



SISTEMA INTELIGENTE DE MANTENIMIENTO PARA SISTEMAS DE GENERACIÓN EÓLICA MEDIANTE REDES DE PETRI COLOREADAS AUTOCRESCIENTES

Tesina Máster Automática e Informática Industrial

*Departamento de Ingeniería de Sistemas y Automática (DISA) Departamento de
Ingeniería de Sistemas y Computadores (DISCA)*



Autora: M^aJosé Pérez Collada

Director: Dr. Emilio García Moreno

Codirector: Dr. Antonio Correcher Salvador

Abril 2011, Valencia

1. INTRODUCCIÓN	4
1.1. Integración en sistema inteligente de seguimiento de estado, diagnóstico de fallos y mantenimiento (SISEDFYM) del proyecto EOLIA	5
1.2. Planteamiento del problema	6
1.3. Objetivos	8
2. SISTEMA DE GESTIÓN DEL MANTENIMIENTO	10
2.1. Descripción del sistema de mantenimiento	11
2.2. Descripción detallada de parámetros de entrada al sistema	14
2.2.1. Parámetros seguimiento de estado	14
2.2.2. Parámetros para diagnóstico de fallos	15
2.2.3. Parámetros mantenimiento basado en calendario	16
2.2.4. Parámetros mantenimiento basado en inspecciones	16
2.2.5. Parámetros mantenimiento proactivo	17
2.3. Consideraciones adicionales	18
3. RED DE PETRI GLOBAL PARA LA GESTIÓN DEL MANTENIMIENTO	19
3.1. Conceptos básicos Redes de Petri Coloreadas	19
3.2. Red de Petri para el Mantenimiento	21
3.3. Red de Petri adaptativa	26
4. DESCRIPCIÓN DEL SISTEMA DE MANTENIMIENTO Y LOS SUBSISTEMAS QUE LO CONFORMAN	30
4.1. Subsistema asignación de prioridades	30
4.1.1. Asignación a partir de tabla.	33
4.1.2. Asignación a partir de control inteligente (fuzzy logic)	34
4.1.3. Asignación a partir de la información de Maximo	35
4.2. Subsistema de agrupación de tareas de mantenimiento	36
4.2.1. Funcionamiento de la herramienta y parámetros de decisión	37
4.3. Subsistema de decisión multiobjetivo de tareas de mantenimiento	38
4.3.1. Método basado en restricciones	40
4.3.2. Método basado en interfaz de preferencias	41
4.3.3. Método basado en control inteligente (lógica fuzzy) y asignación de pesos	45
4.3.4. Solución adoptada y parámetros de implementación	48
5. ANÁLISIS COSTE RIESGO	50
5.1. Metodología.	50
5.2. Optimización Coste-Riesgo (OCR)	52
5.3. Análisis de Software para implementación	55
5.4. Software @Risk y Crystall_Ball	56
5.5. Implementación sistema de optimización coste riesgo en Excel.	61
5.6. Conclusiones Optimización Coste Riesgo	65

6.	SISTEMA GENERAL DE MANTENIMIENTO	66
6.1.	<i>Estructura RdPCA</i>	66
6.2.	<i>Librería RdPCA</i>	70
6.3.	<i>Librería comunicación LABVIEW-EXCEL</i>	73
6.3.1.	Estructuras básicas	73
6.4.	<i>Estructura funcionamiento RdPCA</i>	77
6.4.1.	Inicialización variables	78
6.4.2.	Funcionamiento mediante estructura de secuencias	84
6.5.	<i>Subsistemas implementados para la RdPCA</i>	92
6.5.1.	Creación ordenes nuevas	93
6.5.2.	Información de tareas activas	101
6.5.3.	Informe prioridad alta	104
6.5.4.	Consulta requerimientos	105
6.5.5.	Consulta disponibilidad material y equipo	109
6.5.6.	Eliminación órdenes finalizadas	114
6.6.	<i>Subsistema de Meteorología</i>	116
6.7.	<i>Subsistema de Agrupación de Tareas</i>	122
6.8.	<i>Subsistema de Decisión Multiobjetivo</i>	127
6.9.	<i>Presentación de resultados para Aceptación de experto</i>	129
6.10.	<i>Creación de Informe para Maximo</i>	132
7.	INTERFAZ DE USUARIO GMAO	134
7.1.	<i>Distribución estructural</i>	135
7.2.	<i>Distribución por sectores temáticos</i>	139
8.	CONCLUSIONES	147
9.	REFERENCIAS	150

1. INTRODUCCIÓN

A lo largo de los años, las energías renovables se han consolidado como una fuente de energía a tener en cuenta y han sido, por lo tanto, desarrolladas y mejoradas en muchos aspectos. Dentro de estas energías, cabe destacar la gran evolución de los sistemas de energía eólica, tanto a nivel nacional como internacional. Su presencia en el mundo de la generación ha aumentado de forma más que considerable en los últimos años. Se trata pues de sistemas altamente estudiados y desarrollados.

Es por esto que las nuevas líneas de investigación en este campo se dirigen hacia la generación eólica en entornos marinos, o eólica offshore. Si bien no se trata de una tecnología completamente nueva, si se puede determinar que su presencia es muy limitada y está reducida a determinados países, muy experimentados en la energía eólica. A nivel nacional, no se dispone en la actualidad de parques de generación offshore en funcionamiento, a pesar de existir diversos proyectos de gran envergadura relacionados con el desarrollo de estas nuevas metodologías.

La tesina que se presenta forma parte de uno de estos grandes proyectos de investigación, como veremos más adelante, y tiene como principal objetivo el desarrollo de una herramienta completa de gestión de los sistemas de mantenimiento, totalmente enfocada a sistemas de generación eólica offshore.

Los sistemas de mantenimiento utilizados para proyectos de tan gran tamaño son muy complejos y trabajan con un número de variables y factores muy elevado. Es por esto que se necesita una herramienta completa y precisa, capaz de adaptarse a los requisitos de funcionamiento y las especificaciones del proyecto.

Dadas las características del sistema con el que se ha trabajado, ha sido preciso el desarrollo de una nueva herramienta de implementación para el sistema, que se analizará en profundidad a lo largo de la presente tesina, analizando sus características estructurales y de funcionamiento.

1.1. Integración en sistema inteligente de seguimiento de estado, diagnóstico de fallos y mantenimiento (SISEDFYM) del proyecto EOLIA

Tal y como se ha mencionado en el apartado anterior, esta tesina se engloba en un proyecto nacional de investigación para el desarrollo de herramientas de seguimiento de estado, diagnóstico de fallos y gestión del mantenimiento para aerogeneradores offshore, el proyecto EOLIA.

Este proyecto está dividido en diversos sectores, que precisan trabajar conjuntamente para el correcto funcionamiento de la herramienta global. Esta solución global mostraría la distribución y relación entre los sectores, y se denominará SISEDFYM: sistema inteligente de seguimiento de estado, diagnóstico de fallos y mantenimiento.

La presente tesina se centra exclusivamente en la herramienta de gestión del mantenimiento, si bien en diversos apartados se hace referencia a la importancia de la comunicación con los sistemas de seguimiento de estado y diagnóstico de fallos, por lo que es importante conocer la situación del método desarrollado para el mantenimiento dentro del proyecto SISEDFYM. La relación entre todas las etapas es un punto muy importante que permitirá la correcta relación entre los sistemas, dando lugar a un sistema inteligente completo.

Basándonos en la Fig. 1, vemos que las salidas del sistema de seguimiento de estado así como las de diagnóstico de fallos actúan como entradas al sistema de mantenimiento, relacionando así el módulo de planificación y programación de la gestión de activos con el módulo inteligente de diagnóstico. Estas entradas, además de las propias proporcionadas por el sistema de mantenimiento, servirán para alimentar la herramienta desarrollada para la gestión del mantenimiento. En apartados posteriores se analizarán los requisitos necesarios en cuanto a información que se precisan para el sistema.

Tal y como se puede apreciar de la estructura SISEDFYM, se trata de una herramienta que trabaja con distintas entradas provenientes de sistemas diversos, lo que plantea un problema para el funcionamiento general del sistema, que debe ser capaz de encontrar una forma de mantener un funcionamiento estable a pesar de las diferencias entre las entradas.

