,65	135,92	2,66	0,32	0,37	0,35	0,35	0,62	0,37
	50	0,76	0,73	133,24	3,86	0,41	0,38	0,44	0,72	0,67	0,36
Media 1000		0,44	0,72	103,56	2,41	0,47	0,45	0,48	0,34	0,69	0,46
Media general		0,30	0,50	98,25	2,36	0,23	0,18	0,20	0,20	0,45	0,17

Tabla D.20 – Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo máquinas correlacionadas de los algoritmos de reducción de tamaño para 30 segundos como tiempo de parada.

n	m	CPLEX	DIG	3J	3J2M	3JD	2JDi	2JDi(90)	M-CPLEX	M-DIG	M-2JDi(90)
100	10	0,00	0,22	54,42	1,15	0,06	0,00	0,00	0,00	0,18	0,00
	20	0,00	0,35	74,42	2,39	0,09	0,00	0,00	0,00	0,27	0,00
	30	0,00	0,34	94,09	2,78	0,00	0,00	0,00	0,00	0,21	0,00
	40	0,00	0,15	108,18	2,60	0,00	0,00	0,00	0,00	0,15	0,00
	50	0,00	0,32	91,07	4,77	0,00	0,00	0,00	0,00	0,32	0,00
Media 100		0,00	0,28	84,44	2,74	0,03	0,00	0,00	0,00	0,23	0,00
200	10	0,04	0,46	62,96	1,20	0,27	0,16	0,17	0,02	0,41	0,14
	20	-0,02	0,28	84,78	1,30	0,17	0,00	-0,02	-0,02	0,20	-0,02
	30	0,10	0,17	101,70	1,61	0,06	0,06	0,06	0,00	0,17	0,00
	40	0,00	0,42	113,92	2,37	0,15	0,00	0,00	0,00	0,34	0,00
	50	-0,13	0,26	122,88	3,49	-0,13	-0,05	-0,04	-0,05	0,26	-0,13
Media 200		0,00	0,32	97,25	1,99	0,10	0,04	0,03	-0,01	0,28	0,00
500	10	0,05	0,57	65,37	1,00	0,40	0,33	0,41	0,08	0,56	0,30
	20	0,20	0,63	108,44	2,14	0,39	0,32	0,29	0,22	0,58	0,33
	30	0,22	0,50	118,53	2,41	0,20	0,22	0,22	0,18	0,43	0,14
	40	0,61	0,63	117,69	2,35	0,20	0,09	0,12	0,06	0,48	0,12
	50	0,80	0,60	128,69	3,47	0,26	0,13	0,18	0,43	0,60	0,01
Media 500		0,38	0,58	107,75	2,28	0,29	0,22	0,24	0,19	0,53	0,18
1000	10	0,03	0,52	55,77	0,80	0,40	0,36	0,38	0,03	0,51	0,32
	20	0,23	0,78	91,27	2,01	0,63	0,56	0,60	0,14	0,73	0,54
	30	0,25	0,85	101,61	2,63	0,56	0,52	0,48	0,17	0,83	0,58
	40	0,43	0,62	135,92	2,64	0,32	0,33	0,30	0,27	0,56	0,29
	50	0,44	0,65	133,24	3,83	0,39	0,35	0,42	0,30	0,63	0,30
Media 1000		0,27	0,68	103,56	2,38	0,46	0,42	0,43	0,18	0,65	0,41
Media general		0,16	0,47	98,25	2,35	0,22	0,17	0,18	0,09	0,42	0,15

Tabla D.21 – Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo máquinas correlacionadas de los algoritmos de reducción de tamaño para 60 segundos como tiempo de parada.

n	m	CPLEX	DIG	3J	3J2M	3JD	2JDi	2JDi(90)	M-CPLEX	M-DIG	M-2JDi(90)
100	10	0,00	0,18	54,42	1,15	0,06	0,00	0,00	0,00	0,11	0,00
	20	0,00	0,27	74,42	2,39	0,09	0,00	0,00	0,00	0,27	0,00
	30	0,00	0,21	94,09	2,78	0,00	0,00	0,00	0,00	0,21	0,00
	40	0,00	0,15	108,18	2,60	0,00	0,00	0,00	0,00	0,15	0,00
	50	0,00	0,32	91,07	4,77	0,00	0,00	0,00	0,00	0,32	0,00
Media 100		0,00	0,23	84,44	2,74	0,03	0,00	0,00	0,00	0,21	0,00
200	10	0,01	0,46	62,96	1,20	0,26	0,16	0,10	0,01	0,38	0,07
	20	-0,02	0,17	84,75	1,30	0,17	0,00	-0,02	-0,02	0,14	-0,02
	30	0,06	0,17	101,70	1,61	0,06	0,06	0,06	0,00	0,17	0,00
	40	0,00	0,34	113,92	2,37	0,15	0,00	0,00	0,00	0,34	0,00
	50	-0,13	0,17	122,88	3,49	-0,13	-0,05	-0,04	-0,05	0,17	-0,13
Media 200		-0,02	0,26	97,24	1,99	0,10	0,04	0,02	-0,01	0,24	-0,02
500	10	0,03	0,56	65,37	0,99	0,39	0,32	0,33	0,05	0,54	0,24
	20	0,17	0,57	108,44	2,14	0,38	0,29	0,26	0,08	0,55	0,30
	30	0,20	0,46	118,53	2,41	0,20	0,20	0,16	0,02	0,42	0,09
	40	0,21	0,57	117,69	2,29	0,20	0,06	0,12	0,03	0,45	0,06
	50	0,42	0,60	128,69	3,47	0,26	0,10	0,14	0,15	0,60	0,01
Media 500		0,20	0,55	107,75	2,26	0,29	0,20	0,20	0,07	0,51	0,14
1000	10	0,02	0,51	55,77	0,79	0,40	0,36	0,30	0,02	0,49	0,23
	20	0,15	0,71	91,27	1,99	0,62	0,51	0,51	0,10	0,70	0,48
	30	0,19	0,77	101,61	2,61	0,55	0,52	0,44	0,15	0,77	0,52
	40	0,21	0,60	135,92	2,64	0,32	0,31	0,26	0,10	0,54	0,23
	50	0,26	0,64	133,24	3,80	0,37	0,31	0,34	0,09	0,60	0,25
Media 1000		0,17	0,64	103,56	2,37	0,45	0,40	0,37	0,09	0,62	0,34
Media general		0,09	0,42	98,25	2,34	0,22	0,16	0,15	0,04	0,40	0,12

Tabla D.22 – Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo máquinas correlacionadas de los algoritmos de reducción de tamaño para 120 segundos como tiempo de parada.

n	m	CPLEX	DIG	3J	3J2M	3JD	2JDi	2JDi(90)	M-CPLEX	M-DIG	M-2JDi(90)
100	10	0,00	0,18	54,42	1,15	0,06	0,00	0,00	0,00	0,11	0,00
	20	0,00	0,15	74,42	2,39	0,09	0,00	0,00	0,00	0,15	0,00
	30	0,00	0,13	94,09	2,78	0,00	0,00	0,00	0,00	0,13	0,00
	40	0,00	0,00	108,18	2,60	0,00	0,00	0,00	0,00	0,00	0,00
	50	0,00	0,00	91,07	4,77	0,00	0,00	0,00	0,00	0,00	0,00
Media 100		0,00	0,09	84,44	2,74	0,03	0,00	0,00	0,00	0,08	0,00
200	10	0,01	0,41	62,96	1,20	0,26	0,16	0,06	-0,02	0,33	-0,04
	20	-0,02	0,17	84,75	1,30	0,17	0,00	-0,02	-0,02	0,14	-0,02
	30	0,06	0,17	101,70	1,61	0,06	0,06	0,06	0,00	0,17	0,00
	40	0,00	0,34	113,92	2,37	0,15	0,00	0,00	0,00	0,34	0,00
	50	-0,13	0,17	122,88	3,49	-0,13	-0,05	-0,04	-0,05	0,17	-0,13
Media 200		-0,02	0,25	97,24	1,99	0,10	0,04	0,01	-0,02	0,23	-0,04
500	10	0,02	0,54	65,37	0,98	0,39	0,32	0,23	0,05	0,49	0,11
	20	0,09	0,54	108,44	2,12	0,38	0,29	0,22	0,04	0,50	0,16
	30	0,06	0,44	118,53	2,41	0,20	0,16	0,06	0,00	0,42	0,04
	40	0,09	0,57	117,69	2,29	0,20	0,06	0,06	0,00	0,42	0,06
	50	0,15	0,56	128,69	3,47	0,26	0,06	0,08	0,08	0,56	0,01
Media 500		0,08	0,53	107,75	2,26	0,29	0,18	0,13	0,04	0,48	0,08
1000	10	0,02	0,48	55,77	0,78	0,39	0,36	0,24	0,01	0,46	0,10
	20	0,08	0,67	91,27	1,98	0,62	0,50	0,43	0,04	0,67	0,31
	30	0,16	0,73	101,61	2,60	0,55	0,51	0,41	0,12	0,72	0,41
	40	0,09	0,53	135,92	2,61	0,32	0,29	0,23	0,05	0,51	0,20
	50	0,18	0,59	133,24	3,80	0,37	0,29	0,24	0,08	0,57	0,17
Media 1000		0,10	0,60	103,56	2,35	0,45	0,39	0,31	0,06	0,59	0,24
Media general		0,04	0,37	98,25	2,34	0,22	0,15	0,11	0,02	0,34	0,07

Tabla D.23 – Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo máquinas correlacionadas de los algoritmos de reducción de tamaño para 240 segundos como tiempo de parada.

n	m	CPLEX	DIG	3J	3J2M	3JD	2JDi	2JDi(90)	M-CPLEX	M-DIG	M-2JDi(90)
100	10	0,00	0,15	54,42	1,15	0,06	0,00	0,00	0,00	0,11	0,00
	20	0,00	0,15	74,42	2,39	0,09	0,00	0,00	0,00	0,15	0,00
	30	0,00	0,13	94,09	2,78	0,00	0,00	0,00	0,00	0,13	0,00
	40	0,00	0,00	108,18	2,60	0,00	0,00	0,00	0,00	0,00	0,00
	50	0,00	0,00	91,07	4,77	0,00	0,00	0,00	0,00	0,00	0,00
Media 100		0,00	0,09	84,44	2,74	0,03	0,00	0,00	0,00	0,08	0,00
200	10	-0,02	0,37	62,96	1,20	0,23	0,14	0,02	-0,02	0,32	-0,04
	20	-0,02	0,14	84,75	1,30	0,17	0,00	-0,02	-0,02	0,14	-0,02
	30	0,06	0,17	101,70	1,61	0,06	0,06	0,06	0,00	0,17	0,00
	40	0,00	0,34	113,92	2,37	0,15	0,00	0,00	0,00	0,34	0,00
	50	-0,13	0,17	122,88	3,49	-0,13	-0,05	-0,04	-0,13	0,17	-0,13
Media 200		-0,02	0,24	97,24	1,99	0,10	0,03	0,00	-0,04	0,23	-0,04
500	10	0,01	0,51	65,37	0,97	0,39	0,27	0,23	0,04	0,49	0,05
	20	0,08	0,54	108,44	2,12	0,37	0,27	0,19	0,00	0,50	0,12
	30	0,02	0,44	118,53	2,41	0,20	0,16	0,06	0,00	0,42	0,04
	40	0,09	0,57	117,69	2,29	0,20	0,06	0,06	0,00	0,42	0,03
	50	0,09	0,56	128,69	3,47	0,26	0,06	0,05	0,08	0,56	0,01
Media 500		0,06	0,52	107,75	2,25	0,28	0,16	0,12	0,02	0,48	0,05
1000	10	0,01	0,47	55,77	0,78	0,39	0,35	0,20	0,00	0,45	0,05
	20	0,04	0,67	91,27	1,97	0,62	0,47	0,40	0,02	0,67	0,26
	30	0,13	0,72	101,61	2,60	0,54	0,47	0,40	0,07	0,71	0,37
	40	0,09	0,51	135,92	2,61	0,31	0,29	0,20	0,03	0,48	0,19
	50	0,16	0,58	133,24	3,80	0,37	0,29	0,24	0,08	0,56	0,14
Media 1000		0,09	0,59	103,56	2,35	0,45	0,38	0,29	0,04	0,57	0,20
Media general		0,03	0,36	98,25	2,33	0,21	0,14	0,10	0,01	0,34	0,05

Tabla D.24 – Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo máquinas correlacionadas de los algoritmos de reducción de tamaño para 300 segundos como tiempo de parada.

n	m	CPLEX	DIG	3J	3J2M	3JD	2JDi	2JDi(90)	M-CPLEX	M-DIG	M-2JDi(90)
100	10	0,03	0,29	0,02	0,03	0,02	0,01	0,01	0,02	0,27	0,02
	20	0,15	0,28	0,30	0,08	0,11	0,09	0,04	0,13	0,28	0,06
	30	0,69	-0,10	0,15	0,39	-0,10	-0,15	-0,07	0,17	-0,12	-0,12
	40	0,95	0,00	0,43	0,26	0,03	0,03	0,07	0,07	0,00	0,07
	50	0,00	0,14	0,67	0,00	0,00	0,00	0,00	0,00	0,00	0,00
Media 100		0,36	0,12	0,31	0,15	0,01	0,00	0,01	0,08	0,09	0,00
200	10	0,44	0,39	0,19	0,27	0,16	0,22	0,16	0,35	0,31	0,19
	20	0,27	0,51	0,10	0,12	0,14	0,17	0,14	0,18	0,45	0,17
	30	1,17	-0,31	0,38	0,34	-0,21	-0,21	-0,20	0,54	-0,37	-0,24
	40	0,48	0,33	0,25	0,10	0,06	0,10	0,06	0,37	0,33	0,08
	50	0,73	0,39	0,29	0,07	0,10	0,22	0,12	0,39	0,34	0,05
Media 200		0,62	0,26	0,24	0,18	0,05	0,10	0,06	0,37	0,21	0,05
500	10	0,19	0,23	0,06	0,07	0,03	0,02	0,03	0,18	0,20	0,02
	20	0,85	0,60	0,22	0,19	0,26	0,20	0,28	0,64	0,55	0,24
	30	0,71	0,52	1,19	1,14	0,64	0,59	0,63	0,75	0,49	0,66
	40	0,45	0,26	1,14	1,05	0,33	0,30	0,36	0,41	0,21	0,33
	50	0,73	0,44	0,09	0,15	0,11	0,11	0,08	0,63	0,43	0,02
Media 500		0,59	0,41	0,54	0,52	0,27	0,25	0,28	0,52	0,38	0,26
1000	10	0,21	0,13	0,10	0,08	0,02	0,02	0,02	0,21	0,11	0,03
	20	0,61	0,40	0,33	0,25	0,10	0,11	0,08	0,76	0,37	0,06
	30	0,68	0,43	0,82	0,70	0,56	0,54	0,56	0,74	0,42	0,58
	40	0,90	0,50	0,39	0,60	0,11	0,14	0,14	0,90	0,49	0,07
	50	0,70	0,42	0,36	0,40	0,04	-0,02	0,04	0,62	0,40	0,02
Media 1000		0,62	0,38	0,40	0,41	0,17	0,16	0,17	0,65	0,36	0,15
Media general		0,55	0,29	0,37	0,31	0,13	0,12	0,13	0,40	0,26	0,12

Tabla D.25 – Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo U(100,200) de los algoritmos de reducción de tamaño para 15 segundos como tiempo de parada.

n	m	CPLEX	DIG	3J	3J2M	3JD	2JDi	2JDi(90)	M-CPLEX	M-DIG	M-2JDi(90)
100	10	0,03	0,21	0,01	0,01	0,02	0,01	0,01	0,01	0,19	0,00
	20	0,11	0,26	0,30	0,06	0,11	0,06	0,04	0,08	0,24	0,04
	30	0,25	-0,12	0,10	0,35	-0,12	-0,15	-0,07	0,15	-0,15	-0,12
	40	0,20	0,00	0,33	0,20	0,03	0,00	0,03	0,07	0,00	0,00
	50	0,00	0,10	0,67	0,00	0,00	0,00	0,00	0,00	0,00	0,00
Media 100		0,12	0,09	0,28	0,12	0,01	-0,02	0,00	0,06	0,06	-0,02
200	10	0,26	0,32	0,12	0,13	0,15	0,18	0,12	0,30	0,25	0,14
	20	0,14	0,44	0,09	0,10	0,10	0,11	0,11	0,10	0,37	0,06
	30	0,91	-0,34	0,37	0,29	-0,23	-0,23	-0,20	0,27	-0,39	-0,28
	40	0,31	0,27	0,23	0,08	0,04	0,06	0,04	0,25	0,25	0,00
	50	0,34	0,34	0,24	0,05	0,10	0,12	0,00	0,27	0,24	0,05
Media 200		0,39	0,21	0,21	0,13	0,03	0,05	0,02	0,24	0,14	-0,01
500	10	0,13	0,20	0,04	0,05	0,03	0,01	0,02	0,04	0,16	0,02
	20	0,48	0,54	0,16	0,10	0,24	0,17	0,26	0,29	0,50	0,19
	30	0,62	0,47	1,09	1,06	0,64	0,59	0,63	0,56	0,44	0,65
	40	0,36	0,20	1,08	1,02	0,33	0,30	0,36	0,30	0,14	0,33
	50	0,72	0,39	0,03	0,02	0,05	0,07	0,02	0,36	0,37	0,00
Media 500		0,46	0,36	0,48	0,45	0,26	0,23	0,26	0,31	0,32	0,24
1000	10	0,09	0,10	0,03	0,03	0,02	0,01	0,02	0,02	0,08	0,02
	20	0,46	0,37	0,15	0,09	0,05	0,08	0,04	0,65	0,33	0,01
	30	0,61	0,39	0,73	0,64	0,51	0,53	0,56	0,66	0,37	0,58
	40	0,71	0,44	0,06	0,14	0,04	0,08	0,07	0,78	0,43	0,04
	50	0,66	0,34	0,08	0,05	0,01	-0,06	0,01	0,61	0,33	-0,05
Media 1000		0,51	0,33	0,21	0,19	0,13	0,13	0,14	0,55	0,31	0,12
Media general		0,37	0,25	0,29	0,22	0,11	0,10	0,10	0,29	0,21	0,08

Tabla D.26 – Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo U(100,200) de los algoritmos de reducción de tamaño para 30 segundos como tiempo de parada.

n	m	CPLEX	DIG	3J	3J2M	3JD	2JDi	2JDi(90)	M-CPLEX	M-DIG	M-2JDi(90)
100	10	0,02	0,18	0,01	0,00	0,01	0,01	0,01	0,00	0,15	0,00
	20	0,06	0,24	0,28	0,06	0,11	0,06	0,04	0,08	0,23	0,04
	30	0,12	-0,12	0,05	0,30	-0,12	-0,15	-0,07	0,12	-0,15	-0,15
	40	0,07	-0,03	0,26	0,20	0,03	0,00	0,03	0,03	-0,03	0,00
	50	0,00	0,05	0,67	0,00	0,00	0,00	0,00	0,00	0,00	0,00
Media 100		0,05	0,06	0,25	0,11	0,01	-0,02	0,00	0,05	0,04	-0,02
200	10	0,16	0,30	0,08	0,11	0,14	0,16	0,11	0,21	0,18	0,12
	20	0,09	0,37	0,06	0,09	0,08	0,10	0,08	0,10	0,32	0,06
	30	0,34	-0,41	0,31	0,13	-0,23	-0,38	-0,21	-0,27	-0,45	-0,32
	40	0,21	0,25	0,23	0,06	0,04	0,04	0,04	0,16	0,21	0,00
	50	0,22	0,27	0,24	0,00	0,10	0,12	0,00	0,22	0,22	0,00
Media 200		0,20	0,16	0,19	0,08	0,02	0,01	0,00	0,09	0,10	-0,03
500	10	0,05	0,16	0,03	0,04	0,02	0,01	0,02	0,01	0,13	0,01
	20	0,33	0,50	0,13	0,05	0,24	0,16	0,24	0,21	0,48	0,16
	30	0,55	0,43	1,03	1,06	0,64	0,58	0,60	0,36	0,40	0,59
	40	0,30	0,14	0,97	0,90	0,30	0,30	0,36	0,24	0,11	0,33
	50	0,54	0,32	0,01	-0,02	0,04	0,06	0,00	0,19	0,32	-0,02
Media 500		0,35	0,31	0,43	0,41	0,25	0,22	0,24	0,20	0,29	0,22
1000	10	0,02	0,08	0,01	0,02	0,01	0,01	0,02	0,01	0,07	0,02
	20	0,20	0,34	0,07	0,01	0,02	0,06	0,02	0,15	0,28	0,00
	30	0,49	0,36	0,61	0,59	0,46	0,45	0,56	0,47	0,34	0,52
	40	0,63	0,42	0,02	0,04	0,03	0,07	0,07	0,32	0,40	0,01
	50	0,60	0,32	-0,02	-0,03	-0,09	-0,09	-0,09	0,35	0,30	-0,10
Media 1000		0,39	0,30	0,14	0,13	0,09	0,10	0,12	0,26	0,28	0,09
Media general		0,25	0,21	0,25	0,18	0,09	0,08	0,09	0,15	0,18	0,06

Tabla D.27 – Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo U(100,200) de los algoritmos de reducción de tamaño para 60 segundos como tiempo de parada.

n	m	CPLEX	DIG	3J	3J2M	3JD	2JDi	2JDi(90)	M-CPLEX	M-DIG	M-2JDi(90)
100	10	0,02	0,16	0,01	0,00	0,01	0,00	0,00	0,00	0,14	0,00
	20	0,06	0,17	0,26	0,06	0,11	0,04	0,04	0,08	0,15	0,02
	30	0,10	-0,15	0,02	0,27	-0,15	-0,15	-0,12	0,10	-0,15	-0,15
	40	0,03	-0,03	0,26	0,20	0,03	0,00	0,03	0,00	-0,03	0,00
	50	0,00	0,00	0,67	0,00	0,00	0,00	0,00	0,00	0,00	0,00
Media 100		0,04	0,03	0,25	0,11	0,00	-0,02	-0,01	0,03	0,02	-0,03
200	10	0,12	0,26	0,05	0,08	0,13	0,10	0,06	0,14	0,15	0,01
	20	0,06	0,31	0,05	0,09	0,08	0,09	0,03	0,10	0,28	0,02
	30	-0,15	-0,42	0,27	0,08	-0,27	-0,38	-0,37	-0,57	-0,46	-0,52
	40	0,14	0,19	0,23	0,04	0,04	0,04	0,04	0,16	0,19	0,00
	50	0,19	0,19	0,24	0,00	0,10	0,05	0,00	0,15	0,15	0,00
Media 200		0,07	0,11	0,17	0,06	0,01	-0,02	-0,05	-0,01	0,06	-0,10
500	10	0,03	0,14	0,02	0,03	0,02	0,01	0,01	0,01	0,13	0,00
	20	0,26	0,44	0,08	0,03	0,19	0,15	0,04	0,06	0,44	0,02
	30	0,43	0,37	0,93	1,00	0,64	0,57	0,58	0,28	0,36	0,52
	40	0,27	0,10	0,95	0,89	0,28	0,30	0,31	0,23	0,06	0,31
	50	0,15	0,25	0,00	-0,03	0,04	0,04	-0,05	0,13	0,25	-0,08
Media 500		0,23	0,26	0,40	0,38	0,23	0,21	0,18	0,14	0,25	0,16
1000	10	0,01	0,07	0,01	0,02	0,01	0,01	0,01	0,01	0,06	0,01
	20	0,07	0,30	0,05	-0,01	0,00	0,03	-0,01	0,03	0,23	-0,01
	30	0,39	0,33	0,48	0,44	0,39	0,44	0,53	0,38	0,30	0,32
	40	0,19	0,37	0,01	0,02	0,02	0,01	0,00	0,18	0,35	0,01
	50	0,29	0,29	-0,08	-0,09	-0,10	-0,09	-0,11	0,21	0,25	-0,10
Media 1000		0,19	0,27	0,09	0,07	0,06	0,08	0,08	0,16	0,24	0,04
Media general		0,13	0,17	0,23	0,15	0,08	0,06	0,05	0,08	0,14	0,02

Tabla D.28 – Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo U(100,200) de los algoritmos de reducción de tamaño para 120 segundos como tiempo de parada.

n	m	CPLEX	DIG	3J	3J2M	3JD	2JDi	2JDi(90)	M-CPLEX	M-DIG	M-2JDi(90)
100	10	0,01	0,15	0,01	0,00	0,01	0,00	0,00	0,00	0,12	0,00
	20	0,04	0,11	0,26	0,06	0,11	0,02	0,04	0,04	0,11	0,00
	30	0,07	-0,15	0,00	0,25	-0,15	-0,15	-0,12	0,00	-0,15	-0,15
	40	0,00	-0,03	0,23	0,16	0,03	0,00	0,03	0,00	-0,07	0,00
	50	0,00	0,00	0,67	0,00	0,00	0,00	0,00	0,00	0,00	0,00
Media 100		0,02	0,02	0,23	0,09	0,00	-0,03	-0,01	0,01	0,00	-0,03
200	10	0,08	0,21	0,04	0,07	0,13	0,10	0,02	0,13	0,13	-0,01
	20	0,03	0,29	0,05	0,06	0,08	0,07	0,03	0,10	0,26	0,02
	30	-0,49	-0,48	0,26	0,08	-0,31	-0,38	-0,52	-0,66	-0,53	-0,63
	40	0,10	0,17	0,23	0,04	0,04	0,04	0,04	0,12	0,16	0,00
	50	0,10	0,19	0,24	0,00	0,10	0,02	0,00	0,12	0,10	0,00
Media 200		-0,04	0,08	0,16	0,05	0,01	-0,03	-0,09	-0,04	0,02	-0,12
500	10	0,02	0,11	0,02	0,02	0,02	0,01	0,01	0,01	0,10	-0,01
	20	0,17	0,41	0,02	0,02	0,14	0,15	0,02	0,05	0,39	0,02
	30	0,28	0,35	0,82	0,88	0,64	0,57	0,34	0,17	0,33	0,40
	40	0,22	0,06	0,95	0,87	0,28	0,30	0,28	0,21	0,02	0,28
	50	0,05	0,23	-0,02	-0,04	-0,04	0,01	-0,07	0,08	0,23	-0,09
Media 500		0,15	0,23	0,36	0,35	0,21	0,21	0,11	0,10	0,21	0,12
1000	10	0,01	0,07	0,01	0,01	0,01	0,01	0,01	0,00	0,05	0,00
	20	0,03	0,27	0,03	-0,01	0,00	0,02	-0,01	0,02	0,20	-0,01
	30	0,31	0,27	0,36	0,29	0,38	0,43	0,38	0,25	0,24	0,30
	40	0,14	0,34	0,00	0,00	0,01	0,00	-0,01	0,13	0,32	0,00
	50	0,13	0,25	-0,09	-0,11	-0,12	-0,09	-0,11	0,10	0,22	-0,11
Media 1000		0,12	0,24	0,06	0,04	0,06	0,08	0,05	0,10	0,21	0,04
Media general		0,06	0,14	0,20	0,13	0,07	0,06	0,02	0,04	0,11	0,00

