



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



ESCUELA TÉCNICA
SUPERIOR INGENIEROS
INDUSTRIALES VALENCIA

TRABAJO DE FIN DE MASTER:
INGENIERÍA AVANZADA DE PRODUCCIÓN,
LOGÍSTICA Y CADENA DE SUMINISTRO

TÍTULO:

ESTADO DE ARTE DE LAS APLICACIONES
DEL CONCEPTO DE *LOT STREAMING* A LA
SECUENCIACIÓN DE TALLERES DE FLUJO

AUTOR: SILVIA JANETH VELECELA ROJAS

TUTOR: DR. PEDRO GÓMEZ GASQUET

CURSO ACADÉMICO: 2017-2018

Dedicatoria

Familia, novio y amigos que siempre han estado presente y mucho mas cuando les he necesitado, sabiendo que este proyecto no fue fácil y sin esperar nada a cambio demostraron su amor cariño compartiendo su conocimiento, alegrías y tristezas durante este recorrido para que este sueño se haga realidad.

Agradecimiento

Gracias a Dios por haberme regalado una oportunidad de un logro mas en la etapa profesional, a todos los formadores del master quienes se han esforzado y nos han trasmitido su conocimiento. A mi tutor por su sabiduría y paciencia para la culminación de este trabajo.

Gracias a todas las personas que de una u otra manera aportaron para el desarrollo de este proyecto.

Resumen

El presente estudio, versa en una revisión de los artículos publicados sobre la aplicación del concepto de *Lot Streaming* y su aplicación en la secuenciación de talleres de flujo. Los documentos se clasificaron de acuerdo a los dimensionamientos que se asocian a evidenciar las combinaciones y comparaciones de diversos algoritmos utilizados para mejorar los talleres de flujo a través de la división de Sub-lotes. Lo cual representa un particular enfoque para la revisión de artículos e investigadores que han abordado esta temática, aplicando una metodología con un enfoque cualitativo, por cuanto la información se sistematizó a través del software Atlas_ti en correspondencia a las variables exploradas en cada estudio que identifican la eficacia de la división de lotes en la solución de problemas de talleres de flujo. La mayoría de los estudios identificaron algoritmos genéticos y genéticos híbridos, mostrando algunas desventajas frente a los tradicionales y en pocos estudios evidenciando la eficacia en su aplicación.

Abstract

The present study is based on a review of the articles published on the application of the Lot Streaming concept and its application in the sequencing of flow workshops. The documents were classified according to the sizing that is associated to evidence the combinations and comparisons of different algorithms used to improve the workshops of flow through the division of sublots. This represents a particular approach to the review of articles and researchers that have addressed this issue, applying a methodology with a qualitative approach, as the information is systematized through the Atlas_ti software in correspondence to the variables explored in each study that identify the efficiency of the division of batches in the solution of problems of flow workshops. The majority of the studies identified hybrid genetic and genetic algorithms, showing some disadvantages compared to the traditional ones and in few studies showing the efficacy in their application.

Índice

| | |
|--|----|
| Dedicatoria | 2 |
| Agradecimiento..... | 3 |
| Resumen | 4 |
| Abstract | 5 |
| Índice | 6 |
| Índice de Tablas | 7 |
| Índice de Figuras | 8 |
| Índice de Anexos | 8 |
| CAPÍTULO 1:..... | 9 |
| 1.1. INTRODUCCIÓN..... | 9 |
| 1.1.1. Contextualización | 10 |
| 1.1.2. Objetivos..... | 13 |
| 1.1.2.1. Objetivo General..... | 13 |
| 1.1.2.2. Objetivos Específicos | 13 |
| 1.1.2.3. Resultados Esperados | 13 |
| CAPÍTULO 2: ESTADO DEL ARTE | 14 |
| 2.1. <i>Lot Streaming</i> | 14 |
| 2.2. <i>Lot Streaming</i> en distintos sistemas de producción | 17 |
| CAPÍTULO 3:..... | 33 |

| | |
|--|----|
| 3.1. METODOLOGÍA DE INVESTIGACIÓN | 33 |
| 3.1.1. Primera Fase: Selección de la idea | 34 |
| 3.1.2. Segunda Fase: Establecimiento del ámbito de estudio..... | 36 |
| 3.1.3. Tercera Fase: Identificación de fuentes de datos. | 37 |
| 3.1.4. Cuarta Fase: Recolección, análisis e interpretación de información..... | 40 |
| 3.1.5. Quinta Fase: Estructura de conclusiones | 41 |
| CAPÍTULO 4:..... | 42 |
| ANÁLISIS DE LOS RESULTADOS..... | 42 |
| 4.1 Perspectivas: Líneas Futuras de Investigación | 73 |
| CAPÍTULO 5:..... | 75 |
| CONCLUSIONES y DISCUSIÓN..... | 75 |
| 5.1. Conclusiones..... | 75 |
| BIBLIOGRAFÍA..... | 79 |
| ANEXOS..... | 87 |

Índice de Tablas

| | |
|--|----|
| Tabla 1: Proceso de investigación | 33 |
| Tabla 2: Selección idea a investigar | 34 |
| Tabla 3: Objetivos y dimensionamientos de estudio..... | 36 |
| Tabla 4: Técnica y fuentes de información | 38 |
| Tabla 5: Estructura recopilación de información | 40 |
| Tabla 6: Estructuración conclusiones finales | 41 |

| | |
|---|----|
| Tabla 7: Identificación algoritmo EDA | 49 |
| Tabla 8: Categorización de dimensionamientos de estudio | 42 |
| Tabla 9: Identificación del algoritmo BMI | 45 |
| Tabla 10: Identificación del algoritmo ABC | 52 |
| Tabla 11: Identificación Algoritmo SFLA | 60 |
| Tabla 12: Identificación algoritmo NGA..... | 67 |

Índice de Figuras

| | |
|---|----|
| Figura 1: Red semántica Combinaciones | 48 |
| Figura 2: Red semántica Comparaciones | 58 |
| Figura 3: Red semántica Marchas | 63 |
| Figura 4: Red semántica Máquinas-Trabajos | 64 |
| Figura 5: Red semántica Sub-lotes..... | 70 |
| Figura 6: Red semántica Secuenciación..... | 71 |
| Figura 7: Red Semántica General..... | 72 |

Índice de Anexos

| | |
|--|----|
| Anexo 1: Esquema de matriz de caracterización de variables por estudio analizado | 87 |
| Anexo 2: Imagen Programa Atlas_ti..... | 88 |

CAPÍTULO 1:

1.1. INTRODUCCIÓN

Lot streaming es un proceso que divide un lote de trabajo en otros más pequeños y luego se los procesa de manera intercalada en las máquinas. Es así que, este concepto según Sarin & Jaiprakash (2007) mejora significativamente el rendimiento general del proceso productivo, con ello, hacer que el funcionamiento de fabricación sea más veloz.

Por otro lado, secuenciación (*scheduling*) constituye el proceso de seleccionar entre planes facultativos, determinando recursos y tiempos a las actividades del plan (*schedule*) (Fiol-Zulueta, A., Arzola-Ruiz, J., & da Silva-Carvalho, A., 2010); es decir, optimiza el proceso en el cual recursos limitados son asignados en un periodo establecido.

Con ello, un conveniente modelo de secuenciación permite determinar una buena regla de programación que facilite el trabajo del responsable en una organización multietapa, en el que habitualmente se encuentran varios criterios en la secuenciación de un producto considerando cada etapa.

En las industrias manufactureras, la planificación de uso general y estrategias de programación de las decisiones se orientan a un enfoque jerárquico, en el que la planificación

se resuelve en primer lugar con el propósito de la producción y los temas de programación se resuelven de la mano al cumplir las metas programadas.

1.1.1. Contextualización

Para Mortezaei & Zulkifli (2013) es necesario desarrollar metodologías que integren efectivamente la producción, planificación y producción. Al respecto *Lot Streaming* minimiza el tiempo de fabricación, proporcionando una oportunidad para bajar los costes de explotación (Ramasesh, R. V., Fu, H., Fong, D. K., & Hayya, J. C., 2000).

La evolución de las organizaciones y las peticiones de las mismas con relación a la personalización de productos con base a los requerimientos de los clientes, motiva un cambio profundo en los procesos productivos. Frente a ello, el dimensionamiento de lotes como táctica complementaria de la función de programación de operaciones en las organizaciones, se valora con relación a las medidas de eficacia, el desempeño de un sistema productivo al utilizar diferentes tamaños de lote de transferencia, el nivel de demanda y la mezcla de los productos (López, M. L. N., Hernández, E. A. G., y Rodríguez, G. A. G, 2011).

La configuración productiva plantea distintos modelos para la elaboración de los productos con el propósito de minimizar los retrasos que se presentan en una secuenciación. Algunas veces por disponer de una sola máquina que utiliza tiempos de preparación en su ejecución o tiempos que influyen en la producción e las máquinas. Al respecto, existen diversos modelos de programación que aportan a optimizar la producción.

El problema de la programación de la producción (scheduling), consiste en, establecer una agenda para las principales actividades de manufactura requeridas para cada trabajo, es decir, tiempos de inicio y fin de cada tarea, para con ello, definir qué equipo ejecutará cada operación de maquinado. Al respecto, se considera cada lote atraviesa por todas las operaciones requeridas como una unidad invisible, y cuya división menor no está permitida (Novas, 2016).

La separación de los lotes en Sub-lotes (Lot streaming) más pequeños, permite que todas las operaciones demandadas de forma sucesiva por un mismo lote puedan superponerse de manera temporal, pero, siempre y cuando los Sub-lotes distintos sean asignados a máquinas diferentes, permitiendo completar de forma anticipada todo el proceso de maquinado a un lote determinado, considerando la

composición y cantidad de Sub-lotes mientras la programación se resuelve de manera simultánea a las tareas de manufactura requeridas (Low, HSU, Huang, Mortezaei, 2013).

La formulación general del problema de la programación de la producción, se puede recurrir a un enfoque de programación matemática, dado que se puede procesar en un conjunto de máquinas, dado que todos los trabajos comparten la misma secuencia de procesamiento, a través de equipos y todos los equipos procesan trabajos siguiendo la misma secuencia, dado que este tipo de trabajos mejora la productividad que suelen verse comprometidas en el entorno industrial (Minella, Ruiz, Ciavotta, 2011).

Existen varios estudios que han abordado esta temática, sin embargo, pocos han enfatizado en contrastar los métodos utilizados y la eficiencia de los mismos al momento de reducir los tiempos y optimizar los procesos productivos. Frente a esta particularidad surge la interrogante ¿revisar los métodos empleados en la resolución de problemas de optimización en talleres de flujo aportará a fortalecer los conceptos de *Lot Streaming* que actualmente se aplican? La interrogante planteada orienta a plantearse los siguientes objetivos que persigue el estudio.

1.1.2. Objetivos

1.1.2.1. Objetivo General

Estructurar un estado del arte del concepto de *Lot Streaming* y su aplicación a la secuenciación en talleres de flujo.

1.1.2.2. Objetivos Específicos

- Identificar las variables que influyen en la secuenciación del taller de flujo.
- Clasificar el problema que implica la división de lotes y su mejoramiento en el rendimiento del proceso productivo.
- Comparar las ventajas y desventajas de la aplicabilidad de *Lot Streaming* a la secuenciación en talleres de flujo.

1.1.2.3. Resultados Esperados

- Determinar la metodología aplicada en cada mediante la revisión documental.
- Analizar y clasificar la metodología óptima para la efectividad del rendimiento de lotes de los procesos productivos.
- Establecer las principales ventajas y desventajas de *Lot Streaming* dentro de la secuencia de talleres de flujo

CAPÍTULO 2: ESTADO DEL ARTE

2.1. *Lot Streaming*

Según Chang & Chiu (2005) la distribución de lotes combinados con superposición de operaciones es una de técnica eficaz utilizada para implementar la estrategia basada en el tiempo, considerando la competencia global. Por lo tanto, esta práctica ha sido estudiada extensamente las últimas décadas.