SISEDFYM

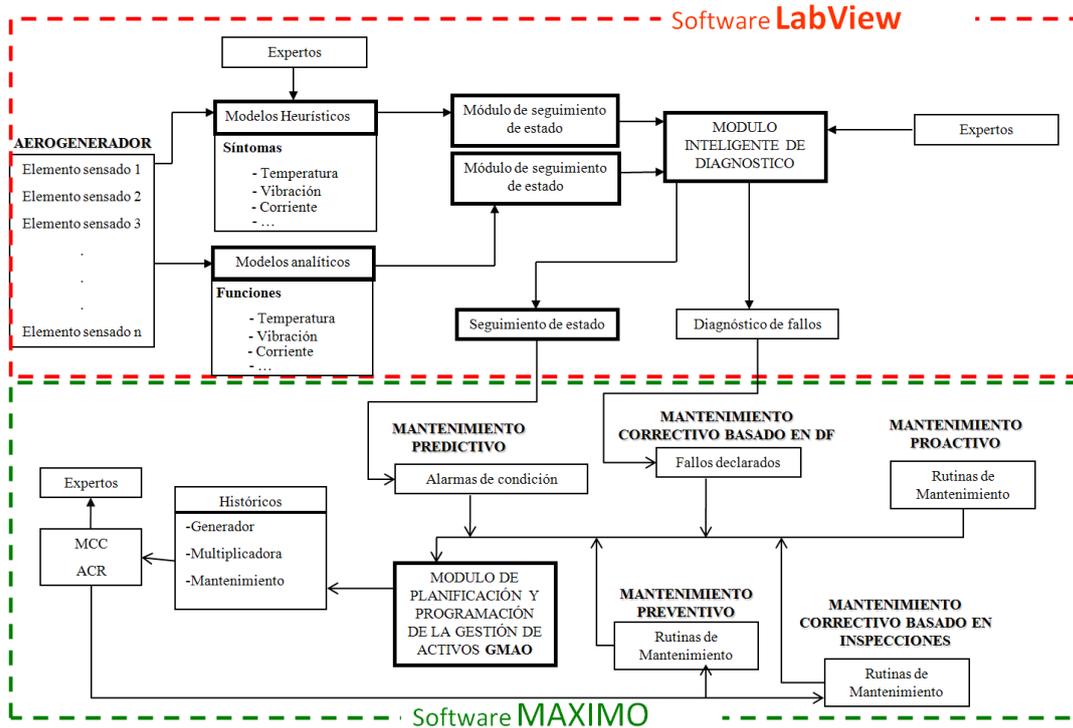


Figura 1: Estructura SISEDFYM

De esta estructura se puede deducir también que el sistema de mantenimiento constituye por sí solo una parte importante del proyecto EOLIA, que no se basa tan sólo en herramientas de gestión sino también de integración de información, y que debe, además, garantizar el funcionamiento continuado del sistema total desarrollado para la tesina.

1.2. Planteamiento del problema

Se van a analizar a continuación, a grandes rasgos, la problemática surgida con esta tesina, que ha dado lugar al desarrollo de una nueva metodología de implementación para el sistema de mantenimiento. Se presentan estos conceptos a modo introductorio, dejando para los correspondientes apartados posteriores un análisis más exhaustivo de los requisitos y problemas del sistema.

Principalmente, la tesina está basada en una herramienta dinámica de gestión inteligente del mantenimiento, con las implicaciones que esto conlleva. Se analiza a continuación la

problemática desde dos aspectos básicos de la tesina: la herramienta de implementación y las características específicas del sistema.

En cuanto a la implementación, debe tratarse de una herramienta automática y dinámica, en constante funcionamiento y susceptible a modificaciones.

Por motivos relacionados con el proyecto global, proyecto EOLIA, para la herramienta de desarrollo se decidió utilizar el software LABVIEW, permitiendo una comunicación adecuada entre los diferentes apartados del sistema.

Para la implementación, tal y como se analizará detalladamente en apartados posteriores, se consideró que la herramienta más adecuada se basaría en Redes de Petri Coloreadas.

Estas decisiones dieron lugar a la problemática que nos ocupa. La implementación de Redes de Petri Coloreadas (RdPC) en LabView se ha desarrollado hasta el momento como sistemas estáticos, que suponen un desarrollo previo de la estructura y un conocimiento exhaustivo de los componentes que participan en el sistema.

Sin embargo, el sistema con el que se trabaja es distinto a las aplicaciones en las que se suelen emplear las Redes de Petri Coloreadas. Se está tratando de modelar un sistema completo de gestión del mantenimiento, lo que supone que la estructura es dinámica ya que en todo momento se introducen nuevas entradas al sistema, con parámetros absolutamente distintos entre ellas, y también se eliminan completamente del sistema una vez las órdenes de trabajo han sido ejecutadas.

Esto dio lugar a las denominadas Redes de Petri Coloreadas Adaptativas (RdPCA), objeto principal de esta tesina de investigación. Esta estructura, específicamente desarrollada para la tesina que nos ocupa, se basa en las Redes de Petri Coloreadas convencionales, pero permitiendo una estructura dinámicamente adaptativa al número de órdenes de trabajo existentes en cada momento.

Por otro lado, en cuanto a las características del sistema de gestión del mantenimiento, se trata de un proyecto aplicado a sistemas de generación eólica offshore y por lo tanto presenta una problemática distinta a los sistemas onshore.

A pesar de que determinadas acciones siguen una estructura convencional de los sistemas de mantenimiento, se debe prestar especial atención a la ejecución de las órdenes de trabajo. Al tratarse de sistemas marítimos, encontramos mayor problemática a la hora de ejecutar las

órdenes, tanto por las condiciones meteorológicas (más restrictivas que en sistemas terrestres) como por los problemas logísticos.

La ejecución de la órdenes de trabajo se debe realizar con un trabajo de planificación logístico muy preciso, ya que los altos costes asociados a los métodos de transporte y los problemas en cuanto a tiempos y disponibilidades pueden llevar a la no viabilidad de la planta de generación.

Por todo esto, la herramienta debe disponer de avanzados sistemas de agrupación de órdenes de trabajo de forma dinámica, que consulten las disponibilidades logísticas y las condiciones meteorológicas, así como los costes y prioridades de las tareas, para tomar las decisiones en cuanto a la ejecución de las tareas, tratando en todo momento de reducir al máximo los costes asociados a estas.

Los sistemas convencionales de gestión del mantenimiento disponen de potentes herramientas informativas de las disponibilidades y las tareas activas en todo momento, y proporcionan sistemas de planificación, si bien es el operario cualificado el que debe considerar las posibles agrupaciones, con el riesgo y alto cargo de responsabilidad que esto conlleva.

Es por esto que la herramienta que se presenta introduce un gran avance en este campo, ya que permite automatizar el proceso de decisión, basándose en un amplio estudio de los requisitos de la empresa.

A modo de resumen, se concluye que el sistema de gestión de mantenimiento a desarrollar consistía en una herramienta completa y automática, que debía incluir mejoras tanto a nivel de funcionamiento y características del sistema de gestión, como en la metodología de implementación.

A lo largo de la presente tesina se presentará la herramienta completa diseñada, analizando durante el proceso las necesidades y mejoras que suponen los conceptos desarrollados en el proyecto para la industria de generación eólica.

1.3. Objetivos

La presente tesina ha sido realizada en el contexto del proyecto europeo EOLIA, cuyo objetivo general consiste en la realización de un sistema global de seguimiento de estado, diagnóstico de fallos y mantenimiento.

Esta tesina se corresponde con la creación del sistema inteligente de mantenimiento. Se van a analizar a continuación los objetivos principales de este sistema.

En primer lugar, se debe analizar la estructura general del proyecto, cumpliendo con los siguientes objetivos:

- Adquirir conocimiento exhaustivo del sistema de mantenimiento a diseñar y sus parámetros característicos.
- Análisis general de las características y posibles herramientas para la implementación del sistema.
- Desarrollo de estructura general de mantenimiento
- Implementación y comprobación de la herramienta diseñada
- Análisis exhaustivo del funcionamiento de la herramienta
- Desarrollo de subsistemas y programas complementarios al sistema de mantenimiento
- Creación del interfaz de usuario adaptado a los requisitos del proyecto

Por otro lado, el sistema de mantenimiento y la herramienta diseñada para tal fin debe cumplir con unos objetivos específicos, que se detallan a continuación:

- El sistema de mantenimiento debe funcionar de forma automática y continuada
- Se debe facilitar la comunicación con los demás sistemas del proyecto EOLIA, así como con los sistemas de mantenimiento empleados por la empresa: Maximo y Excel
- Permitir la introducción y eliminación de órdenes de trabajo sin que afecten al funcionamiento normal del sistema
- Cumplir con las estructuras generales de los sistemas de mantenimiento industriales
- Ofrecer un sistema experto de agrupación de órdenes de trabajo y toma de decisiones.
- Actualización constante de parámetros temporales y meteorológicos

A lo largo de la presente tesina se analizarán con mayor precisión los objetivos aquí planteados, así como las posibilidades que se estudiaron para su implementación y las soluciones adoptadas en cada caso.

2. SISTEMA DE GESTIÓN DEL MANTENIMIENTO

Este documento pretende describir el sistema de gestión de mantenimiento desarrollado por el grupo de supervisión y diagnóstico de fallos de la Universidad Politécnica, englobado en el proyecto EOLIA.

Los objetivos que se pretenden conseguir son:

- Describir del sistema global de gestión del mantenimiento.
- Englobar el sistema en el diagrama general del proyecto (SISEDFYM)
- Definir la relación entre el sistema de mantenimiento y los demás sistemas del proyecto.
- Analizar los diferentes subsistemas y describir la metodología aplicada en cada caso.

El proceso de desarrollo de los subsistemas ha sido el siguiente:

- Estudio de las características principales de cada subsistema
- Análisis de las diversas metodologías aplicables
- Selección de la metodología más adecuada
- Desarrollo de la metodología aplicada

El sistema general de gestión del mantenimiento es un proceso muy amplio y complicado, que se relaciona con otros procesos, por lo que se debe prestar especial atención al desarrollo de las conexiones entre sistemas que permitirán la futura implementación del sistema global.