Tabla D.29 – Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo U(100,200) de los algoritmos de reducción de tamaño para 240 segundos como tiempo de parada.

n	m	CPLEX	DIG	3J	3J2M	3JD	2JDi	2JDi(90)	M-CPLEX	M-DIG	M-2JDi(90)
100	10	0,01	0,13	0,01	0,00	0,01	0,00	0,00	0,00	0,12	0,00
	20	0,04	0,11	0,26	0,06	0,11	0,02	0,04	0,04	0,11	0,00
	30	-0,02	-0,15	0,00	0,15	-0,15	-0,15	-0,15	-0,05	-0,15	-0,15
	40	0,00	-0,03	0,23	0,16	0,03	-0,03	0,00	-0,03	-0,07	0,00
	50	0,00	0,00	0,67	0,00	0,00	0,00	0,00	0,00	0,00	0,00
Media 100		0,00	0,01	0,23	0,07	0,00	-0,03	-0,02	-0,01	0,00	-0,03
200	10	0,06	0,21	0,02	0,05	0,10	0,08	0,01	0,11	0,13	-0,01
	20	0,01	0,29	0,04	0,04	0,07	0,04	0,03	0,05	0,25	0,00
	30	-0,57	-0,48	-0,17	-0,28	-0,53	-0,57	-0,56	-0,70	-0,55	-0,67
	40	0,08	0,14	0,23	0,02	0,04	0,02	0,02	0,10	0,12	0,00
	50	0,05	0,17	0,24	0,00	0,10	0,02	0,00	0,10	0,10	0,00
Media 200		-0,07	0,07	0,07	-0,03	-0,05	-0,08	-0,10	-0,07	0,01	-0,14
500	10	0,01	0,11	0,01	0,02	0,01	0,01	0,00	0,00	0,10	-0,01
	20	0,14	0,41	0,02	0,01	0,06	0,02	0,01	0,03	0,37	0,02
	30	0,18	0,33	0,49	0,65	0,34	0,39	0,30	0,13	0,30	0,33
	40	0,17	0,05	0,77	0,72	0,27	0,27	0,28	0,15	0,00	0,27
	50	0,03	0,21	-0,02	-0,04	-0,07	-0,04	-0,08	0,06	0,19	-0,09
Media 500		0,11	0,22	0,25	0,27	0,12	0,13	0,10	0,07	0,19	0,11
1000	10	0,01	0,06	0,00	0,00	0,01	0,01	0,00	0,00	0,05	0,00
	20	0,02	0,26	0,02	-0,01	-0,01	0,02	-0,01	0,01	0,20	-0,01
	30	0,29	0,26	0,28	0,21	0,25	0,36	0,30	0,20	0,23	0,30
	40	0,13	0,33	0,00	-0,01	0,00	-0,01	-0,01	0,09	0,31	0,00
	50	0,11	0,25	-0,09	-0,11	-0,12	-0,12	-0,11	0,05	0,20	-0,12
Media 1000		0,11	0,23	0,04	0,02	0,03	0,05	0,03	0,07	0,20	0,03
Media general		0,04	0,13	0,15	0,08	0,03	0,02	0,00	0,02	0,10	-0,01

Tabla D.30 – Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo U(100,200) de los algoritmos de reducción de tamaño para 300 segundos como tiempo de parada.

n	m	CPLEX	DIG	3J	3J2M	3JD	2JDi	2JDi(90)	M-CPLEX	M-DIG	M-2JDi(90)
100	10	0,03	0,08	0,04	0,03	0,02	0,03	0,03	0,05	0,08	0,01
	20	0,02	0,04	0,08	0,04	0,02	0,02	0,04	0,02	0,04	0,00
	30	0,03	0,00	0,03	0,05	0,00	0,00	0,00	0,03	0,00	0,00
	40	0,03	0,00	0,00	0,03	0,00	0,00	0,00	0,00	0,00	0,00
	50	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
Media 100		0,02	0,02	0,03	0,03	0,01	0,01	0,01	0,02	0,02	0,00
200	10	0,04	0,05	0,00	0,00	0,02	0,02	0,02	0,04	0,05	0,01
	20	0,08	0,09	0,03	0,01	-0,01	0,01	-0,01	0,09	0,07	0,02
	30	0,29	0,01	0,16	0,16	0,01	0,00	-0,01	0,09	0,00	0,00
	40	0,04	0,02	0,02	0,00	0,00	0,00	0,00	0,04	0,02	0,00
	50	0,02	0,02	0,00	0,00	0,00	0,00	0,00	0,02	0,00	0,00
Media 200		0,09	0,04	0,04	0,03	0,01	0,01	0,00	0,06	0,03	0,01
500	10	0,05	0,09	0,01	0,01	0,02	0,02	0,01	0,03	0,07	0,01
	20	0,17	0,07	0,01	0,02	0,01	0,00	0,00	0,05	0,06	0,01
	30	0,33	0,09	0,32	0,33	0,11	0,11	0,11	0,33	0,09	0,11
	40	0,06	-0,02	0,12	0,15	-0,02	-0,02	-0,02	0,05	-0,02	-0,02
	50	0,08	0,01	-0,03	-0,03	-0,02	-0,04	-0,03	0,09	0,01	-0,04
Media 500		0,14	0,05	0,09	0,10	0,02	0,02	0,02	0,11	0,04	0,01
1000	10	0,15	0,07	0,02	0,03	0,01	0,02	0,02	0,04	0,06	0,01
	20	0,19	0,08	0,04	0,03	0,01	0,02	0,02	0,10	0,07	0,02
	30	0,09	0,03	0,19	0,20	0,05	0,04	0,04	0,09	0,02	0,04
	40	0,14	0,04	0,01	0,01	-0,01	-0,01	-0,01	0,12	0,04	-0,01
	50	0,04	0,01	-0,02	-0,02	-0,04	-0,04	-0,04	0,06	0,00	-0,05
Media 1000		0,12	0,04	0,05	0,05	0,00	0,00	0,00	0,08	0,04	0,00
Media general		0,09	0,04	0,05	0,05	0,01	0,01	0,01	0,07	0,03	0,01

Tabla D.31 – Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo U(100,120) de los algoritmos de reducción de tamaño para 15 segundos como tiempo de parada.

n	m	CPLEX	DIG	3J	3J2M	3JD	2JDi	2JDi(90)	M-CPLEX	M-DIG	M-2JDi(90)
100	10	0,02	0,08	0,04	0,02	0,02	0,03	0,01	0,05	0,08	0,00
	20	0,02	0,04	0,08	0,04	0,02	0,02	0,02	0,02	0,04	0,00
	30	0,00	0,00	0,03	0,05	0,00	0,00	0,00	0,00	0,00	0,00
	40	0,00	0,00	0,00	0,03	0,00	0,00	0,00	0,00	0,00	0,00
	50	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
Media 100		0,01	0,02	0,03	0,03	0,01	0,01	0,01	0,01	0,02	0,00
200	10	0,02	0,05	0,00	0,00	0,01	0,02	0,02	0,03	0,05	0,00
	20	0,05	0,09	0,00	0,01	-0,01	0,01	-0,01	0,08	0,07	0,00
	30	0,20	0,01	0,11	0,11	0,00	-0,04	-0,03	0,04	0,00	-0,03
	40	0,02	0,02	0,02	0,00	0,00	0,00	0,00	0,04	0,02	0,00
	50	0,00	0,02	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
Media 200		0,06	0,04	0,03	0,02	0,00	0,00	0,00	0,04	0,03	-0,01
500	10	0,01	0,07	0,01	0,01	0,02	0,01	0,01	0,01	0,06	0,01
	20	0,06	0,06	0,01	0,01	0,00	0,00	0,00	0,02	0,06	0,01
	30	0,32	0,09	0,32	0,33	0,11	0,11	0,11	0,32	0,09	0,11
	40	0,06	-0,02	0,12	0,12	-0,02	-0,02	-0,02	0,03	-0,02	-0,02
	50	0,07	0,01	-0,03	-0,03	-0,03	-0,04	-0,03	0,04	0,01	-0,04
Media 500		0,10	0,04	0,08	0,09	0,02	0,01	0,02	0,09	0,04	0,01
1000	10	0,05	0,07	0,00	0,01	0,01	0,01	0,02	0,02	0,05	0,01
	20	0,17	0,06	0,02	0,02	0,01	0,01	0,01	0,06	0,06	0,01
	30	0,07	0,02	0,15	0,20	0,05	0,04	0,04	0,06	0,01	0,04
	40	0,12	0,03	0,00	0,00	-0,01	-0,01	-0,02	0,08	0,03	-0,01
	50	0,03	0,00	-0,04	-0,05	-0,04	-0,04	-0,04	0,06	-0,01	-0,05
Media 1000		0,09	0,04	0,03	0,04	0,00	0,00	0,00	0,05	0,03	0,00
Media general		0,06	0,04	0,04	0,04	0,01	0,01	0,00	0,05	0,03	0,00

Tabla D.32 – Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo U(100,120) de los algoritmos de reducción de tamaño para 30 segundos como tiempo de parada.

n	m	CPLEX	DIG	3J	3J2M	3JD	2JDi	2JDi(90)	M-CPLEX	M-DIG	M-2JDi(90)
100	10	0,02	0,06	0,04	0,02	0,02	0,03	0,01	0,03	0,05	0,00
	20	0,00	0,04	0,08	0,02	0,02	0,02	0,02	0,00	0,04	0,00
	30	0,00	0,00	0,00	0,03	0,00	0,00	0,00	0,00	0,00	0,00
	40	0,00	0,00	0,00	0,03	0,00	0,00	0,00	0,00	0,00	0,00
	50	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
Media 100		0,00	0,02	0,02	0,02	0,01	0,01	0,01	0,01	0,02	0,00
200	10	0,01	0,05	0,00	0,00	0,01	0,02	0,01	0,03	0,05	0,00
	20	0,04	0,08	-0,01	0,01	-0,01	0,01	-0,01	0,07	0,07	0,00
	30	0,09	0,01	0,09	0,10	-0,03	-0,04	-0,06	-0,01	-0,01	-0,04
	40	0,02	0,02	0,02	0,00	0,00	0,00	0,00	0,02	0,02	0,00
	50	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
Media 200		0,03	0,03	0,02	0,02	-0,01	0,00	-0,01	0,02	0,02	-0,01
500	10	0,00	0,06	0,00	0,00	0,02	0,01	0,01	0,00	0,05	0,01
	20	0,02	0,06	0,01	0,01	0,00	0,00	0,00	0,02	0,06	0,01
	30	0,31	0,09	0,32	0,32	0,11	0,11	0,11	0,32	0,08	0,11
	40	0,04	-0,02	0,11	0,12	-0,02	-0,02	-0,02	0,03	-0,02	-0,02
	50	0,00	0,01	-0,03	-0,03	-0,03	-0,04	-0,03	0,02	0,01	-0,04
Media 500		0,07	0,04	0,08	0,08	0,02	0,01	0,02	0,08	0,04	0,01
1000	10	0,02	0,06	0,00	0,00	0,01	0,01	0,02	0,01	0,04	0,01
	20	0,08	0,05	0,01	0,00	0,01	0,01	0,00	0,03	0,05	0,00
	30	0,06	0,01	0,14	0,20	0,05	0,04	0,04	0,02	0,01	0,03
	40	0,11	0,02	-0,01	-0,01	-0,01	-0,01	-0,02	0,04	0,02	-0,02
	50	0,03	-0,02	-0,04	-0,05	-0,04	-0,04	-0,05	0,03	-0,02	-0,05
Media 1000		0,06	0,02	0,02	0,03	0,00	0,00	0,00	0,03	0,02	-0,01
Media general		0,04	0,03	0,04	0,04	0,01	0,01	0,00	0,03	0,02	0,00

Tabla D.33 – Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo U(100,120) de los algoritmos de reducción de tamaño para 60 segundos como tiempo de parada.

n	m	CPLEX	DIG	3J	3J2M	3JD	2JDi	2JDi(90)	M-CPLEX	M-DIG	M-2JDi(90)
100	10	0,02	0,06	0,04	0,02	0,02	0,03	0,00	0,03	0,04	0,00
	20	0,00	0,04	0,08	0,02	0,02	0,02	0,02	0,00	0,04	0,00
	30	0,00	0,00	0,00	0,03	0,00	0,00	0,00	0,00	0,00	0,00
	40	0,00	0,00	0,00	0,03	0,00	0,00	0,00	0,00	0,00	0,00
	50	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
Media 100		0,00	0,02	0,02	0,02	0,01	0,01	0,00	0,01	0,02	0,00
200	10	0,01	0,04	0,00	0,00	0,01	0,01	0,00	0,02	0,03	0,00
	20	0,02	0,07	-0,01	0,01	-0,01	0,01	-0,02	0,06	0,05	-0,03
	30	-0,03	-0,01	0,09	0,10	-0,03	-0,07	-0,09	-0,04	-0,03	-0,07
	40	0,02	0,02	0,02	0,00	0,00	0,00	0,00	0,02	0,02	0,00
	50	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
Media 200		0,01	0,02	0,02	0,02	-0,01	-0,01	-0,02	0,01	0,01	-0,02
500	10	0,00	0,06	0,00	0,00	0,02	0,01	0,00	0,00	0,05	0,00
	20	0,02	0,06	0,01	0,00	0,00	0,00	0,00	0,01	0,06	0,00
	30	0,29	0,09	0,32	0,32	0,11	0,11	0,11	0,32	0,08	0,09
	40	0,03	-0,02	0,10	0,11	-0,02	-0,02	-0,02	0,03	-0,02	-0,02
	50	0,00	0,00	-0,03	-0,04	-0,04	-0,04	-0,04	0,01	0,00	-0,04
Media 500		0,07	0,04	0,08	0,08	0,01	0,01	0,01	0,07	0,03	0,01
1000	10	0,01	0,04	0,00	0,00	0,01	0,01	0,01	0,00	0,04	0,00
	20	0,02	0,04	0,00	0,00	0,00	0,01	0,00	0,02	0,03	0,00
	30	0,04	0,01	0,14	0,20	0,05	0,04	0,04	0,02	0,01	0,02
	40	0,04	0,02	-0,01	-0,01	-0,01	-0,01	-0,02	0,02	0,01	-0,02
	50	0,02	-0,03	-0,04	-0,05	-0,05	-0,05	-0,05	-0,01	-0,03	-0,05
Media 1000		0,03	0,02	0,02	0,03	0,00	0,00	0,00	0,01	0,01	-0,01
Media general		0,03	0,02	0,04	0,04	0,00	0,00	0,00	0,03	0,02	-0,01

Tabla D.34 – Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo U(100,120) de los algoritmos de reducción de tamaño para 120 segundos como tiempo de parada.

n	m	CPLEX	DIG	3J	3J2M	3JD	2JDi	2JDi(90)	M-CPLEX	M-DIG	M-2JDi(90)
100	10	0,02	0,05	0,04	0,02	0,01	0,03	0,00	0,01	0,02	0,00
	20	0,00	0,04	0,08	0,02	0,02	0,02	0,00	0,00	0,04	0,00
	30	0,00	0,00	0,00	0,03	0,00	0,00	0,00	0,00	0,00	0,00
	40	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
	50	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
Media 100		0,00	0,02	0,02	0,01	0,01	0,01	0,00	0,00	0,01	0,00
200	10	0,01	0,03	0,00	0,00	0,01	0,01	0,00	0,02	0,02	0,00
	20	0,01	0,05	-0,01	0,01	-0,01	0,01	-0,02	0,04	0,03	-0,03
	30	-0,04	-0,01	0,09	0,10	-0,03	-0,07	-0,09	-0,04	-0,03	-0,10
	40	0,02	0,02	0,02	0,00	0,00	0,00	0,00	0,02	0,02	0,00
	50	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
Media 200		0,00	0,02	0,02	0,02	-0,01	-0,01	-0,02	0,01	0,01	-0,03
500	10	0,00	0,04	0,00	0,00	0,02	0,01	0,00	0,00	0,04	0,00
	20	0,02	0,06	0,01	0,00	0,00	0,00	0,00	0,01	0,04	-0,02
	30	0,23	0,08	0,29	0,32	0,11	0,11	0,11	0,19	0,08	0,08
	40	0,03	-0,02	0,08	0,09	-0,02	-0,02	-0,02	0,03	-0,02	-0,03
	50	-0,02	-0,01	-0,03	-0,04	-0,04	-0,04	-0,04	0,00	-0,01	-0,04
Media 500		0,05	0,03	0,07	0,07	0,01	0,01	0,01	0,05	0,03	0,00
1000	10	0,00	0,03	0,00	0,00	0,01	0,01	0,00	0,00	0,03	0,00
	20	0,01	0,03	0,00	0,00	0,00	0,01	0,00	0,01	0,03	0,00
	30	0,03	0,00	0,14	0,19	0,05	0,04	0,03	0,02	0,00	0,02
	40	0,02	0,01	-0,01	-0,02	-0,01	-0,01	-0,02	0,01	0,00	-0,03
	50	0,00	-0,03	-0,04	-0,05	-0,05	-0,05	-0,05	-0,03	-0,03	-0,06
Media 1000		0,01	0,01	0,02	0,02	0,00	0,00	-0,01	0,00	0,00	-0,01
Media general		0,02	0,02	0,03	0,03	0,00	0,00	-0,01	0,01	0,01	-0,01

Tabla D.35 – Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo U(100,120) de los algoritmos de reducción de tamaño para 240 segundos como tiempo de parada.

n	m	CPLEX	DIG	3J	3J2M	3JD	2JDi	2JDi(90)	M-CPLEX	M-DIG	M-2JDi(90)
100	10	0,02	0,04	0,04	0,02	0,01	0,01	0,00	0,00	0,02	0,00
	20	0,00	0,04	0,08	0,02	0,02	0,02	0,00	0,00	0,04	0,00
	30	0,00	0,00	0,00	0,03	0,00	0,00	0,00	0,00	0,00	0,00
	40	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
	50	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
Media 100		0,00	0,02	0,02	0,01	0,01	0,01	0,00	0,00	0,01	0,00
200	10	0,01	0,02	0,00	0,00	0,01	0,00	0,00	0,01	0,02	0,00
	20	0,01	0,04	-0,01	0,01	-0,01	0,00	-0,02	0,02	0,03	-0,03
	30	-0,04	-0,01	0,03	0,09	-0,06	-0,07	-0,09	-0,04	-0,03	-0,10
	40	0,02	0,00	0,02	0,00	0,00	0,00	0,00	0,00	0,00	0,00
	50	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
Media 200		0,00	0,01	0,01	0,02	-0,01	-0,01	-0,02	0,00	0,01	-0,03
500	10	0,00	0,04	0,00	0,00	0,01	0,01	0,00	0,00	0,04	0,00
	20	0,02	0,05	0,01	0,00	0,00	0,00	0,00	0,00	0,04	-0,02
	30	0,20	0,08	0,12	0,20	0,11	0,11	0,11	0,14	0,08	0,08
	40	0,02	-0,02	0,01	0,03	-0,02	-0,02	-0,02	0,02	-0,02	-0,03
	50	-0,02	-0,01	-0,03	-0,04	-0,04	-0,04	-0,04	0,00	-0,01	-0,04
Media 500		0,04	0,03	0,02	0,04	0,01	0,01	0,01	0,03	0,03	0,00
1000	10	0,00	0,03	0,00	0,00	0,01	0,01	0,00	0,00	0,03	0,00
	20	0,00	0,03	0,00	0,00	0,00	0,01	0,00	0,00	0,03	0,00
	30	0,03	0,00	0,12	0,17	0,04	0,03	0,03	0,02	0,00	0,02
	40	0,02	0,01	-0,01	-0,02	-0,01	-0,01	-0,02	0,01	0,00	-0,03
	50	0,00	-0,03	-0,04	-0,05	-0,05	-0,05	-0,05	-0,03	-0,03	-0,06
Media 1000		0,01	0,01	0,01	0,02	0,00	0,00	-0,01	0,00	0,00	-0,01
Media general		0,01	0,02	0,02	0,02	0,00	0,00	-0,01	0,01	0,01	-0,01

Tabla D.36 – Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo U(100,120) de los algoritmos de reducción de tamaño para 300 segundos como tiempo de parada.

n	m	CPLEX	DIG	3J	3J2M	3JD	2JDi	2JDi(90)	M-CPLEX	M-DIG	M-2JDi(90)
100	10	0,03	0,02	0,02	0,03	0,02	0,02	0,01	0,03	0,02	0,02
	20	0,02	0,01	0,02	0,01	0,00	0,01	0,01	0,02	0,01	0,01
	30	0,10	-0,01	0,04	0,05	-0,02	-0,01	-0,01	0,02	-0,02	-0,02
	40	0,13	-0,01	0,05	0,07	0,00	0,00	0,00	0,06	-0,01	0,00
	50	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,01	0,00	0,00
Media 100		0,06	0,00	0,03	0,03	0,00	0,00	0,00	0,03	0,00	0,00
200	10	0,02	0,04	0,02	0,02	0,01	0,02	0,01	0,03	0,04	0,01
	20	0,04	0,05	0,02	0,01	0,03	0,03	0,04	0,04	0,05	0,02
	30	0,08	-0,12	-0,03	-0,02	-0,10	-0,10	-0,10	0,01	-0,12	-0,10
	40	0,03	0,01	0,05	0,00	0,00	0,00	0,01	0,04	0,01	-0,01
	50	0,06	0,02	0,04	0,00	0,00	0,00	0,00	0,05	0,02	0,00
Media 200		0,05	0,00	0,02	0,00	-0,01	-0,01	-0,01	0,03	0,00	-0,02
500	10	0,05	0,04	0,01	0,01	0,01	0,01	0,00	0,03	0,04	0,00
	20	0,08	0,03	0,01	0,01	0,00	0,00	0,01	0,06	0,03	0,01
	30	0,02	0,01	0,11	0,13	0,02	0,02	0,02	0,01	0,01	0,02
	40	0,03	0,01	0,08	0,07	0,02	0,02	0,02	0,02	0,01	0,02
	50	0,08	0,04	0,01	0,00	0,01	0,03	0,02	0,06	0,04	0,02
Media 500		0,05	0,03	0,04	0,05	0,01	0,01	0,01	0,04	0,03	0,01
1000	10	0,06	0,03	0,01	0,02	0,01	0,01	0,01	0,03	0,03	0,01
	20	0,10	0,04	0,02	0,02	0,00	0,00	0,00	0,08	0,04	0,00
	30	0,13	0,05	0,13	0,13	0,05	0,05	0,05	0,06	0,04	0,05
	40	0,07	0,04	0,05	0,05	0,00	0,00	0,00	0,07	0,04	0,01
	50	0,06	0,04	0,04	0,05	0,00	0,00	0,00	0,06	0,03	0,01
Media 1000		0,08	0,04	0,05	0,05	0,01	0,01	0,01	0,06	0,04	0,02
Media general		0,06	0,02	0,04	0,03	0,00	0,01	0,00	0,04	0,02	0,00

Tabla D.37 – Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo U(1000,1100) de los algoritmos de reducción de tamaño para 15 segundos como tiempo de parada.

n	m	CPLEX	DIG	3J	3J2M	3JD	2JDi	2JDi(90)	M-CPLEX	M-DIG	M-2JDi(90)
100	10	0,02	0,02	0,02	0,03	0,01	0,02	0,01	0,01	0,01	0,02
	20	0,01	0,01	0,02	0,00	0,00	0,01	0,01	0,01	0,01	0,01
	30	0,06	-0,01	0,04	0,05	-0,02	-0,01	-0,01	0,01	-0,02	-0,02
	40	0,07	-0,01	0,05	0,05	0,00	0,00	0,00	0,03	-0,01	0,00
	50	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,01	0,00	0,00
Media 100		0,03	0,00	0,03	0,03	0,00	0,00	0,00	0,02	0,00	0,00
200	10	0,02	0,03	0,02	0,02	0,01	0,01	0,01	0,02	0,03	0,01
	20	0,03	0,04	0,01	0,01	0,02	0,02	0,03	0,03	0,04	0,02
	30	0,05	-0,12	-0,03	-0,02	-0,10	-0,10	-0,10	0,00	-0,12	-0,10
	40	0,02	0,01	0,05	0,00	-0,01	0,00	0,01	0,02	0,01	-0,01
	50	0,02	0,01	0,04	0,00	-0,01	0,00	0,00	0,03	0,01	-0,01
Media 200		0,03	-0,01	0,02	0,00	-0,02	-0,01	-0,01	0,02	-0,01	-0,02
500	10	0,02	0,03	0,00	0,01	0,01	0,00	0,00	0,01	0,03	0,00
	20	0,06	0,03	0,00	0,00	0,00	0,00	0,00	0,03	0,03	0,00
	30	0,01	0,01	0,11	0,12	0,02	0,02	0,02	0,01	0,01	0,02
	40	0,02	0,01	0,08	0,07	0,02	0,02	0,02	0,01	0,01	0,02
	50	0,07	0,03	0,00	-0,01	0,00	0,02	0,01	0,05	0,03	0,01
Media 500		0,04	0,02	0,04	0,04	0,01	0,01	0,01	0,02	0,02	0,01
1000	10	0,04	0,03	0,01	0,01	0,00	0,01	0,00	0,02	0,03	0,00
	20	0,09	0,04	0,01	0,01	0,00	0,00	0,00	0,06	0,04	0,00
	30	0,10	0,04	0,12	0,12	0,05	0,05	0,05	0,06	0,04	0,05
	40	0,07	0,03	0,01	0,01	-0,01	-0,01	-0,01	0,06	0,03	0,01
	50	0,06	0,03	0,00	0,00	0,00	0,00	-0,01	0,06	0,03	0,01
Media 1000		0,07	0,03	0,03	0,03	0,01	0,01	0,01	0,05	0,03	0,01
Media general		0,04	0,01	0,03	0,02	0,00	0,00	0,00	0,03	0,01	0,00

Tabla D.38 – Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo U(1000,1100) de los algoritmos de reducción de tamaño para 30 segundos como tiempo de parada.