El término transmisión por lotes fue presentado por primera vez por Reiter (1966). El concepto luego fue postulado por Lundrigan (1986). Un lote de transferencia, es decir, un Sub-lotes no puede, y a menudo no debería ser igual a un lote de procesamiento. Claramente, *streaminges* relaciona una técnica, en la que, un lote de procesamiento se divide en varios Sub-lotes y se superpone realizando operaciones en diferentes estaciones de trabajo de fabricación, de esta forma, la producción puede acelerarse.

Hasta fines de la década de 1980, la transmisión por lotes era investigado independientemente de la producción tecnológica de optimización (OPT). Desde entonces, esta técnica ha sido extensamente investigada en campos académicos e industriales, demostrando que es eficaz

técnica para comprimir MLT (Kher, H. V., Malhotra, M. K., & Steele, D. C., 2000).

Es así que los métodos se clasifican de manera aproximada y exacta, siendo el primero es tipo meta-heurística, como es el caso de un caso de dos máquinas, en el cuál se establece un supuesto, en otras dimensiones para compararlos con métodos tradicionales.

Para Bagchi, Gupta & Sriskandarajah (2006) en el entorno de producción, es necesario programar muchos productos. Unidades idénticas de un producto, que se agrupan en lotes de producción, es decir, un lote se transfiere de una máquina a otra, solo cuando todos los artículos del lote han sido procesados. Se divide en varios Sub-lotes para que las operaciones puedan superponerse; se lo conoce como transmisión de lotes, procedimiento que reduce los tiempos de inactividad en las máquinas siguientes.

El autor Castillo & Anselmo (2014), considera que una correcta planificación y programación de actividades (scheduling) son procesos de toma de decisiones para optimizar los objetivos propuestos, utilizando algoritmos evolutivos, buscando la efectividad del tiempo total en la ejecución de las tareas, el tiempo de respuesta. Estos algoritmos incluyen versiones secuenciales e

implementaciones en paralelo, el procedimiento que emplea es de la estimación de distribuciones para resolver un problema real en programación de tareas, describiendo eventos discretos con algoritmos de estimación de distribuciones, adaptando técnicas heurísticas de optimización por enjambre de partículas.

Un estudio de Lot streaming en tienda de flujo híbrido, trata sobre la transmisión de lotes, que mediante el flujo de hibridación considera desarrollar un cronograma de tareas, cuyo objetivo es minimizar el makespan; consiste en una máquina en cada una de las etapas (N . etapas - 1) y fases (N). Propone un modelo de programación lineal mixta (MILP), enfocado de manera heurístico en la división y secuenciación de lotes; basados en tiempos promedios, los resultados muestran que la desviación porcentual de la solución optimiza y brinda una solución óptima a los procesos (Lazmi, Naru, Madhusudanan, 2017).

Los autores (Hong, Ke-Pan, Duan, 2018) Una representación de la permutación heurística adaptada al nivel de calidad, diseña un algoritmo de dispersión espacial basada en la distribución normal en función tangente para mejorar de la productividad global de los procesos, cuyos resultados dentro de los experimentos numéricos con el uso de los algoritmos constituye una solución para los

problemas de flowshop de lot-streaming con tiempos reducidos dentro del criterio makespan.

El problema de programación de bloqueo del flujo de lotes (BLSFS), tienen aplicaciones considerables en varios sistemas industriales, un modelo de optimización el que propone este estudio, es un algoritmo mejorado para la optimización de aves migratorias (IMBO) para minimizar el makespan, utilizando permutaciones discretas para representar soluciones y aplicar múltiples operaciones de inserción, empleando un sistema de distribución el algoritmo (EDA) para la mejora en la capacidad de explotación local en el conocido benchmark de lot streaming, según los autores (Han, Hongyan, Yun Bao, Qun sun, 2018).

2.2. Lot Streaming en distintos sistemas de producción

Resulta pertinente para iniciar este apartado hacer un acercamiento de la opinión de autores con respecto a la gestión, considerando el enfoque, para el estudio se tomó la postulación de (Becerra, Cárdenas, Castrillón, García, Ibarra y Zapata (2008) quienes destacan que la producción puede constituirse desde un sistema, como una función organizacional y como la realización con base a un conjunto jerárquico de decisiones, teniendo calificaciones en niveles estratégicos, tácticos y operativos.

Existen varios sistemas de producción, según Chase, Aquilano, Ciociano, García y Jacobs (2000) entre los cuales se encuentra el *Flow Shop* que constituye el tipo de fabricación con enfoque repetitivo, determinado por series de trabajo y para que se ejecute tiene que pasar por todas las máquinas en el mismo orden con respecto a los procesos y la secuenciación.

Es así que, Ramasesh, Fu, Fong & Hayya (2000) propusieron un modelo para calcular el tiempo de ciclo de fabricación de un elemento en un sistema de producción, al igual que, el tamaño del lote que minimizara económicamente el coste total cuando se utiliza *lot streaming*. Planteando un prototipo que toma en cuenta los costos asociados con la preparación, transporte, producto en proceso y terminado inventariado; luego establece procedimientos para encontrar el tamaño del lote de producción óptimo que logre el objetivo. A través de ejemplos numéricos se muestrea costes significativos en las comparaciones con enfoques tradicionales.

En el prototipo propuesto se realiza un sistema de inventario general para un solo elemento de un sistema de multietapas; adicionando el costo de inventario de productos en procesos. Constituye un modelo robusto en los componentes de fabricación del tiempo de espera, al igual

para asignar un valor al tiempo que requiere para la configuración, facilitando su aplicación para el *lot streaming* a sistemas multietapas.

Un estudio realizado Kalir & Sarin (2001) muestra resultados analíticos relativos a los beneficios potenciales de *lot streaming* en sistemas de taller de flujo, utilizando medidas como el *makespan*, el flujo de medio tiempo y nivel promedio de WIP. Para cada expresión el índice de medida de *lot streaming* a la medida sin *lot streaming* es desarrollada, utilizando los beneficios en determinadas condiciones; ya que, en casos extremos las expresiones dependen del parámetro del problema (número de máquinas número de lotes el tamaño del lote) Luego del análisis tiene la hipótesis que la transferencia y los tiempos de preparación son insignificantes.

Otra investigación realizada por Kumar, Bagchi & Sriskandarajah (2000) en la programación de lotes, buscó minimizar el *makespan* con máquinas sin espera en el taller de flujo con múltiples productos. Cuando hay un número fijo de Sub-lotes se utiliza una heurística y se obtiene Sub-lotes óptimos de tamaño continuo. Cuando se tiene esta condición de un solo producto se resuelve mediante el método del comerciante viajero. Se construyen heurística (S &W) para Sub-lotes de tamaño entero.

Se evalúa el uso de meta-heurísticas en un problema de permutación NP para minimizar el *makespan*, posteriormente se realiza una heurística MHEU combinada (S&W y TPS) para encontrar la secuencia de los lotes; analizando con la meta-heurística GA optimizar cada una de las tres etapas.

Según las comparaciones realizadas por diferentes criterios utilizados el número de Sub-lotes puede optimizarse por el algoritmo genético que ayuda a mejorar el *makespan* ligeramente con respecto a las heurísticas especiales y optimiza el número de Sub-lotes para cada tarea.

Por otro lado, un estudio relacionado con la heurística óptima en lotes múltiples realizado por Kalir & Sarin (2001) identificó que el problema de secuencia *lot streaming* en un ambiente multi-lotes, multi-máquinas en un taller de flujo; los Sub-lotes no pueden entremezclarse, tomando en cuenta que se ha desarrollado una heurística casi óptima para esta solución.

Con ello, se propone una heurística de mínima inactividad de cuello de botella (BMI), que identifica y utiliza propiedades irregulares en los problemas de taller de flujo. Esta heurística intenta maximizar el *buffer* antes de la máquina de cuello de botella y a la vez secuenciar los lotes.

Los resultados generados se acercan a las soluciones óptimas comparadas a las generadas a través del uso de la heurística de la inserción rápida, que es considerada una de las mejores para la resolución de programación de taller de flujo cuando se aplica manualmente.

En lo concerniente a problemas de programación de “n” trabajo y “n” maquina *lot streaming* que considera varios tipos de *buffers* y Sub-lotes. Al respecto Yoon & Ventura (2002) sostiene que el propósito es minimizar la desviación absoluta media ponderada de las fechas de vencimiento cuando los trabajos se programan en un taller de flujo *lot streaming*.

Considerando que no es regular que se necesite una búsqueda entre la programación de inactividad insertados para encontrar la solución óptima. Por ello, se presenta una programación lineal obteniendo tiempos de finalización óptimos en caso de que los *buffers* en máquinas sucesivas limiten las capacidades “p” infinitas; y los Sub-lotes tengan el mismo tamaño.

Incluso se presentan algoritmos mediante la combinación de reglas, resultando algoritmos heurísticos con un método de intercambio de pares no adyacentes; y la regla de tiempo de

holgura global más pequeña generando el inicial con la heurística del más pequeño tiempo de holgura global (OSL). El intercambio de pares no adyacentes (API); si bien es cierto, una es para problemas de programación con una sola máquina, adaptada para que al compararla genera resultados favorables en el promedio.

Así mismo, otra investigación desarrollada por Chiu, Chang & Lee (2004) respecto a modelos de lotes con un número limitado de transportadores en sistemas de producción multietapas, evidenciaron un inconveniente de programación de taller de flujo con *lot streaming* con un sistema de varias etapas que incorpora las actividades de transporte.

Se propone un modelo matemático orientado al tiempo de inicio óptimo, el número de lotes y la asignación de transferencia, por otro lado, valora con las dos alternativas heurísticas eficientes, la utilidad en la práctica y la cantidad de tiempo que tarda en resolver el modelo matemático. Este experimento tuvo dos fases: medir la eficacia y la veracidad de los resultados.

Incluso otros análisis versan en modelos de coste para transmisión de lotes en una tienda de flujo multietapa, como el propuesto por Chiu & Chang (2005) constituido por dos

modelos de costos para la solución de problemas de *lot streaming* en un taller de flujo en etapas múltiples con el objetivo de establecer el tamaño recomendable de procesamiento y número de lotes de transferencia que reduzcan al mínimo el coste total.

En realidad, el primer modelo es más completo y preciso para calcular costos de materia prima, trabajo en proceso y los inventarios de producto terminado. Por lo que atañe, al segundo modelo adiciona el coste asociado con el tiempo de *makespan*; realizando un experimento que explora la solución extremada cuando se cambia el valor de un parámetro a la vez, indicando que algunos parámetros tiene efectos significativos en la solución óptima.

De cualquier manera, cuando se explora alternativas de optimización de partículas para el problema de programación *flowshop* por lotes, estudios como el de Tseng & Liao (2008) destacan talleres de flujo con “n” trabajos y “m” máquinas con Sub-lotes de similar tamaño con la finalidad de reducir el nivel de anticipación y tardanza, con base al algoritmo de beneficio neto de movimiento; en donde se asume que es mejor que la programación lineal, incorporando el método *swarm* y un algoritmo genético híbrido.

Cuando se aplica el algoritmo genético híbrido y el de beneficio neto de movimiento, resulta más rápido y se ahorra tiempo en el cálculo. Las derivaciones muestran que es competitivo con el sistema de herencia de puntos; puesto que, la construcción de partículas es rápida y de mejor calidad debido al equilibrio que se obtiene en la exploración de lasas nuevas y tomando en cuenta las anteriores.

Al considerar algoritmos para solucionar problemas de programación, se observó estudios con distintas variantes como el desarrollado por Marimuthu, Ponnambalam & Jawahar (2008) con una propuesta de algoritmos evolutivos para una tienda de flujo “m” máquinas con demasiada transmisión, estudio que abordó la toma de decisión de secuenciación en el taller de trabajo: a) con “n” trabajos y “n” máquinas de igual tamaño de lote, el objetivo consistió en valorar dos heurísticas evolutivas como son: el algoritmo genético y algoritmo genético híbrido con criterios de tiempo de preparación, secuencia y mejora del *makespan*.