Por el momento, la conexión a partir de informes programados entre Maximo y Labview no está disponible, pero a partir del trabajo de estudio de funcionamiento de Maximo se ha preparado la herramienta diseñada para que mantenga la estructura que seguirán dichos informes, facilitando así la futura conexión entre los programas.

Se han estudiado, en todos los casos, las diversas metodologías susceptibles de ser aplicadas al sistema. Se presentan las características de todas ellas, así como la selección de la metodología seleccionada, buscando siempre la opción más adecuada y que permita un desarrollo más completo del sistema de gestión del mantenimiento.

Se presenta también en este informe la programación en los entornos Matlab y Simulink que permitirán la implementación de los subsistemas. Estos entornos son los más adecuados para

los subsistemas estudiados y facilitan también, como se ha dicho antes, la relación con los otros sistemas del proyecto final.

Es importante destacar que, a pesar de disponer de los subsistemas implementados en estos programas, quedaría como trabajo pendiente a desarrollar en futuras tareas la implementación en el software Maximo, así como el desarrollo de la comunicación entre los subsistemas para completar el sistema global de gestión de mantenimiento.

Se analiza también la estructura del sistema global de gestión, presentando la estructura seleccionada para la implementación, la problemática que presenta y las nuevas tecnologías desarrolladas para el proyecto.

El sistema global se ha desarrollado en la herramienta Labview, por lo que se presenta los requisitos de esta herramienta para la programación del sistema de gestión y la comunicación entre subsistemas, así como la comunicación entre programas necesaria para el correcto desarrollo del sistema.

2.1. Descripción del sistema de mantenimiento

El sistema general de gestión de mantenimiento está compuesto por diversos subsistemas relacionados entre sí. En este apartado se pretende dar una visión global del sistema para, en apartados posteriores, describir detalladamente los subsistemas que lo conforman.

Se trata también de relacionar el sistema con los diversos sistemas que se integran dentro del proyecto EOLIA (Seguimiento de estado, Diagnóstico de fallos, etc.), tal y como se determinó en el apartado 1.1.

En primer lugar, se definirán los parámetros que intervienen en el sistema. A partir del estudio de los sistemas de gestión, se determina que los tipos de mantenimiento aplicables son los que presenta la Tabla 1.

Podemos ver así que el sistema debe considerar estos tipos de mantenimiento como entradas, teniendo en cuenta las características de la señal en cada caso. El grupo de supervisión y diagnóstico de fallos de la UPV ha desarrollado, englobado en el proyecto EOLIA, los sistemas de mantenimiento predictivo (seguimiento de estado) y correctivo basado en diagnóstico de

fallos, quedando los sistemas preventivos, proactivos y correctivos basados en inspecciones como tarea a desarrollar por la empresa contratante, Ingeteam.

Tabla 1: Tipos de mantenimiento

TIPO	DESCRIPCIÓN
Preventivo	Mantenimiento programado basado en calendario
Predictivo	A partir del seguimiento de estado, basándonos en estado de salud del componente.
Correctivo (diagnóstico fallos)	A partir del diagnóstico de fallos, basándonos en fallos declarados.
Correctivo (inspecciones)	Basado en las conclusiones de los técnicos tras revisiones periódicas.
Proactivo	Estudios de mejora de componentes, piezas, máquinas, etc.

Pasamos ahora a estudiar la estructura del sistema de gestión desarrollado. Para introducir el modelo general, partimos del modelo creado inicialmente (Fig. 2), en el que se puede ver los distintos tipos de mantenimiento como entradas al sistema y el proceso de gestión hasta la creación de la orden de trabajo que incluye las diversas tareas de mantenimiento.

La información relativa a las órdenes de trabajo procederá del software de gestión Maximo, mientras que los subsistemas que forman el proyecto y la Red de Petri se implementarán en Labview, debido a las ventajas que esto supone para relacionar el sistema de mantenimiento con los sistemas de seguimiento de estado, y por la posibilidad de utilizar Excel para comunicar con Maximo.

Como se ha dicho anteriormente, las entradas al sistema serán las proporcionadas por los diversos métodos de mantenimiento. Estas entradas deberán indicar la criticidad del fallo y el tiempo disponible para la reparación, además de otros factores relevantes que se especificarán en apartados posteriores. Este tiempo disponible será el tiempo que se considere adecuado para realizar la reparación en el caso de que la alarma provenga del sistema de diagnóstico de fallos, o el tiempo disponible antes de que se produzca un posible fallo si la alarma proviene de alguno de los demás sistemas.

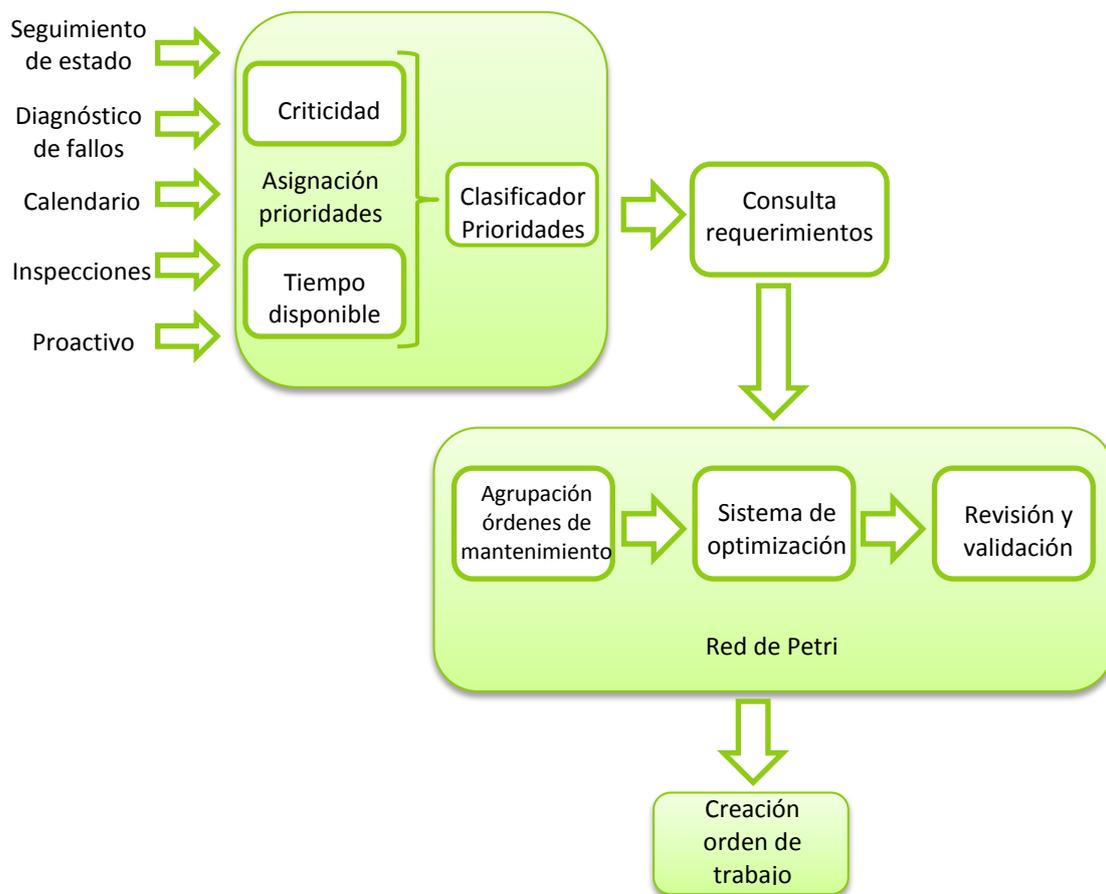


Figura 2: Esquema general sistema de mantenimiento

Una vez disponemos de la información necesaria, pasamos a asignar una prioridad a cada una de las alarmas de mantenimiento que se genera. A esta etapa nos referiremos en adelante como subsistema de asignación de prioridades, y será detallado en el apartado 4.1.

El siguiente paso dentro del esquema general, sería clasificar las alarmas según sus prioridades, para poder trabajar posteriormente con ellas de una forma más clara y accesible. Se propone por lo tanto la agrupación en pilas de cada prioridad, y dentro de cada clasificador organizar las tareas según el tiempo disponible. Este apartado será considerado como clasificador de prioridades.

Pasamos a continuación al subsistema de agrupación de órdenes de mantenimiento, que consiste en analizar las posibilidades de transporte, órdenes de mantenimiento, costes, etc. y agrupar las órdenes según estos parámetros.

Una vez disponemos de estas opciones, el subsistema de optimización seleccionará, a partir de diversos métodos de optimización multiobjetivo, la opción más adecuada de orden de trabajo. Será el subsistema de creación de órdenes de trabajo el que se encargue de recopilar la información de los apartados anteriores y establecer la orden de trabajo.

2.2. Descripción detallada de parámetros de entrada al sistema

En este apartado se analizan las entradas del sistema, que como hemos visto en el punto 2.1 provienen del seguimiento de estado, del diagnóstico de fallos, del mantenimiento programado por calendario, de las inspecciones periódicas y de los estudios proactivos.