n	m	CPLEX	DIG	3J	3J2M	3JD	2JDi	2JDi(90)	M-CPLEX	M-DIG	M-2JDi(90)
100	10	0,02	0,01	0,02	0,02	0,01	0,01	0,01	0,01	0,01	0,02
	20	0,01	0,01	0,02	0,00	0,00	0,01	0,01	0,00	0,01	0,00
	30	0,03	-0,02	0,03	0,04	-0,02	-0,01	-0,02	0,00	-0,02	-0,02
	40	0,04	-0,01	0,05	0,04	0,00	0,00	0,00	0,02	-0,01	0,00
	50	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,01	0,00	0,00
Media 100		0,02	0,00	0,02	0,02	0,00	0,00	0,00	0,01	0,00	0,00
200	10	0,01	0,03	0,02	0,01	0,01	0,01	0,01	0,02	0,02	0,01
	20	0,02	0,04	0,01	0,01	0,01	0,02	0,02	0,02	0,04	0,01
	30	0,05	-0,13	-0,03	-0,02	-0,10	-0,10	-0,10	-0,01	-0,13	-0,10
	40	0,01	0,00	0,04	0,00	-0,01	0,00	0,00	0,00	0,00	-0,01
	50	0,01	0,00	0,04	0,00	-0,01	0,00	0,00	0,00	0,00	-0,01
Media 200		0,02	-0,01	0,02	0,00	-0,02	-0,01	-0,02	0,01	-0,01	-0,02
500	10	0,01	0,03	0,00	0,01	0,00	0,00	0,00	0,00	0,03	0,00
	20	0,03	0,03	0,00	0,00	0,00	-0,01	0,00	0,01	0,03	0,00
	30	0,01	0,01	0,10	0,09	0,02	0,01	0,01	0,01	0,01	0,02
	40	0,01	0,00	0,06	0,06	0,02	0,02	0,02	0,01	0,00	0,02
	50	0,05	0,02	0,00	-0,01	0,00	0,01	0,00	0,03	0,02	0,00
Media 500		0,02	0,02	0,03	0,03	0,01	0,01	0,01	0,01	0,02	0,01
1000	10	0,03	0,02	0,00	0,00	0,00	0,00	0,00	0,01	0,02	0,00
	20	0,06	0,03	0,00	0,00	-0,01	0,00	0,00	0,05	0,03	-0,01
	30	0,06	0,04	0,12	0,11	0,05	0,05	0,05	0,05	0,03	0,05
	40	0,07	0,03	0,00	0,00	-0,01	-0,01	-0,01	0,04	0,02	-0,01
	50	0,06	0,02	0,00	0,00	-0,01	0,00	-0,01	0,04	0,02	0,00
Media 1000		0,05	0,03	0,02	0,02	0,01	0,01	0,01	0,04	0,03	0,01
Media general		0,03	0,01	0,02	0,02	0,00	0,00	0,00	0,02	0,01	0,00

Tabla D.39 – Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo U(1000,1100) de los algoritmos de reducción de tamaño para 60 segundos como tiempo de parada.

n	m	CPLEX	DIG	3J	3J2M	3JD	2JDi	2JDi(90)	M-CPLEX	M-DIG	M-2JDi(90)
100	10	0,01	0,01	0,01	0,01	0,01	0,01	0,00	0,01	0,00	0,00
	20	0,01	0,01	0,02	0,00	0,00	0,01	0,00	0,00	0,01	0,00
	30	0,02	-0,02	0,02	0,02	-0,02	-0,01	-0,02	-0,01	-0,02	-0,02
	40	0,03	-0,01	0,03	0,04	0,00	0,00	0,00	0,01	-0,01	0,00
	50	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,01	0,00	0,00
Media 100		0,01	0,00	0,02	0,02	0,00	0,00	0,00	0,00	0,00	-0,01
200	10	0,01	0,03	0,01	0,01	0,01	0,01	0,01	0,01	0,02	0,01
	20	0,01	0,03	0,01	0,01	0,01	0,01	0,00	0,02	0,03	0,00
	30	0,02	-0,13	-0,03	-0,02	-0,10	-0,10	-0,10	-0,05	-0,13	-0,10
	40	0,00	-0,01	0,04	-0,01	-0,01	-0,01	-0,01	0,00	-0,01	-0,01
	50	0,00	0,00	0,04	0,00	-0,01	0,00	-0,01	0,00	0,00	-0,01
Media 200		0,01	-0,02	0,01	0,00	-0,02	-0,02	-0,02	-0,01	-0,02	-0,02
500	10	0,01	0,02	0,00	0,01	0,00	0,00	0,00	0,00	0,02	0,00
	20	0,01	0,02	0,00	0,00	0,00	-0,01	-0,01	0,01	0,02	-0,01
	30	0,01	0,00	0,10	0,07	0,02	0,01	0,01	0,01	0,00	0,02
	40	0,01	0,00	0,06	0,06	0,02	0,02	0,02	0,00	0,00	0,02
	50	0,03	0,02	-0,01	-0,01	0,00	0,00	-0,01	0,02	0,02	-0,01
Media 500		0,01	0,01	0,03	0,02	0,01	0,01	0,00	0,01	0,01	0,00
1000	10	0,00	0,02	0,00	0,00	0,00	0,00	0,00	0,00	0,02	0,00
	20	0,05	0,03	0,00	0,00	-0,01	0,00	0,00	0,04	0,03	-0,01
	30	0,05	0,03	0,11	0,09	0,05	0,04	0,05	0,03	0,03	0,05
	40	0,05	0,02	-0,01	-0,01	-0,01	-0,01	-0,01	0,03	0,02	-0,01
	50	0,05	0,02	-0,01	-0,01	-0,01	-0,01	-0,01	0,03	0,02	-0,01
Media 1000		0,04	0,02	0,02	0,01	0,01	0,01	0,01	0,03	0,02	0,00
Media general		0,02	0,00	0,02	0,01	0,00	0,00	0,00	0,01	0,00	-0,01

Tabla D.40 – Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo U(1000,1100) de los algoritmos de reducción de tamaño para 120 segundos como tiempo de parada.

n	m	CPLEX	DIG	3J	3J2M	3JD	2JDi	2JDi(90)	M-CPLEX	M-DIG	M-2JDi(90)
100	10	0,00	0,01	0,01	0,01	0,01	0,01	0,00	0,01	0,00	0,00
	20	0,00	0,01	0,02	0,00	0,00	0,01	0,00	0,00	0,00	0,00
	30	0,01	-0,02	0,01	0,02	-0,02	-0,01	-0,02	-0,01	-0,02	-0,02
	40	0,01	-0,02	0,03	0,04	0,00	0,00	0,00	0,01	-0,02	-0,01
	50	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,01	0,00	0,00
Media 100		0,01	-0,01	0,02	0,01	0,00	0,00	0,00	0,00	-0,01	-0,01
200	10	0,01	0,03	0,01	0,01	0,01	0,01	0,00	0,01	0,02	0,00
	20	0,01	0,02	0,00	0,00	0,01	0,01	0,00	0,01	0,02	0,00
	30	-0,03	-0,14	-0,03	-0,02	-0,10	-0,10	-0,10	-0,07	-0,14	-0,10
	40	-0,01	-0,01	0,04	-0,01	-0,01	-0,01	-0,01	0,00	-0,01	-0,01
	50	0,00	0,00	0,04	0,00	-0,01	0,00	-0,01	0,00	0,00	-0,01
Media 200		0,00	-0,02	0,01	0,00	-0,02	-0,02	-0,03	-0,01	-0,02	-0,03
500	10	0,00	0,02	0,00	0,00	0,00	0,00	0,00	0,00	0,02	0,00
	20	0,01	0,02	0,00	0,00	0,00	-0,01	-0,01	0,01	0,02	-0,01
	30	0,00	-0,01	0,09	0,05	0,02	0,01	0,01	0,01	-0,01	0,01
	40	0,01	-0,01	0,05	0,06	0,02	0,01	0,02	0,00	-0,01	0,01
	50	0,02	0,02	-0,01	-0,01	0,00	0,00	-0,01	0,01	0,02	-0,01
Media 500		0,01	0,01	0,03	0,02	0,01	0,00	0,00	0,01	0,01	0,00
1000	10	0,00	0,02	0,00	0,00	0,00	0,00	0,00	0,00	0,02	0,00
	20	0,02	0,02	0,00	-0,01	-0,01	0,00	-0,01	0,01	0,02	-0,01
	30	0,03	0,03	0,10	0,09	0,05	0,04	0,04	0,02	0,02	0,05
	40	0,04	0,02	-0,01	-0,01	-0,01	-0,01	-0,01	0,02	0,02	-0,01
	50	0,04	0,02	-0,01	-0,01	-0,01	-0,01	-0,01	0,02	0,01	-0,01
Media 1000		0,03	0,02	0,02	0,01	0,01	0,00	0,00	0,02	0,02	0,00
Media general		0,01	0,00	0,02	0,01	0,00	0,00	-0,01	0,00	0,00	-0,01

Tabla D.41 – Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo U(1000,1100) de los algoritmos de reducción de tamaño para 240 segundos como tiempo de parada.

n	m	CPLEX	DIG	3J	3J2M	3JD	2JDi	2JDi(90)	M-CPLEX	M-DIG	M-2JDi(90)
100	10	0,00	0,00	0,01	0,01	0,00	0,00	0,00	0,00	0,00	0,00
	20	0,00	0,01	0,02	0,00	0,00	0,01	0,00	0,00	0,00	0,00
	30	0,00	-0,02	0,01	0,02	-0,02	-0,02	-0,02	-0,01	-0,02	-0,02
	40	0,01	-0,02	0,03	0,03	-0,01	0,00	0,00	0,01	-0,02	-0,01
	50	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
Media 100		0,00	-0,01	0,01	0,01	0,00	0,00	0,00	0,00	-0,01	-0,01
200	10	0,00	0,02	0,01	0,01	0,01	0,01	0,00	0,01	0,02	0,00
	20	0,01	0,02	0,00	0,00	0,00	0,00	0,00	0,01	0,02	0,00
	30	-0,06	-0,14	-0,03	-0,02	-0,10	-0,10	-0,10	-0,09	-0,14	-0,10
	40	-0,01	-0,01	0,03	-0,01	-0,01	-0,01	-0,01	0,00	-0,01	-0,01
	50	0,00	0,00	0,04	-0,01	-0,01	0,00	-0,01	0,00	0,00	-0,01
Media 200		-0,01	-0,02	0,01	-0,01	-0,02	-0,02	-0,03	-0,02	-0,02	-0,03
500	10	0,00	0,02	0,00	0,00	0,00	0,00	0,00	0,00	0,02	0,00
	20	0,00	0,02	0,00	-0,01	-0,01	-0,01	-0,01	0,01	0,02	-0,01
	30	0,00	-0,01	0,05	0,02	0,01	0,01	0,01	0,01	-0,01	0,01
	40	0,00	-0,01	0,05	0,05	0,02	0,01	0,02	0,00	-0,01	0,01
	50	0,02	0,02	-0,01	-0,01	0,00	-0,01	-0,01	0,01	0,02	-0,01
Media 500		0,01	0,01	0,02	0,01	0,00	0,00	0,00	0,00	0,01	0,00
1000	10	0,00	0,02	0,00	0,00	0,00	0,00	0,00	0,00	0,02	0,00
	20	0,02	0,02	0,00	-0,01	-0,01	0,00	-0,01	0,01	0,02	-0,01
	30	0,03	0,03	0,09	0,08	0,05	0,03	0,04	0,02	0,02	0,04
	40	0,03	0,02	-0,01	-0,01	-0,01	-0,01	-0,01	0,01	0,02	-0,01
	50	0,03	0,01	-0,01	-0,01	-0,01	-0,01	-0,01	0,02	0,01	-0,01
Media 1000		0,02	0,02	0,01	0,01	0,01	0,00	0,00	0,01	0,02	0,00
Media general		0,01	0,00	0,01	0,01	0,00	0,00	-0,01	0,00	0,00	-0,01

Tabla D.42 – Tabla porcentual de la media de las diferencias respecto a la referencia para el intervalo U(1000,1100) de los algoritmos de reducción de tamaño para 300 segundos como tiempo de parada.

D.2. ANOVAS de algoritmos de reducción de tamaño

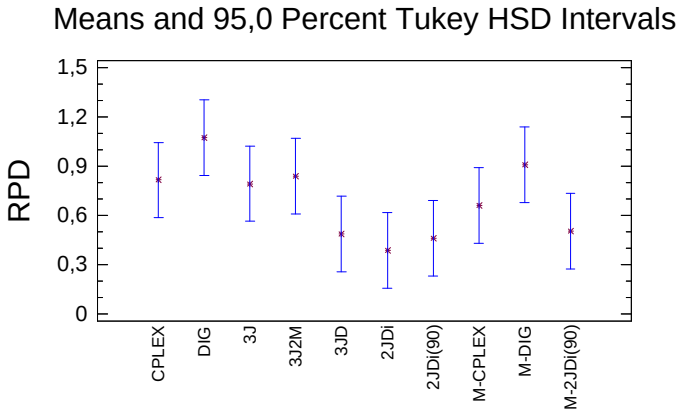

Figura D.1 – Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo U(1,100) y 15 segundos.

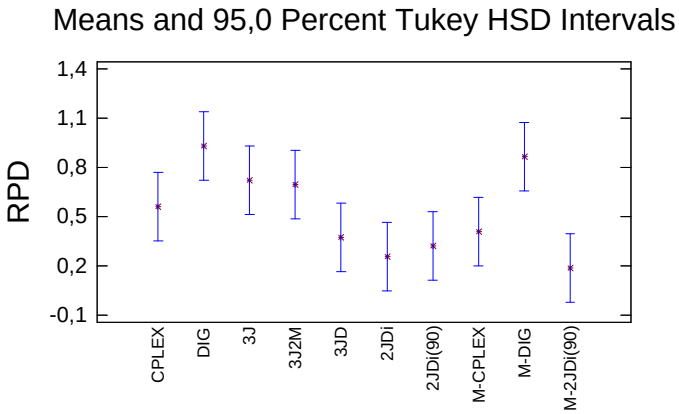

Figura D.2 – Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo U(1,100) y 30 segundos.

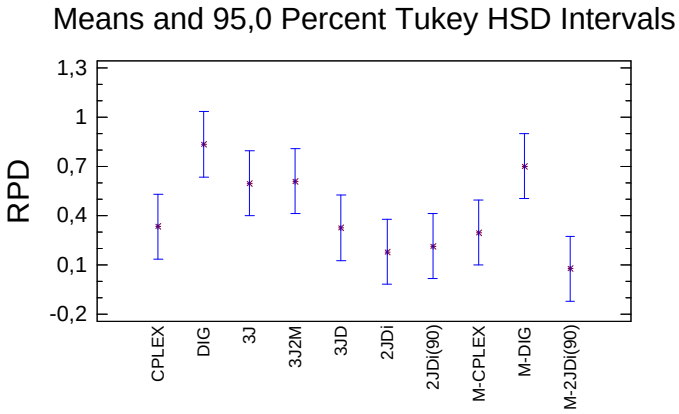

Figura D.3 – Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo U(1,100) y 60 segundos.

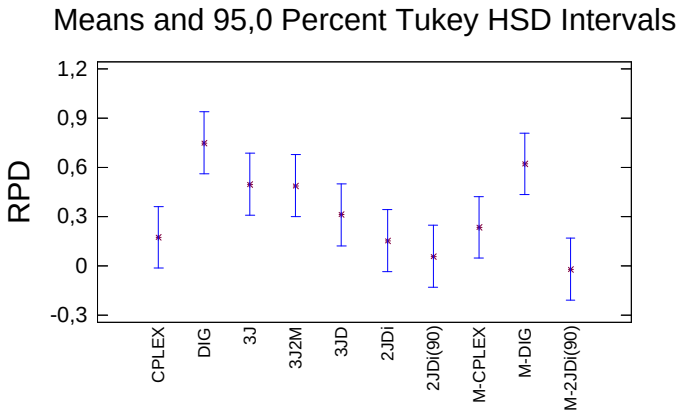

Figura D.4 – Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo U(1,100) y 120 segundos.

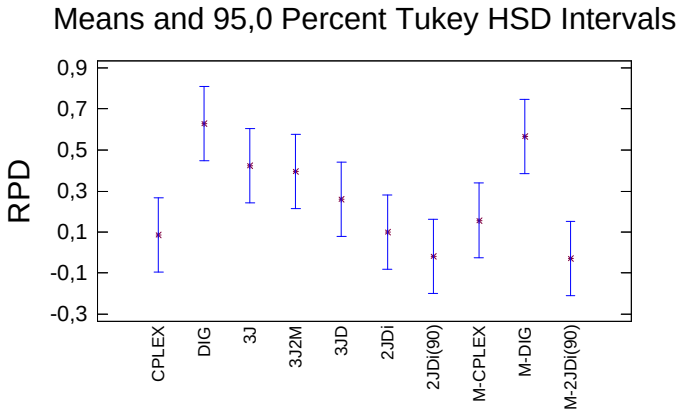

Figura D.5 – Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo U(1,100) y 240 segundos.

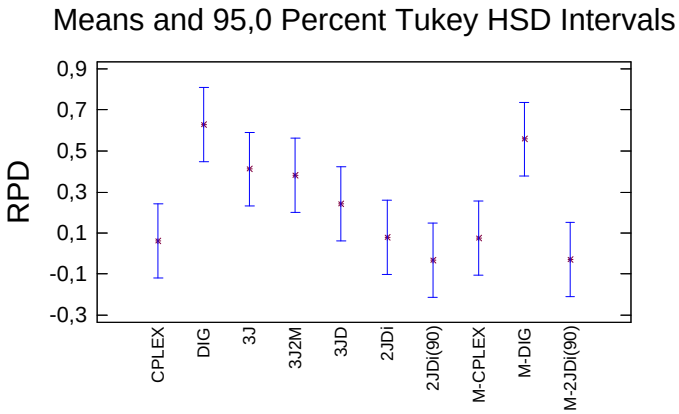

Figura D.6 – Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo U(1,100) y 300 segundos.

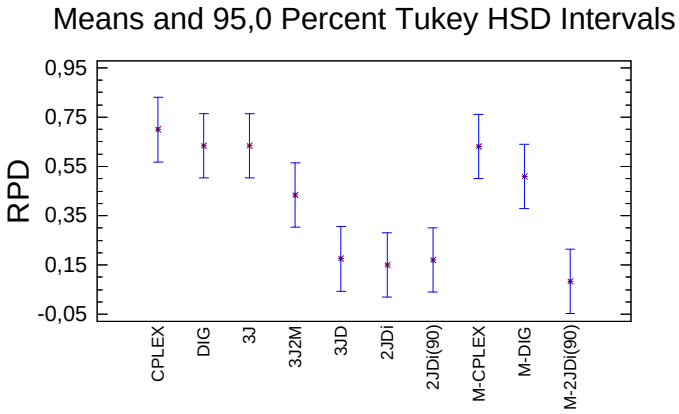

Figura D.7 – Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo U(10,100) y 15 segundos.

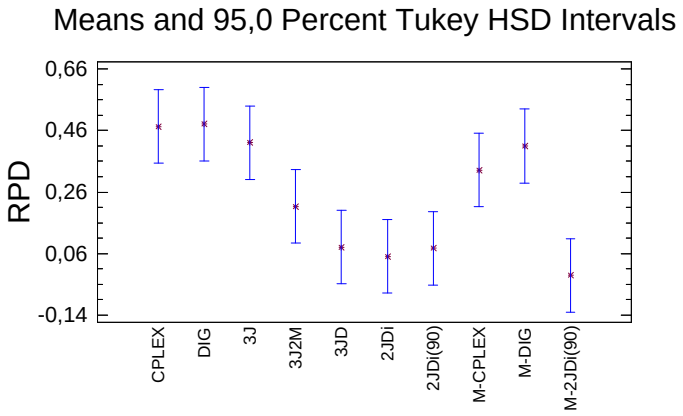

Figura D.8 – Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo U(10,100) y 30 segundos.

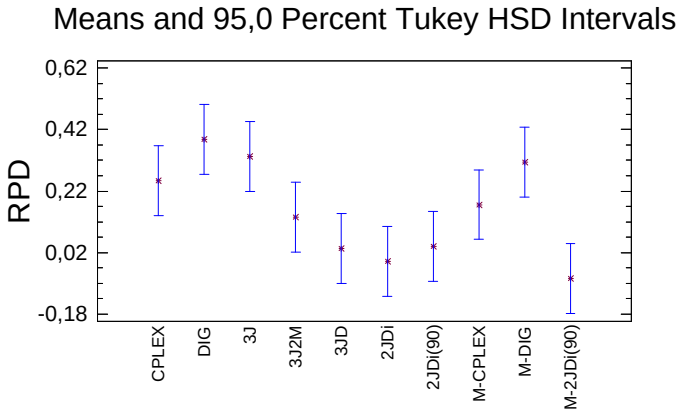

Figura D.9 – Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo U(10,100) y 60 segundos.

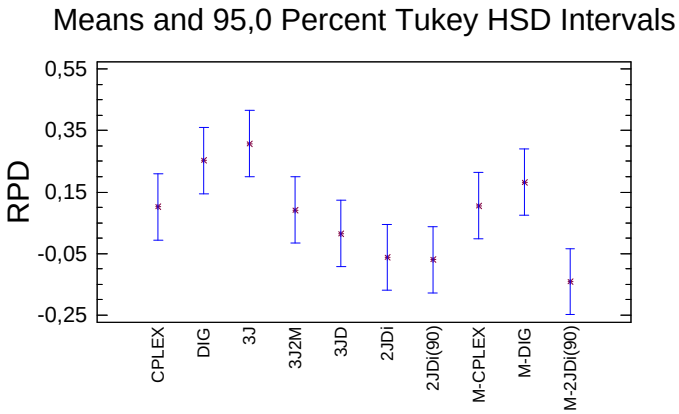

Figura D.10 – Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo U(10,100) y 120 segundos.

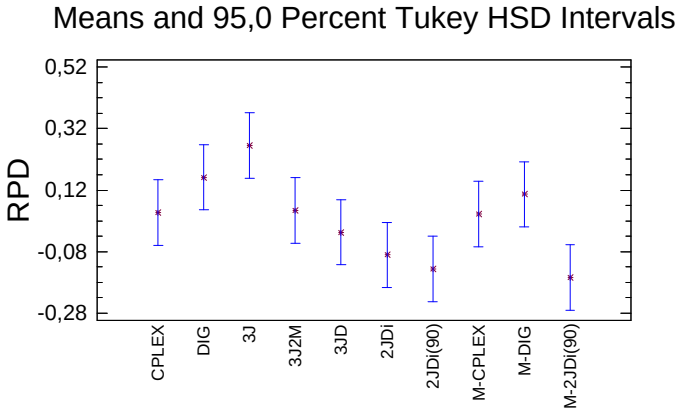

Figura D.11 – Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo U(10,100) y 240 segundos.

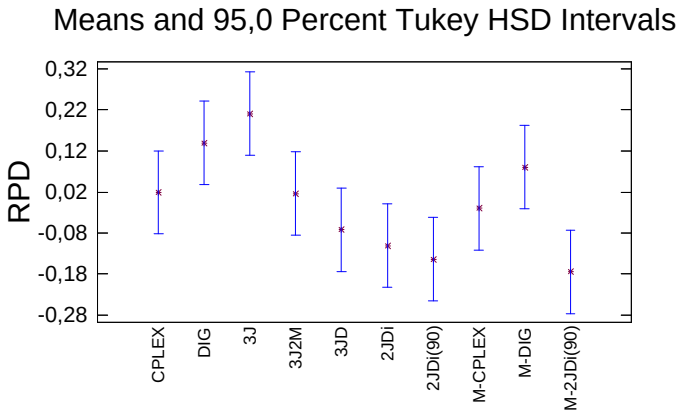

Figura D.12 – Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo U(10,100) y 300 segundos.

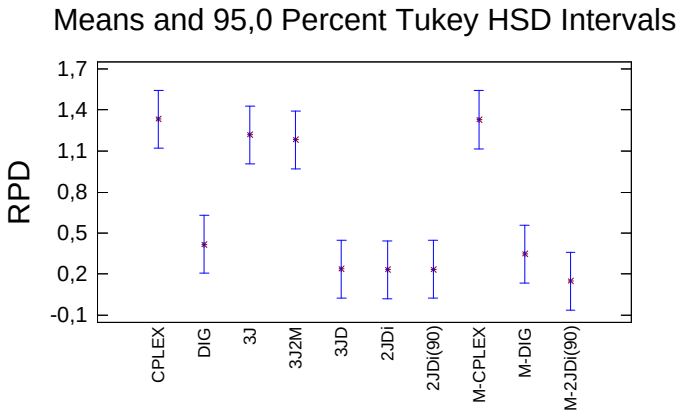

Figura D.13 – Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo Trabajos correlacionados y 15 segundos.

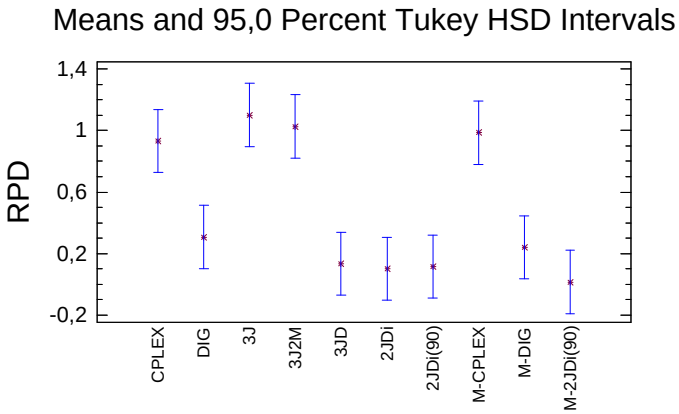

Figura D.14 – Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo Trabajos correlacionados y 30 segundos.

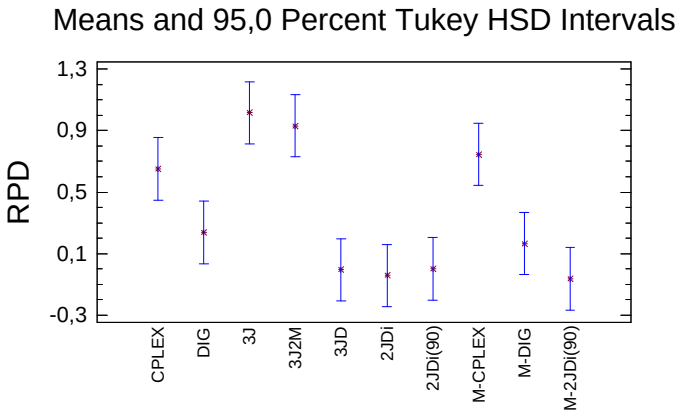

Figura D.15 – Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo Trabajos correlacionados y 60 segundos.

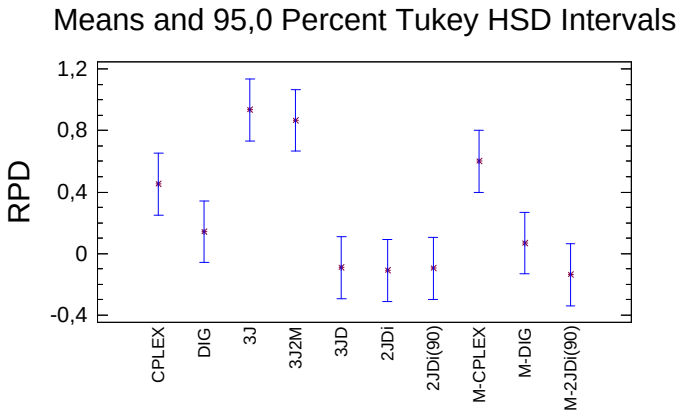

Figura D.16 – Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo Trabajos correlacionados y 120 segundos.

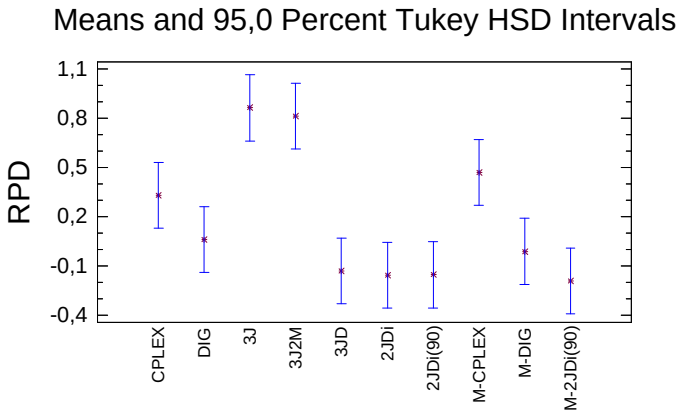

Figura D.17 – Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo Trabajos correlacionados y 240 segundos.