Al respecto, se utilizó una metodología de generación aleatoria para el flujo total de tiempo de trabajo, comparada con el algoritmo de *Baker* (AB) y el recocido simulado (SA). Observándose, que el algoritmo genético y genético híbrido son capaces de aportar soluciones optimas, porque, se comportan mejor que los otros algoritmos.

De igual forma un estudio realizado por Martin (2009) con notaciones genéticas Ahíbridas con orientación matemática para el problema de *Flowshop* con transmisión de lotes, encontró la vinculación de una heurística algoritmo genético híbrido con programación matemática que ayuden a minimizar el *makespan*, incluyendo los tiempos de preparación dependientes de la secuencia, con la finalidad de entrelazar los Sub-lotes de los puestos de trabajo en la secuencia.

Con ello, se inicia con un AG para determinar el número de Sub-lotes y la secuencia, luego se utiliza el modelo matemático para definir el tamaño de los Sub-lotes; es así que, dentro de los resultados se logra reducir el *makespan*, porque, favorece el entrelazado, siendo beneficioso y aplicable a otros problemas.

Aparte se indagó otras opciones en la programación de flujos de tiempos; es más, para Kim & Jeong (2009) el algoritmo genético adaptativo minimiza el *makespan* con un problema de programación de taller de flujo sin esperar que *lot streaming* sean flexibles.

En el estudio se enfatiza, que los Sub-lotes no pueden mezclarse; rescatando que el método funciona con cruce de

productos y mutación basada en la búsqueda local, o sea, aplicando un algoritmo iterativo de subida de pendientes para mejorar la generación actual, así como la regulación del cruce y las tasas de mutación; notándose, resultados experimentales que, al contrarrestarlos con el algoritmo genético tradicional, es prometedor.

Con estos antecedentes, es relevante incorporar en la revisión bibliográfica, estudios como el de Pan, Tasgetiren, Suganthan & Chua (2011) que aporta con un algoritmo de colonia artificial de abejas para el problema de programación de flujo de lotes, que se direcciona utilizando el criterio de anticipación ponderada y sanciones de tardanza con casos de máquinas sin parada y en trabajo continuo.

Este algoritmo se emplea en espacios discretos combinatorios, fundamentado en la permutación de generación de nuevas fuentes, considerando criterios de fecha de vencimiento, tiempo y normas de holgura más pequeño con respecto a la última máquina y la generación de la población inicial; siendo los resultados eficaces en el rendimiento.

Pan, Wang, Gao & Li (2011) resuelven un problema de programación de *lot streaming* en el taller de flujo con Sub-

lotes de tamaño iguales para minimizar el *makespan* con criterios idle y no idle; proponiendo un algoritmo shuffle frog-leaping (SFLA). El cual, funciona como una representación con base a permutaciones de trabajo discretos y utiliza un operador de cruce fundamentado en condiciones de generar nuevas soluciones.

Se construyó una población inicial aplicando la heurística Nawaz – Ensore – Ham, experimentando distintas comparaciones, para tener una efectividad de la propuesta. Considerando que los talleres de flujo de lotes tienen aplicaciones en distintas actividades industriales, un estudio realizado por Pan & Ruiz (2012) abordaron el problema de programación de flujo de lotes con “n” máquinas y trabajos *lot streaming* con tiempos de configuración relacionados con la secuencia.

Un estudio relevante fue el realizado por Ventura & Yoon (2013) al postular un algoritmo genético (NGA) para programar flujo de lotes con buffers de capacidad limitada, teniendo como finalidad que el algoritmo reemplace los operadores de selección y apareamiento a algoritmos genéticos.

De igual forma, Mortezaei & Zulkifli (2013) desarrollo un modelo matemático para integrar el dimensionamiento de

lotes y la programación de Flow Shop con la transmisión de lotes, probando ocho inconvenientes, el primero con Sub-lotes entremezclados; consistentes y sin mezcla; el tercero subotes iguales con entremezcla; el siguiente iguales y sin entremezcla; el quinto no-wait de iguales con entremezcla; el sexto n-wait consistentes sin mezclar; y el último no-wait con Sub-lotes iguales sin entremezclar. Evidenciando que el mejor *makespan* se logra con Sub-lotes constantes. Al contrario de la investigación de Sang (2010) que propuso un algoritmo de evolución diferencial discreto (DDE) con base a permutaciones de trabajo, utilizado para realizar una explotación global con un procedimiento de búsqueda local que considera la estructura de vecindad de inserción y canje, utilizado para mejorar la capacidad de explotación, demostrando a través de los análisis computacionales la efectividad del algoritmo.

Davendra, Senkerik, Zelinka, Pluhacek & Bialic-Davendra (2014) utilizando un algoritmo de migración auto organizado (DSOMA) como generador pseudoaleatorio¹, el cual se aplicó al problema de programación Flowshop y flot-lot con base a la restricción, mostrando que el algoritmo de la variante del inconveniente mejora significativamente el rendimiento con el algoritmo propuesto.

¹ Proceso que genera un número al azar, pero no lo hace efectivamente, pues las secuencias no muestran ningún patrón

Han, Gong, Sun & Pan (2014) an una alternativa de mejora a un algoritmo de cruce y mutaciones (NSGA-II) orientado a resolver el problema de programación de flujo de lotes a través de cuatro criterios: a) variantes de la heurística NEH; b) incorporando la estimación del algoritmo de distribución; c) reemplazando los operadores tradicionales; y d) se aplicó una estrategia de reinicio. Dando como resultado luego de los experimentos, que el algoritmo postulado supera a los algoritmos base.

Incluso un estudio realizado por Chakaravarthy, Marimuthu, Ponnambalam & Kanagaraj (2014) desarrollo una heurística de búsqueda inteligente para resolver los problemas de talleres de flujo, siendo el propósito restar el tiempo el flujo total y de makespan de manera independiente, los que mostraron mejor rendimiento de los algoritmos ISFHA y ABC.

Zhang & Wang (2014) por otro lado, analizaron la transmisión de lotes, abordando el problema de programación de flujo continuo sin espera con la restricción de la interconexión de Sub-lotes de tamaño mediante un algoritmo de optimización mosca agrupada (GFOA), el cual, al aplicar las pruebas respectivas muestra efectividad.

El problema de programación en Job-shop, trata de sistemas de manufactura flexibles, pretendiendo resolver el criterio de minimizar el tiempo máximo de culminación de todas las tareas, comparando diversos operadores y mutaciones con el uso de permutaciones y un modelo probabilístico utilizado en una nueva generación de soluciones, cuyos resultados, para la optimización de sistemas de manufactura capaces de procesar diversos trabajos compartidas por diferentes máquinas en la distribución y división de lotes en la programación de tareas (Wang, Xu, Zhou, Liu, 2015).

Dentro de las particularidades de los modelos estándares de programación de la producción, cuya importancia sea la reducción de costos en la fabricación textil, este estudio propone una nueva tecnología basada en algoritmos genéticos (secuencia en tiempos de montaje, no relaciones entre máquinas paralelas, fechas de entrega establecidas); cuyos resultados muestran un reducción completa del porcentaje de cumplimiento de entregas, siendo una metodología efectiva en la aplicación de sectores industriales, considerando las complejidades como los lotes de transferencia variables, entrada dinámica y la maleabilidad, cuyos procesos se pueden considerar para futuras investigaciones (Arango, 2015).

Para el autor Novas J. (2016), analiza la programación de las tareas de manufactura en ambientes de Job-Shop, la cual, considera la posible división de lotes de trabajo en sub lotes de menor tamaño, dicho modelo permite desarrollar y establecer la distribución necesaria de cada lote, considerando en conjunto de restricciones. La propuesta determina los tiempos de inicio y fin de operaciones de manufactura que se realizan los Sub-lotes y los equipos adecuados, mediante el modelo misto lineal (MILP), cuyos resultados favorecieron a las actividades de los procesos minimizando efectivamente los tiempos de la productividad.

Todavía más, Sang, Pan, Duan & Li (2018) propuso un algoritmo de optimización de hierbas invasivas discretas (DIWO) asociada a una permutación de trabajo y heurística adaptada NEH, aplicado a través del diseño de experimentos comparado con algoritmos de mejor rendimiento en total 280 momentos, generados aleatoriamente, dando resultados significativamente mejores que los algoritmos de referencia.

Han, Gong, Jin & Pan (2016) presentaron una posible solución a los problemas de programación de flujo de lotes, con base a la aplicación de un algoritmo de planificación robusto evolutivo multiobjetivo (REMO), el cual fue aplicado a 22 estimaciones de prueba y se comparó con algoritmos

de referencia, resultados empíricos que demuestran su eficiencia.

Al contrario, Rossit, Tohmé, Frutos, Bard & Broz (2016) investigaron el uso de la transmisión por lotes en problemas de programación de flowshop que no son de permutación. Con base a las experimentaciones computacionales realizadas muestran que la transmisión por lotes reduce hasta un 43% para una amplia comparación con el caso de qué no se aplica la división del trabajo.

CAPÍTULO 3:

3.1. METODOLOGÍA DE INVESTIGACIÓN

Tomando en cuenta que el objeto de estudio y los supuestos teóricos de la investigación orientar a escoger la metodología a utilizar, se escogió una metodología con enfoque cualitativo por su orientación exploratoria.

Para el desarrollo del estudio se consideraron las siguientes etapas de investigación: a) diseño de la investigación; b) identificación de las técnicas de estudio; c) acceso de información y ámbito de estudio; d) selección de los documentos, revistas, artículos académicos y científicos que aportarán información; e) procesamiento de información y estructuración de las conclusiones.

Al respecto, la investigación se esquematiza considerando las siguientes fases:

Tabla 1: Proceso de investigación

| Fases | Detalle |
|--------------|---|
| 1 | Selección de la idea a investigar |
| 2 | Establecimiento del ámbito de estudio |
| 3 | Identificación de fuentes de datos |
| 4 | Recolección, análisis e interpretación de información |
| 5 | Estructura de conclusiones finales |

3.1.1. Primera Fase: Selección de la idea

En esta etapa es necesario desarrollar metodologías que integren efectivamente la producción, planificación y producción, descritas en la sección de Contextualización (Capítulo I), para lo cual, se identificó el tema de estudio y las características que motivaron analizar la aplicación de *Lot Streaming* en la secuenciación de talleres de flujo, describiendo su estado actual y contexto con base al objeto de estudio, describiendo información relevando de la aplicabilidad de esta herramienta en la optimización de los procesos productivos.

Está revisión se enfocó específicamente en el *Lot Streaming*, que aportó a la comprensión de su utilización en la gestión de producción, analizado dentro de los objetivos establecidos, actividades y productos, como muestra en la Tabla 2:

Tabla 2: Selección idea a investigar

| Objetivo | Actividad | Producto |
|---|--|---|
| Compilación de datos con relación a la aplicación del <i>Lot Streaming</i> | Revisión bibliográfica en libros, revistas, documentales, etc., revisados en plataformas virtuales | Identificación de la problemática, estructuración y contextualización |

| | |
|--|--|
| | <p>como: IEEE, de la idea de sciencedirect, scopus, investigación dialnet, redalyc, google scholar.</p> |
| | <p>Revisión bibliográfica considerando el siguiente proceso:</p> <ol style="list-style-type: none"> a) Identificación de documentos b) Selección de documentos c) Extracción de información relevante d) Recopilación, resumen y trasmisión de datos |
| <p>Identificación de los aportes de autores que han aportado sobre la aplicación del <i>Lot Streaming</i></p> | |
| <p>Particularidades del proceso de secuenciación de talleres de flujo</p> | <p>Revisión conceptual del proceso y su alcance en el mejoramiento de producción</p> |

3.1.2. Segunda Fase: Establecimiento del ámbito de estudio

Luego de analizar los objetivos de estudio, y lo supuestos teóricos de la investigación, una correcta orientación para establecer la metodología a emplear, se escoge un enfoque cualitativo, para lo cual, tornará a una orientación exploratoria para identificar los distintos procesos, actividades, programación operativa para aplicación del Lot streaming; de acuerdo al Estado del Arte, siendo el análisis de los estudios relacionados y de teorías aplicadas en diversos estudios, para lo cual, se determina el enfoque de los objetivos y dimensiones.