Se van a analizar por separado los parámetros que debe transmitir cada una de estas entradas al sistema. Estos parámetros deben proporcionar toda la información necesaria para que el módulo de mantenimiento pueda funcionar de un modo completo y adecuado. La comunicación entre módulos se realizará a partir de un vector, que incluya los parámetros seleccionados para cada tipo de entrada.

2.2.1. Parámetros seguimiento de estado

En este punto, se relaciona el módulo de seguimiento de estado con el módulo de mantenimiento. Sabemos que el módulo de seguimiento de estado examina el estado de los componentes a través de diversos estudios, adecuándose en cada momento a la señal analizada. A partir del conocimiento de la situación de cada componente estudiado, el módulo determina algunos parámetros como el tiempo de vida, los posibles fallos, etc. Cuando un posible fallo esté próximo a ocurrir, en el dominio del tiempo, se generará un aviso para el sistema de mantenimiento.

Estudiando este bloque y las necesidades del sistema de mantenimiento, se ha considerado necesario que la información transmitida entre los módulos sea la siguiente:

- Tiempo estimado disponible hasta la ocurrencia del fallo
- Criticidad del fallo
- Componente al que nos referimos
- Situación del componente dentro del aerogenerador
- Situación del aerogenerador dentro del parque
- Código causa-solución
- Material y unidades necesarias
- Cualificación del personal necesario, cantidad y horas de cada tipo
- Coste/hora estimado de la tarea
- Condiciones meteorológicas

2.2.2. Parámetros para diagnóstico de fallos

El sistema de diagnóstico de fallos generará avisos para el sistema de mantenimiento una vez ha ocurrido el fallo. A partir de la ocurrencia del fallo, el módulo hace un análisis para determinar su procedencia, evitando así falsos diagnósticos en componentes generados por fallos en otros componentes.

La información que debe transmitir al sistema de mantenimiento será:

- Tiempo estimado adecuado para la resolución del problema
- Criticidad del fallo
- Componente al que nos referimos
- Situación del componente dentro del aerogenerador
- Situación del aerogenerador dentro del parque
- Código causa-solución
- Material y unidades necesarias
- Cualificación del personal necesario, cantidad y horas de cada tipo
- Coste/hora estimado de la tarea
- Condiciones meteorológicas

2.2.3. Parámetros mantenimiento basado en calendario

El sistema de mantenimiento basado en calendario utiliza las reparaciones y revisiones programadas y planificadas para generar los avisos que servirán de entrada al módulo de gestión del mantenimiento. Es un módulo más sencillo que los anteriores, ya que no realiza ningún tipo de análisis. El disponer de estos avisos con tiempo suficiente permite planificar adecuadamente el mantenimiento.

Los parámetros de comunicación entre los módulos serán:

- Tiempo disponible hasta la reparación
- Criticidad del fallo
- Componente al que nos referimos
- Situación del componente dentro del aerogenerador
- Situación del aerogenerador dentro del parque
- Código causa-solución
- Material y unidades necesarias
- Cualificación del personal necesario, cantidad y horas de cada tipo
- Coste/hora estimado de la tarea
- Condiciones meteorológicas

2.2.4. Parámetros mantenimiento basado en inspecciones

En este módulo se tratan los avisos generados a partir de la detección de componentes en estado defectuoso, o elementos en funcionamiento inadecuado. Esta detección se realiza a partir de las inspecciones periódicas que tanto la maquinaria como los elementos debe cumplir. Se trata de una detección inesperada, en la que el tiempo disponible puede ser muy variable, en función del tipo de fallo y tipo de componente.

Al igual que en los módulos anteriores, se deben definir los parámetros comunicados entre los sistemas:

- Tiempo estimado adecuado para la resolución del problema
- Criticidad del fallo
- Componente al que nos referimos

- Situación del componente dentro del aerogenerador
- Situación del aerogenerador dentro del parque
- Código causa-solución
- Material y unidades necesarias
- Cualificación del personal necesario, cantidad y horas de cada tipo
- Coste/hora estimado de la tarea
- Condiciones meteorológicas

2.2.5. Parámetros mantenimiento proactivo

En el caso del mantenimiento proactivo se realizan estudios sobre los componentes, las piezas, etc. con el objetivo de determinar posibles mejoras. Estas mejoras serían ampliamente estudiadas y analizadas, y en caso de ser aceptadas y precisar algún cambio, producirían el aviso para el sistema de mantenimiento.

En este caso, el tiempo disponible es relativo y dependerá de si el resultado del estudio determina un cambio inmediato o a largo plazo.

La información necesaria para el sistema de mantenimiento será la siguiente:

- Tiempo disponible para realizar las modificaciones.
- Criticidad del fallo
- Componente al que nos referimos
- Situación del componente dentro del aerogenerador
- Situación del aerogenerador dentro del parque
- Código causa-solución
- Material y unidades necesarias
- Cualificación del personal necesario, cantidad y horas de cada tipo
- Coste/hora estimado de la tarea
- Condiciones meteorológicas

2.3. Consideraciones adicionales

Los parámetros determinados para cada tipo de sistema de mantenimiento son sólo indicativos, aunque recomendables. Con esta afirmación pretendemos indicar que se trata de los parámetros necesarios para el correcto funcionamiento del sistema, si bien el sistema se puede adaptar y modificar en función a los requisitos marcados por la empresa y las necesidades del sistema de gestión de mantenimiento.

Los parámetros se heredan del software Maximo, ya que es el programa maestro en el que deben estar reflejadas las órdenes de trabajo y toda su información.

Según el diseño del sistema, la implementación en Labview sería la encargada de consultar las bases de datos correspondientes para determinar si existe disponibilidad de material y personal, pero al igual que el resto de parámetros, al tratarse de un sistema de diseño propio, se podría modificar la programación para que la Red de Petri heredara esta disponibilidad de Maximo.

En cuanto a la estimación de coste/hora, se trata de un parámetro fácilmente calculable en Maximo y muy necesario para el cálculo de los costes para el sistema de agrupación y el de optimización. Es más interesante el coste/hora que el coste total de la tarea, ya que al agrupar las órdenes el tiempo de ejecución puede aumentar y por lo tanto los costes también.

Inicialmente, se consideró que Maximo informaría al sistema de la criticidad y el tiempo disponible para la reparación y el sistema de asignación de prioridades la determinaría en función a estos datos, si bien tras estudiar las posibilidades del programa se vio que podía determinar la prioridad directamente e informar al sistema desarrollado en Labview. Queda implementado pues el sistema de asignación de prioridades, pero con la posibilidad de eliminar este subsistema y determinar la prioridad a partir de la información heredada de Maximo.

3. RED DE PETRI GLOBAL PARA LA GESTIÓN DEL MANTENIMIENTO

En este apartado se va a analizar la herramienta que se ha optado por utilizar para la gestión global del sistema de mantenimiento, ya que se precisa una herramienta capaz de relacionar todos los subsistemas entre sí, gestionar la información del sistema y dirigir el flujo de trabajo.

Para el sistema estudiado, la gestión de mantenimiento de aerogeneradores offshore, y dada la experiencia del grupo de investigación en aplicaciones basadas en Redes de Petri, se ha decidido analizar la adecuación de Redes de Petri Coloreadas para el sistema en cuestión.

3.1. Conceptos básicos Redes de Petri Coloreadas

Se presentan a continuación diversos conceptos de Redes de Petri y redes de Petri Coloreadas, con objeto de justificar su adecuación al tipo de sistema que se pretende implementar.

Las Redes de Petri (RdP) convencionales son una herramienta gráfica que se utiliza para modelar distintos tipos de sistemas. Han demostrado ser adecuadas para sistemas con recursos compartidos, sincronizaciones y concurrencias. Sin embargo, si queremos diferenciar las marcas en el mismo lugar recurriremos a Redes de Petri Coloreadas (RdPC). Esto es necesario para el sistema que se presenta ya que trabajaremos con gran número de órdenes de trabajo que han de ser identificadas. Además, las RdPC resultan muy útiles cuando una RdP o un patrón dentro de esta se repiten. En este caso, al aumentar las redes, aumenta el tamaño del sistema y si trabajamos con RdPC trabajamos sobre la misma red distinguiendo las marcas por separado.

Veamos la definición de una RdP convencional, según la cual una RdP es el quinteto:

$$PN \equiv (P, T, F, W, M_0)$$

Donde P es el conjunto finito de lugares, T el conjunto finito de transiciones, F el conjunto de arcos, W la función de peso y M_0 el marcado inicial.

Por otro lado, vamos a ver la diferencia con las RdPC, que están formadas por el sexteto:

$$CPN \equiv (P, T, Pre, Post, M_0, C)$$

Donde P , T y M_0 se corresponden con las RdP, C es el conjunto de colores y Pre y $Post$ son funciones relativas al disparo de las marcas coloreadas.

Vamos a ver a continuación un ejemplo de RdP y RdPC, para mostrar la simplificación que se produce al introducir marcas coloreadas en las redes convencionales. El caso está formado con dos sistemas idénticos, para dos vagonetas. Cada uno de ellos se desplaza a derechas o izquierdas hasta alcanzar el final del recorrido, donde cambia de sentido. El sistema requeriría dos RdP iguales, una para cada vagón. (Fig. 3). Los superíndices b y v hacen referencia a la vagoneta azul y amarilla, respectivamente.