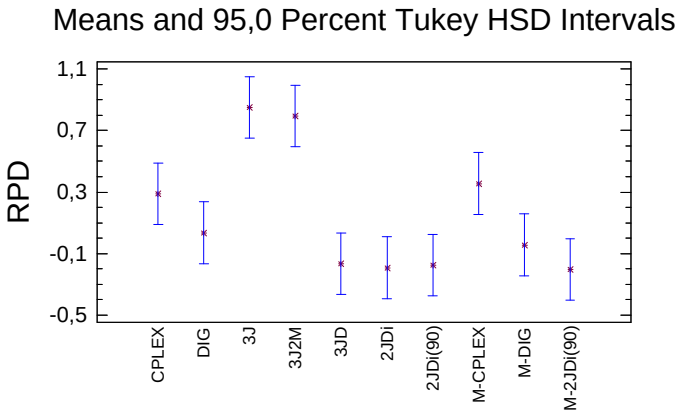

Figura D.18 – Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo Trabajos correlacionados y 300 segundos.

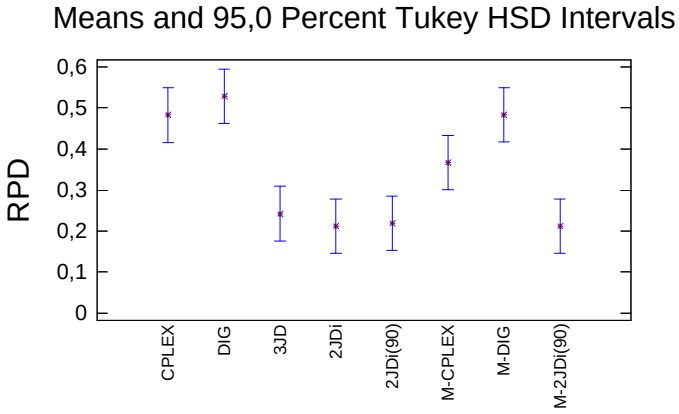

Figura D.19 – Gráfico de medias de los algoritmos de reducción de tamaño (excepto 3J y 3J2M) para el intervalo Máquinas correlacionadas y 15 segundos.

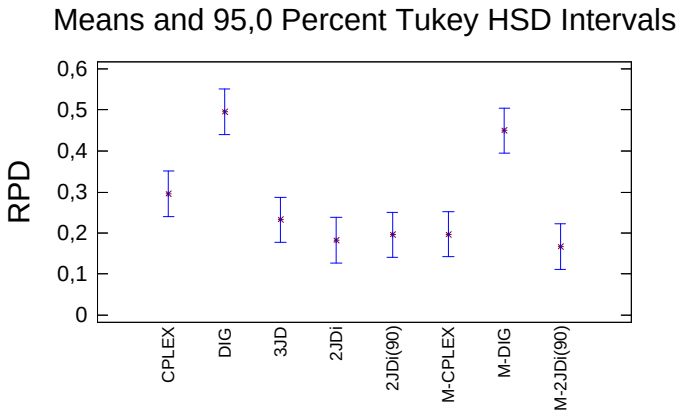

Figura D.20 – Gráfico de medias de los algoritmos de reducción de tamaño (excepto 3J y 3J2M) para el intervalo Máquinas correlacionadas y 30 segundos.

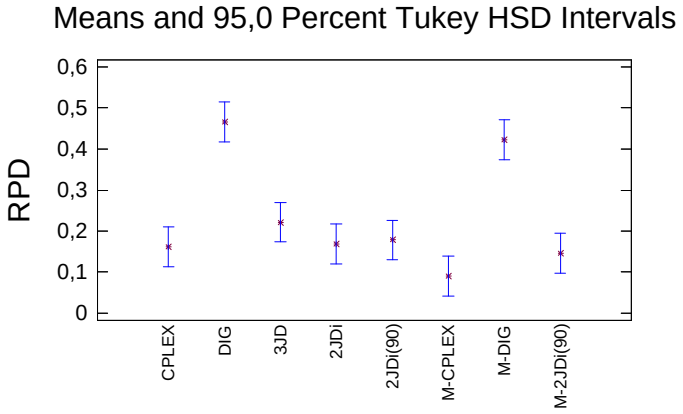

Figura D.21 – Gráfico de medias de los algoritmos de reducción de tamaño (excepto 3J y 3J2M) para el intervalo Máquinas correlacionadas y 60 segundos.

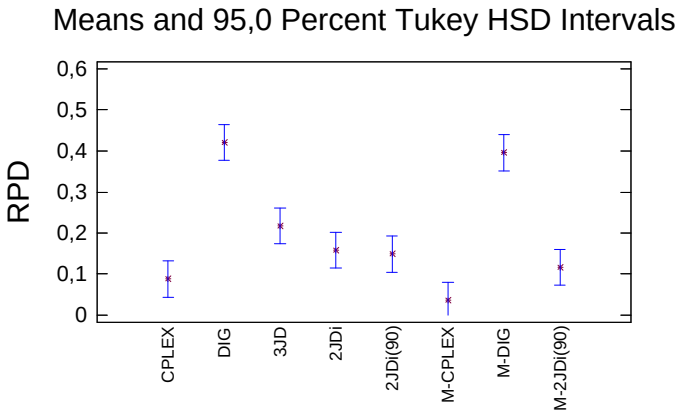

Figura D.22 – Gráfico de medias de los algoritmos de reducción de tamaño (excepto 3J y 3J2M) para el intervalo Máquinas correlacionadas y 120 segundos.

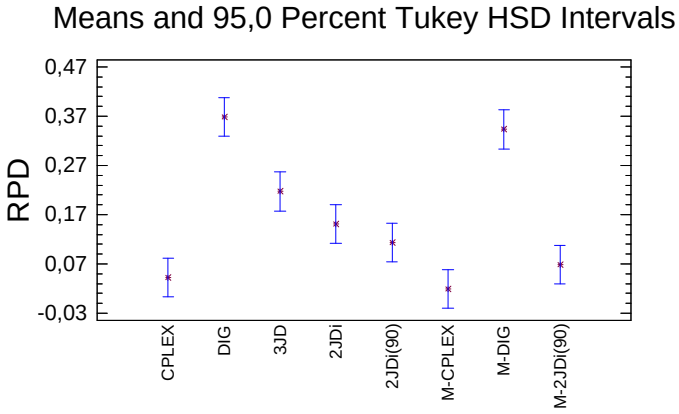

Figura D.23 – Gráfico de medias de los algoritmos de reducción de tamaño (excepto 3J y 3J2M) para el intervalo Máquinas correlacionadas y 240 segundos.

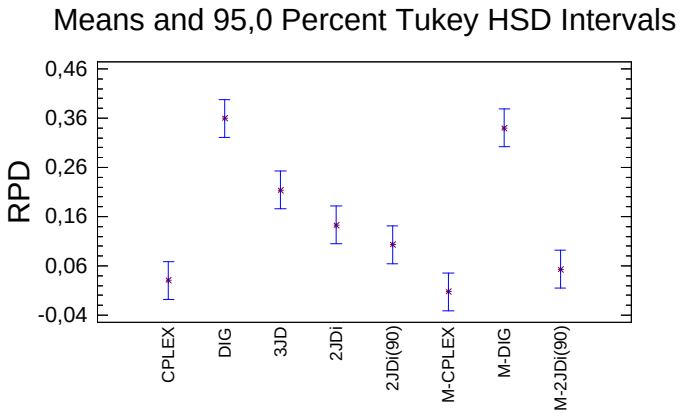

Figura D.24 – Gráfico de medias de los algoritmos de reducción de tamaño (excepto 3J y 3J2M) para el intervalo Máquinas correlacionadas y 300 segundos.

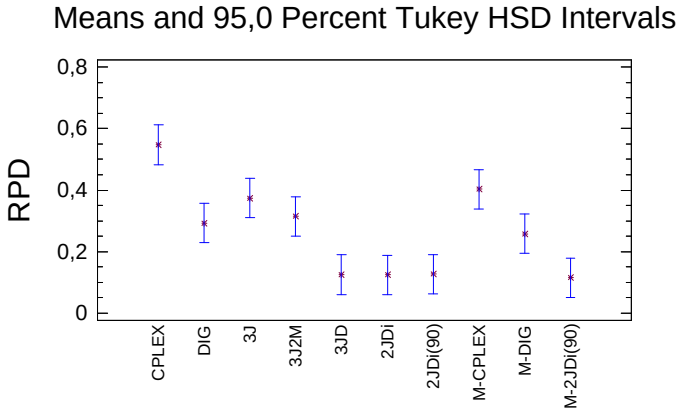

Figura D.25 – Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo U(100,200) y 15 segundos.

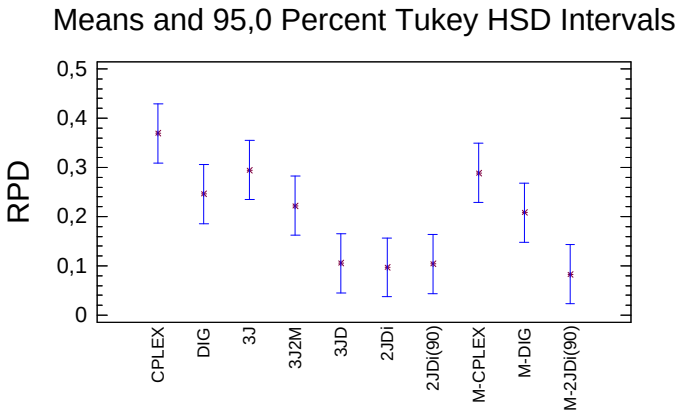

Figura D.26 – Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo U(100,200) y 30 segundos.

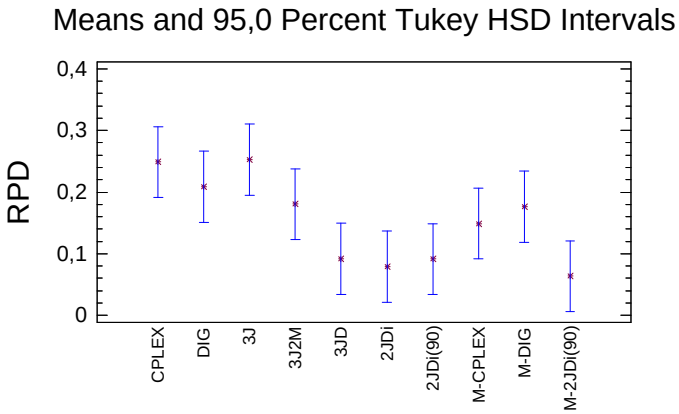

Figura D.27 – Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo U(100,200) y 60 segundos.

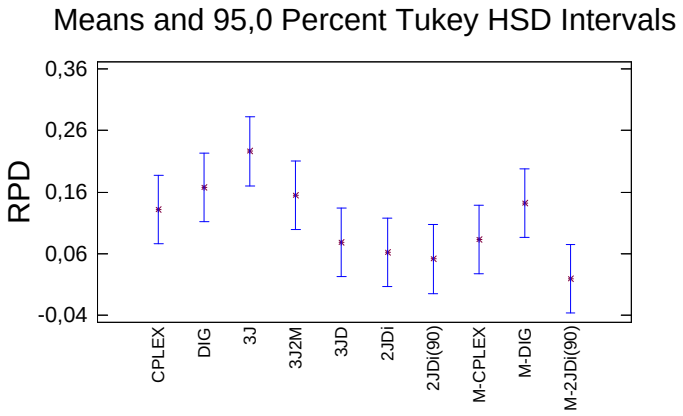

Figura D.28 – Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo U(100,200) y 120 segundos.

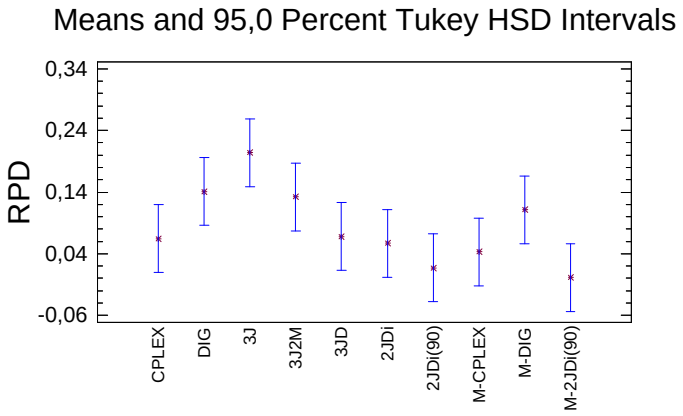

Figura D.29 – Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo U(100,200) y 240 segundos.

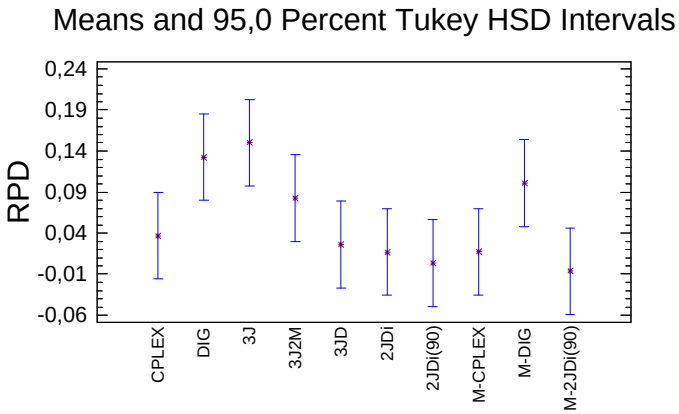

Figura D.30 – Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo U(100,200) y 300 segundos.

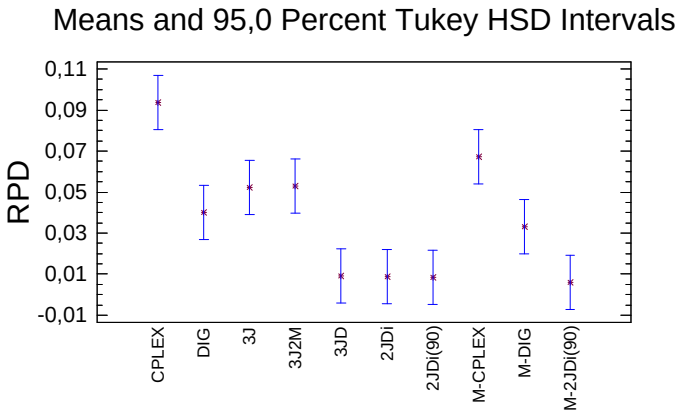

Figura D.31 – Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo U(100,120) y 15 segundos.

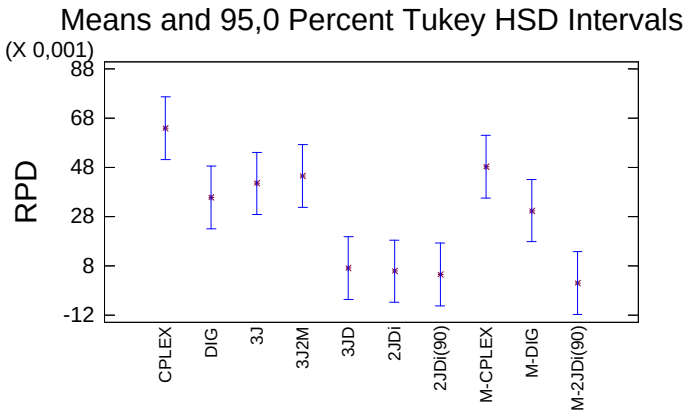

Figura D.32 – Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo U(100,120) y 30 segundos.

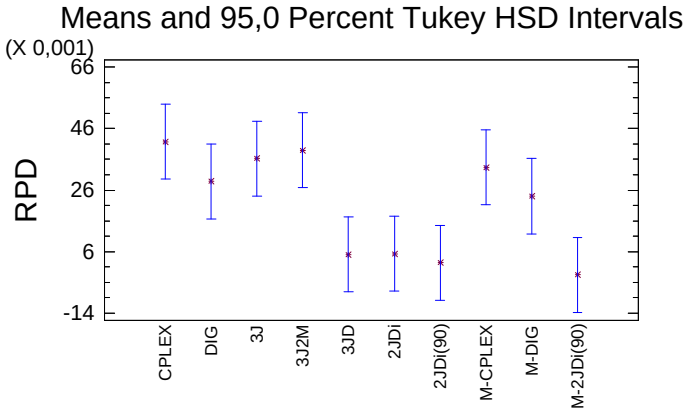

Figura D.33 – Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo U(100,120) y 60 segundos.

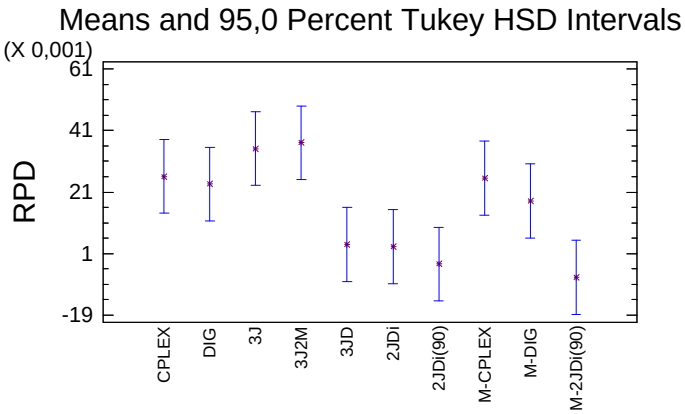

Figura D.34 – Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo U(100,120) y 120 segundos.

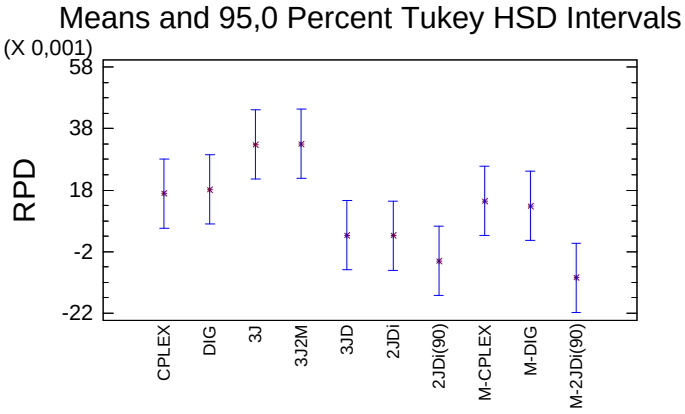

Figura D.35 – Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo U(100,120) y 240 segundos.

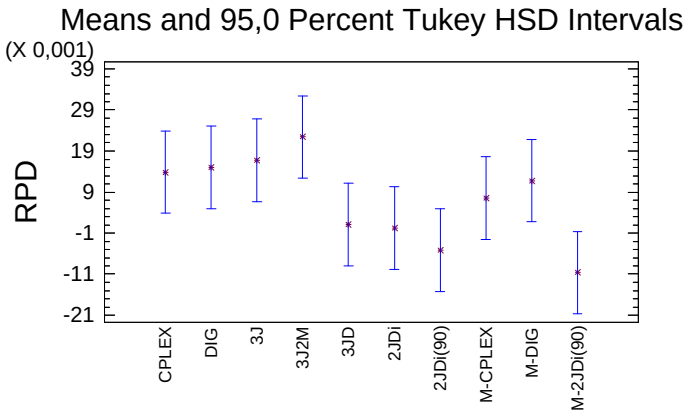

Figura D.36 – Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo U(100,120) y 300 segundos.

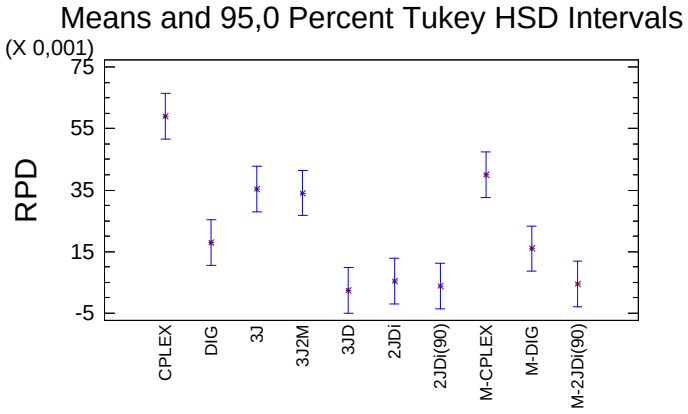

Figura D.37 – Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo U(1000,1100) y 15 segundos.

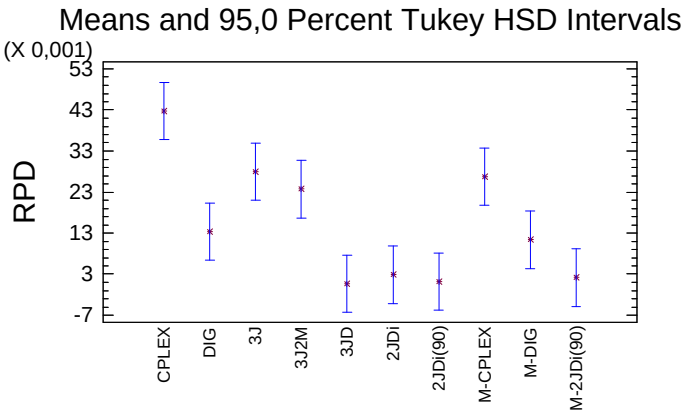

Figura D.38 – Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo U(1000,1100) y 30 segundos.

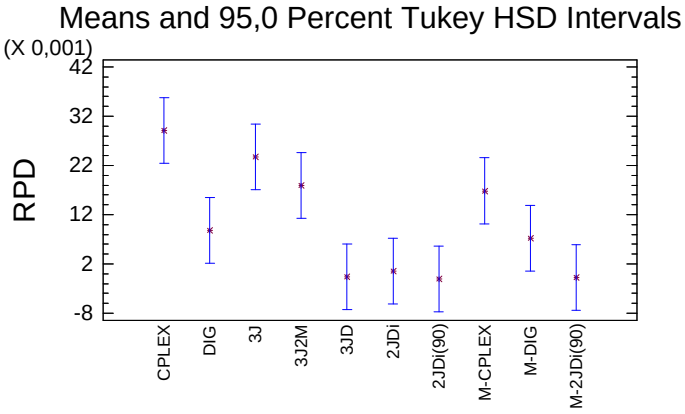

Figura D.39 – Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo U(1000,1100) y 60 segundos.

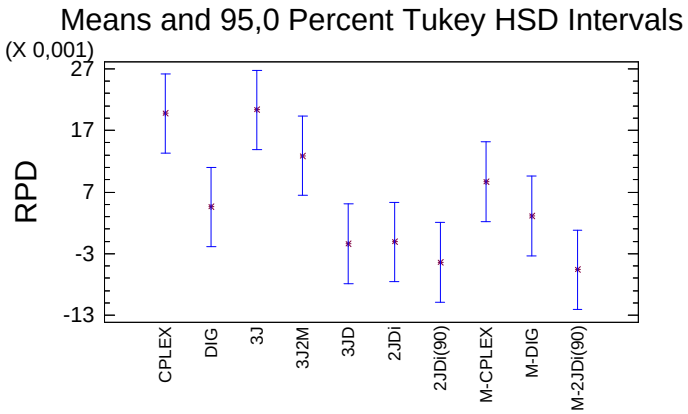

Figura D.40 – Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo U(1000,1100) y 120 segundos.

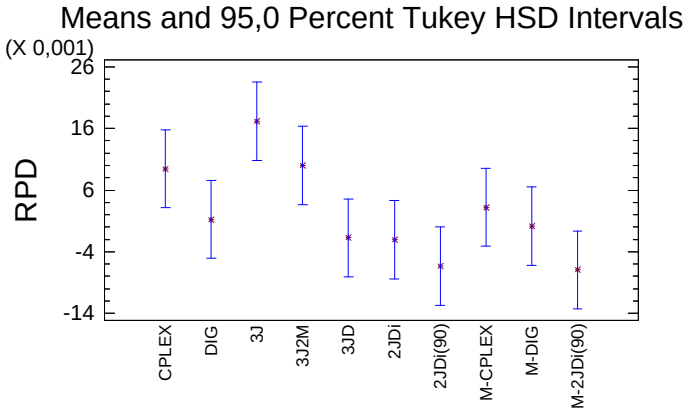

Figura D.41 – Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo U(1000,1100) y 240 segundos.

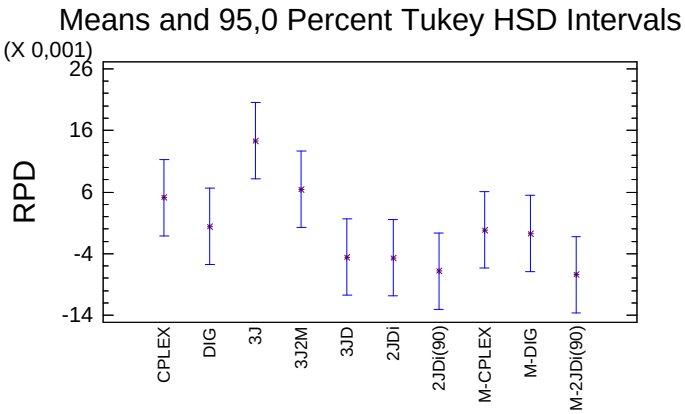

Figura D.42 – Gráfico de medias de los algoritmos de reducción de tamaño para el intervalo U(1000,1100) y 300 segundos.

ANEXO E

TABLAS Y ANOVAS PARA EL PROBLEMA DE MÁQUINAS OPCIONALES

En este anexo presentaremos todas las tablas porcentuales de las diferencias con los mejores resultados obtenidos, así como sus ANOVAS correspondientes. Todo ello para los siete intervalos de tiempos de proceso estudiados estudiados, a saber U(1,100), U(10,100), trabajos correlacionados, máquinas correlacionadas, U(100,200), U(100,120) y U(1000,1100), para los tiempos de ejecución de 300 segundos y 60 segundos y para los porcentajes de máquinas sin usar de 20 %, 50 % y 80 %.