Bajo este contexto, en la Tabla 3 se muestra la relación entre los objetivos y dimensiones teóricas de análisis establecidas en la investigación.

Tabla 3: Objetivos y dimensionamientos de estudio

| Objetivos | Dimensiones |
|---|----------------------|
| Identificar las variables que influyen en la secuenciación del taller de flujo | ▪ Combinación |
| | ▪ Comparación |
| | ▪ Idling |
| | ▪ No Idling |
| | ▪ Número de máquinas |
| | ▪ Número de trabajos |
| | ▪ Tipo de Sub-lotes |

| | |
|---|---|
| | <ul style="list-style-type: none"> ▪ Tipo de preparación ▪ Dependencia de la secuencia ▪ Tiempo de traslado ▪ No wait |
| <p>Clasificar el problema que implica la división de lotes y su mejoramiento en el rendimiento del proceso productivo</p> | <ul style="list-style-type: none"> ▪ Cualidades de optimización ▪ Acoplamiento del concepto de <i>Lot Streaming</i> a la secuenciación |
| <p>Comparar las ventajas y desventajas de la aplicabilidad de <i>Lot Streaming</i> a la secuenciación en talleres de flujo</p> | <ul style="list-style-type: none"> ▪ Análisis de información a través de programa de datos cualitativos Atlas_ti ▪ Valoración de las ventajas y desventajas a través de una red semántica |

3.1.3. Tercera Fase: Identificación de fuentes de datos.

Una vez identificado el ámbito de estudio, y con base a los metodologías de los objetivos específicos planteados y la correspondencia con los fundamentos teóricos se definió las técnicas de recolección de información, teniendo como la primordial, basados en el análisis documental, de las

teorías, fundamentaciones y análisis de los procesos, identificados en las dimensionalidad, técnicas y fuentes de información, en referencia de los objetivos de estudio.

Con respecto a esta técnica se identificaron las fuentes fiables para la obtención de información, en la Tabla 4 se muestra la relación entre las etapas descritas y analizadas dentro del Estado del Arte de la investigación.

Tabla 4: Técnica y fuentes de información

| Objetivos | Dimensiones | Técnicas | Fuente de información |
|--|--|--|---|
| Identificar las variables que influyen en la secuenciación del taller de flujo | <ul style="list-style-type: none"> ▪ Combinación ▪ Comparación ▪ Idling ▪ No Idling ▪ Número de máquinas ▪ Número de trabajos ▪ Tipo de Sub-lotes | Revisión bibliográfica y análisis documental | Ranga V. Ramasesh!,*, Haizhen Fu", Duncan K.H. Fong#, Jack C. Hayya# <hr/> Adar A. Kalir, Subhash C. Sarin* |

| | | |
|---|---|--|
| | <ul style="list-style-type: none"> ▪ Tipo de preparación ▪ Dependencia de la secuencia ▪ Tiempo de traslado ▪ No wait | <p>Subodha Kumara,*, Tapan P. Bagchib, C. Sriskandarajaha</p> <hr/> <p>Suk-Hun Yoon, Jose A. Ventura</p> |
| <p>Clasificar el problema que implica la división de lotes y su mejoramiento en el rendimiento</p> | <ul style="list-style-type: none"> ▪ Cualidades de optimización ▪ Acoplamiento del concepto de <i>Lot Streaming</i> a la secuenciación | <p>Quan-Ke Pan a, M. Fatih Tasgetiren b, P.N. Suganthan c,*, T.J. Chua d</p> <hr/> <p>Quan-Ke Pan a, P.N. Suganthan b,†, J.J. Liang c, M. Fatih Tasgetiren</p> |
| <p>Comparar las ventajas y desventajas de la aplicabilidad de <i>Lot Streaming</i> a la secuenciación</p> | <ul style="list-style-type: none"> ▪ Análisis de información a través de programa de datos cualitativos Atlas_ti ▪ Valoración de las ventajas y desventajas a través de una red semántica | <p>Resultados Atlas_ti</p> |

3.1.4. Cuarta Fase: Recolección, análisis e interpretación de información.

En esta fase, luego de analizar, determinar y evaluar los estudios y revisiones documentales de artículos relacionados a nuestro tema de investigación, en este punto, se obtuvo información relevante y posteriormente se ingresó al Software Atlas_ti para una reestructuración de red semántica y conocer su enlace entre estudios y su importancia dentro del Lot streaming en la programación de producción.

Se ejecutó la recopilación de información e interpretación de la misma, que posteriormente dieron origen a los resultados del estudio. En la Tabla 5 se muestra la estructura de recopilación de datos.

Tabla 5: Estructura recopilación de información

| Objetivo | Actividad | Producto |
|------------------------------------|----------------------------------|---|
| Recopilación de información | Análisis de documentos escogidos | Respuesta a la interrogante planteada y |
| Análisis | Ingreso al programa Atlas_ti | alcance de los objetivos |
| Interpretación | Estructuración Red semántica | planteados |

3.1.5. Quinta Fase: Estructura de conclusiones

En este punto, al conocer los resultados de las diferentes estudios relevantes y estructurados por la red semántica del Software empleado, se obtiene las conclusiones, las cuales muestran el interés de estudio, y aplicación para futuras investigaciones relacionadas con revisión de estados del arte y poder conocer y evaluar, el impacto de los diferentes algoritmos empleados y su desenvolvimiento dentro de los procesos industriales productivos y en cumplimiento de los objetivos empresariales.

Con base a los resultados obtenidos se establecieron las conclusiones del estudio.

Tabla 6: Estructuración conclusiones finales

| Objetivo | Actividad | Producto |
|--------------------------------|---|--------------------------|
| Estructura conclusiones | Inferir los resultados y revisar las conclusiones | Trabajo de Fin de Master |
| Ajustes | Correcciones pertinentes | |

CAPÍTULO 4:

ANÁLISIS DE LOS RESULTADOS

A continuación, se describen los resultados cualitativos de la revisión bibliográfica realizada a través de los buscadores utilizados como: Scopus, Google Scholar, Science Direct, Polibuscador, IEEE, utilizando como filtro de la literatura palabras claves como: Lot streaming, Flowshop, Secuenciación, Sublots, además de incorporar un período de análisis desde el año 2000 al 2016. La revisión fue aplicada a 70 artículos, considerando los idóneos con respecto al alcance del presente estudio, se escogió 30 documentos, seleccionando los aspectos relevantes con relación a las categorías identificadas con la temática abordada:

Tabla 7: Categorización de dimensionamientos de estudio

| Categoría principal | |
|------------------------------|----------------|
| Combinación-Comparación | |
| Categorías emergentes | Códigos |
| Combinación | Comb EPL b= 5 |
| | Comb EPL b= 1 |
| | Comb-RRV |
| | Comb-BNM |
| | Comb-SFA |
| | Comb-PMX |

| | |
|------------------------------|----------------|
| | Comb-Heu |
| | Comb-EDA |
| | Comb-TSP |
| | Comp=API |
| | Comp=LSL |
| | Comp=Heu |
| Comparación | Comp=Alg.genet |
| | Comp=NEH |
| | Comp=ACO |
| | Comp=TA |
| | Comp=NEH |
| Categoría principal | |
| Marchas | |
| Categorías emergentes | Códigos |
| Idling | Idl |
| No Idling | No-Idl |
| Categoría principal | Códigos |
| Máquinas-Trabajos | |
| Categorías emergentes | Códigos |
| Número de máquinas | Num-maq |
| Número de trabajos | Num-tr |
| Categoría principal | |
| Sub-lotes | |
| Categorías emergentes | Códigos |
| Tipo de Sub-lotes | Tip-subl |
| Tipo de preparación | Tip-pre-setup |
| Tipo Sub-lotes sin espera | Tip-sp |
| Categoría principal | |

| Secuenciación | |
|-----------------------------|---------|
| Categorías emergentes | Códigos |
| Dependencia de la secuencia | Dep-sec |
| Tiempo de traslado | Ttras |
| No wait | Nw |

Categoría “Combinación-Comparación”

Estudios como el de Ramasesh et al., (2000) destacan que la transmisión de lotes es un procedimiento en el que un lote de producción se divide en Sub-lotes más pequeños y se desplaza al siguiente procesamiento. Para que las operaciones en las etapas sucesivas de un sistema de fabricación, que tiene varias fases puedan superponer en el tiempo.

Ramasesh et al., (2000): “Método utilizado tamaño de lote de producción ELQ- Tamaño de lote óptimo WIP.

Kumar et al., (2000): “aplicó S&W, TPS, NP, GA”

Yoon et al., (2002): “utilizó método OSL/NAPI; con una combinación API, NAPI, EDD”

Kalir & Sarin (2001): “estudio que aplicó una metodología heurística de mínima inactividad del cuello de botella (BMI)

Tabla 8: Identificación del algoritmo BMI

| Algoritmo | Características |
|--|---|
| BMI (Cuello de botella de inactividad mínima) | Explora las características de significativas y esenciales para minimizar el tiempo de inactividad de la máquina de cuello, sujeta a tamaños pequeños |

La heurística de BMI para el LSSP tiene los siguientes detalles:

Paso 1: Inicia k ; S conjunto de lotes $S \equiv \{1; 2; \dots; N\}$;
Calcular:

$$T_j = \sum_{i=1}^N Q_i p_{ij} \forall j, \quad \bar{T} = \frac{\sum_{j=1}^m T_j}{m}, \quad \sigma_T = \frac{s}{\sqrt{m}},$$

Donde,

$D = \{u: Tu \geq T + kt\}$ (conjunto ordenado de máquinas candidatas)

Paso 2: Sea, j el índice de la máquina (i) en el conjunto D ;
establecer: $D \leftarrow D - \{j\}$.

Paso 3: $BN = j$. Si $j \neq m$, calcule:

$$\text{tail}(i) = \max_{\text{BN}+1 \leq r \leq m} \{p_{ir}\} \quad \forall i.$$

Paso 4: S1, conjunto de lotes con cuello de botella dominante y S2 conjunto complementario

$$S_1 = \left\{ i: p_{i,\text{BN}} - \max_{1 \leq r < \text{BN}} \{p_{i,r}\} \geq 0 \right\},$$

$$S_2 = \left\{ i: p_{i,\text{BN}} - \max_{1 \leq r < \text{BN}} \{p_{i,r}\} < 0 \right\}.$$

Paso 5: Nivel secundario máquinas de cuello de botella, máquina (BN-1) hasta la máquina 1.

Paso 6: Conjunto ordenado S, los lotes en orden decreciente de la máquina secundaria de cuello de botella a BN.

Paso 7: Matriz bidimensional y el tipo de conjunto: ("1" para S1 y "2" para S2).

Paso 8: Construye la secuencia de la siguiente manera:

- Programe el lote tipo "1" en la matriz
- Retrocede cualquier tipo de "2" que lo preceda en la matriz Lote de "1".

Matemáticamente, el Las definiciones son:

$$S_1 = \left\{ i: p_{i,BN} - \max_{1 \leq r < BN} \{p_{i,r}\} \geq 0 \right\},$$

$$S_2 = \left\{ i: p_{i,BN} - \max_{1 \leq r < BN} \{p_{i,r}\} < 0 \right\}.$$

El estudio experimental, muestra que la heurística del BMI supera a la FIH, considerada la heurística más conocida para la programación de ow-shops hasta ahora, cuando aplicado al problema, siendo el algoritmo BMI el más eficiente.