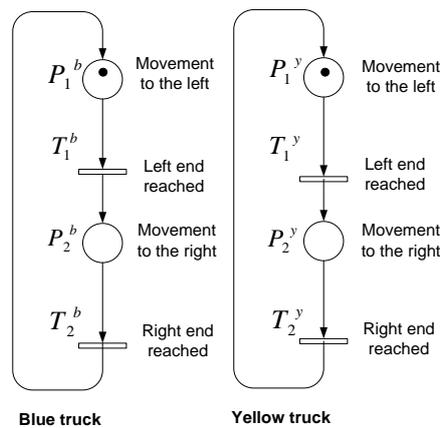


Figura 3: Ejemplo Red de Petri para vagonetas

Los lugares P_1 y P_2 muestran el desplazamiento a izquierda o derecha, y las transiciones T_1 y T_2 con la llegada al correspondiente final de carril.

Como ya se ha dicho anteriormente, las RdPC son muy útiles para sistemas duplicados, como es este caso. Mediante una RdPC podríamos simplificar el sistema a una sola red, distinguiendo las marcas entre ellas mediante el color.

Veamos a continuación dos formas distintas de plantear este sistema mediante RdPC. En la Fig. 4(a) se ha creado la red asignando el color a los distintos vagones (b , y). La red sigue la misma estructura que la anterior, pero las marcas han pasado a ser coloreadas y las transiciones están relacionadas con el color. A cada arco se le asocia una función (Id) para traducir la relación entre el color asociado con la transición (color de disparo) y el color asociado con el lugar correspondiente (color de marcado).

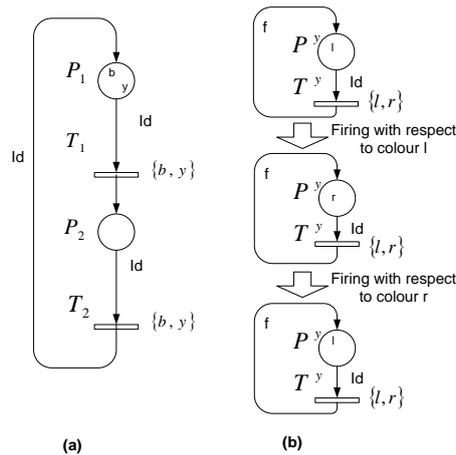


Figura 4 (a) y (b): Ejemplo vagonetas con Red de Petri Coloreada

En la Fig. 4(b) vemos la red implementada para el camión amarillo, asociando el color a la dirección de movimiento. Se ha representado la secuencia de disparo de las transiciones para observar el cambio en el color de la marca, para hacer referencia al movimiento en cada instante.

A través de este ejemplo se pretende comprender mejor las RdPC y su adecuación al sistema que se pretende diseñar. En el apartado anterior se ha dicho que la cantidad de órdenes de trabajo que se manejan con un sistema de gestión es un factor importante para el desarrollo de la herramienta. Es por esto que se plantea la utilización de RdPC, ya que la estructura del sistema de mantenimiento es el mismo para todas las órdenes, con diferencias en las transiciones, por lo que si se implementa una RdPC con colores asociados a las órdenes de mantenimiento y se definen correctamente los colores de disparo se puede simplificar y generalizar el sistema.

3.2. Red de Petri para el Mantenimiento

En este apartado se presenta una visión global del sistema, que nos permita conocer el funcionamiento del sistema y los subsistemas que lo forman, que serán analizados posteriormente con más detalle. Se analizan también las características de la Red de Petri

implementada en Labview, con las mejoras que esto supone para la creación de un sistema automático de gestión del mantenimiento.

Veamos a continuación la estructura base de la red de Petri que seguirán cada una de las tareas creadas e introducidas en el sistema. La red se puede dividir en dos sistemas:

- Subsistema de consulta de requerimientos
- Subsistema de agrupación, decisión y revisión

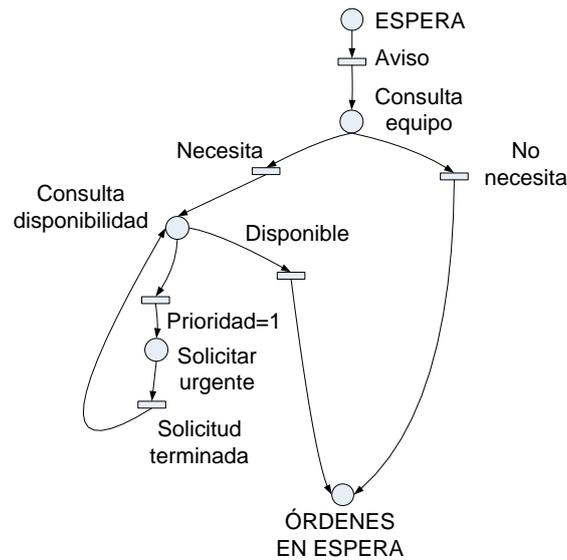


Figura 5: Consulta de requerimientos equipo de trabajo

El primer subsistema (Figuras 5 y 6) realiza de forma simultánea una consulta en los datos de información de la orden acerca de las necesidades en cuando a material para la reparación y el personal cualificado necesario, determinando en primer lugar si es necesario alguno de estos parámetros y en una etapa posterior las características de los requisitos (horas, cantidad, etc.).

Durante el proceso la prioridad de la tarea está en constante actualización, en función del tiempo disponible. Si se da el caso de llegar niveles urgentes de realización, se realiza solicitud urgente de los requisitos que no han sido cumplidos.

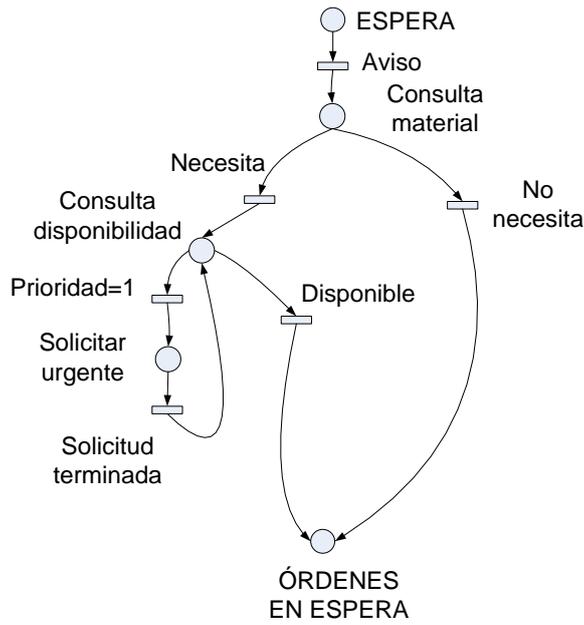


Figura 6: Consulta de requerimientos material

Una vez las necesidades de preparación de la orden están cubiertas, la orden queda en una etapa de espera, pendiente de que las condiciones meteorológicas sean favorables para pasar al segundo subsistema.

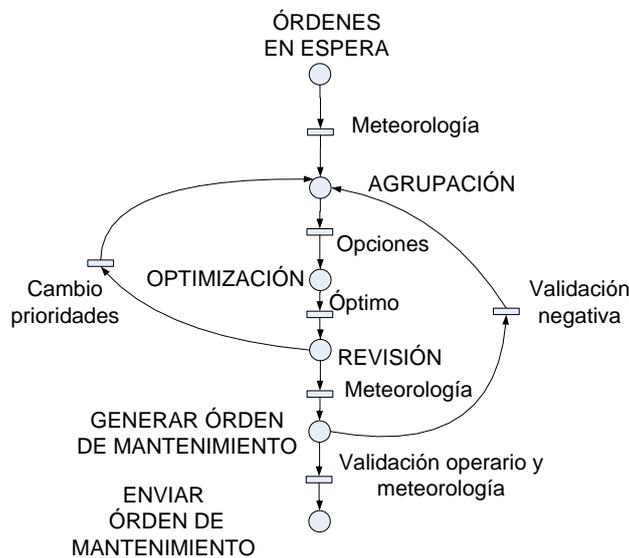


Figura 7: Red de Petri para las fases de agrupación, decisión y revisión

Esta segunda etapa de la red de Petri (Fig. 7) se encarga de realizar las posibles agrupaciones en función de las tareas activas y decidir, mediante diversas metodologías, la solución óptima

en función de los parámetros marcados por el sistema. Posteriormente, se precisa validación por parte del operario, que permite recalcular las opciones en caso de que se considere que la opción no es adecuada. A lo largo de todo el proceso se mantiene la actualización de prioridades y la revisión de las condiciones meteorológicas.

La fase de agrupación, como se verá en el apartado 4.2, tiene que considerar las posibilidades de transporte hasta la planta de aerogeneradores (tipo de barco, tamaño, capacidad, posibilidad de utilizar helicóptero, etc.) Debe considerarse también los costes asociados a la realización de las tareas y al envío de la embarcación. Con estos parámetros el sistema calcula las posibles opciones de agrupación de tareas, que se trasladarán al subsistema de decisión multiobjetivo.