E.1. Tablas para un 50 % de máquinas sin usar

n	m	CPLEX	NAM+CPLEX	NAM+ST+DIG	NAM+ST+CPLEX
100	10	0,00	0,00	0,06	0,00
100	20	2,28	0,49	3,34	0,79
100	30	2,67	0,53	4,31	1,05
100	40	2,42	1,93	3,21	0,43
100	50	1,96	5,40	9,45	4,01
200	10	0,00	0,00	0,27	0,00
200	20	3,73	0,12	1,04	0,18
200	30	7,77	0,27	2,00	0,13
200	40	9,97	1,80	1,83	0,00
200	50	11,80	1,37	3,03	0,00
500	10	0,32	0,00	0,02	0,00
500	20	4,42	0,26	0,16	0,05
500	30	7,91	1,52	0,81	0,05
500	40	9,64	2,39	0,79	0,00
500	50	10,71	2,88	1,30	0,00
1000	10	0,67	0,00	0,02	0,00
1000	20	3,92	0,11	0,29	0,06
1000	30	6,73	0,27	0,71	0,12
1000	40	6,81	0,66	0,41	0,00
1000	50	8,49	1,01	0,51	0,00
Media		5,11	1,05	1,68	0,34

Tabla E.1 – Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo U(1,100) y 50 % de máquinas quitadas con 300 segundos.

n	m	CPLEX	NAM+CPLEX	NAM+ST+DIG	NAM+ST+CPLEX
100	10	0,00	0,00	0,06	0,00
100	20	5,79	0,62	3,34	1,01
100	30	7,41	3,84	4,31	1,05
100	40	9,52	8,09	3,21	0,99
100	50	11,24	8,75	9,45	4,73
200	10	0,07	0,00	0,27	0,00
200	20	7,25	1,49	1,04	0,30
200	30	11,31	2,44	2,00	0,92
200	40	16,62	5,28	1,83	1,11
200	50	19,06	5,06	3,03	1,72
500	10	2,95	0,00	0,02	0,00
500	20	7,21	0,74	0,16	0,16
500	30	11,06	3,24	0,81	1,48
500	40	13,20	3,80	0,79	2,03
500	50	24,79	4,19	1,30	2,11
1000	10	1,85	0,00	0,02	0,00
1000	20	6,65	0,91	0,29	0,77
1000	30	15,38	1,19	0,71	1,07
1000	40	27,68	1,65	0,41	0,66
1000	50	58,42	1,53	0,51	0,63
Media		12,87	2,64	1,68	1,04

Tabla E.2 – Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo U(1,100) y 50 % de máquinas quitadas con 60 segundos.

n	m	CPLEX	NAM+CPLEX	NAM+ST+DIG	NAM+ST+CPLEX
100	10	0,00	0,00	0,11	0,00
100	20	2,25	0,00	0,86	0,06
100	30	3,61	0,10	1,82	0,21
100	40	3,75	0,76	1,21	0,30
100	50	4,55	0,94	1,51	0,00
200	10	0,01	0,00	0,14	0,00
200	20	1,95	0,30	0,18	0,09
200	30	4,28	1,32	0,53	0,00
200	40	4,51	1,80	0,38	0,15
200	50	5,53	0,79	0,00	0,79
500	10	0,18	0,00	0,07	0,00
500	20	2,58	0,18	0,10	0,03
500	30	3,29	0,76	0,12	0,04
500	40	4,81	0,87	0,18	0,03
500	50	5,14	0,44	0,08	0,16
1000	10	1,05	0,00	0,03	0,00
1000	20	2,36	0,10	0,09	0,01
1000	30	2,97	0,65	0,39	0,00
1000	40	3,36	0,55	0,27	0,00
1000	50	2,56	0,26	0,12	0,04
Media		2,94	0,49	0,41	0,10

Tabla E.3 – Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo U(10,100) y 50 % de máquinas quitadas con 300 segundos.

n	m	CPLEX	NAM+CPLEX	NAM+ST+DIG	NAM+ST+CPLEX
100	10	0,02	0,00	0,11	0,00
100	20	3,96	0,85	0,84	0,42
100	30	5,82	1,99	1,82	0,75
100	40	6,01	1,81	1,05	0,90
100	50	7,94	0,94	1,71	1,30
200	10	0,33	0,00	0,15	0,00
200	20	3,81	0,96	0,33	0,18
200	30	6,36	1,89	0,58	0,63
200	40	6,70	2,18	0,53	0,75
200	50	10,87	1,58	0,40	1,88
500	10	2,08	0,04	0,07	0,03
500	20	4,51	0,74	0,21	0,21
500	30	5,50	1,09	0,60	0,37
500	40	5,72	1,21	0,60	0,69
500	50	41,98	1,03	0,47	0,51
1000	10	1,87	0,01	0,04	0,00
1000	20	3,30	0,37	0,38	0,26
1000	30	22,49	1,11	0,66	0,57
1000	40	30,78	1,01	0,48	0,54
1000	50	35,84	0,73	0,30	0,49
Media		10,29	0,98	0,57	0,52

Tabla E.4 – Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo U(10,100) y 50 % de máquinas quitadas con 60 segundos.

n	m	CPLEX	NAM+CPLEX	NAM+ST+DIG	NAM+ST+CPLEX
100	10	0,02	0,01	0,28	0,11
100	20	0,66	0,16	0,00	0,39
100	30	1,92	0,20	0,09	0,69
100	40	2,79	0,46	0,04	0,94
100	50	3,28	0,78	0,00	1,36
200	10	0,01	0,00	0,14	0,10
200	20	0,45	0,01	0,04	0,21
200	30	0,75	0,10	0,10	0,44
200	40	1,05	0,22	0,02	0,41
200	50	1,51	0,60	0,00	0,72
500	10	0,19	0,00	0,07	0,04
500	20	0,35	0,02	0,03	0,08
500	30	0,45	0,05	0,02	0,13
500	40	0,78	0,14	0,01	0,18
500	50	0,98	0,23	0,00	0,26
1000	10	0,28	0,00	0,03	0,01
1000	20	0,59	0,02	0,05	0,03
1000	30	3,46	0,02	0,05	0,05
1000	40	8,87	0,08	0,00	0,10
1000	50	44,69	0,11	0,04	0,13
Media		3,65	0,16	0,05	0,32

Tabla E.5 – Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo Trabajos Correlacionados y 50 % de máquinas quitadas con 300 segundos.

n	m	CPLEX	NAM+CPLEX	NAM+ST+DIG	NAM+ST+CPLEX
100	10	0,09	0,06	0,34	0,11
100	20	0,99	0,24	0,16	0,39
100	30	2,15	0,61	0,29	0,69
100	40	3,18	0,89	0,22	0,94
100	50	4,23	1,32	0,48	1,36
200	10	0,31	0,02	0,15	0,10
200	20	0,67	0,15	0,08	0,22
200	30	1,53	0,37	0,16	0,47
200	40	2,62	0,57	0,14	0,43
200	50	3,83	1,01	0,17	0,72
500	10	0,47	0,01	0,08	0,04
500	20	0,66	0,08	0,05	0,09
500	30	7,81	0,14	0,10	0,13
500	40	19,03	0,21	0,05	0,18
500	50	45,76	0,35	0,08	0,26
1000	10	0,51	0,00	0,03	0,01
1000	20	19,85	0,05	0,06	0,03
1000	30	33,43	0,08	0,10	0,05
1000	40	34,35	0,25	0,05	0,10
1000	50	56,62	0,37	0,08	0,13
Media		11,90	0,34	0,14	0,32

Tabla E.6 – Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo Trabajos Correlacionados y 50 % de máquinas quitadas con 60 segundos.

n	m	CPLEX	NAM+CPLEX	NAM+ST+DIG	NAM+ST+CPLEX
100	10	0,03	0,00	0,25	0,00
100	20	0,09	0,00	0,69	0,00
100	30	0,00	0,00	1,37	0,00
100	40	0,00	0,00	0,39	0,00
100	50	0,00	0,00	0,25	0,14
200	10	0,03	0,00	0,28	0,02
200	20	0,21	0,07	0,64	0,04
200	30	0,12	0,00	0,73	0,04
200	40	0,18	0,05	0,64	0,06
200	50	0,22	0,00	0,81	0,00
500	10	0,32	0,01	0,31	0,01
500	20	0,24	0,05	0,58	0,10
500	30	0,55	0,00	0,71	0,08
500	40	1,81	0,00	0,89	0,24
500	50	3,96	0,00	0,85	0,21
1000	10	0,16	0,00	0,55	0,00
1000	20	1,01	0,01	0,71	0,03
1000	30	2,94	0,02	0,77	0,12
1000	40	6,60	0,05	0,65	0,04
1000	50	6,51	0,01	0,78	0,06
Media		1,25	0,01	0,64	0,06

Tabla E.7 – Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo Máquinas Correlacionados y 50 % de máquinas quitadas con 300 segundos.

n	m	CPLEX	NAM+CPLEX	NAM+ST+DIG	NAM+ST+CPLEX
100	10	0,03	0,00	0,27	0,00
100	20	0,09	0,04	0,81	0,00
100	30	0,17	0,07	1,34	0,00
100	40	0,00	0,00	0,39	0,00
100	50	0,00	0,00	0,52	0,25
200	10	0,06	0,00	0,40	0,03
200	20	0,35	0,32	0,71	0,26
200	30	0,45	0,09	0,91	0,53
200	40	0,47	0,25	0,87	0,40
200	50	1,60	0,15	1,00	0,34
500	10	0,81	0,02	0,40	0,01
500	20	1,61	0,21	0,66	0,16
500	30	4,56	1,16	0,79	0,20
500	40	3,61	0,92	0,99	0,43
500	50	8,23	1,41	0,95	0,45
1000	10	0,82	0,01	0,59	0,00
1000	20	3,76	0,09	0,76	0,04
1000	30	13,94	0,24	0,85	0,27
1000	40	17,49	0,44	0,78	0,31
1000	50	16,53	0,44	0,87	0,39
Media		3,73	0,29	0,74	0,20

Tabla E.8 – Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo Máquinas Correlacionados y 50 % de máquinas quitadas con 60 segundos.

n	m	CPLEX	NAM+CPLEX	NAM+ST+DIG	NAM+ST+CPLEX
100	10	0,04	0,00	0,19	0,00
100	20	1,15	0,02	0,46	0,07
100	30	1,68	0,01	0,40	0,04
100	40	1,11	0,27	0,31	0,04
100	50	1,37	0,31	0,51	0,17
200	10	0,02	0,00	0,12	0,03
200	20	1,22	0,06	0,36	0,10
200	30	1,33	0,24	0,23	0,38
200	40	1,51	0,15	0,12	0,01
200	50	2,12	0,21	0,30	0,04
500	10	0,33	0,01	0,06	0,00
500	20	0,73	0,16	0,13	0,03
500	30	1,29	0,10	0,11	0,30
500	40	2,79	0,08	0,22	0,03
500	50	2,98	0,06	0,26	0,01
1000	10	0,34	0,01	0,01	0,00
1000	20	0,83	0,10	0,06	0,05
1000	30	0,96	0,05	0,12	0,12
1000	40	11,67	0,05	0,10	0,18
1000	50	32,99	0,10	0,09	0,08
Media		3,32	0,10	0,21	0,08

Tabla E.9 – Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo U(100,200) y 50 % de máquinas quitadas con 300 segundos.

n	m	CPLEX	NAM+CPLEX	NAM+ST+DIG	NAM+ST+CPLEX
100	10	0,19	0,00	0,20	0,04
100	20	1,78	0,26	0,59	0,21
100	30	2,04	0,12	0,48	0,65
100	40	2,26	0,42	0,35	0,15
100	50	3,40	0,58	0,58	0,39
200	10	0,23	0,08	0,13	0,05
200	20	1,85	0,41	0,45	0,28
200	30	2,05	0,84	0,31	0,86
200	40	4,82	0,29	0,22	0,11
200	50	6,99	0,29	0,41	0,24
500	10	0,81	0,03	0,06	0,03
500	20	1,43	0,23	0,15	0,13
500	30	18,84	0,40	0,25	0,36
500	40	35,12	0,40	0,31	0,17
500	50	34,13	0,55	0,27	0,34
1000	10	0,78	0,02	0,02	0,02
1000	20	22,55	0,26	0,11	0,11
1000	30	41,66	0,34	0,16	0,34
1000	40	31,51	0,48	0,18	0,40
1000	50	45,24	0,55	0,18	0,37
Media		12,88	0,33	0,27	0,26

Tabla E.10 – Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo U(100,200) y 50 % de máquinas quitadas con 60 segundos.

n	m	CPLEX	NAM+CPLEX	NAM+ST+DIG	NAM+ST+CPLEX
100	10	0,02	0,00	0,09	0,00
100	20	0,22	0,03	0,06	0,00
100	30	0,47	0,04	0,06	0,04
100	40	0,30	0,04	0,06	0,02
100	50	0,15	0,07	0,12	0,00
200	10	0,03	0,00	0,05	0,00
200	20	0,26	0,08	0,05	0,01
200	30	0,25	0,09	0,04	0,02
200	40	0,42	0,06	0,09	0,00
200	50	0,46	0,04	0,06	0,01
500	10	0,17	0,00	0,03	0,01
500	20	0,34	0,04	0,05	0,00
500	30	0,25	0,03	0,02	0,14
500	40	0,71	0,01	0,02	0,01
500	50	0,64	0,01	0,02	0,01
1000	10	0,24	0,00	0,03	0,03
1000	20	0,72	0,03	0,04	0,01
1000	30	4,68	0,07	0,00	0,07
1000	40	29,60	0,02	0,02	0,02
1000	50	47,97	0,01	0,00	0,01
Media		4,40	0,03	0,05	0,02

Tabla E.11 – Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo U(100,120) y 50 % de máquinas quitadas con 300 segundos.

n	m	CPLEX	NAM+CPLEX	NAM+ST+DIG	NAM+ST+CPLEX
100	10	0,07	0,00	0,10	0,00
100	20	0,42	0,03	0,08	0,02
100	30	0,48	0,04	0,09	0,17
100	40	0,60	0,08	0,06	0,02
100	50	0,65	0,12	0,15	0,12
200	10	0,16	0,00	0,05	0,00
200	20	0,46	0,13	0,06	0,06
200	30	0,68	0,18	0,06	0,16
200	40	1,35	0,13	0,11	0,09
200	50	1,31	0,12	0,10	0,10
500	10	0,35	0,01	0,03	0,03
500	20	7,17	0,06	0,05	0,05
500	30	25,21	0,08	0,06	0,14
500	40	40,11	0,06	0,05	0,04
500	50	48,88	0,03	0,03	0,05
1000	10	9,26	0,01	0,04	0,04
1000	20	20,91	0,05	0,06	0,05
1000	30	32,46	0,08	0,03	0,08
1000	40	33,49	0,11	0,04	0,09
1000	50	60,01	0,09	0,02	0,09
Media		14,20	0,07	0,06	0,07

Tabla E.12 – Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo U(100,120) y 50 % de máquinas quitadas con 60 segundos.

n	m	CPLEX	NAM+CPLEX	NAM+ST+DIG	NAM+ST+CPLEX
100	10	0,01	0,00	0,03	0,00
100	20	0,18	0,01	0,03	0,01
100	30	0,21	0,02	0,02	0,02
100	40	0,17	0,02	0,02	0,01
100	50	0,23	0,01	0,02	0,01
200	10	0,01	0,00	0,02	0,00
200	20	0,15	0,03	0,02	0,01
200	30	0,10	0,04	0,01	0,06
200	40	0,20	0,02	0,01	0,00
200	50	0,36	0,03	0,01	0,01
500	10	0,04	0,00	0,02	0,00
500	20	0,16	0,01	0,01	0,00
500	30	0,12	0,02	0,03	0,03
500	40	0,15	0,02	0,01	0,00
500	50	0,13	0,02	0,01	0,00
1000	10	0,12	0,01	0,01	0,00
1000	20	0,90	0,01	0,01	0,00
1000	30	0,09	0,03	0,00	0,02
1000	40	0,12	0,02	0,01	0,00
1000	50	0,11	0,02	0,00	0,01
Media		0,18	0,02	0,01	0,01

Tabla E.13 – Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo U(1000,1100) y 50 % de máquinas quitadas con 300 segundos.

n	m	CPLEX	NAM+CPLEX	NAM+ST+DIG	NAM+ST+CPLEX
100	10	0,03	0,00	0,03	0,02
100	20	0,30	0,04	0,05	0,02
100	30	0,27	0,06	0,03	0,07
100	40	0,31	0,05	0,03	0,02
100	50	0,59	0,06	0,03	0,03
200	10	0,09	0,00	0,02	0,01
200	20	0,22	0,04	0,02	0,02
200	30	0,24	0,15	0,01	0,08
200	40	0,38	0,05	0,02	0,04
200	50	3,01	0,06	0,02	0,03
500	10	0,20	0,01	0,02	0,01
500	20	1,99	0,02	0,02	0,01
500	30	0,17	0,07	0,06	0,07
500	40	0,18	0,05	0,02	0,03
500	50	0,14	0,05	0,02	0,03
1000	10	4,67	0,02	0,01	0,01
1000	20	1,43	0,03	0,01	0,02
1000	30	7,62	0,05	0,02	0,03
1000	40	0,14	0,04	0,02	0,02
1000	50	16,80	0,04	0,01	0,02
Media		1,94	0,04	0,02	0,03

Tabla E.14 – Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo U(1000,1100) y 50 % de máquinas quitadas con 60 segundos.

E.2. Tablas para un 20 % de máquinas sin usar

n	m	CPLEX	NAM+CPLEX	NAM+ST+DIG	NAM+ST+CPLEX
100	10	0,00	0,00	0,61	3,18
100	20	0,00	0,22	2,83	7,86
100	30	0,00	0,50	3,08	8,67
100	40	0,00	1,54	2,37	5,53
100	50	0,00	1,11	4,13	7,35
200	10	0,07	0,00	0,14	1,90
200	20	0,39	0,13	1,57	5,74
200	30	0,88	0,87	2,51	7,54
200	40	1,46	0,93	3,77	10,68
200	50	0,67	0,71	3,22	9,61
500	10	0,02	0,00	0,18	1,12
500	20	0,27	0,37	0,21	2,78
500	30	0,67	1,10	0,79	4,09
500	40	0,19	1,72	1,52	5,88
500	50	0,27	2,76	1,94	7,45
1000	10	0,09	0,00	0,09	0,39
1000	20	0,55	0,18	0,16	1,17
1000	30	0,39	1,15	0,33	2,20
1000	40	1,77	0,56	0,46	3,06
1000	50	3,05	0,00	0,27	3,73
Media		0,54	0,69	1,51	5,00

Tabla E.15 – Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo U(1,100) y 20 % de máquinas quitadas con 300 segundos.

n	m	CPLEX	NAM+CPLEX	NAM+ST+DIG	NAM+ST+CPLEX
100	10	0,00	0,00	0,68	3,18
100	20	0,49	0,22	3,29	7,86
100	30	0,00	0,50	4,61	8,24
100	40	0,00	2,25	2,37	5,53
100	50	0,00	1,11	5,13	7,35
200	10	0,14	0,07	0,21	1,90
200	20	1,45	1,42	2,09	5,74
200	30	2,26	1,38	2,51	8,07
200	40	2,86	3,28	4,23	10,68
200	50	2,59	0,71	3,84	10,32
500	10	0,23	0,06	0,26	1,12
500	20	0,79	1,11	0,63	2,78
500	30	1,98	1,87	1,12	4,42
500	40	4,56	2,11	1,90	6,27
500	50	6,30	3,31	2,76	8,83
1000	10	0,40	0,04	0,11	0,40
1000	20	2,98	0,93	0,57	1,30
1000	30	4,95	1,70	0,98	2,79
1000	40	7,06	1,77	1,21	3,90
1000	50	9,07	0,95	0,55	4,15
Media		2,41	1,24	1,95	5,24

Tabla E.16 – Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo U(1,100) y 20 % de máquinas quitadas con 60 segundos.

n	m	CPLEX	NAM+CPLEX	NAM+ST+DIG	NAM+ST+CPLEX
100	10	0,04	0,00	0,81	3,09
100	20	0,87	0,00	1,63	4,74
100	30	0,35	0,18	1,06	6,34
100	40	0,48	0,00	0,95	4,97
100	50	0,88	0,00	0,88	5,93
200	10	0,08	0,00	0,37	1,71
200	20	0,11	0,43	0,21	2,47
200	30	1,18	0,27	0,36	3,80
200	40	1,00	0,51	0,25	4,41
200	50	1,16	0,16	0,32	4,74
500	10	0,02	0,02	0,03	0,55
500	20	0,22	0,37	0,02	1,06
500	30	0,77	0,33	0,18	1,91
500	40	0,99	0,21	0,05	2,34
500	50	1,62	0,47	0,13	2,97
1000	10	0,09	0,01	0,04	0,26
1000	20	0,34	0,27	0,03	0,52
1000	30	0,37	0,29	0,11	0,99
1000	40	1,22	0,05	0,08	1,17
1000	50	1,87	0,10	0,20	1,60
Media		0,68	0,18	0,39	2,78

Tabla E.17 – Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo U(10,100) y 20 % de máquinas quitadas con 300 segundos.

n	m	CPLEX	NAM+CPLEX	NAM+ST+DIG	NAM+ST+CPLEX
100	10	0,17	0,00	0,85	3,09
100	20	1,74	1,51	1,96	4,85
100	30	0,69	0,70	1,59	6,70
100	40	0,95	0,00	0,71	4,97
100	50	1,19	0,29	0,88	5,93
200	10	0,19	0,12	0,39	1,71
200	20	0,65	1,08	0,32	2,79
200	30	2,26	1,18	0,64	3,98
200	40	2,52	1,51	0,38	4,54
200	50	1,48	0,33	0,65	4,90
500	10	0,21	0,08	0,09	0,55
500	20	0,58	0,71	0,13	1,17
500	30	1,70	0,74	0,44	1,99
500	40	2,55	1,20	0,31	2,39
500	50	3,91	2,02	0,68	3,17
1000	10	0,13	0,02	0,06	0,27
1000	20	1,22	0,51	0,24	0,66
1000	30	1,26	0,88	0,44	1,15
1000	40	2,79	0,84	0,37	1,33
1000	50	15,77	1,22	0,54	1,60
Media		2,10	0,75	0,58	2,89

Tabla E.18 – Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo U(10,100) y 20 % de máquinas quitadas con 60 segundos.

n	m	CPLEX	NAM+CPLEX	NAM+ST+DIG	NAM+ST+CPLEX
100	10	0,09	0,06	0,18	1,08
100	20	0,25	0,13	0,09	0,84
100	30	1,29	0,50	0,00	1,56
100	40	2,12	0,70	0,00	3,02
100	50	1,81	0,36	0,53	7,62
200	10	0,01	0,09	0,11	0,62
200	20	0,24	0,11	0,04	0,36
200	30	0,62	0,44	0,02	0,71
200	40	1,34	0,78	0,00	1,09
200	50	1,44	0,83	0,00	1,67
500	10	0,01	0,02	0,07	0,24
500	20	0,14	0,06	0,02	0,20
500	30	0,44	0,23	0,01	0,23
500	40	0,84	0,31	0,00	0,21
500	50	1,19	0,46	0,00	0,49
1000	10	0,04	0,01	0,02	0,10
1000	20	0,12	0,02	0,06	0,14
1000	30	0,25	0,10	0,01	0,08
1000	40	0,61	0,36	0,00	0,09
1000	50	0,98	0,54	0,00	0,13
Media		0,69	0,31	0,06	1,02

Tabla E.19 – Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo Trabajos Correlacionados y 20 % de máquinas quitadas con 300 segundos.

n	m	CPLEX	NAM+CPLEX	NAM+ST+DIG	NAM+ST+CPLEX
100	10	0,33	0,21	0,26	1,10
100	20	0,98	0,49	0,39	0,95
100	30	1,98	0,78	0,18	2,15
100	40	2,76	1,58	0,51	3,73
100	50	2,69	0,65	1,47	9,95
200	10	0,15	0,13	0,15	0,62
200	20	0,47	0,41	0,15	0,39
200	30	1,19	0,71	0,20	0,83
200	40	1,84	1,49	0,28	1,46
200	50	2,05	1,33	0,08	2,24
500	10	0,12	0,06	0,08	0,24
500	20	0,28	0,15	0,06	0,21
500	30	0,54	0,31	0,07	0,27
500	40	1,63	0,48	0,01	0,32
500	50	3,76	0,93	0,03	0,63
1000	10	0,11	0,01	0,03	0,10
1000	20	0,23	0,10	0,10	0,17
1000	30	0,72	0,27	0,05	0,11
1000	40	11,58	1,15	0,05	0,12
1000	50	1,82	1,86	0,08	0,22
Media		1,76	0,66	0,21	1,29

Tabla E.20 – Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo Trabajos Correlacionados y 20 % de máquinas quitadas con 60 segundos.

n	m	CPLEX	NAM+CPLEX	NAM+ST+DIG	NAM+ST+CPLEX
100	10	0,00	0,00	0,32	2,87
100	20	0,00	0,00	0,34	2,31
100	30	0,00	0,00	0,13	1,68
100	40	0,00	0,00	0,29	1,61
100	50	0,00	0,00	0,60	2,37
200	10	0,02	0,03	0,46	2,13
200	20	0,03	0,00	0,36	1,67
200	30	0,04	0,00	0,57	1,71
200	40	0,06	0,00	0,45	1,84
200	50	0,00	0,00	0,51	1,56
500	10	0,02	0,03	0,37	1,18
500	20	0,04	0,11	0,67	1,60
500	30	0,14	0,09	0,46	1,37
500	40	0,03	0,08	0,61	1,53
500	50	0,07	0,06	0,45	1,38
1000	10	0,01	0,02	0,51	1,02
1000	20	0,03	0,02	0,70	1,28
1000	30	0,06	0,04	0,74	1,41
1000	40	0,30	0,03	0,52	1,09
1000	50	1,72	0,09	0,47	1,09
Media		0,13	0,03	0,48	1,63

Tabla E.21 – Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo Máquinas Correlacionados y 20 % de máquinas quitadas con 300 segundos.

n	m	CPLEX	NAM+CPLEX	NAM+ST+DIG	NAM+ST+CPLEX
100	10	0,00	0,00	0,37	2,87
100	20	0,00	0,00	0,41	2,31
100	30	0,00	0,00	0,13	1,87
100	40	0,00	0,00	0,27	1,61
100	50	0,00	0,00	0,61	2,37
200	10	0,09	0,16	0,62	2,13
200	20	0,07	0,00	0,51	1,67
200	30	0,04	0,12	0,58	1,71
200	40	0,19	0,23	0,63	1,84
200	50	0,08	0,15	0,59	1,79
500	10	0,04	0,11	0,42	1,18
500	20	0,26	0,39	0,77	1,60
500	30	0,70	0,49	0,60	1,39
500	40	0,59	0,73	0,71	1,53
500	50	2,37	0,90	0,48	1,41
1000	10	0,03	0,10	0,54	1,02
1000	20	0,35	0,32	0,78	1,28
1000	30	1,13	0,32	0,82	1,41
1000	40	2,81	0,38	0,60	1,09
1000	50	5,82	0,60	0,60	1,11
Media		0,73	0,25	0,55	1,66

Tabla E.22 – Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo Máquinas Correlacionados y 20 % de máquinas quitadas con 60 segundos.

n	m	CPLEX	NAM+CPLEX	NAM+ST+DIG	NAM+ST+CPLEX
100	10	0,17	0,00	0,60	2,16
100	20	0,00	0,00	0,10	0,71
100	30	0,00	0,00	0,06	0,32
100	40	0,02	0,00	0,00	0,17
100	50	0,20	0,10	0,07	0,86
200	10	0,14	0,05	0,18	1,01
200	20	0,40	0,01	0,30	1,21
200	30	0,20	0,35	0,02	0,71
200	40	0,23	0,06	0,07	0,57
200	50	0,47	0,18	0,02	1,20
500	10	0,04	0,09	0,08	0,42
500	20	0,30	0,08	0,19	0,73
500	30	0,41	0,00	0,18	0,63
500	40	0,26	0,13	0,01	0,41
500	50	0,26	0,16	0,00	0,43
1000	10	0,08	0,04	0,03	0,22
1000	20	0,29	0,00	0,13	0,35
1000	30	0,36	0,05	0,03	0,34
1000	40	0,54	0,16	0,00	0,39
1000	50	0,96	0,12	0,04	0,43
Media		0,27	0,08	0,11	0,66