El algoritmo genético (GA) y el algoritmo genético híbrido (HEA) también conocido como algoritmo evolutivo híbrido caen bajo heurística evolutiva (Kalir, A. A., & Sarin, S. C. , 2001).

Marimuthu et al., (2008): “métodos aplicados con algoritmos genético y genético híbrido”

Investigaciones también abordan el problema de establecer el número de Sub-lotes al ser divididos, así **Martin (2009):** abordó el estudio aplicando “métodos con referencia a algoritmo genético híbrido sumado a programación matemática”

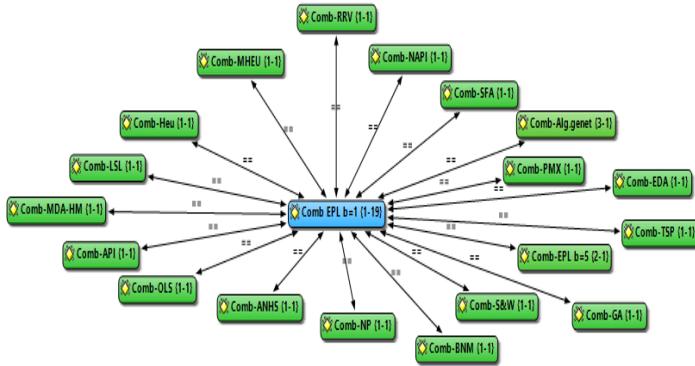


Figura 1: Red semántica Combinaciones

La red semántica estructurada, con base a la revisión bibliográfica muestra una asociación con las categorías emergentes evidenciadas como: Algoritmos genéticos que tienen presencia en 3 estudios de los 19 analizados, el resto de combinaciones son individuales, propias de las particularidades de cada estudio; es decir, las combinaciones observadas son: EPL-tamaño de lote óptimo, TPS, NP con heurística NP, MHEU, GA, PMX, EDA, S&W, BNM, ANHS, OLS, API, MDA-HM, LSL, Heu, RRV, NAPI y SFA.

Con respecto a las comparaciones observadas en los estudios, evidenciando que 6 investigaciones compararon sus análisis con la heurística NEH; de 7 que se pudo identificar este dimensionamiento.

Así pues, **Kalir & Sarin (2001)**: “aplicaron una heurística de inserción rápida”

Así mismo, se identificó 3 comparaciones de algoritmos genéticos y 2 DPSO existentes, tal es el caso de autores como **Tseng & Liao (2008)**: “utilizó DPSO existente y Algoritmo genético Híbrido”

Marimuthu et al., (2008): “Comparación asociada al reconocido simulado, Baker, Algoritmo genético y Algoritmo genético híbrido”

Aplicando un algoritmo EDA y asociado a la heurística NEH, mostrando resultados efectivos luego del análisis computacional y estadísticos completos.

Tabla 9: Identificación algoritmo EDA

| Algoritmo | Características | Ventajas | Desventajas |
|--|---|----------|-------------|
| EDA (Algoritmo Genético de Poblaciones que evolucionan) | Metodología meteorológica propuesta, que se basa en la población que evoluciona dentro de los procesos de búsqueda y tener una fundación teórica de probabilidad. | x | |

El algoritmo propuesto utiliza la siguiente metodología:

Paso 1: Establecer los parámetros de calificador PS, Q, M, Pls, e, y, g; dejar Gen=1.

Paso 2: Inicializar la población y evaluar cada individuo.

Paso 3: Realizar una búsqueda local en la mejor ilustración individual en el inicio población.

Paso 4: Seleccione que individuo tiene la mejor probabilidad estadística

Paso 5: Muestra y genera M descendencia del modelo probabilístico

Paso 6: Evalúe el desenlace y actualice la población.

Paso 7: Verificar la universidad de la población si gen mayor a 100. Si el nivel de diversidad es mayor que el de g, el procedimiento de reinicio y el ajuste gen=0.

Paso 8: Si se supera el criterio, devuelve el mejor la solución se encuentra en la parte superior.

Calcula el porcentaje de aumento relativo (RPI):

$$\text{RPI}(c_i) = (c_i - c^*) / c^* \times 100$$

Donde,

C_i : valor objetivo generado en la i -ésima réplica

c^* : valor objetivo encontrado por cualquiera de los algoritmos comparados

$$\text{div} = \left(\frac{\alpha - n}{n \times \min(n, PS - 1)} + \frac{\beta - (n - 1)}{(n - 1) \times \min(n - 1, PS - 1)} \right) / 2$$

El objetivo estudiado es la minimización de *makespan*. Este problema tiene aplicaciones importantes en textiles, plásticos, químicos, semiconductores y muchas otras industrias donde los trabajos son en realidad lotes de muchos productos idénticos para fabricar. Se ha propuesto una estimación de algoritmo de distribución (EDA) novedosa para el problema tanto en casos de deslizamiento como de no ralenti (Pan, Q. K., Suganthan, P. N., Liang, J. J., & Tasgetiren, M. F., 2011)

Pan et al., (2011): “Algoritmo genético Híbrido, DPSO, HDPSO)

De igual manera, se evidenció algoritmos de discreta optimización de partículas (DPSO) en 2 estudios, el resto de comparaciones fueron unitarias y propias de cada estudio, siendo las más relevantes las siguientes: LSL, Heu, API, TA y ACO.

Tabla 10: Identificación del algoritmo ABC

| Algoritmo | Características/Componentes | Ventajas | Desventajas |
|---|---|----------|-------------|
| <p>PSO (Optimización por Enjambres de Partículas), es una técnica de optimización/búsqueda, con la finalidad de facilitar la visualización, es decir, cada partícula es atraída hacia la mejor localización o hacia la mejor localización que ha sido encontrada por el grupo.</p> | <p>Inicia un arreglo de partículas con velocidades y posiciones aleatorias</p> <hr/> <p>Evalúa y compara la función objetivo para cada partícula</p> <hr/> <p>Identifica cada partícula del enjambre con el valor más óptimo de la función objetivo</p> <hr/> <p>Ajusta velocidad, posición de acuerdo al criterio de terminación</p> | | x |

| | |
|---|---|
| <p>ABC (Artificial Bee Colony), proceso de optimización que imita a las abejas por fuentes importantes de alimento, resultando un proceso análogo para encontrar soluciones óptimas.</p> | <p>Inicialización de la población.- fuentes de alimento para las abejas obreras que representa las variables de decisión, siendo sus límites aleatoriamente definidos.</p> |
| | <p>Enviar abejas obreras.- operación que cada abeja obrera genera como nueva fuente de alimento de los parámetros seleccionados aleatoriamente y satisfaciendo las condiciones, cuyo objetivo es minimizar la función objetivo.</p> |
| | <p>Selección de fuentes de alimento.- es la probabilidad de que una fuente de alimento sea seleccionada.</p> |
| | <p>Determinar abejas exploradoras.- si la solución candidata no puede ser mejorada, la fuente de alimento se abandona y la abeja se</p> |

| | | |
|--|--|---|
| | convierte en exploradora. | |
| HGA (Genético), promete convergencia pero no optimidad e identifica las observaciones donde se encuentran esos valores óptimos; mediante, representación genética del dominio de la solución y evaluación del dominio de la solución. | <p>Inicialización.- es el tamaño de la solución la cual se genera aleatoriamente, permitiendo toda la gama de posibles soluciones.</p> <p>Selección.- evalúan la actitud de cada solución y seleccionan las mejores soluciones.</p> <p>Operadores genéticos.- genera una población de segunda generación, a través de entrecruzamiento, recombinación y mutación.</p> <p>Terminación.- Encuentra solución que satisface criterios mínimos, alcanza fijado número de observaciones y presupuesto asignado dependiendo de sucesivas iteraciones.</p> | x |

El algoritmo aplicado al mismo tiempo computacional, el DABC es el más óptimo que los PDO y HDA, porque minimiza la precocidad y tardanza total de *Lot Streaming*, con Sub-lotes igual al tamaño.

Inicialización de la Población

La población inicial de soluciones se llena con el número SN de vectores n-dimensionales de valor real generados aleatoriamente. Deje $X_i = \{x_{i1}, x_{i2}, \dots, x_{in}\}$ representan la i-ésima fuente de alimento en la población.

$$x_{ij} = LB_j + (UB_j - LB_j) * r \quad \text{for } j = 1; 2; \dots; n \text{ and } i = 1; 2; \dots; SN;$$

Donde,

r: número aleatorio uniforme, rango [0, 1]

LB_j y UB_j: límites inferior y superior de la dimensión j,

Estas fuentes de alimentos se asignan aleatoriamente al número de PS de las abejas empleadas y se evalúan.

Inicialización de la fase de la abeja

En esta etapa, cada abeja empleada x_i genera una nueva fuente de alimento en el vecindario de su posición actual como sigue:

$$X_{new(j)} = x_{ij} + (x_{ij} - x_{kj}) * r;$$

Donde,

K: $\{1, 2, \dots, PS\}$ ^ k no puede i y j pertenece $\{1, 2, \dots, n\}$ aleatoriamente.

R: número real uniforme $[0, 1]$.

Una vez que se obtiene X_{new} , se evaluará y se comparará con x_i .

Fase de la abeja observadora

Evalúa la información del néctar empleadas y selecciona una fuente de alimento X_i dependiendo de su valor de probabilidad (P_i) calculado por:

$$P_i = \frac{f_i}{\sum_{i=1}^{SN} f_i},$$

Donde,

F_i : cantidad de néctar (valor de aptitud) de la i -ésima fuente de alimento x_i ; cuanto mayor es el índice, mayor es la probabilidad (alimento) es seleccionada.

X_i y convertirse en un nuevo miembro de la población.

$$x_{ij} = LB_j + (UB_j - LB_j) \times r \quad \text{for } j = 1, 2, \dots, n.$$

Donde,

r : número aleatorio uniforme en el rango $[0, 1]$.

En el algoritmo ABC básico, en cada ciclo, como máximo, un explorador sale a buscar una nueva fuente de alimento y es el más eficiente (Pan et al., 2011).

A continuación en la Figura 2 se muestra la red semántica estructurada con base a los resultados de los artículos científicos analizados, considerando la dimensión: "comparación".

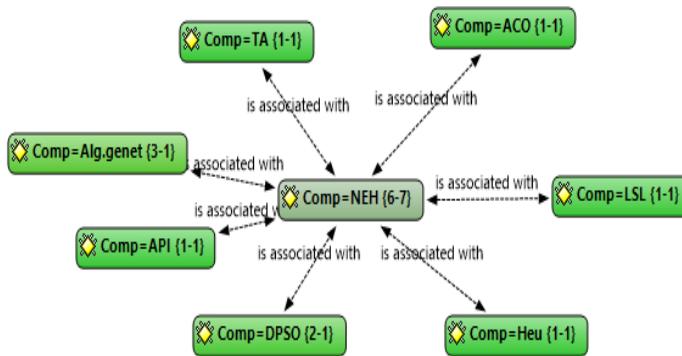


Figura 2: Red semántica Comparaciones

La red semántica resultante del análisis de la categoría “comparaciones” observa 7 estudios que identificaban esta condición, relacionado básicamente con autores que postulan mayoritariamente la heurística NEH, luego los algoritmos genéticos y posteriormente los DPSO, siendo los más relevantes y utilizados por los investigadores para considerar la programación de flujo de lotes.

Categoría “Marchas”

Las comparaciones con marchas en vacío o no, también influyen en las características y resultados en las investigaciones consultadas; 9 estudios realizaron las pruebas de sus combinaciones algorítmicas para mejorar talleres de flujo, de los cuales 6 fueron “no Idling” y 3 “Idling”, siendo características de la secuenciación al aplicar *lot streaming*.

Algunos de los estudios que utilizaron las comparaciones antes descritas son:

Ramasesh et al., (2000): “No-Idling” estudio que tomó para el análisis un conjunto de suposiciones como la tasa de la demanda constante, el lote de producción se produce en una configuración única en cada máquina, teniendo todos los Sub-lotes el mismo tamaño.