Para la decisión multiobjetivo se han propuesto varios métodos diferentes para el cálculo de la agrupación de tareas. Es importante disponer de diversas opciones de decisión multiobjetivo, ya que la precisión de la información con la que se trabaja depende en gran medida de la empresa o el tipo de máquina con el que se trabaja, y ya que las metodologías están basadas en diferentes parámetros, las soluciones pueden no ser iguales.

Los métodos que se proponen en este sistema son los siguientes:

- Basado en restricciones
- Basado en interfaz de preferencias
- Lógica Fuzzy

El último paso de la red consiste en pasar por la revisión y aceptación del operario encargado, al mando del sistema de mantenimiento y cumpliendo con los parámetros establecidos por la empresa correspondiente. Para cumplir este objetivo, el sistema debe informar al usuario a través de Maximo de las posibles agrupaciones y sus parámetros, así como de la opción considerada óptima. Aparte de la validación del operario, es preciso que las condiciones meteorológicas continúen cumpliéndose, ya que se trata de un parámetro susceptible a cambios.

Una vez que el usuario ha determinado la opción más adecuada, el resto de órdenes que no han sido ejecutadas debe volver al estado de espera para permitir su futura agrupación.

Durante todo el proceso debe mantenerse la actualización de prioridades de cada tarea, ya que este parámetro varía en función del tiempo y determina la urgencia de reparación, por lo que puede modificar las agrupaciones.

La unión de todos los subsistemas que han sido explicados y analizados da lugar a la estructura general de la red de Petri Coloreada Adaptativa, a implementar en la herramienta Labview, que podemos ver en la Fig. 8.

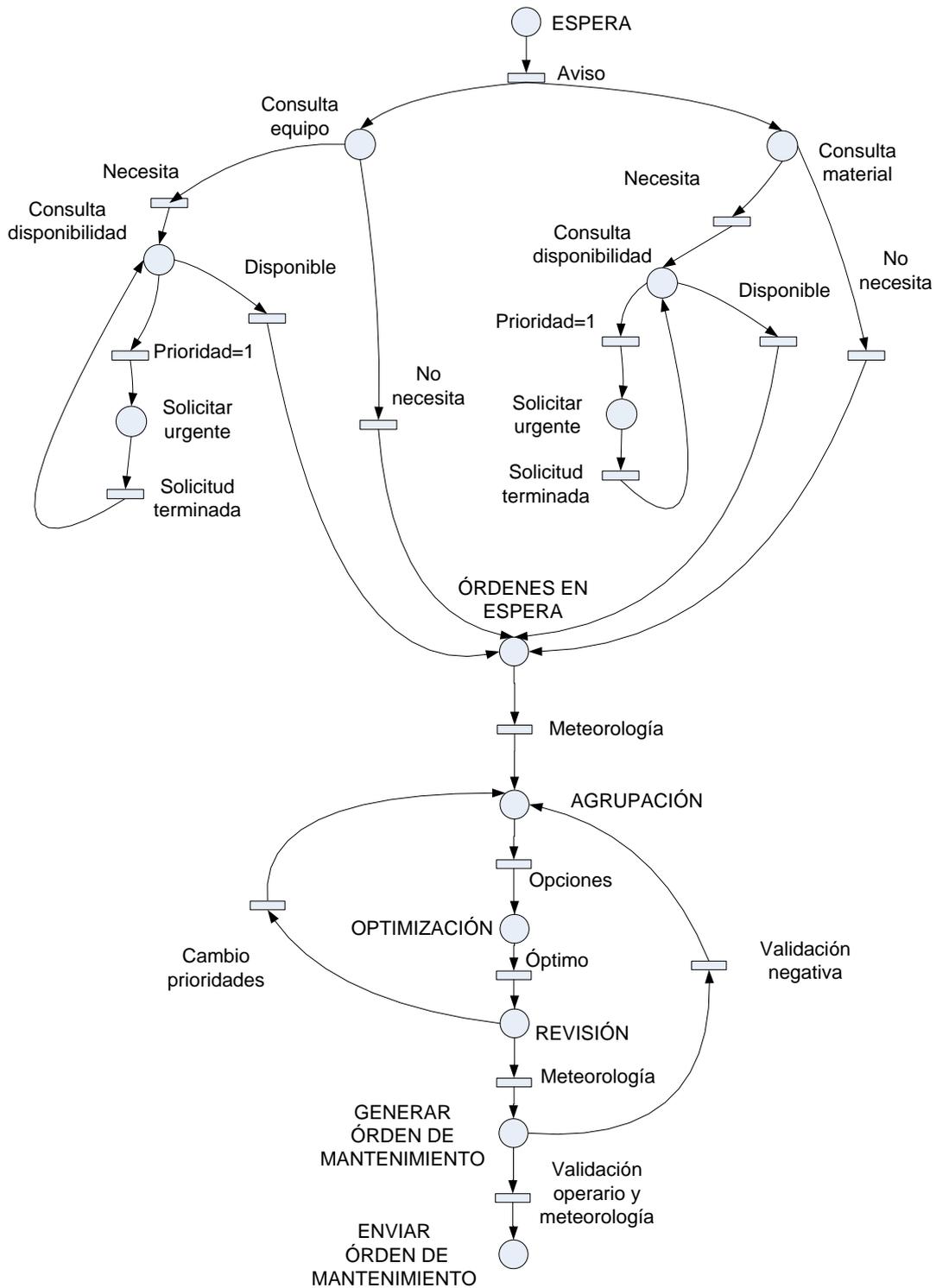


Figura 8: Estructura global RdPC para el mantenimiento

3.3. Red de Petri adaptativa

Una vez analizada la herramienta y su funcionamiento, pasamos a analizar la estructura de programación que requiere la implementación de la red de Petri Coloreada en Labview. A partir del estudio de las características del sistema de gestión de mantenimiento que se pretende implementar para trabajar conjuntamente con el software Maximo, se determinó que la red diseñada debía trabajar de forma totalmente automática, a partir de la información obtenida de Maximo y proporcionando en su fase final un apoyo al mencionado programa.

Por otro lado, se planteó un problema que no se había considerado hasta el momento para el desarrollo de la aplicación: se debía crear un sistema que trabajara con tantas órdenes de trabajo como estuvieran activas en ese momento, teniendo en cuenta que la evolución de las mismas sigue parámetros distintos y su permanencia en la red es variable.

Debido a estas conclusiones, se decidió investigar una nueva metodología de redes de Petri que fuera capaz de añadir nuevas órdenes de trabajo (nuevas marcas en la red) y borrar las ya finalizadas, sin que ninguno de estos factores alterara de modo alguno al funcionamiento global del sistema. La metodología desarrollada debe ser totalmente genérica en todos los puntos, permitiendo trabajar con las órdenes activas, sin importar el tamaño del sistema y el número de órdenes.

Veamos a continuación una pequeña explicación, a modo de introducción, del funcionamiento del sistema a implementar. La red de Petri trabaja conjuntamente con Excel, para utilizar este programa como base de datos tanto de las órdenes activas como de la información correspondiente a cada una de ellas.

El sistema utiliza una red de Petri para evaluar las condiciones de las diferentes tareas. Los diversos tipos de mantenimiento alimentan el sistema, incluyendo en cada caso la información relevante a la tarea. Esta información se introduce en el archivo Excel de nueva orden de mantenimiento, incluyendo un identificador del sistema de origen de la tarea. De este modo, el sistema implementado en Labview es capaz de generar una nueva orden de trabajo totalmente independiente de las anteriores, modificando la estructura inicial de la red de Petri implementada.

El funcionamiento es el siguiente: se define una estructura de Red de Petri base, que se ha analizado en detalle en el apartado anterior, con las transiciones y entradas del sistema. A

partir de este punto, cada nueva orden de trabajo repite la estructura anterior, manteniendo siempre un código identificador de la tarea.

De este modo, se trabaja con diversas redes de Petri duplicadas pero independientes entre sí. Esto supone una gran ventaja frente a métodos tradicionales de redes de Petri: se pueden generar tantas órdenes de mantenimiento como sean necesarias, ya que todo el sistema es automático y funciona en función del identificador de la orden.

Se va a presentar a continuación la metodología empleada para la implementación de la Red de Petri en la herramienta Labview. Se utiliza esta herramienta ya que el proyecto trabaja con ella en otros apartados y permite así una fácil comunicación entre sistemas.

Previamente se ha estudiado la estructura general de la RdPC que se va a utilizar para el sistema de mantenimiento, y se ha analizado el problema del número desconocido de órdenes en ejecución. El sistema debe ser capaz de funcionar con las órdenes que estén activas en ese momento, que puede muy reducido o muy amplio en función del momento. Esto va a suponer que la RdPC no tenga un tamaño determinado, sino que debe tratarse de una estructura dinámica.

Para poder implementar esta estructura se ha desarrollado una librería propia en Labview, a partir de la cual se puede trabajar de forma automática con la RdPC. La metodología empleada consiste en definir la estructura principal de la RdPC, y cada vez que se crea una orden de trabajo nueva crear otra nueva estructura igual, funcionando conjuntamente con las ordenes anteriores como una sola red. Como marca de color para la red, se utilizará el identificador de la orden de trabajo.