Tabla E.23 – Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo U(100,200) y 20 % de máquinas quitadas con 300 segundos.

n	m	CPLEX	NAM+CPLEX	NAM+ST+DIG	NAM+ST+CPLEX
100	10	0,37	0,04	0,70	2,16
100	20	0,01	0,00	0,10	0,73
100	30	0,14	0,00	0,04	0,38
100	40	0,10	0,00	0,00	0,22
100	50	0,26	0,16	0,13	0,89
200	10	0,34	0,22	0,26	1,01
200	20	0,69	0,65	0,46	1,24
200	30	0,43	1,22	0,14	0,77
200	40	0,48	0,16	0,10	0,59
200	50	0,91	0,25	0,25	1,26
500	10	0,18	0,11	0,11	0,43
500	20	0,67	0,53	0,18	0,74
500	30	0,87	0,44	0,26	0,64
500	40	0,73	0,30	0,09	0,44
500	50	0,57	0,30	0,09	0,49
1000	10	0,20	0,08	0,05	0,22
1000	20	0,61	0,28	0,18	0,40
1000	30	0,67	0,40	0,12	0,39
1000	40	2,58	0,42	0,08	0,44
1000	50	6,07	0,37	0,08	0,47
Media		0,84	0,30	0,17	0,69

Tabla E.24 – Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo U(100,200) y 20 % de máquinas quitadas con 60 segundos.

n	m	CPLEX	NAM+CPLEX	NAM+ST+DIG	NAM+ST+CPLEX
100	10	0,07	0,00	0,14	0,56
100	20	0,00	0,01	0,06	0,17
100	30	0,00	0,00	0,00	0,00
100	40	0,00	0,00	0,00	0,00
100	50	0,00	0,00	0,00	0,10
200	10	0,01	0,00	0,07	0,24
200	20	0,03	0,05	0,03	0,21
200	30	0,00	0,06	0,03	0,17
200	40	0,00	0,00	0,00	0,06
200	50	0,04	0,00	0,00	0,18
500	10	0,12	0,00	0,09	0,32
500	20	0,12	0,02	0,02	0,16
500	30	0,07	0,06	0,00	0,10
500	40	0,06	0,10	0,01	0,13
500	50	0,09	0,00	0,05	0,13
1000	10	0,05	0,00	0,02	0,14
1000	20	0,12	0,04	0,00	0,11
1000	30	0,09	0,01	0,00	0,07
1000	40	0,09	0,02	0,00	0,08
1000	50	0,23	0,02	0,00	0,08
Media		0,06	0,02	0,03	0,15

Tabla E.25 – Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo U(100,120) y 20 % de máquinas quitadas con 300 segundos.

n	m	CPLEX	NAM+CPLEX	NAM+ST+DIG	NAM+ST+CPLEX
100	10	0,11	0,00	0,16	0,56
100	20	0,03	0,01	0,07	0,17
100	30	0,00	0,00	0,00	0,00
100	40	0,00	0,00	0,00	0,00
100	50	0,00	0,00	0,00	0,10
200	10	0,03	0,04	0,09	0,24
200	20	0,08	0,28	0,05	0,24
200	30	0,10	0,14	0,03	0,17
200	40	0,03	0,09	0,00	0,09
200	50	0,10	0,00	0,02	0,18
500	10	0,31	0,13	0,15	0,33
500	20	0,20	0,11	0,05	0,19
500	30	0,21	0,13	0,01	0,11
500	40	0,16	0,17	0,02	0,13
500	50	0,18	0,04	0,05	0,13
1000	10	0,16	0,03	0,02	0,14
1000	20	0,16	0,07	0,03	0,11
1000	30	21,83	0,06	0,01	0,08
1000	40	5,85	0,06	0,02	0,09
1000	50	3,04	0,08	0,01	0,10
Media		1,63	0,07	0,04	0,16

Tabla E.26 – Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo U(100,120) y 20 % de máquinas quitadas con 60 segundos.

n	m	CPLEX	NAM+CPLEX	NAM+ST+DIG	NAM+ST+CPLEX
100	10	0,01	0,00	0,07	0,23
100	20	0,00	0,00	0,01	0,06
100	30	0,00	0,00	0,00	0,03
100	40	0,00	0,00	0,00	0,02
100	50	0,02	0,01	0,00	0,09
200	10	0,01	0,01	0,03	0,14
200	20	0,02	0,08	0,01	0,10
200	30	0,02	0,06	0,00	0,08
200	40	0,03	0,03	0,00	0,06
200	50	0,08	0,00	0,00	0,11
500	10	0,01	0,05	0,07	0,16
500	20	0,05	0,00	0,05	0,11
500	30	0,03	0,01	0,01	0,05
500	40	0,03	0,00	0,01	0,05
500	50	0,04	0,00	0,00	0,05
1000	10	0,00	0,01	0,01	0,04
1000	20	0,03	0,01	0,01	0,04
1000	30	0,03	0,00	0,00	0,03
1000	40	0,03	0,01	0,00	0,05
1000	50	0,04	0,01	0,00	0,03
Media		0,02	0,01	0,02	0,08

Tabla E.27 – Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo U(1000,1100) y 20 % de máquinas quitadas con 300 segundos.

n	m	CPLEX	NAM+CPLEX	NAM+ST+DIG	NAM+ST+CPLEX
100	10	0,03	0,06	0,06	0,23
100	20	0,01	0,00	0,00	0,06
100	30	0,01	0,00	0,00	0,03
100	40	0,01	0,00	0,00	0,02
100	50	0,04	0,08	0,00	0,09
200	10	0,03	0,01	0,04	0,14
200	20	0,05	0,16	0,02	0,11
200	30	0,05	0,13	0,01	0,08
200	40	0,05	0,05	0,00	0,06
200	50	0,11	0,03	0,01	0,11
500	10	0,11	0,11	0,08	0,16
500	20	0,09	0,04	0,06	0,12
500	30	0,07	0,02	0,02	0,06
500	40	0,06	0,01	0,02	0,06
500	50	0,05	0,02	0,01	0,05
1000	10	0,07	0,02	0,02	0,04
1000	20	0,05	0,03	0,02	0,04
1000	30	0,07	0,02	0,01	0,04
1000	40	0,76	0,03	0,01	0,05
1000	50	0,23	0,03	0,01	0,04
Media		0,10	0,04	0,02	0,08

Tabla E.28 – Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo U(1000,1100) y 20 % de máquinas quitadas con 60 segundos.

E.3. Tablas para un 80 % de máquinas sin usar

n	m	CPLEX	NAM+CPLEX	NAM+ST+DIG	NAM+ST+CPLEX
100	10	0,00	0,00	0,26	0,00
100	20	0,00	0,23	0,17	0,08
100	30	5,28	0,17	0,27	2,44
100	40	13,55	0,00	3,37	5,20
100	50	24,67	0,43	4,30	5,78
200	10	0,00	0,00	0,15	0,00
200	20	1,24	0,16	0,02	0,60
200	30	7,95	0,60	0,59	1,29
200	40	14,53	0,52	0,37	2,05
200	50	18,43	2,93	0,58	3,02
500	10	1,43	0,00	0,05	0,03
500	20	6,64	0,14	0,03	0,26
500	30	11,21	0,82	0,09	0,62
500	40	12,58	1,88	0,24	0,65
500	50	18,66	4,20	0,25	0,68
1000	10	8,24	0,00	0,03	0,13
1000	20	17,62	0,00	0,13	0,25
1000	30	25,55	0,10	0,28	0,29
1000	40	33,36	0,66	0,87	0,09
1000	50	38,07	0,56	0,49	0,28
Media		12,95	0,67	0,63	1,19

Tabla E.29 – Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo U(1,100) y 80 % de máquinas quitadas con 300 segundos.

n	m	CPLEX	NAM+CPLEX	NAM+ST+DIG	NAM+ST+CPLEX
100	10	0,00	0,00	0,24	0,00
100	20	2,40	0,23	0,17	1,34
100	30	13,81	0,43	1,08	2,55
100	40	23,25	2,50	3,37	5,20
100	50	31,69	7,07	4,30	5,78
200	10	0,10	0,00	0,12	0,09
200	20	8,27	0,16	0,07	0,64
200	30	14,01	1,04	0,59	1,85
200	40	22,48	4,93	0,92	2,46
200	50	28,55	9,74	3,63	4,37
500	10	8,87	0,00	0,05	0,08
500	20	23,99	0,34	0,17	0,41
500	30	35,37	2,66	1,97	1,27
500	40	43,73	3,36	2,99	2,22
500	50	62,01	4,76	4,42	5,51
1000	10	10,05	0,00	0,04	0,13
1000	20	20,96	0,00	0,37	0,32
1000	30	28,92	1,08	1,91	1,43
1000	40	38,02	1,44	1,68	1,78
1000	50	46,99	1,11	1,21	1,37
Media		23,17	2,04	1,47	1,94

Tabla E.30 – Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo U(1,100) y 80 % de máquinas quitadas con 60 segundos.

n	m	CPLEX	NAM+CPLEX	NAM+ST+DIG	NAM+ST+CPLEX
100	10	0,00	0,00	0,36	0,00
100	20	0,00	0,02	0,12	0,14
100	30	6,20	0,52	0,80	2,10
100	40	11,20	0,20	1,12	2,44
100	50	11,30	0,06	1,63	2,68
200	10	0,00	0,00	0,15	0,00
200	20	1,49	0,02	0,06	0,63
200	30	4,99	0,01	0,38	1,38
200	40	8,73	1,61	0,21	1,29
200	50	10,90	2,01	0,07	1,22
500	10	1,72	0,00	0,08	0,04
500	20	4,09	0,01	0,02	0,20
500	30	6,42	0,25	0,17	0,39
500	40	8,11	1,60	0,01	0,58
500	50	9,07	1,61	0,04	0,42
1000	10	2,87	0,00	0,01	0,05
1000	20	10,25	0,00	0,07	0,16
1000	30	15,98	0,60	0,16	0,10
1000	40	17,82	0,67	0,18	0,13
1000	50	26,83	0,31	0,19	0,13
Media		7,90	0,47	0,29	0,70

Tabla E.31 – Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo U(10,100) y 80 % de máquinas quitadas con 300 segundos.

n	m	CPLEX	NAM+CPLEX	NAM+ST+DIG	NAM+ST+CPLEX
100	10	0,00	0,00	0,36	0,00
100	20	2,68	0,11	0,13	0,81
100	30	9,75	0,88	0,83	2,20
100	40	14,31	3,68	1,12	2,44
100	50	14,00	3,67	1,70	2,94
200	10	0,00	0,00	0,15	0,08
200	20	5,56	0,02	0,07	0,63
200	30	9,91	1,89	0,39	1,41
200	40	12,46	3,35	0,26	1,29
200	50	13,71	3,33	0,39	1,44
500	10	8,05	0,00	0,07	0,14
500	20	16,26	0,10	0,06	0,23
500	30	27,90	0,83	0,28	0,46
500	40	29,15	1,80	1,42	1,32
500	50	33,61	1,89	1,52	1,79
1000	10	7,72	0,00	0,02	0,05
1000	20	11,24	0,05	0,20	0,17
1000	30	17,68	0,82	0,86	0,94
1000	40	18,21	0,72	0,58	0,74
1000	50	27,12	0,69	0,58	0,75
Media		13,97	1,19	0,55	0,99

Tabla E.32 – Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo U(10,100) y 80 % de máquinas quitadas con 60 segundos.

n	m	CPLEX	NAM+CPLEX	NAM+ST+DIG	NAM+ST+CPLEX
100	10	0,00	0,00	0,24	0,00
100	20	0,00	0,00	0,22	0,31
100	30	0,87	0,15	0,06	0,40
100	40	0,92	0,17	0,06	0,36
100	50	1,31	0,23	0,00	0,48
200	10	0,00	0,00	0,18	0,00
200	20	0,38	0,01	0,07	0,25
200	30	0,82	0,10	0,06	0,37
200	40	0,86	0,18	0,03	0,31
200	50	0,97	0,20	0,00	0,32
500	10	0,37	0,00	0,03	0,07
500	20	0,76	0,00	0,05	0,14
500	30	0,66	0,03	0,01	0,15
500	40	0,83	0,16	0,00	0,11
500	50	0,73	0,10	0,01	0,10
1000	10	0,38	0,00	0,01	0,04
1000	20	6,50	0,00	0,04	0,16
1000	30	7,85	0,02	0,05	0,13
1000	40	12,37	0,06	0,00	0,07
1000	50	12,48	0,03	0,02	0,08
Media		2,45	0,07	0,06	0,19

Tabla E.33 – Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo Trabajos Correlacionados y 80 % de máquinas quitadas con 300 segundos.

n	m	CPLEX	NAM+CPLEX	NAM+ST+DIG	NAM+ST+CPLEX
100	10	0,00	0,00	0,24	0,00
100	20	0,85	0,02	0,18	0,46
100	30	1,25	0,24	0,17	0,47
100	40	1,29	0,26	0,13	0,46
100	50	1,58	0,40	0,09	0,54
200	10	0,13	0,00	0,19	0,07
200	20	1,29	0,04	0,06	0,26
200	30	1,34	0,12	0,12	0,38
200	40	1,33	0,28	0,10	0,31
200	50	1,87	0,31	0,14	0,36
500	10	2,22	0,00	0,05	0,09
500	20	9,39	0,01	0,06	0,14
500	30	11,84	0,13	0,07	0,16
500	40	15,34	0,22	0,14	0,20
500	50	15,92	0,16	0,12	0,25
1000	10	3,66	0,00	0,02	0,04
1000	20	7,64	0,00	0,04	0,17
1000	30	8,70	0,09	0,12	0,21
1000	40	12,82	0,09	0,09	0,14
1000	50	14,48	0,04	0,07	0,11
Media		5,65	0,12	0,11	0,24

Tabla E.34 – Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo Trabajos Correlacionados y 80 % de máquinas quitadas con 60 segundos.

n	m	CPLEX	NAM+CPLEX	NAM+ST+DIG	NAM+ST+CPLEX
100	10	0,00	0,00	0,44	0,00
100	20	0,00	0,00	0,08	0,02
100	30	0,48	0,00	0,19	0,51
100	40	1,31	0,00	0,38	2,07
100	50	1,30	0,00	0,47	2,24
200	10	0,00	0,00	0,23	0,00
200	20	0,18	0,00	0,54	0,19
200	30	0,80	0,00	0,20	0,68
200	40	1,92	0,00	0,43	1,57
200	50	3,68	0,05	0,53	2,78
500	10	0,04	0,00	0,18	0,07
500	20	8,58	0,00	0,29	0,85
500	30	8,21	0,00	0,38	1,10
500	40	10,30	0,00	0,48	1,64
500	50	11,31	0,00	0,55	1,74
1000	10	2,36	0,00	0,40	0,50
1000	20	18,15	0,00	0,43	1,11
1000	30	53,27	0,00	0,49	1,07
1000	40	220,96	0,00	0,44	1,10
1000	50	263,84	0,00	0,62	1,28
Media		30,33	0,00	0,39	1,03

Tabla E.35 – Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo Máquinas Correlacionados y 80 % de máquinas quitadas con 300 segundos.

n	m	CPLEX	NAM+CPLEX	NAM+ST+DIG	NAM+ST+CPLEX
100	10	0,00	0,00	0,38	0,00
100	20	0,13	0,00	0,08	0,34
100	30	1,40	0,00	0,19	1,84
100	40	1,79	0,05	0,52	2,69
100	50	2,79	0,00	0,57	3,54
200	10	0,01	0,00	0,17	0,06
200	20	0,49	0,00	0,56	0,29
200	30	1,76	0,05	0,22	1,49
200	40	4,68	0,17	0,57	2,00
200	50	7,36	0,19	0,61	2,88
500	10	3,05	0,00	0,20	0,21
500	20	37,19	0,00	0,36	1,03
500	30	57,92	0,01	0,42	1,34
500	40	59,32	0,03	0,57	1,64
500	50	70,16	0,11	0,74	1,76
1000	10	41,15	0,00	0,40	0,83
1000	20	85,78	0,00	0,46	1,14
1000	30	111,73	0,02	0,55	1,07
1000	40	306,19	0,03	0,49	1,10
1000	50	339,90	0,07	0,69	1,28
Media		56,64	0,04	0,44	1,33

Tabla E.36 – Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo Máquinas Correlacionados y 80 % de máquinas quitadas con 60 segundos.

n	m	CPLEX	NAM+CPLEX	NAM+ST+DIG	NAM+ST+CPLEX
100	10	0,00	0,00	0,07	0,00
100	20	0,01	0,00	0,14	0,31
100	30	2,10	0,04	0,25	0,92
100	40	2,55	0,02	0,37	1,17
100	50	3,46	0,09	0,24	0,67
200	10	0,00	0,00	0,07	0,00
200	20	0,64	0,00	0,08	0,30
200	30	1,60	0,39	0,05	0,49
200	40	2,61	0,28	0,06	0,57
200	50	3,40	0,26	0,13	0,61
500	10	0,90	0,00	0,01	0,04
500	20	1,25	0,04	0,03	0,14
500	30	1,57	0,22	0,02	0,22
500	40	3,94	0,46	0,01	0,30
500	50	1,95	0,25	0,03	0,26
1000	10	0,71	0,00	0,01	0,02
1000	20	3,73	0,02	0,02	0,09
1000	30	11,85	0,16	0,00	0,09
1000	40	11,63	0,23	0,01	0,12
1000	50	25,11	0,09	0,00	0,14
Media		3,95	0,13	0,08	0,32

Tabla E.37 – Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo U(100,200) y 80 % de máquinas quitadas con 300 segundos.

n	m	CPLEX	NAM+CPLEX	NAM+ST+DIG	NAM+ST+CPLEX
100	10	0,00	0,00	0,07	0,00
100	20	1,33	0,01	0,12	0,40
100	30	3,61	0,27	0,33	1,04
100	40	3,78	0,43	0,47	1,16
100	50	4,97	0,68	0,25	0,73
200	10	0,04	0,00	0,07	0,03
200	20	2,57	0,02	0,08	0,34
200	30	2,69	0,45	0,08	0,52
200	40	3,21	0,55	0,22	0,57
200	50	5,27	0,43	0,14	0,64
500	10	0,91	0,00	0,01	0,04
500	20	11,19	0,18	0,03	0,18
500	30	12,79	0,33	0,14	0,27
500	40	16,86	0,53	0,33	0,40
500	50	15,90	0,34	0,15	0,49
1000	10	5,23	0,00	0,01	0,02
1000	20	5,95	0,06	0,08	0,10
1000	30	12,11	0,19	0,16	0,20
1000	40	11,72	0,23	0,20	0,33
1000	50	34,22	0,19	0,04	0,21
Media		7,72	0,24	0,15	0,38

Tabla E.38 – Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo U(100,200) y 80 % de máquinas quitadas con 60 segundos.

n	m	CPLEX	NAM+CPLEX	NAM+ST+DIG	NAM+ST+CPLEX
100	10	0,00	0,00	0,05	0,00
100	20	0,16	0,00	0,06	0,16
100	30	0,87	0,01	0,13	0,35
100	40	0,80	0,00	0,14	0,30
100	50	0,74	0,02	0,05	0,21
200	10	0,01	0,00	0,03	0,00
200	20	0,48	0,00	0,06	0,12
200	30	0,71	0,00	0,19	0,38
200	40	0,86	0,04	0,04	0,12
200	50	1,03	0,10	0,01	0,12
500	10	0,14	0,00	0,02	0,03
500	20	0,55	0,02	0,03	0,06
500	30	2,39	0,05	0,02	0,16
500	40	8,11	0,02	0,05	0,16
500	50	14,65	0,04	0,01	0,08
1000	10	0,17	0,00	0,01	0,02
1000	20	5,91	0,02	0,02	0,04
1000	30	10,04	0,04	0,01	0,08
1000	40	11,33	0,06	0,01	0,06
1000	50	24,84	0,02	0,01	0,06
Media		4,19	0,02	0,05	0,13

Tabla E.39 – Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo U(100,120) y 80 % de máquinas quitadas con 300 segundos.

n	m	CPLEX	NAM+CPLEX	NAM+ST+DIG	NAM+ST+CPLEX
100	10	0,00	0,00	0,05	0,00
100	20	0,76	0,02	0,06	0,16
100	30	1,12	0,06	0,15	0,41
100	40	0,85	0,03	0,15	0,30
100	50	0,97	0,14	0,06	0,21
200	10	0,19	0,00	0,03	0,05
200	20	1,03	0,00	0,06	0,14
200	30	1,69	0,14	0,22	0,37
200	40	8,56	0,13	0,04	0,12
200	50	20,25	0,17	0,05	0,14
500	10	0,24	0,00	0,02	0,03
500	20	7,37	0,03	0,04	0,07
500	30	9,72	0,16	0,06	0,17
500	40	14,82	0,12	0,11	0,19
500	50	17,01	0,09	0,08	0,12
1000	10	2,39	0,00	0,01	0,02
1000	20	6,55	0,04	0,03	0,04
1000	30	10,58	0,06	0,05	0,12
1000	40	12,13	0,08	0,04	0,13
1000	50	28,35	0,06	0,05	0,13
Media		7,23	0,07	0,07	0,14

Tabla E.40 – Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo U(100,120) y 80 % de máquinas quitadas con 60 segundos.

n	m	CPLEX	NAM+CPLEX	NAM+ST+DIG	NAM+ST+CPLEX
100	10	0,00	0,00	0,05	0,00
100	20	0,12	0,00	0,02	0,07
100	30	0,44	0,01	0,06	0,17
100	40	0,37	0,00	0,04	0,14
100	50	0,43	0,07	0,00	0,06
200	10	0,01	0,00	0,02	0,01
200	20	0,33	0,00	0,02	0,06
200	30	0,42	0,02	0,05	0,14
200	40	0,42	0,05	0,00	0,04
200	50	0,42	0,06	0,00	0,04
500	10	0,08	0,00	0,00	0,01
500	20	0,24	0,00	0,01	0,02
500	30	3,93	0,01	0,03	0,10
500	40	6,27	0,01	0,02	0,08
500	50	8,44	0,03	0,00	0,02
1000	10	0,10	0,00	0,01	0,01
1000	20	4,01	0,01	0,00	0,01
1000	30	4,94	0,02	0,01	0,03
1000	40	8,53	0,02	0,00	0,01
1000	50	15,67	0,02	0,00	0,01
Media		2,76	0,02	0,02	0,05

Tabla E.41 – Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo U(1000,1100) y 80 % de máquinas quitadas con 300 segundos.

n	m	CPLEX	NAM+CPLEX	NAM+ST+DIG	NAM+ST+CPLEX
100	10	0,00	0,00	0,05	0,00
100	20	0,38	0,00	0,02	0,07
100	30	0,48	0,05	0,09	0,18
100	40	0,43	0,02	0,06	0,13
100	50	0,49	0,10	0,01	0,06
200	10	0,08	0,00	0,02	0,02
200	20	0,45	0,01	0,02	0,06
200	30	0,70	0,06	0,06	0,15
200	40	8,70	0,08	0,01	0,05
200	50	8,93	0,07	0,01	0,05
500	10	0,16	0,00	0,00	0,01
500	20	5,41	0,00	0,02	0,03
500	30	5,80	0,09	0,05	0,11
500	40	7,51	0,08	0,06	0,09
500	50	10,29	0,05	0,02	0,04
1000	10	1,39	0,00	0,01	0,01
1000	20	5,17	0,02	0,01	0,01
1000	30	5,57	0,08	0,05	0,07
1000	40	9,16	0,04	0,03	0,04
1000	50	38,26	0,03	0,02	0,03
Media		5,47	0,04	0,03	0,06

Tabla E.42 – Tabla porcentual de la media de las diferencias respecto al mejor resultado obtenido para el intervalo U(1000,1100) y 80 % de máquinas quitadas con 60 segundos.

E.4. Anovas para un 50 % de máquinas sin usar

Se presentan los gráficos procedentes de la eliminación de un 50 % de máquinas, para los tiempos de estudio de 60 segundos y 300 segundos y los intervalos de tiempos de proceso de $U(1, 100)$, $U(10, 100)$, trabajos correlacionados, máquinas correlacionadas, $U(100, 200)$, $U(100, 120)$ y $U(1000, 1100)$.

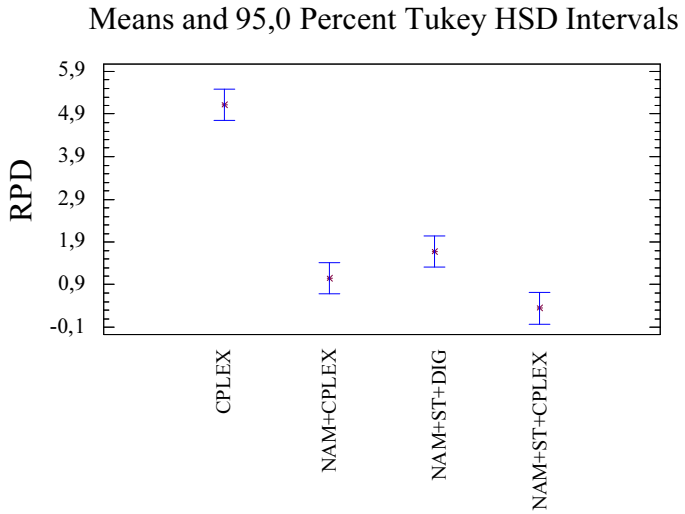

Figura E.1 – ANOVA para el intervalo U(1,100) con 50 % máquinas quitadas y 300 segundos.

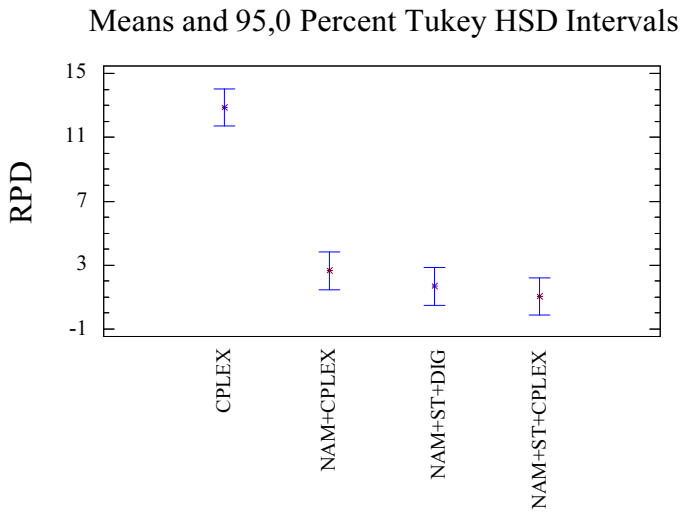

Figura E.2 – ANOVA para el intervalo U(1,100) con 50 % máquinas quitadas y 60 segundos.