Pan et al., (2011): En su estudio aplicado con un algoritmo de colonia artificial de abejas, para resolver el problema de flujo de lotes, utilizó “Idling” y “No-Idling” en las comparaciones. Aplicando el algoritmo ABC demostrando la efectividad y eficiencia de un algoritmo de búsqueda asociado simple DABC.

Pan, Wang, Gao & Li (2011): En la aplicación de un algoritmo de salto de rana articulado con el problema de flujos de lotes, utilizó “Idling” y “No-Idling”, el algoritmo fue diseñado para mejorar la explotación y capacidad demostrando a través de las simulaciones computacionales la efectividad de la SFLA con respecto a otros algoritmos ya existentes.

La formulación de *makespan* es avanzar y retroceder, una permutación = { 1, 2, ..., n} representar un cronograma de trabajos a procesar, $ST(k, j, e)$ el tiempo de inicio más temprano de la máquina k , $CT(k, j, e)$ sea el tiempo de finalización más temprano del Sub-lotes del trabajo j en la

máquina k , y C_{max} el cálculo del pase hacia adelante, el cual, se describe de la siguiente manera:

$$\begin{cases} ST(1, \pi_1, 1) = 0 \\ CT(1, \pi_1, l(\pi_1)) = ST(1, \pi_1, 1) + s(1, \pi_1) + l(\pi_1) \times p(1, \pi_1) \end{cases}$$

Calcula el porcentaje de aumento relativo (RPI):

$$RPI(c_i) = (c_i - c^*) / c^* \times 100$$

Donde,

C_i : valor objetivo generado en la i -ésima réplica

c^* : valor objetivo encontrado por cualquiera de los algoritmos comparados

Tabla 11: Identificación Algoritmo SFLA

| Algoritmo | Características | Ventajas | Desventajas |
|--|---|----------|-------------|
| SFLA (Algoritmo salto de Ranas) | Método de optimización metaheurística natural basado en un grupo de ranas cuando se busca la ubicación que tiene el máximo cantidad de alimento disponible, combina ventajas de ambos algoritmos genéticos basados en la optimización de enjambre de partículas algoritmos. | x | |
| HGA (Algoritmo genético) | Un algoritmo genético puede presentar diversas variaciones, dependiendo de cómo se aplican los operadores genéticos (cruzamiento, mutación), de cómo se realiza la selección y de cómo se decide el reemplazo de los individuos para formar la nueva población | | x |
| ACO (Algoritmo de colonia de hormigas) | Proviene de la observación de la explotación de los recursos alimentarios entre hormigas, en el que las habilidades cognitivas de las hormigas son individualmente | | x |

limitadas y en conjunto son capaces
de buscar el menor camino
existente entre la fuente de comida
y su nido o colonia.

Para probar el rendimiento de la SFLA propuesta para minimizar *makespan* para el flujo de lotes, se llevan a cabo las replicaciones independientes de la programación de la planta, y para cada replicación; cuanto más pequeño RPI, el mejor resultado que produce el algoritmo.

El objetivo es minimizar *makespan* para el flujo problema de programación de compras en un entorno de transmisión de lotes con Sub-lotes de igual tamaño. Se una nueva SFLA y considerado el problema tanto en ralentí como sin ralentí casos de producción.

Con base a permutaciones de trabajo discretas, SFLA para encontrar soluciones prometedoras en el todo el espacio de búsqueda mediante el uso de una población efectiva (Pan, Q. K., Tasgetiren, M. F., Suganthan, P. N., & Chua, T. J., 2011).

Sang (2010): abordó los problemas de tardanza y penalización para los problemas de programación de flujo de lotes, aplicando

“No-Idling”, proponiendo un algoritmo de evolución diferencial discreta.

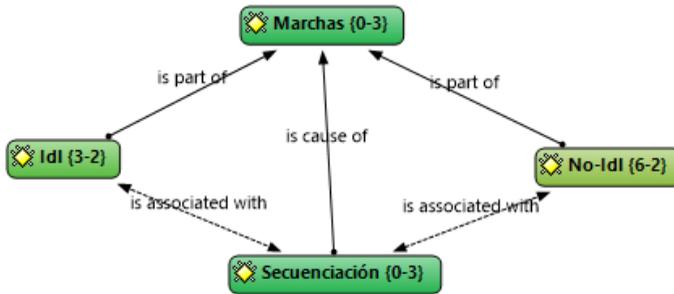


Figura 3: Red semántica Marchas

Al revisar la categoría “Marchas” se evidencia que los estudios mayoritariamente fundamentan las comparaciones con “No-Idling”, tal es el caso, de la investigación realizada por Davendra, Senkerik, Zelinka, Pluhacek & Bialic-Davendra (2014) quién abordó esta temática a través de un algoritmo discreto de auto organización migratoria (DSOMA) el cual se aplica en programación Flowshop de flow-lot, el cual, con base a los resultados evidenció una mejora significativa en el rendimiento.

Categoría Máquinas – Trabajos

Con relación máquinas y trabajos empleados en los distintos estudios revisados, 15 se analizaron con “n” número de máquinas; 11 con “n” número de trabajos y 1 estudio con “igual” número de trabajos.

Estudios como el de **Ramasesh et al., (2000)**: consideraron “n” máquinas y trabajos en las aplicaciones.

Kalir & Sarin (2000): considera “n” máquinas y trabajos, en el que enfatiza la probabilidad de que los beneficios disminuyan con base al número de lotes, es decir, si aumenta. En otro caso, tomando como referencia el número de máquinas sea más grande con respecto a los lotes más pequeños.

Por otro lado, en el estudio de **Kumar et al., (2000)**: hizo su investigación con “n” trabajos, destacando que los métodos meta heurísticos pueden superar a los de búsqueda aleatoria de lo óptimo en la secuencia simultánea de lotes y problemas de secuenciación.

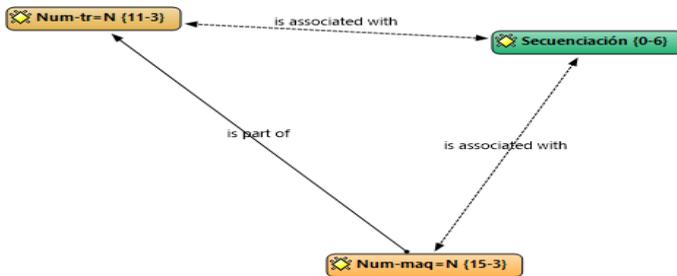


Figura 4: Red semántica Máquinas-Trabajos

La red semántica del contraste del uso de máquinas y trabajos, observa una prevalencia en las investigaciones con “n” número de máquinas al analizar la secuenciación de

talleres de flujo, es decir, se relaciona básicamente con investigadores que muestran una tendencia de la alternativa, evidenciando que tanto las dos dimensiones están asociadas a la secuenciación, mientras que, son parte de cada una tanto las máquinas como el número de trabajos al analizar la optimización de la secuencia simultánea de lotes.

Categoría “Sub-lotes”

Los resultados observados en los artículos científicos analizados, muestran que los Sub-lotes iguales son los que mayor aceptación tienen al momento de analizarlos en la secuenciación de talleres de flujo, 10 de entre los 19 estudios muestran esta condición; no obstante, 3 fueron consistentes y setup, respectivamente.

Ramasesh et al., (2000): “muestra un tipo de Sub-lotes igual”

Realiza una planificación compatible con las restricciones tecnológicas, es decir, que trate de ser factible; la programación y secuenciación son modelados como job shop, mediante la secuenciación de datos de información presentes:

- Máquinas de cada tipo, no existen varias máquinas para realizar las operaciones
- La maquinaria no puede procesar más de una operación a la vez
- Cada trabajo incluye una operación de cada máquina

- Los tiempos son independientes de la secuencia seguida
- No pueden procesar operaciones de un mismo trabajo simultáneamente.

Los algoritmos genéticos empleados, en un carácter estocástico y no exacto, al evaluar no solamente la eficacia y la eficiencia sino también la estabilidad de los problemas de scheduling facilita la comparación de distintos métodos para la resolución, mostrando la efectividad como método de búsqueda de soluciones en problemas de naturaleza combinatoria, incluyendo valores óptimos para minimizar el camino máximo, que permita avalar la calidad de las soluciones obtenidas.

Kumar, Bagchi & Sriskandarajah (2000): “trabajaron con un tipo de Sub-lotes consistente” Los resultados después de los experimentos computacionales muestran que los NGA superan a los GA en un 9,86%.

Los parámetros analizados, se determina una eficiente atención debido al aumento de la cantidad de la información electrónica y de la necesidad de acceder en el menor tiempo posible y con la mayor eficacia, en la aplicabilidad de los procesos, dando resultados mejores en tiempos óptimos.

Dentro de los criterios para medir el nivel de desempeño relacionados con el algoritmo genético, se basa en el criterio de eficiencia hacia las medidas relacionadas con los tiempos a llevar a cabo los procesos de construcción y clasificador; y, el criterio de efectividad sobre la capacidad de medir el comportamiento de los

procesos dentro de la calidad de los resultados obtenidos, para evaluar la aplicación de los algoritmos genéticos.

Yoon & Ventura (2002): “aplicaron algoritmos con Sub-lotes iguales y consistentes”

Tabla 12: Identificación algoritmo NGA

| Algoritmo | Características |
|--|--|
| NGA (Nuevo algoritmo genético) | Resuelve optimizaciones combinatorias de problemas de diversas áreas de aplicación de productos y diseños, dentro de la planificación integrada de procesos y programación de dimensiones de lotes |
| LP (Formulación de programación Lineal) | Determina el valor de la actitud del individuo $j = \{1, \dots, n\}$, siendo S_j el número de Sub-lotes. |

Dentro de los determinantes del algoritmo se tiene:

Nuevo algoritmo genético (NGA)

Paso 1: Inicialización), genera una población inicial con w individuos aleatorios.

Paso 2: Cálculo de la forma física del individuo (función objetivo por LP) y cálculo de valores óptimos.

Paso 3: Cambio de cada individuo con otro aleatoriamente.

Paso 4: Embarazo, selección de ruleta para elegir un par de reproducción.

Paso 5: Reproducción, aplicar PMX con dominio; y tasa de cruce constante

Paso 6: Prueba de terminación, si NGA alcanza el número máximo de generaciones, deténgalo.

La función objetivo, por LP son sus valores de aptitud.

En NGA, dos tipos: normalización y procedimiento de clasificación. El valor normalizado de la aptitud del individuo (f_{norm}) puede calcularse usando la siguiente ecuación:

$$f_{norm}(\sigma_\ell) = \frac{z_{max} - z(\sigma_\ell) + z_{min}}{z_{avg}}, \quad \ell = 1, \dots, w,$$

Donde,

z_{max} , z_{min} y z_{avg} : máximo, mínimo y valores objetivos promedio en la población actual

$z(\sigma)$: valor objetivo del individuo.

Se muestran que el promedio la mejora de NGA sobre GA, para diferenciar problemas de tamaño existen numerosas

posibilidades para mejorar NGA, ya que, inicialmente las poblaciones pueden ser creadas heurísticamente en lugar de aleatoriamente, dando a NGA un mejor lugar para comenzar (Yoon, S. H., & Ventura, J. A. , 2002).

Por lo anterior expuesto, al finalizar la fase principal del NGA sobre GA, de la evolución se puede conocer con una probabilidad significativa las mejores decisiones posibles para una correcta evolución y nos ayuda a pensar que se cree una estructura que implica el algoritmo aleatoriamente, resolviendo optimizaciones combinatorias de problemas de diversos procesos de aplicación en la planificación de productos y diseños, dentro de la programación de dimensiones de lotes. Para con ello, observar una función de sistema de continua evolución, que defina parámetros que se van utilizar y relacionando en cómo afectan en las diferentes modificaciones.