Tabla 2: Estructura RdPC en Labview

Origen	Número transición	Marcado inicial	Destino	Marcado final
--------	----------------------	--------------------	---------	------------------

Para definir los diferentes estados de la RdPC se utiliza la estructura de la tabla 2, definiendo estos parámetros para cada una de las transiciones asociadas. En ella, podemos diferenciar origen, número de transición, marcado inicial, destino, y marcado final. Origen y destino hacen referencia al lugar de la RdPC del que se extrae y deposita la marca al activarse la transición, que viene determinado por el número de transición. Mediante el marcado inicial y final

determinamos la marca de color que debe haber en el lugar de origen para que se produzca el disparo, y el color de la marca en el lugar de destino.

Para el correcto funcionamiento de la herramienta deben estar disponibles también los vectores de entradas y transiciones, que al igual que la estructura anterior cambiará de tamaño en función de las órdenes de trabajo activas.

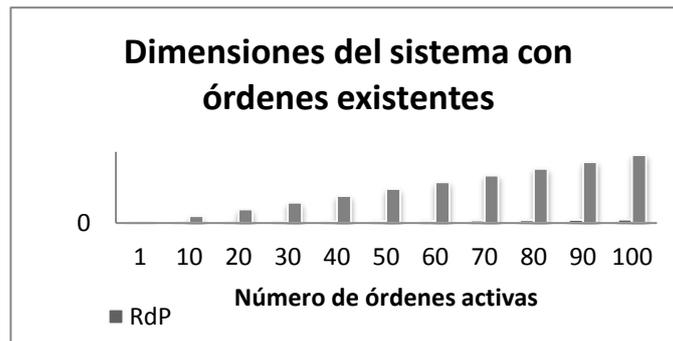


Figura 9: Dimensiones RdPC

En la Fig. 9 se puede ver el aumento de tamaño del sistema en función del número de órdenes activas. Como se puede apreciar, en función del aumento de dicho número, aumenta el número de duplicados de la red original y crece también el número de entradas y transiciones. Mediante esta imagen se pretende mostrar la diferencia entre trabajar con 20 órdenes y menos de 500 entradas, o con 10 órdenes y más de 2000 entradas.

El sistema que se ha diseñado está totalmente preparado para aumentar toda la estructura cada vez que se recibe una orden de trabajo, manteniendo las órdenes activas previamente en el estado en que se encontraban, se trata por lo tanto de un crecimiento dinámico, no de aumentar la estructura y reiniciar el sistema. Mediante la misma metodología, una vez la orden ha sido ejecutada y debe ser extraída del sistema de mantenimiento, la herramienta es capaz de borrar su estructura, actualizando los parámetros correspondientes, sin que esto afecte al funcionamiento normal del sistema.

Además, la metodología está diseñada para admitir distintas estructuras de red. Es decir, si sabemos que determinada orden de trabajo no precisa alguno de los subsistemas, se puede crear la nueva estructura de red que se añadirá a la estructura básica sin incluir esa parte de la red, y al ser un sistema que trabaja por referencias de posición en las transiciones, en sistema funciona igual que cuando se trabaja con la estructura completa.

El sistema se basa en gran medida en la comunicación Labview-Excel. El funcionamiento del sistema está implementado en Labview, pero la definición de la estructura de la red y de los vectores de entradas y transiciones se realiza a partir de Excel. Por este motivo, se ha desarrollado ampliamente la librería de lectura-escritura entre estos dos programas. Además, esta librería aumenta las posibilidades de desarrollo de la herramienta, ya que facilita el autodesarrollo de la red.

Además, esta conexión entre programas es necesaria para la consulta de requisitos de cada orden de trabajo, ya que se utilizarán archivos Excel como bases de datos, tanto para el sistema como para el programa de gestión Maximo, con el que se trabaja conjuntamente. A partir de Maximo se informará al sistema implementado mediante informes Excel, para facilitar la comunicación.

El funcionamiento de la metodología desarrollada, así como la estructura de la red y las librerías implementadas se desarrollará ampliamente en apartados posteriores.

4. DESCRIPCIÓN DEL SISTEMA DE MANTENIMIENTO Y LOS SUBSISTEMAS QUE LO CONFORMAN

En el apartado anterior se ha presentado en detalle la estructura de la Red de Petri utilizada para el sistema de mantenimiento, lo que ha permitido obtener una idea general del funcionamiento del sistema desarrollado.

Para analizar dicha estructura, se ha hecho referencia a diversos subsistemas que, empleados de forma conjunta con los diversos apartados de la red, dan lugar al sistema global de mantenimiento.

Estos subsistemas son:

- Subsistema asignación de prioridades
- Subsistema de agrupación de tareas de mantenimiento
- Subsistema de decisión multiobjetivo

En el presente apartado se procede a analizar en detalle la estructura de los subsistemas planteados, desarrollando para ello un estudio conceptual de las necesidades que presenta cada subsistema y las posibilidades de implementación, para llegar a la toma de decisión de metodología a utilizar. La programación de dichos subsistemas se presentará en apartados posteriores.

4.1. Subsistema asignación de prioridades

Tal y como se ha descrito anteriormente, este subsistema se encarga de asignar las prioridades a las alarmas de mantenimiento en función del tiempo disponible y la criticidad del fallo, ambos parámetros procedentes del vector de comunicación de cada entrada con el módulo de mantenimiento.

El primer punto importante será definir los rangos de criticidad y tiempo disponible, así como de prioridades.

Como vemos en la Tabla 3, los rangos de criticidad irán del 1 al 10, siendo el nivel más crítico el 10. La tabla incluye también una descripción de cada nivel.

Tabla 3: Parámetros Criticidad

CRITICIDAD	EFEECTO	DESCRIPCIÓN
1	Ninguno	No hay razón para esperar que el fallo tenga algún efecto sobre seguridad, salud, medio ambiente o misión.
2	Muy bajo	Trastorno menor de la función de habilidad. La reparación del fallo puede realizarse durante el aviso de problema.
3	Bajo	Trastorno menor de la función de habilidad. La reparación del fallo puede ser más larga que el aviso de problema, pero no retrasa la misión.
4	Bajo a moderado	Trastorno moderado de la función de habilidad. Una parte de la misión puede necesitar modificaciones o retraso en el proceso.
5	Moderado	Trastorno moderado de la función de habilidad. 100% de la misión puede necesitar modificaciones o retraso en el proceso.
6	Moderado a alto	Trastorno moderado de la función de habilidad. Una parte de la misión se pierde. Retardo moderado en la función de restablecimiento.
7	Alto	Trastorno alto de la función de habilidad. Una parte de la misión se pierde. Retardo significativo en la función de restablecimiento.
8	Muy alto	Trastorno alto de la función de habilidad. Toda la misión se pierde. Retardo significativo en la función de restablecimiento.
9	Peligro	Consecuencia potencial en seguridad, salud o medio ambiente. El fallo ocurrirá con advertencia.
10	Peligro	Consecuencia potencial en seguridad, salud o medio ambiente. El fallo ocurrirá sin advertencia.

Para el desarrollo de este subsistema se han tomado unas consideraciones del tiempo disponible de la forma:

- Menos de 24 horas
- De 24h a 3 días
- Entre 3 y 7 días
- Más de 1 semana, menos de 4 semanas
- Más de 1 mes

A estos rangos se les asocia un nivel de tiempo disponible, al igual que se ha hecho con las criticidades.

Esta asociación es de la forma:

Tabla 4: Parámetros Tiempo Disponible

NIVEL	DESCRIPCIÓN	APLICACIÓN
1	< 24 horas	Urgente. Implica parada del aerogenerador.
2	24horas a 3días	Reparación a corto plazo.
3	3 a 7 días	Reparación próxima en el tiempo, no urgente
4	1 a 4 semanas	Planificación a corto plazo.
5	>4 semanas	Planificación a largo plazo.

En la Tabla 5 podemos ver los niveles de prioridad utilizados en el campo de la energía eólica. Estos niveles van del 1 al 5, siendo el número 1 la más prioritaria. Esta clasificación se realizará a partir de la criticidad y el tiempo disponible.

Al depender del tiempo, que es un parámetro variable, la prioridad podrá ser modificada durante el resto del proceso si es necesario.

Tabla 5: Parámetros Prioridades

PRIORIDAD	DESCRIPCIÓN	APLICACIÓN
1	Emergencia	Seguridad
2	Urgente	Funcionamiento continuo de las instalaciones de riesgo
3	Prioritaria	Soporte a la misión, cumplimiento de plazos
4	Rutinaria	Cumplimiento sobre la base de "primero que llega, primero servido"
5	Discrecional	Deseable, pero no esencial

Para el subsistema de asignación de prioridades se proponen dos opciones: asignación a partir de tabla y asignación a partir de control inteligente (lógica borrosa). Se presentan a continuación estos dos métodos, ofreciendo así distintas posibilidades de resolución del subsistema.