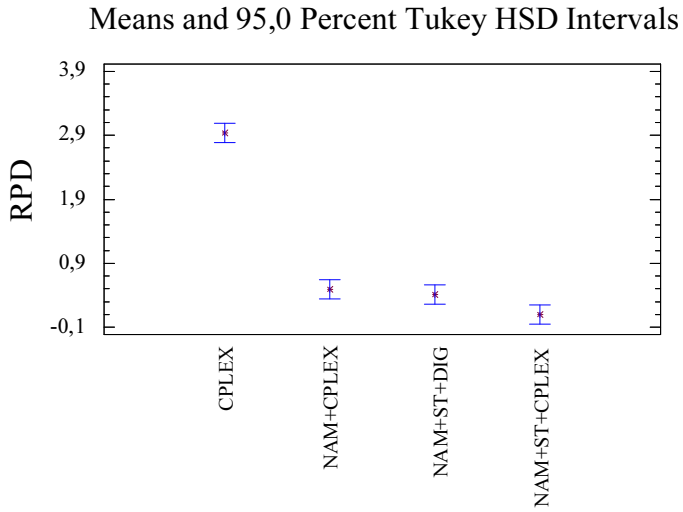

Figura E.3 – ANOVA para el intervalo U(10,100) con 50 % máquinas quitadas y 300 segundos.

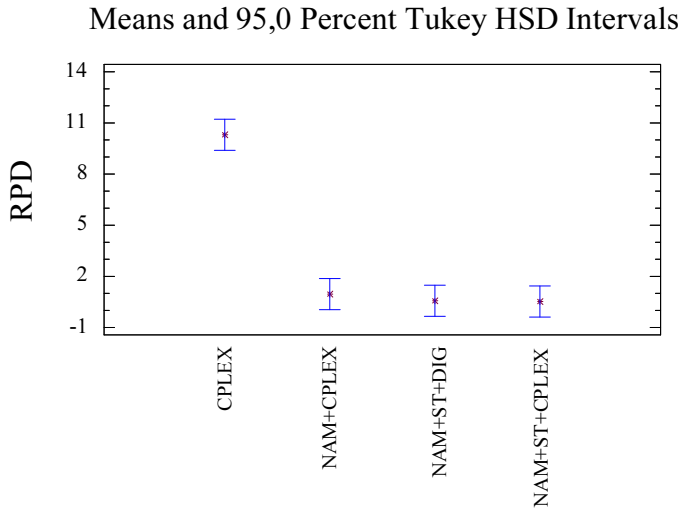

Figura E.4 – ANOVA para el intervalo U(10,100) con 50 % máquinas quitadas y 60 segundos.

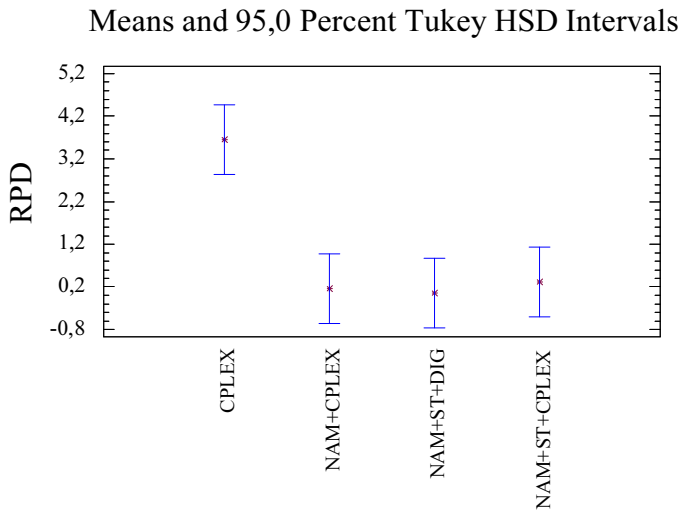

Figura E.5 – ANOVA para el intervalo Trabajos Correlacionados con 50 % máquinas quitadas y 300 segundos.

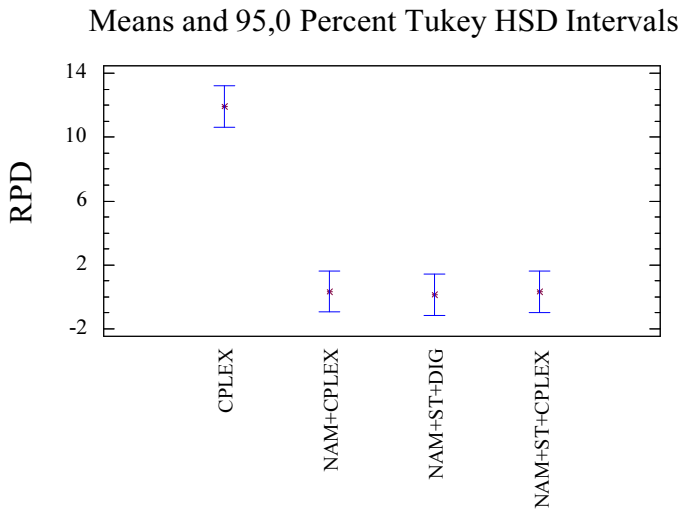

Figura E.6 – ANOVA para el intervalo Trabajos Correlacionados con 50 % máquinas quitadas y 60 segundos.

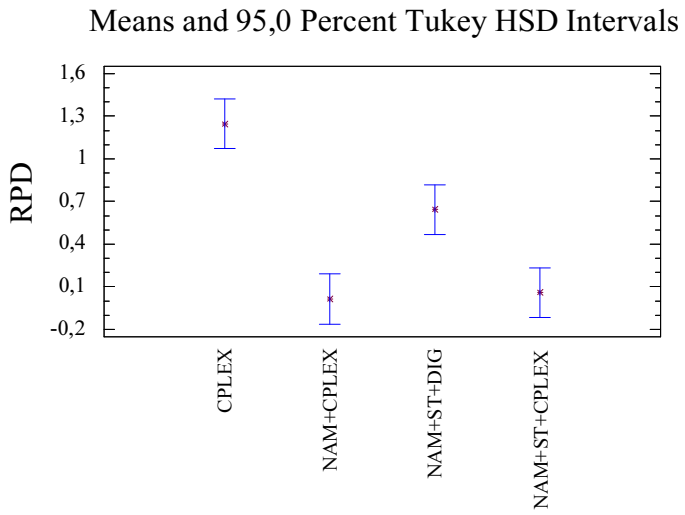

Figura E.7 – ANOVA para el intervalo Máquinas Correlacionadas con 50 % máquinas quitadas y 300 segundos.

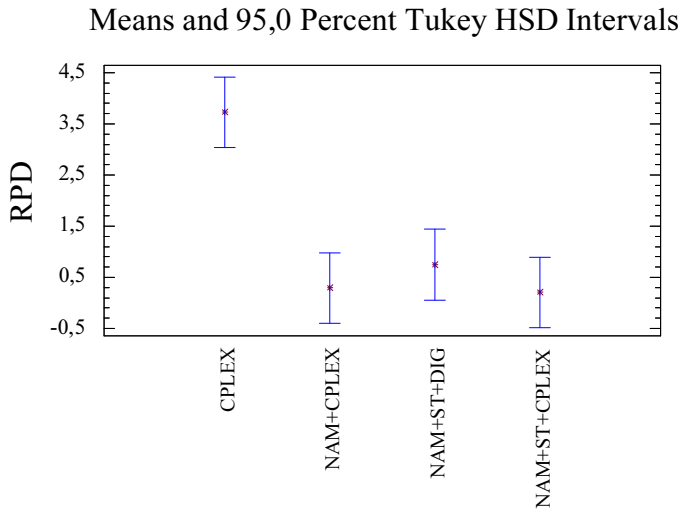

Figura E.8 – ANOVA para el intervalo Máquinas Correlacionadas con 50 % máquinas quitadas y 60 segundos.

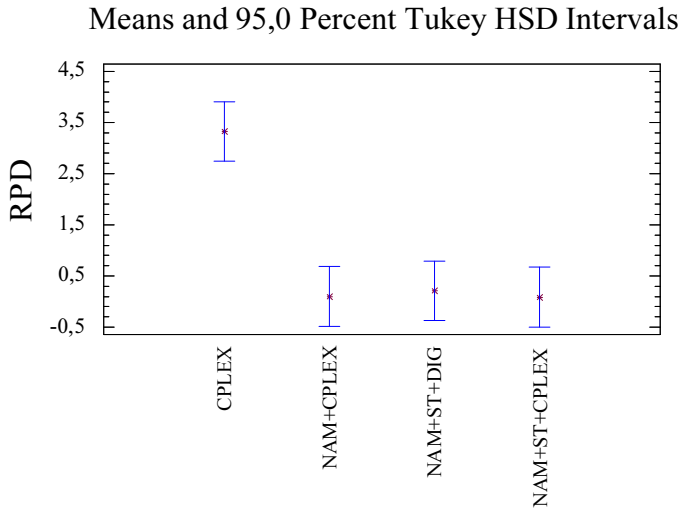

Figura E.9 – ANOVA para el intervalo U(100,200) con 50 % máquinas quitadas y 300 segundos.

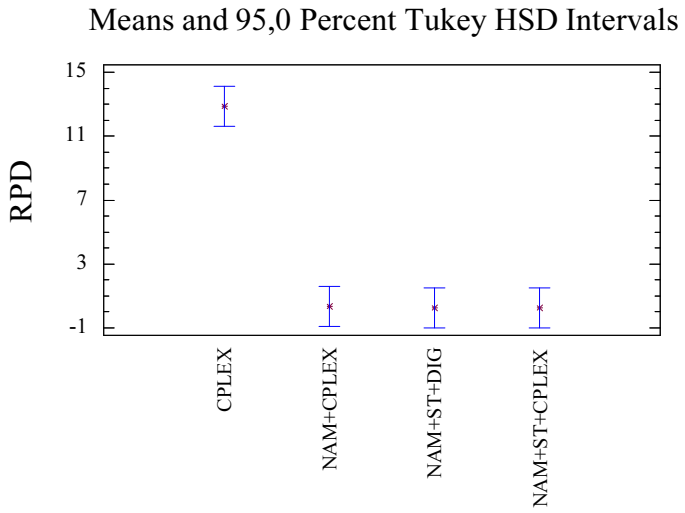

Figura E.10 – ANOVA para el intervalo U(100,200) con 50 % máquinas quitadas y 60 segundos.

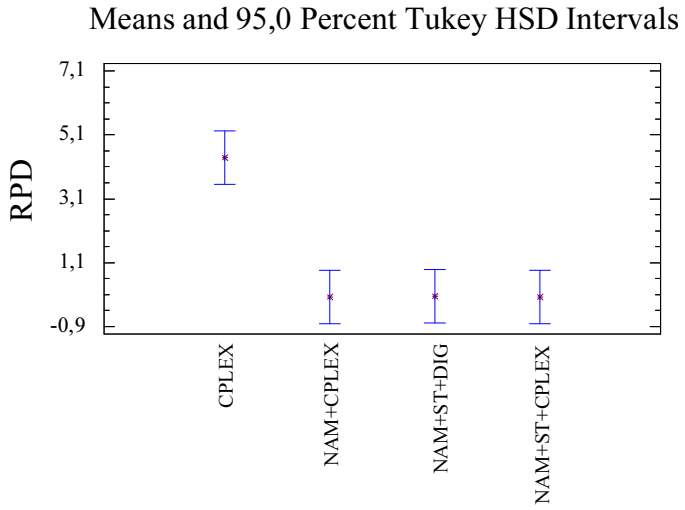

Figura E.11 – ANOVA para el intervalo U(100,120) con 50 % máquinas quitadas y 300 segundos.

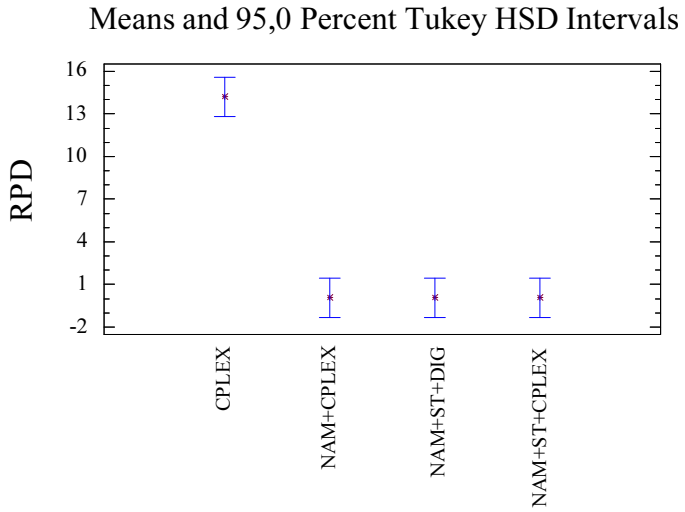

Figura E.12 – ANOVA para el intervalo U(100,120) con 50 % máquinas quitadas y 60 segundos.

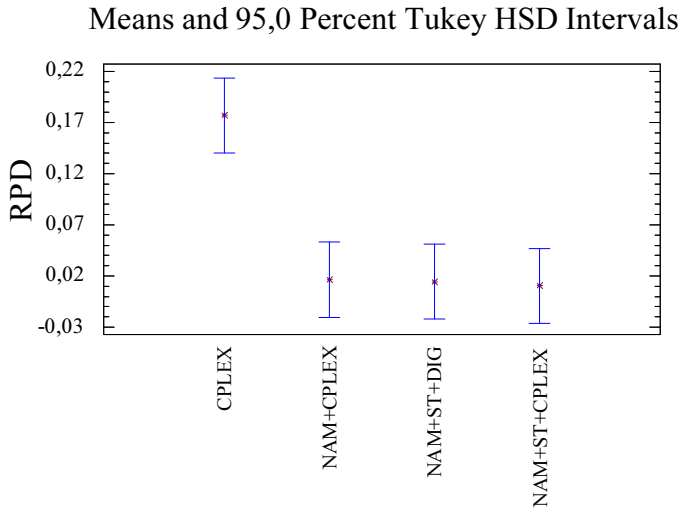

Figura E.13 – ANOVA para el intervalo U(1000,1100) con 50 % máquinas quitadas y 300 segundos.

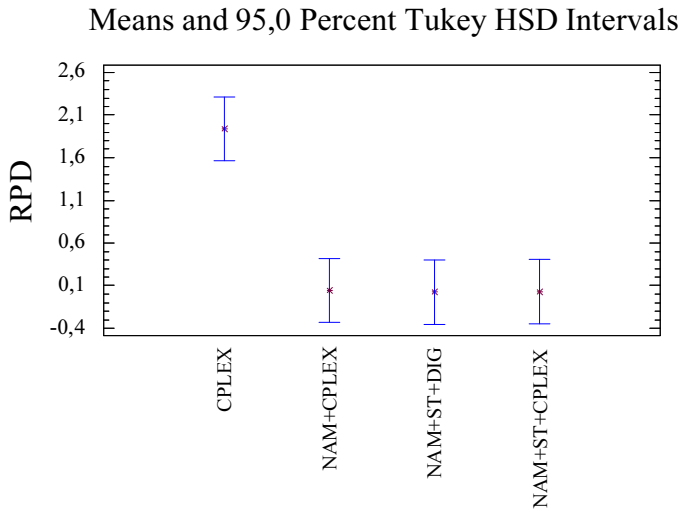

Figura E.14 – ANOVA para el intervalo U(1000,1100) con 50 % máquinas quitadas y 60 segundos.

E.5. Anovas para un 20 % de máquinas sin usar

Se presentan los gráficos procedentes de la eliminación de un 20 % de máquinas, para los tiempos de estudio de 60 segundos y 300 segundos y los intervalos de tiempos de proceso de $U(1, 100)$, $U(10, 100)$, trabajos correlacionados, máquinas correlacionadas, $U(100, 200)$, $U(100, 120)$ y $U(1000, 1100)$.

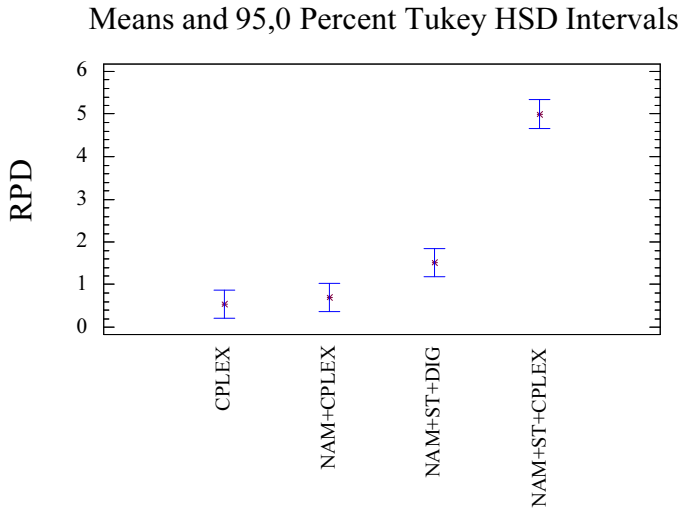

Figura E.15 – ANOVA para el intervalo U(1,100) con 20 % máquinas quitadas y 300 segundos.

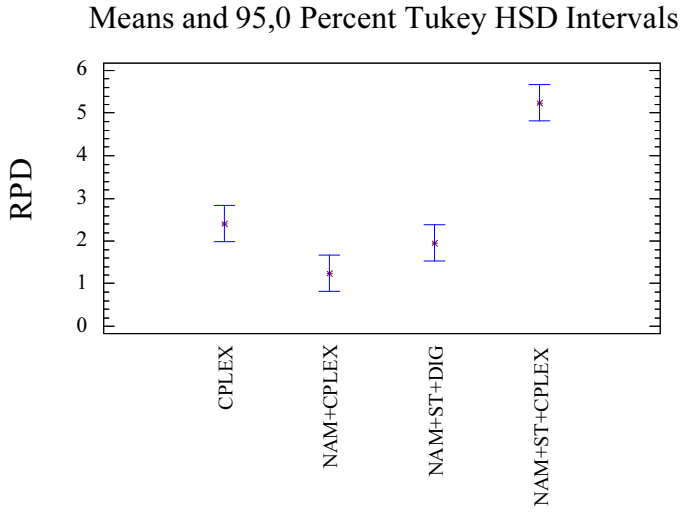

Figura E.16 – ANOVA para el intervalo U(1,100) con 20 % máquinas quitadas y 60 segundos.

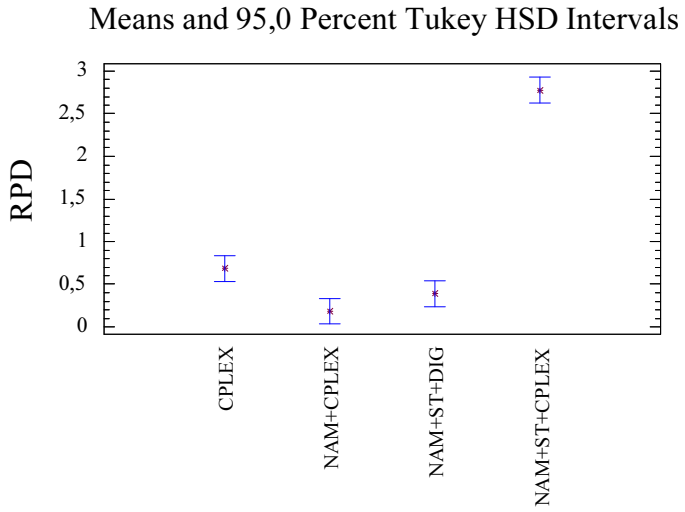

Figura E.17 – ANOVA para el intervalo U(10,100) con 20 % máquinas quitadas y 300 segundos.

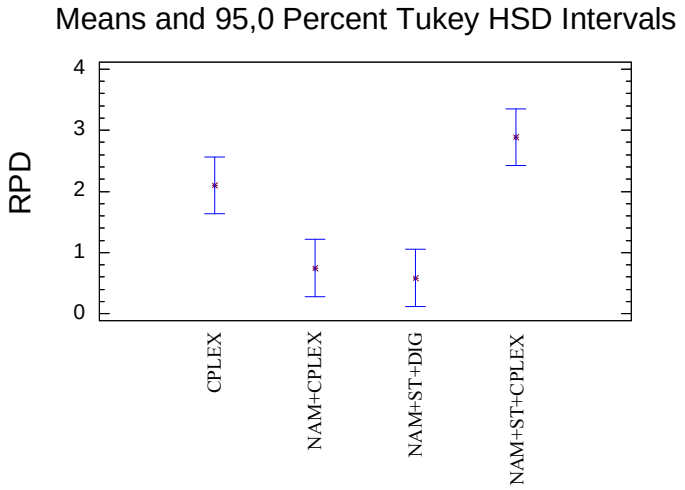

Figura E.18 – ANOVA para el intervalo U(10,100) con 20 % máquinas quitadas y 60 segundos.

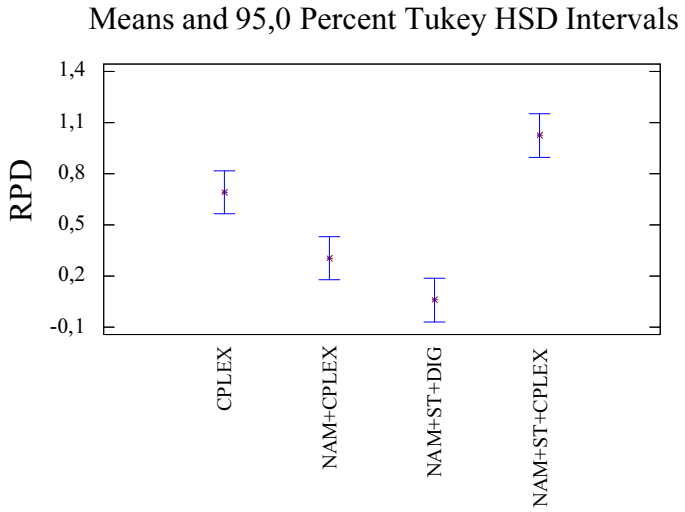

Figura E.19 – ANOVA para el intervalo Trabajos Correlacionados con 20 % máquinas quitadas y 300 segundos.

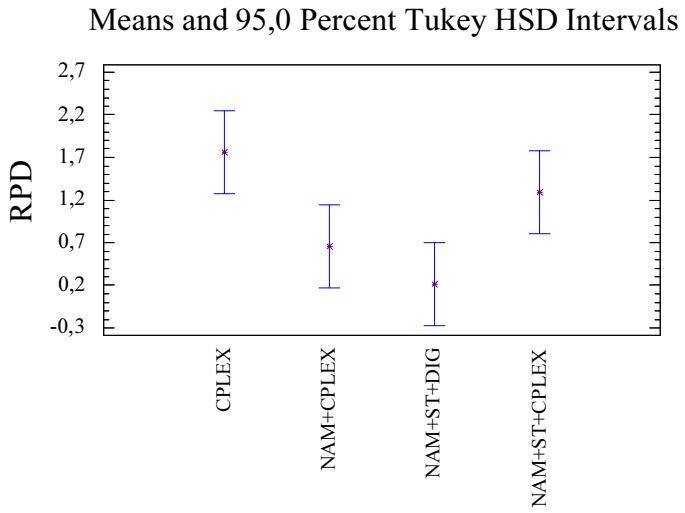

Figura E.20 – ANOVA para el intervalo Trabajos Correlacionados con 20 % máquinas quitadas y 60 segundos.

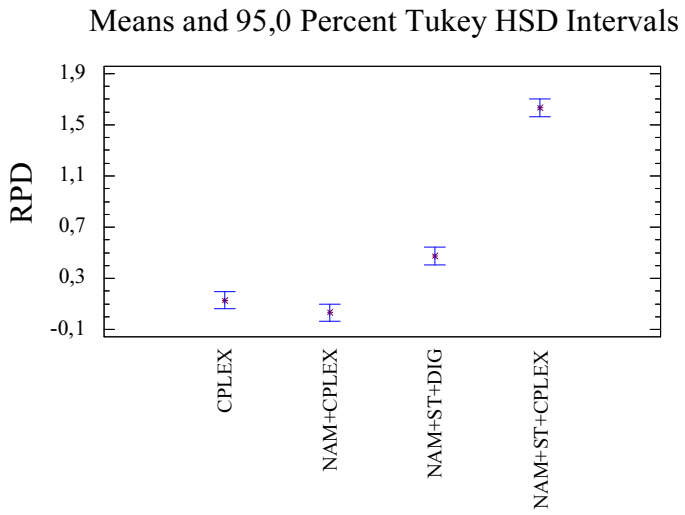

Figura E.21 – ANOVA para el intervalo Máquinas Correlacionadas con 20 % máquinas quitadas y 300 segundos.

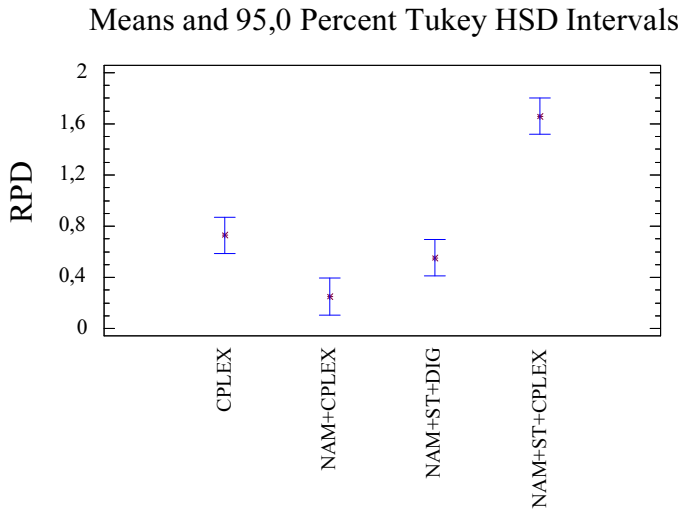

Figura E.22 – ANOVA para el intervalo máquinas Correlacionadas con 20 % máquinas quitadas y 60 segundos.

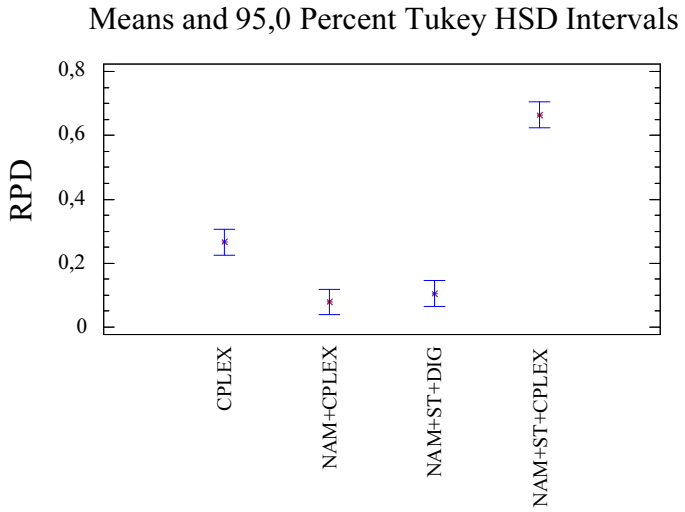

Figura E.23 – ANOVA para el intervalo U(100,200) con 20 % máquinas quitadas y 300 segundos.