Zhang & Wang (2014): “en su estudio aplicando un algoritmo de la mosca de la fruta agrupada, considerando Sub-lotes iguales y consistentes”

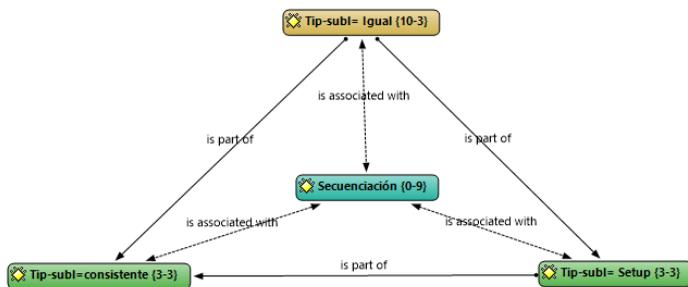


Figura 5: Red semántica Sub-lotes

La red semántica resultante de la comparación con respecto a los tipos de Sub-lotes, muestra que, 3 se analizaron con tipología consistente y setup, respectivamente. Utilizando esta característica en actividades industriales como: carpintería, mecánica, etc.; aplicando tiempos de preparación dependientes de la secuenciación.

Considerando que los Sub-lotes consistentes no varían su conformación en todo el proceso, o sea, tienen la misma la misma cantidad de unidades. No obstante, en la mayoría de estudios analizados predominan los Sub-lotes iguales

Este algoritmo, resulta una forma en que la mosca de fruta organiza sus diminutos pelos utilizados como sensores para “captar” su alrededor, lo cual, puede ser relacionado a los problemas con redes inalámbricas en los sistemas de información, y no necesariamente requiere conocer a detalle

la forma en que se organiza la red. Dentro de los procesos productivos de Lot streaming esta metodología podrá ayudar a la mejora de los procesos de información optimizando el tiempo de transmisión de sistemas informáticos de las industrias.

Categoría “Secuenciación”

En esta categoría se identificaron 3 estudios aplicados con ninguna espera y 1 con tiempo de traslado.

Ramasesh et al., (2000): “Consideró secuencias particulares como el tiempo de traslado en su análisis)

Kumar et al., (2000): “Utilizó No-wait en su análisis”

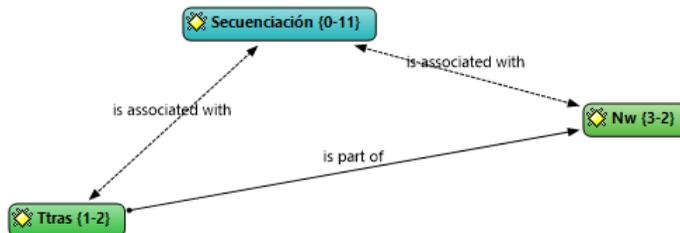


Figura 6: Red semántica Secuenciación

La caracterización “Secuenciación” mostró una red semántica dependiente y asociada con el tiempo de traslado

piezas que debe ser abandonada inmediatamente, para una vez terminada la operación, la máquina procesa en la etapa inicial para pasar a la siguiente; este algoritmo básico separa los procedimientos desarrollados si varía o no el número de máquinas dentro de las diferentes etapas, identificando la existencia de tiempos de preparación de producción dependientes del tipo de pieza a fabricar y de la programación de operaciones.

4.1 Perspectivas: Líneas Futuras de Investigación

Dentro de las perspectivas para futuros estudios relacionados con el *Lot Streaming*, es introducir la secuenciación del entorno real de la empresa en periodos determinados según el caso, los modelos y algoritmos analizados y propuestos deben seguir evolucionando dentro del campo de la programación de la producción, con ello, las industrias puedan escoger la metodología apropiada y comenzar a desarrollar él o los modelos para conseguir la optimización, integración y maniobrabilidad total y eficiente de sus procesos.

Es crucial tomar decisiones óptimas, pero sería más eficiente trabajar con heurísticas o metaheurística, ya que, permiten minimizar el espacio de soluciones al problema,

como señala el algoritmo genético, siendo el más empleado por las investigaciones analizadas para el desarrollo óptimo de los procesos, actividades y tareas de las empresas, generando un entorno analista y orientando a la eliminación de riesgos de perturbación de a solución.

La distribución de tareas por naturaleza estocásticas, las cuales actúen y regulen los controles de eventos a incurrir por desviaciones de la solución óptima durante el proceso de las actividades productivas, el poder construir y utilizar herramientas de origen matemático permite la reprogramación de la producción, proporcionando menos afectación en el flujo productivo; por ello, que un buen sistema de secuenciación permite un control eficiente que sirva de retroalimentación para ajustes en el modelo.

Por lo anterior expuesto, para futuras líneas investigativas, un conjunto de herramientas apropiadas que dan respuestas a los imprevistos, que utilice metodologías de simulación discreta, que permita aplicar de manera efectiva técnicas que aseguren la viabilidad del modelo.

CAPÍTULO 5:

CONCLUSIONES y DISCUSIÓN

5.1. Conclusiones

Luego de la revisión bibliográfica de los artículos relacionados a la secuenciación de talleres de flujo, se identificó las variables que influyen para las valoraciones que diferentes estudios asocian para demostrar la eficacia de la división de lotes, los cuales son: combinación, comparación, Idling, No-Idling, número de máquinas y trabajos, tipo de Sub-lotes y preparación y dependencia de la secuencia.

El Algoritmo genético de poblaciones que evolucionan (EDA) es una metodología propuesta que se fundamenta en la población que evoluciona dentro de los procesos de búsqueda, la cual a través de los estudios muestra una ventaja frente a los algoritmos tradicionales.

PSO (Optimización por enjambres de partículas) es un algoritmo que inicia con el arreglo de partículas con el propósito de minimizar la función objetivo, a pesar de ello muestra desventajas en su aplicación.

El algoritmo genético (HGA) puede presentar diversas variaciones, dependiendo de cómo se aplican los operadores genéticos (cruzamiento, mutación), de cómo se realiza la selección y de cómo se decide el reemplazo de los

individuos para formar la nueva población, lo cual genera desventajas en su aplicación

Así mismo, el algoritmo genético (HGA) propone una convergencia, pero, no optimidad e identifica las observaciones, siendo el tamaño de la solución que se genera aleatoriamente, presentando desventajas en su aplicación.

El algoritmo ACO (Algoritmo de colonia de hormigas) proviene de la observación de la explotación de los recursos alimentarios entre hormigas, en el que, las habilidades cognitivas de las hormigas son individualmente limitadas y en conjunto son capaces de buscar el menor camino existente entre la fuente de comida y su nido o colonia, no obstante, genera algunas desventajas en su aplicación.

Por otro lado, el algoritmo ABC (Artificial Bee Colony) es un proceso de optimización que imita a las abejas por fuentes importantes de alimento, siendo las variables de decisión cuyos límites son aleatoriamente definidos, cuya aplicación mostro ventajas en el proceso de optimización de talleres de flujo.

Los estudios realizados aplicando el algoritmo SFLA, cuyo método de optimización metaheurística natural basado en un grupo de ranas, cuando se busca la ubicación que tiene la máxima cantidad de alimento disponible, combina ventajas de ambos algoritmos genéticos basados en la optimización de enjambre de partículas algoritmos.

5.2 Discusión

Las diversas revisiones literarias, se pudo evidenciar diversas técnicas matemáticas o también denominadas algoritmos para aclarar que no solo existe una heurística basada en la evolución de algoritmos genéticos, la cual trata de optimizar la cantidad de los procesos, en relación a las restricciones y estructuras dimensionales, sino, aportando con soluciones generadas y las principales ventajas que se generan por su ejecución.

La acertada representación dentro de los algoritmos genéticos, basado en la estructura del cromosoma para las variantes de los distintos procesos analizados, resulta de mucha importancia el poder reconocer frente a que variante del problema de scheduling.

Se encuentra la programación de las máquinas dentro de un taller maquinado, asignando la generación adecuada de los procesos o tareas, proporcionando soluciones factibles y valores óptimos en el cumplimiento de los objetivos planteados por las industrias. Por otro lado, la explosión exploratoria debe ser resuelta de una manera efectiva dependiendo del método de solución a emplear, ya que la representación cromosómica es de vital importancia para la aplicación del algoritmo genético.

Dentro de las soluciones analizadas, al comparar con las teorías de los algoritmos, y mediante una codificación binaria la convergencia del algoritmo, presenta un comportamiento óptimo para encontrar las mejores soluciones, mejorando el desempeño y realizando nuevas modificaciones para los diversos operadores genéticos, mostrando una mayor flexibilidad de modificación del algoritmo genético con mayor número de máquinas con diversas restricciones, mejorando notablemente el comportamiento de las tareas.

En relación al algoritmo de colonia de hormigas que propone reducir los costes anuales, las diferencias la disminución en la transferencia de lotes, provoca un aumento en la productividad de los procesos, afectando principalmente en el capital de inversión.

Permite observar que el algoritmo influye de manera significativa en la situación económica del diseño de las tareas, proporcionando un sistema de costes adecuados para la organización, permitiendo la optimización de sus procesos.

El algoritmo ABC o de colonia de abejas artificiales, el esquema de selección estocástica, determina un proceso de evolución auto adaptativo, generando soluciones que afectan e intervienen directamente con el operador de mutación basados en los sistemas de actividades y una efectiva comunicación entre miembros generando una optimización multimodal, dentro de los procesos de intensificación; permitiendo un rendimiento local y global,

este algoritmo ABC permite la flexibilidad de las operaciones tanto de problemas multimodales y multivariables.

La metodología del algoritmo SFLA, metaheurística natural basado en un grupo de ranas, tiene la iniciativa de emplear un pensamiento crítico y sistemático, que permita el análisis de clásica regla de decisión (vecino más cercano), para conocer y determinar los comportamientos de causa y efecto de los procesos operacionales y productivos y destacando su optimización.

BIBLIOGRAFÍA

- Arango. (2015). *Metodología de Programación de Producción para reducción de costos en un Flow Shop híbrido flexible mediante el uso de algoritmos genéticos*. Manizales: Innovación y Desarrollo Tecnológico.
- Bagchi, T. P., Gupta, J. N., & Sriskandarajah, C. (2006). A review of TSP based approaches for flowshop scheduling. *European Journal of Operational Research*, 169(3), 816-854.
- Becerra Rodríguez, F., Cárdenas Aguirre, D. M., Castrillón Gómez, Ó. D., García, A., Giraldo García, J. A., Ibarra Mirón, S., ... & Zapata Gómez, A. (2008). *Gestión de la producción: Una aproximación Conceptual*.