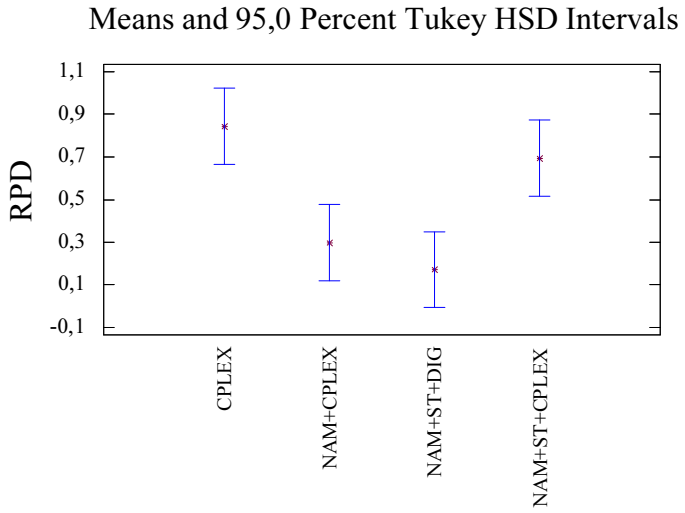

Figura E.24 – ANOVA para el intervalo U(100,200) con 20 % máquinas quitadas y 60 segundos.

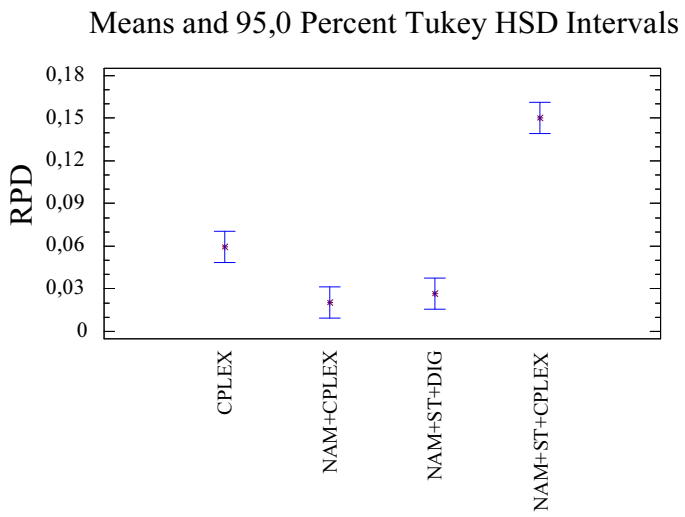

Figura E.25 – ANOVA para el intervalo U(100,120) con 20 % máquinas quitadas y 300 segundos.

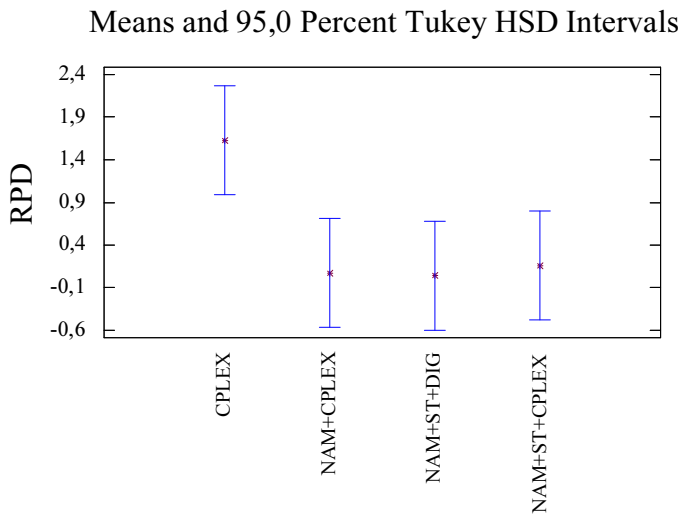

Figura E.26 – ANOVA para el intervalo U(100,120) con 20 % máquinas quitadas y 60 segundos.

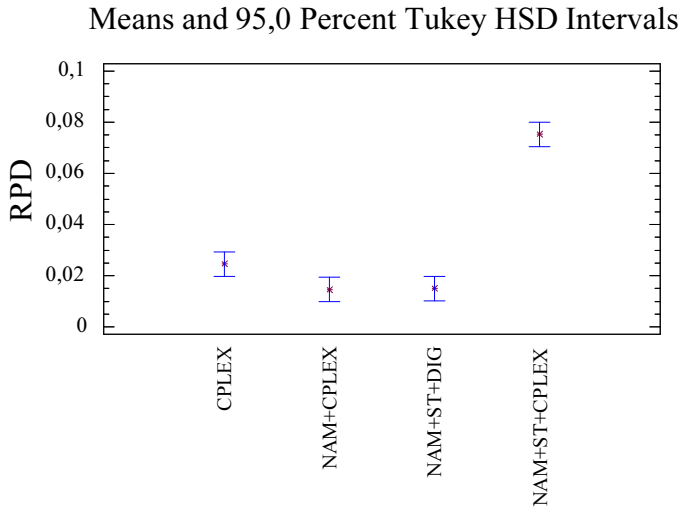

Figura E.27 – ANOVA para el intervalo U(1000,1100) con 20 % máquinas quitadas y 300 segundos.

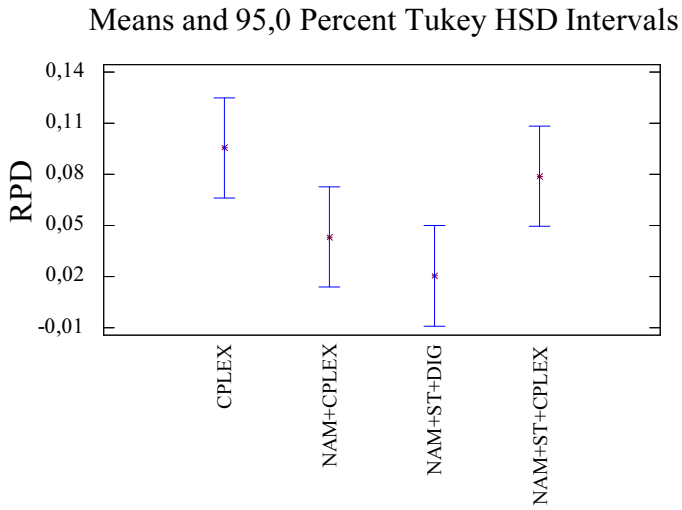

Figura E.28 – ANOVA para el intervalo U(1000,1100) con 20 % máquinas quitadas y 60 segundos.

E.6. Anovas para un 80 % de máquinas sin usar

Se presentan los gráficos procedentes de la eliminación de un 80 % de máquinas, para los tiempos de estudio de 60 segundos y 300 segundos y los intervalos de tiempos de proceso de $U(1, 100)$, $U(10, 100)$, trabajos correlacionados, máquinas correlacionadas, $U(100, 200)$, $U(100, 120)$ y $U(1000, 1100)$.

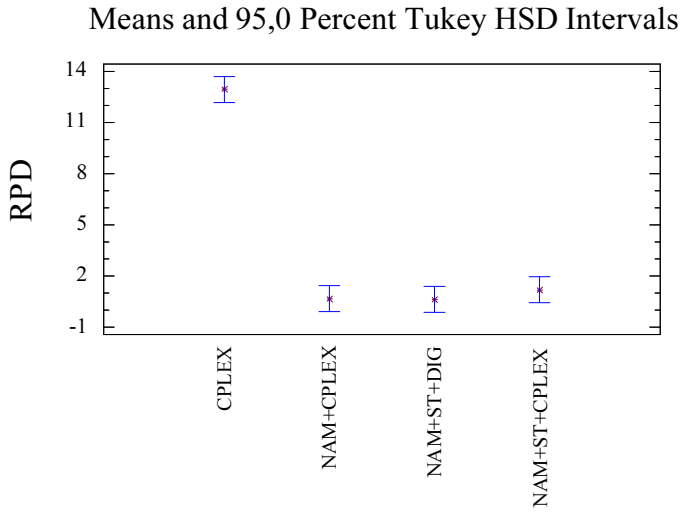

Figura E.29 – ANOVA para el intervalo U(1,100) con 80 % máquinas quitadas y 300 segundos.

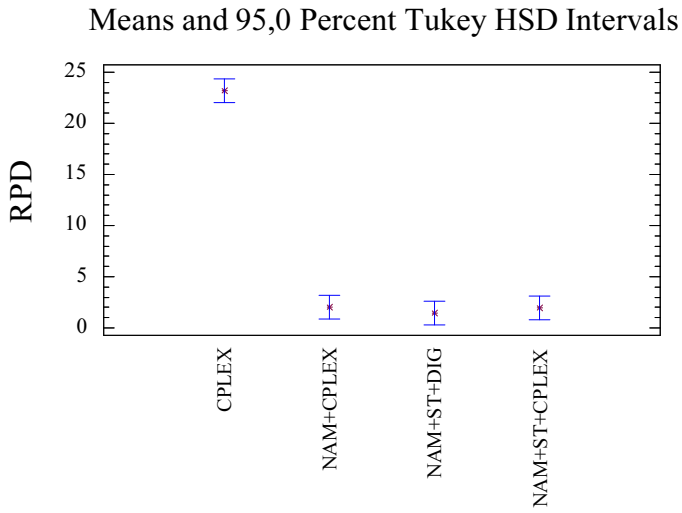

Figura E.30 – ANOVA para el intervalo U(1,100) con 80 % máquinas quitadas y 60 segundos.

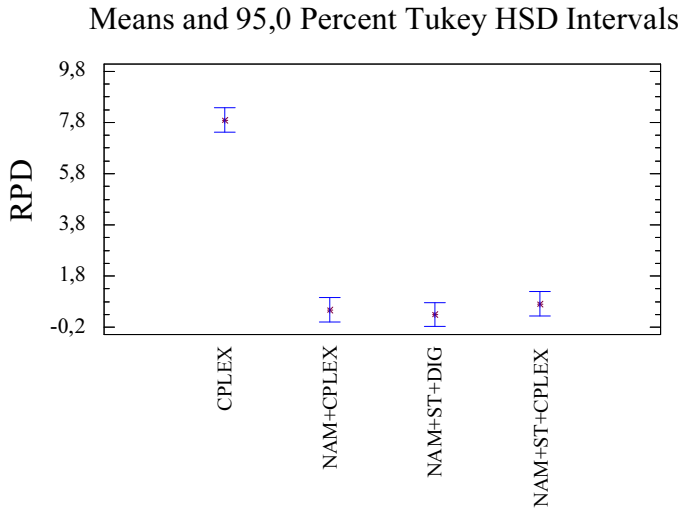

Figura E.31 – ANOVA para el intervalo U(10,100) con 80 % máquinas quitadas y 300 segundos.

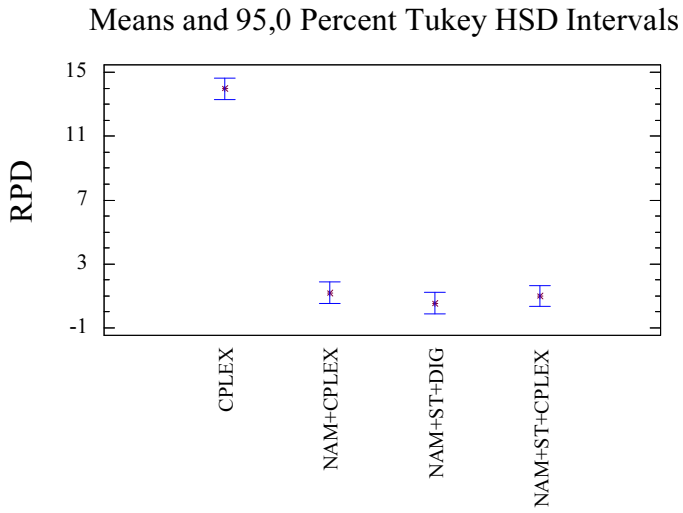

Figura E.32 – ANOVA para el intervalo U(10,100) con 80 % máquinas quitadas y 60 segundos.

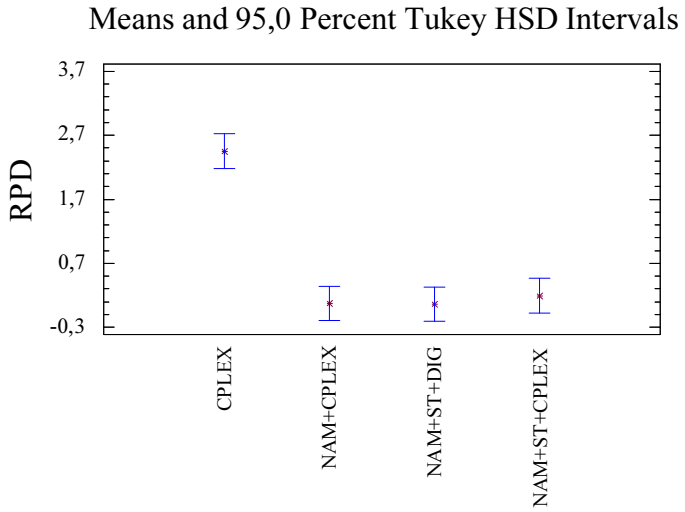

Figura E.33 – ANOVA para el intervalo Trabajos Correlacionados con 80 % máquinas quitadas y 300 segundos.

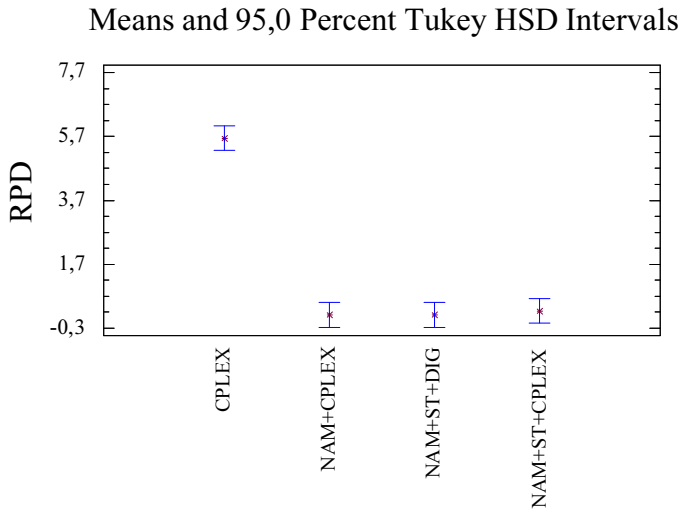

Figura E.34 – ANOVA para el intervalo Trabajos Correlacionados con 80 % máquinas quitadas y 60 segundos.

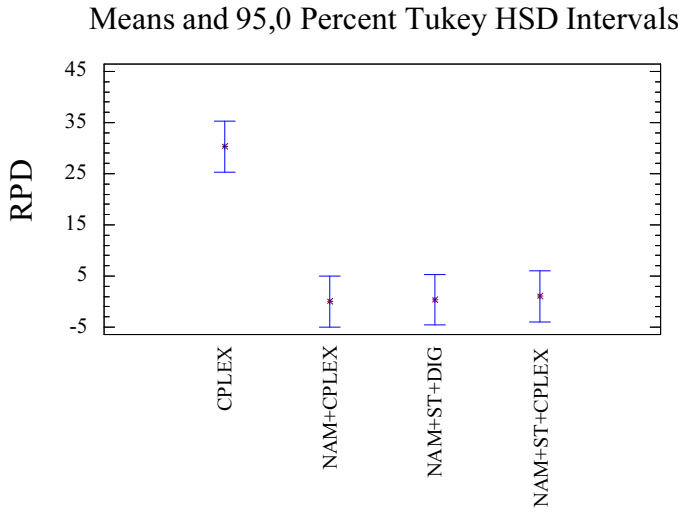

Figura E.35 – ANOVA para el intervalo Máquinas Correlacionadas con 80 % máquinas quitadas y 300 segundos.

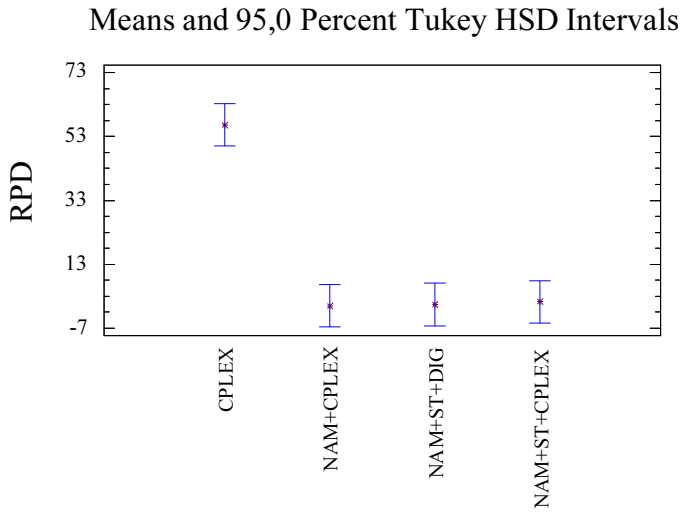

Figura E.36 – ANOVA para el intervalo Máquinas Correlacionadas con 80 % máquinas quitadas y 60 segundos.

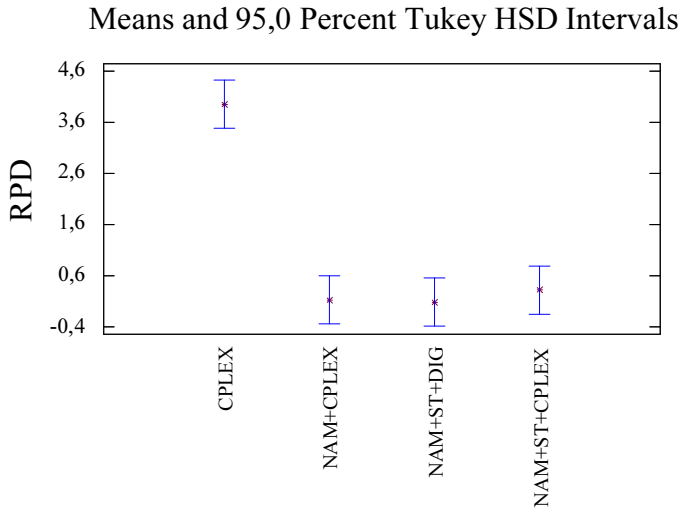

Figura E.37 – ANOVA para el intervalo U(100,200) con 80 % máquinas quitadas y 300 segundos.

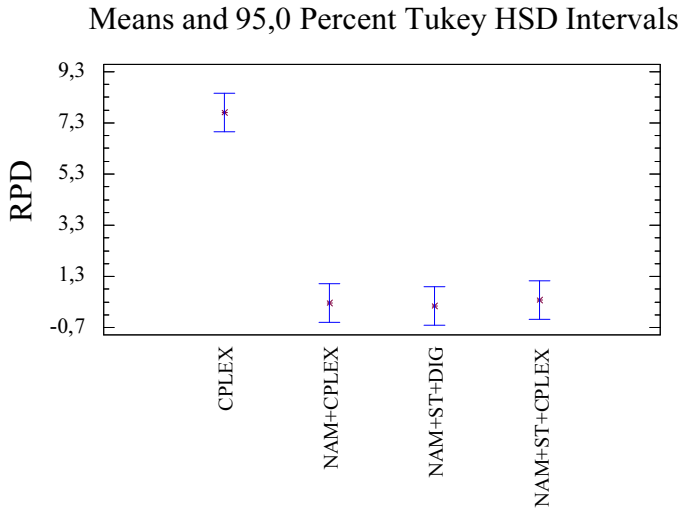

Figura E.38 – ANOVA para el intervalo U(100,200) con 80 % máquinas quitadas y 60 segundos.

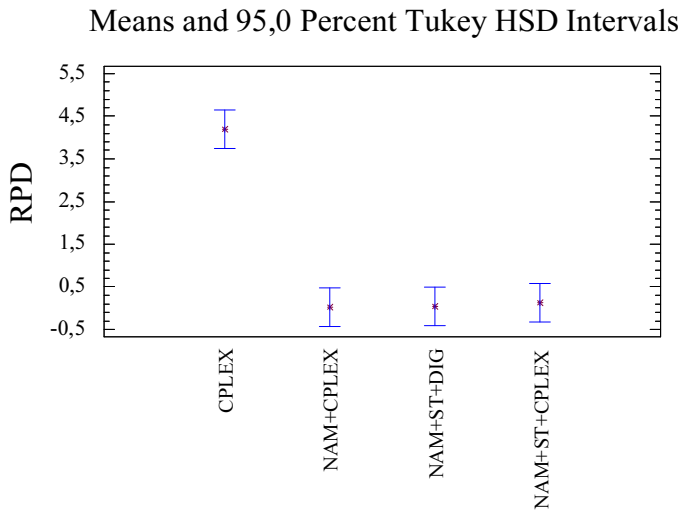

Figura E.39 – ANOVA para el intervalo U(100,120) con 80 % máquinas quitadas y 300 segundos.

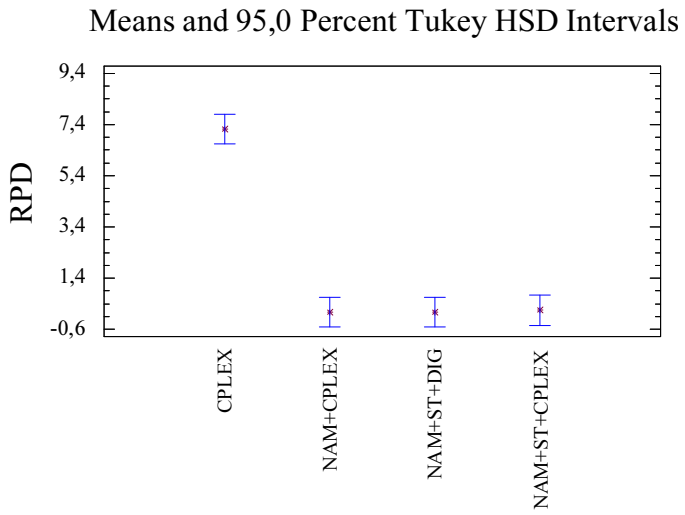

Figura E.40 – ANOVA para el intervalo U(100,120) con 80 % máquinas quitadas y 60 segundos.

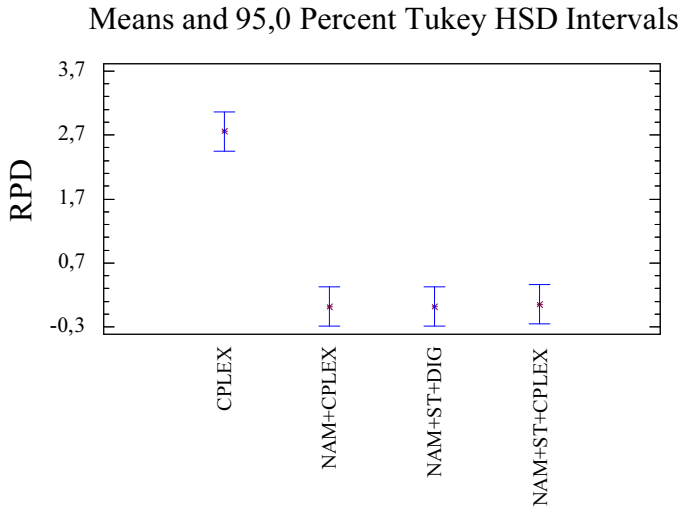

Figura E.41 – ANOVA para el intervalo U(1000,1100) con 80 % máquinas quitadas y 300 segundos.

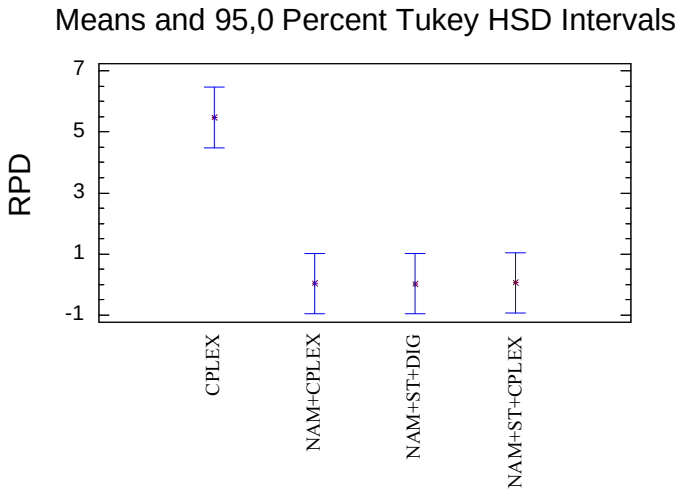

Figura E.42 – ANOVA para el intervalo U(1000,1100) con 80 % máquinas quitadas y 60 segundos.

ANEXO F

ARTÍCULOS ACEPTADOS Y CONGRESOS

Como resultado de todo lo desarrollado en la presente Tesis Doctoral se han presentado ponencias en dos congresos nacionales y en tres congresos internacionales además de tener dos artículos en revistas de alto impacto pendientes de publicación o ya publicados. Los congresos a los que asistimos fueron los siguientes:

- The 21st Conference of European Chapter on Combinatorial Optimization (ECCO XXI), Drubrovnik, mayo 2008. Titulo: "Iterated greedy algorithms for the unrelated parallel machine problem". En el se expuso los resultados de los algoritmos iniciales. Capítulo 3.
- XXXI Congreso nacional de estadística e investigación operativa, Murcia, febrero 2009. Titulo: "Nuevos algoritmos para la resolución del problema de máquinas paralelas no relacionadas". A este congreso nacional se llevo una ponencia relativa a los algoritmos de selección dirigida. Capítulo 4.

- 23rd European Conference on Operational Research (EURO), Bonn, julio 2009. Titulo: “New algorithms for the unrelated parallel machine scheduling problem”. Se mostraron internacionalmente los resultados de los algoritmos de selección dirigida. Capítulo 4.

- 4th International Conference on Industrial Engineering and Industrial Management, San Sebastián, septiembre 2010. Titulo: “Size-reduction methods for the unrelated parallel machines problem and makespan criterion”. Exposición de los algoritmos de reducción de tamaño. Capítulo 5.

- VII Congreso Español sobre Metaheurísticas, Algoritmos Evolutivos y Bioinspirados (MAEB 2010), Valencia, septiembre 2010. Titulo: “Secuenciación en máquinas paralelas no relacionadas con máquinas opcionales y selección de trabajos”. Ponencia sobre el problema más general de máquinas opcionales y selección de trabajos. Capítulo 6.

También exponemos aquí los artículos resultantes de nuestras investigaciones a lo largo del desarrollo de esta Tesis:

En primer lugar esta “Iterated greedy local search methods for unrelated parallel machine scheduling”, ya publicado en la revista EJOR (European Journal of Operational Research) (Fanjul-Peyro y Ruiz (2010a)).

En segundo lugar el articulo "Size-reduction heuristics for the unrelated parallel machines scheduling problem", ya en prensa a la espera de su publicación en la revista C&OR (Computers & Operations Research) (Fanjul-Peyro y Ruiz (2010b)).