- Castillo & Anselmo . (2014). *Adaptación de la técnica heurística optimización por enjambres de partículas para resolver problemas de empaquetamiento con restricciones de precedencia*. México: Komputer Spaiens.
- Chakaravarthy, G. V., Marimuthu, S., Ponnambalam, S. G., & Kanagaraj, G. (2014). Improved sheep flock heredity algorithm and artificial bee colony algorithm for scheduling m-machine flow shops lot streaming with equal size sub-lot problems. *International Journal of Production Research*, 52(5), 1509-1527.
- Chang*, J. H., & Chiu, H. N. . (2005). A comprehensive review of lot streaming. *International Journal of Production Research*, 43(8), 1515-1536.
- Chase, R. B., Aquilano, N. J., CIOCIANO GONZALEZ, M. I. L. D. R. E. D., GARCIA ROCHA, A. N. G. E. L. A., & JACOBS, F. R. (2000). *Administración de producción y operaciones: manufactura y servicios*.
- Chiu, H. N., & Chang, J. H. (2005). Cost models for lot streaming in a multistage flow shop. *Omega*, 33(5), 435-450.
- Chiu, H. N., Chang, J. H., & Lee, C. H. (2004). Lot streaming models with a limited number of capacitated transporters in multistage batch production systems. *Computers & Operations Research*, 31(12), 2003-2020.
- Davendra, D., Senkerik, R., Zelinka, I., Pluhacek, M., & Bialic-Davendra, M. (2014). Utilising the chaos-induced discrete self organising migrating algorithm to solve the lot-streaming flowshop scheduling

- problem with setup time. *Soft Computing*, 18(4), 669-681.
- Fiol-Zulueta, A., Arzola-Ruíz, J., & da Silva-Carvalho, A. (2010). Evaluación del comportamiento de un algoritmo para la secuenciación en un taller de flujo con tiempos dependientes de la secuencia. *Ingeniería Mecánica*, 13(1), 67-78.
- Han, Hongyan, Yun Bao, Qun sun. (2018). Mejora de la optimización de las aves migratorias discretas para el flujo de lotes Flujo de compras Programación de problemas con el bloqueo. *International Conference on Intelligent Computing*, Pag. 780-791.
- Han, Y. Y., Gong, D. W., Sun, X. Y., & Pan, Q. K. (2014). An improved NSGA-II algorithm for multi-objective lot-streaming flow shop scheduling problem. *International Journal of Production Research*, 52(8), 2211-2231.
- Han, Y., Gong, D., Jin, Y., & Pan, Q. K. (2016). Evolutionary multi-objective blocking lot-streaming flow shop scheduling with interval processing time. *Applied Soft Computing*, 42, 229-245.
- Hong, Ke-Pan, Duan. (2018). Algoritmo de optimización de malas hierbas invasoras para problemas de lot streaming en flowshop. *Journal of Intelligent Manufacturing*, Vol. 9, pag.1337-1349.
- Kalir, A. A., & Sarin, S. C. (2000). Evaluation of the potential benefits of lot streaming in flow-shop systems. *International Journal of Production Economics*, 66(2), 131-142.

- Kalir, A. A., & Sarin, S. C. . (2001). A near-optimal heuristic for the sequencing problem in multiple-batch flowshops with small equal sublots. *Omega*, 29(6), 577-584.
- Kalir, A. A., & Sarin, S. C. (2001). A near-optimal heuristic for the sequencing problem in multiple-batch flowshops with small equal sublots. *Omega*, 29(6), 577-584.
- Kher, H. V., Malhotra, M. K., & Steele, D. C. (2000). The effect of push and pull lot splitting approaches on lot traceability and material handling costs in stochastic flow shop environments. *International Journal of Production Research*, 38(1), 141-160.
- Kim, K., & Jeong, I. J. . (2009). Flow shop scheduling with no-wait flexible lot streaming using an adaptive genetic algorithm. *The International Journal of Advanced Manufacturing Technology*, 44(11-12), 1181-1190.
- Kumar, S., Bagchi, T. P., & Srisandarajah, C. (2000). Lot streaming and scheduling heuristics for m-machine no-wait flowshops. *Computers & Industrial Engineering*, 38(1), 149-172.
- Lazmi, Naru, Madhusudanan. (2017). Lot streaming en $[N - 1] (1) + N (m)$ tienda de flujo híbrido. *Journal of Manufacturing Systems*, Pags. 12-21.
- López, M. L. N., Hernández, E. A. G., y Rodríguez, G. A. G. (2011). Relación entre el tamaño de lote de transferencia y el desempeño de un sistema de manufactura tipo "Flow-Shop". *Revista UIS Ingenierías*, 10(2), 151-167.

- Low, HSU, Huang, Mortezaei. (2013). *Programación de Operaciones: Producción*. Florianópolis: Iberoamerican Journal of Industrial Engineering.
- Marimuthu, S., Ponnambalam, S. G., & Jawahar, N. (2008). Evolutionary algorithms for scheduling m-machine flow shop with lot streaming. *Robotics and Computer-Integrated Manufacturing*, 24(1), 125-139.
- Marimuthu, S., Ponnambalam, S. G., & Jawahar, N. (2008). Evolutionary algorithms for scheduling m-machine flow shop with lot streaming. *Robotics and Computer-Integrated Manufacturing*, 24(1), 125-139.
- Martin, C. H. (2009). A hybrid genetic algorithm/mathematical programming approach to the multi-family flowshop scheduling problem with lot streaming. *Omega*, 37(1), 126-137.
- Minella, Ruiz, Ciavotta. (2011). *Restarted Iterated Pareto Greedy algorithm for multi-objective*. Comput Oper.
- Mortezaei, N., & Zulkifli, N. (2013). Integration of lot sizing and flow shop scheduling with lot streaming. *Journal of Applied Mathematics*, 1(10).
- Mortezaei, N., & Zulkifli, N. (2013). Integration of lot sizing and flow shop scheduling with lot streaming. *Journal of Applied Mathematics*.
- Novas. (2016). *Modelo MILP para la Programación de la Producción en ambientes Job-Shop flexibles con división de lotes*. Florianópolis: Iberoamerican Journal of Industrial Engineering.

- Novas. (2016). *Programación de la Producción en ambientes Job-Shop flexibles con División de Lotes*. Córdoba: Consejo Nacional de Investigaciones Científicas y Técnicas.
- Pan, Q. K., & Ruiz, R. (2012). An estimation of distribution algorithm for lot-streaming flow shop problems with setup times. *Omega*, 40(2), 166-180.
- Pan, Q. K., Suganthan, P. N., Liang, J. J., & Tasgetiren, M. F. . (2011). A local-best harmony search algorithm with dynamic sub-harmony memories for lot-streaming flow shop scheduling problem. *Expert Systems with Applications*, 38(4), 3252-3259.
- Pan, Q. K., Tasgetiren, M. F., Suganthan, P. N., & Chua, T. J. (2011). A discrete artificial bee colony algorithm for the lot-streaming flow shop scheduling problem. *Information sciences*, 181(12), 2455-2468.
- Pan, Q. K., Wang, L., Gao, L., & Li, J. (2011). An effective shuffled frog-leaping algorithm for lot-streaming flow shop scheduling problem. *The International Journal of Advanced Manufacturing Technology*, 52(5-8), 699-713.
- Pan, Q. K., Wang, L., Gao, L., & Li, J. (2011). An effective shuffled frog-leaping algorithm for lot-streaming flow shop scheduling problem. *The International Journal of Advanced Manufacturing Technology*, 52(5-8), 699-713.
- Ramasesh, R. V., Fu, H., Fong, D. K., & Hayya, J. C. (2000). Lot streaming in multistage production systems. *International Journal of Production Economics*, 66(3), 199-211.

- Ramasesh, R. V., Fu, H., Fong, D. K., & Hayya, J. C. (2000). Lot streaming in multistage production systems. *International Journal of Production Economics*, 66(3), 199-211.
- Rossit, D., Tohmé, F., Frutos, M., Bard, J., & Broz. (2016). A non-permutation flowshop scheduling problem with lot streaming. *International Journal of Industrial Engineering Computations*, 7(3), 507-516.
- Sang, H. Y. (2010). A discrete differential evolution algorithm for lot-streaming flow shop scheduling problems. *In Natural Computation (ICNC), 2010 Sixth International Conference on*, 10-13.
- Sang, H. Y., Pan, Q. K., Duan, P. Y., & Li, J. Q. (2018). An effective discrete invasive weed optimization algorithm for lot-streaming flowshop scheduling problems. *ournal of Intelligent Manufacturing*, , 29(6), 1337-1349.
- Sarin, S. C., & Jaiprakash, P. (2007). *Flow shop lot streaming*. Springer Science & Business Media.
- Tseng, C. T., & Liao, C. J. (2008). A discrete particle swarm optimization for lot-streaming flowshop scheduling problem. *European Journal of Operational Research*, 191(2), 360-373.
- Ventura, J. A., & Yoon, S. H. (2013). A new genetic algorithm for lot-streaming flow shop scheduling with limited capacity buffers. *. Journal of Intelligent Manufacturing*, 24(6), 1185-1196.
- Wang, Xu, Zhou, Liu. (2015). *Bi-Population estimation of distribution algorithm for the flexible job-shop*

scheduling problem. México: Computers and Industrial.

Yoon, S. H., & Ventura, J. A. . (2002). Minimizing the mean weighted absolute deviation from due dates in lot-streaming flow shop scheduling. *Computers & Operations Research*, 29(10), 1301-1315.

Zhang, P., & Wang, L. (2014). Grouped fruit-fly optimization algorithm for the no-wait lot streaming flow shop scheduling. *International Conference on Intelligent Computing*, 664-674.

Anexo 2: Imagen Programa Atlas_ti

Analisis Silvia Velicelea_31082018-p - ATLAS.ti

Proyecto Edición Documentos Citas Códigos Memos Redes Analisis Herramientas Visualizaciones Ventanas Ayuda

DPs P 1: A discrete artificial Citas Códigos Comb-Alg-genet (3-1) Memo

P 1: A discrete artificial bee colony algorithm for the lot-streaming flow.pdf

Q.-K. Pan et al. / Information Sciences 187 (2011) 2465–2468 2467

Table 6
Comparison of DABC with and without local search under different cases

| n × m | DABC | | DABC | | Ratio |
|---------|--------|--------|--------|----------------|-------|
| | ABN | SD | ABN | SD (DABC DABC) | |
| 10 × 3 | 0.3852 | 0.3900 | 0 | 0 | 1 |
| 10 × 5 | 0 | 0 | 0 | 0 | 0 |
| 10 × 7 | 0 | 0 | 0 | 0 | 0 |
| 10 × 10 | 0 | 0 | 0 | 0 | 0 |
| 20 × 3 | 0.1176 | 0 | 0.1399 | 0 | 0 |
| 20 × 5 | 0.9119 | 0.3994 | 0.1075 | 0.2594 | 1 |
| 20 × 7 | 0.1425 | 0.2359 | 0.022 | 0.051 | 1 |
| 20 × 10 | 0.3887 | 0.4017 | 0.067 | 0.1891 | 1 |
| 30 × 3 | 0.2084 | 0.2119 | 0.038 | 0.0735 | 1 |
| 30 × 5 | 1.1887 | 1.037 | 1.067 | 1.287 | 1 |
| 30 × 7 | 1.6282 | 0.8877 | 1.024 | 0.5827 | 1 |
| 30 × 10 | 1.3326 | 1.4917 | 1.512 | 0.888 | 1 |
| 40 × 3 | 1.2281 | 0.4278 | 0.4889 | 0.3622 | 1 |
| 40 × 5 | 1.6113 | 0.9676 | 0.911 | 0.3479 | 1 |
| 40 × 7 | 2.0889 | 0.8011 | 1.2752 | 0.4854 | 1 |
| 40 × 10 | 1.6052 | 1.1381 | 1.6475 | 0.7166 | 1 |
| 40 × 7 | 2.0782 | 1.8248 | 0.934 | 0.4768 | 1 |
| 40 × 10 | 1.2281 | 1.8805 | 0.8888 | 0.8352 | 1 |
| 40 × 18 | 1.8779 | 0.8118 | 0.8227 | 0.2877 | 1 |
| Mean | 1.2022 | 0.7121 | 0.8148 | 0.2858 | |

Finally, the computational results produced by the DABC algorithm with and without local search in Table 6 further suggest that the superiority in terms of search ability and efficiency of the DABC algorithm should be attributed to the combination of global search and local search with an appropriate balance between exploration and exploitation.

In a nutshell, it can be concluded that, at the same computational time, the proposed DABC algorithm performs much better than the existing GA and HOPSO algorithms to minimize total earliness and tardiness penalties for the lot-streaming problem with equal sized sublots under both the rising and non-rising cases.

8. Conclusions

This paper aimed at minimizing total weighted earliness and tardiness penalties for the lot-streaming flow shop scheduling problems with equal sized sublots. We examined the problem under both the rising and non-rising cases and proposed a novel discrete artificial bee colony (DABC) algorithm. To the best of our knowledge, this was the first reported application of the ABC algorithm for solving the problem under consideration. In the proposed algorithm, the food sources were represented as discrete job permutations. The ABC-based searching mechanism with an effective population initialization approach and a self-adaptive neighboring food source generation strategy were developed to perform exploration for promising solutions within the entire solution space. Furthermore, a simple but effective local search algorithm was em-

Special-hw
Tip-subli-Igual