4.1.1. Asignación a partir de tabla.

El método de asignación a partir de tabla consiste en crear una tabla con dos ejes, en el primero la criticidad y en el segundo el tiempo disponible. De este modo, en cada casilla asignamos una prioridad en función de sus características y el valor de los parámetros de entrada.

A partir del estudio de las entradas, se realiza una asignación en cada estado de la tabla. Se debe tener cuidado a la hora de la asignación y ser consecuentes con la relación con el resto de estados, ya que para que el método sea útil esta asignación debe seguir una lógica clara y coherente. De este modo, se ha utilizado un razonamiento del tipo:

A mayor criticidad, mayor prioridad

A menor tiempo disponible, mayor prioridad

Combinando estas dos reglas, se asignan las prioridades tal y como vemos en la Tabla 6.

Tabla 6: Asignación de Prioridades

Tiempo Criticidad	5	4	3	2	1
1	5	5	5	4	4
2	5	5	4	4	4
3	5	4	4	4	3
4	4	4	4	3	3
5	4	4	3	3	2
6	4	3	3	2	2
7	3	3	2	2	2
8	3	2	2	2	1
9	2	2	2	1	1
10	2	2	1	1	1

Las ventajas de este método son:

- Implementación más sencilla que método fuzzy.
- Accesible y de fácil comprensión (Las modificaciones se aplican directamente a la tabla)
- Método muy transparente (a primera vista vemos la lógica que se utiliza)

Las desventajas con las que nos encontramos son:

- El resultado es poco preciso
- Requiere análisis cuidadoso de la asignación en cada caso particular

4.1.2. Asignación a partir de control inteligente (fuzzy logic)

El método de asignación de prioridades a partir de la utilización de lógica borrosa consiste en definir los rangos de los valores de entrada vistos anteriormente (tiempo disponible y criticidad) y utilizar metodología fuzzy para determinar la prioridad de forma más exacta.

Según la lógica borrosa, se deben definir las funciones de pertenencia que cubran todo el rango de entradas determinadas, así como las reglas que relacionan las dos entradas para determinar la salida.

Las ventajas de este método frente al anterior son las siguientes:

- Los valores de criticidad y tiempo disponible no deben ser números enteros sino que permiten determinar valores intermedios.
- La salida no es un parámetro tan rígido, permite un ajuste más preciso.

Respecto a las desventajas, podemos determinar:

- La programación de estos sistemas requiere un buen conocimiento de los parámetros y la relación entre ellos.
- Se precisa el trabajo conjunto de los programadores de lógica borrosa y los expertos en asignación de prioridad.
- No es un método tan claro visualmente, requiere conocimientos de lógica borrosa.

Para la programación de este sistema, se utilizó inicialmente el programa Matlab, debido a las ventajas que aporta trabajar directamente con la librería específica de lógica borrosa “Fuzzy Toolbox”. En esta librería, se definen las entradas y sus rangos y parámetros. Posteriormente, se procede de igual manera con la variable de salida, en este caso la prioridad de la orden de mantenimiento. Por último, se definen las reglas que relacionan las dos entradas a utilizar y determinan la salida.

En la Fig. 10 vemos la estructura de programación dentro de la librería “Fuzzy Toolbox”, para el subsistema de asignación de prioridades en función del tiempo disponible para la reparación y de la criticidad del fallo.

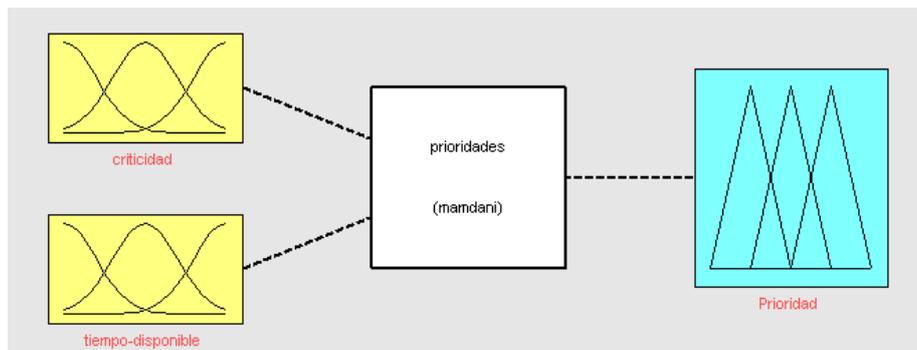


Figura 10: Sistema Fuzzy para asignación de prioridades

4.1.3. Asignación a partir de la información de Maximo

A lo largo del desarrollo de la presente tesina, se ha desarrollado un conocimiento de la herramienta Maximo a partir del estudio de sus posibilidades y características. Es por esto que se ha podido comprobar la amplia capacidad de dicho programa.

Entre las características de funcionamiento de Maximo, existe un parámetro de prioridad que se asigna a la creación de cada tarea de mantenimiento.

Si bien se han desarrollado las técnicas presentadas en apartados anteriores para la asignación de prioridades, estas herramientas precisan de un estudio exhaustivo de cada componente y tarea de mantenimiento susceptible a ser incluido en el sistema.

Debido a términos de confidencialidad, no se ha podido disponer de la totalidad de la información precisada para el correcto funcionamiento de las herramientas.

Es por esto que se planteó la necesidad de emplear el parámetro de prioridad disponible en Maximo, dejando el sistema de mantenimiento habilitado para la inclusión de las herramientas desarrolladas en caso de disponer de la información necesaria.

Por lo tanto, de ahora en adelante se trabajará con el valor de prioridad establecido en el momento de la creación de la orden de trabajo en Maximo, pero manteniendo en todo momento la actualización temporal dentro del sistema, ya que como se ha determinado el paso del tiempo hace incrementar la prioridad de la tarea.

4.2. Subsistema de agrupación de tareas de mantenimiento

A continuación, se procede a explicar el subsistema diseñado para la agrupación de las tareas de mantenimiento. Este subsistema tiene una gran importancia dentro del sistema global, ya que trabaja conjuntamente con los subsistemas anteriores y la información heredada del programa de gestión Maximo, determinando las posibles agrupaciones de las distintas ordenes de trabajo preparadas para su ejecución, teniendo como objetivo mayor la reducción de costes y optimización de los sistemas de mantenimiento, y utilizar estas agrupaciones como entradas para el subsistema de decisión multiobjetivo.

Los sistemas informáticos de gestión de mantenimiento no disponen de este tipo de aplicaciones de agrupación de tareas, dejando para el usuario la decisión de las órdenes a ejecutar y las posibilidades a adjuntar a la orden de trabajo. Sin embargo, cuando se trabaja con sistemas de seguimiento de estado que son capaces de proporcionar información del componente y el tiempo disponible para la reparación con tiempo suficiente para la planificación de las tareas de mantenimiento, es especialmente importante poder contar con una herramienta de ayuda a la decisión de la agrupación más adecuada, teniendo también en cuenta que los altos costes de las embarcaciones de mantenimiento hacen que sea imprescindible una correcta planificación y la agrupación de tantas órdenes como sea posible y adecuado para el funcionamiento del sistema.

4.2.1. Funcionamiento de la herramienta y parámetros de decisión

La aplicación diseñada para este subsistema de agrupación de órdenes de mantenimiento parte de la base de que las posibles embarcaciones y sus parámetros son conocidos por la empresa de mantenimiento, en base al contrato establecido con la naviera correspondiente.

De este modo, conocidos estos parámetros y utilizando la información relativa a las órdenes de trabajo heredada del programa Maximo, se pretende determinar las órdenes que cada embarcación podría llevar a cabo. Para esto, es necesario que las órdenes de trabajo estén correctamente clasificadas en función de su prioridad y tiempo disponible, para dar prioridad en la agrupación a las tareas más restrictivas.

Una vez las órdenes están clasificadas, el subsistema analiza los requisitos de la orden de trabajo en cuanto a personal necesario (número de personas que precisarían ir en la embarcación) y material (en términos de capacidad de espacio) para, comparando estos valores con los parámetros de cada una de las embarcaciones, determinar si la orden en cuestión puede incluirse en la agrupación de esa embarcación.

A partir del subsistema, se calcula determinados parámetros que se utilizarán como entrada para el sistema de decisión multiobjetivo, que será el encargado de decidir en función de los requisitos establecidos por la empresa cuál de las agrupaciones calculadas presenta mayores ventajas frente a las demás.

En la Fig. 11 podemos ver la estructura esquemática de este procedimiento, determinando 4 posibles embarcaciones. Sus capacidades totales están basadas en datos reales de embarcaciones empleadas para trabajos en alta mar, y se incluye también la posibilidad de emplear helicóptero para tareas que no requieran material.

En la imagen se presenta un caso en el que diversas órdenes de trabajo de distintas prioridades (ver código de color leyenda superior izquierda) han sido agrupadas, y se procede a analizar una nueva orden de trabajo. Como se puede ver, en este punto trabajamos con una embarcación de reducido tamaño, cuya ocupación ya está al 100% y no podría agruparla, al igual que el helicóptero, ya que la orden precisa un volumen de 70m^3 . Esta orden, en caso de cumplir los requisitos de las embarcaciones 2 y 3, se introduciría en la pila de órdenes agrupadas o, por el contrario, en la de órdenes que no pueden ejecutarse con esa agrupación.

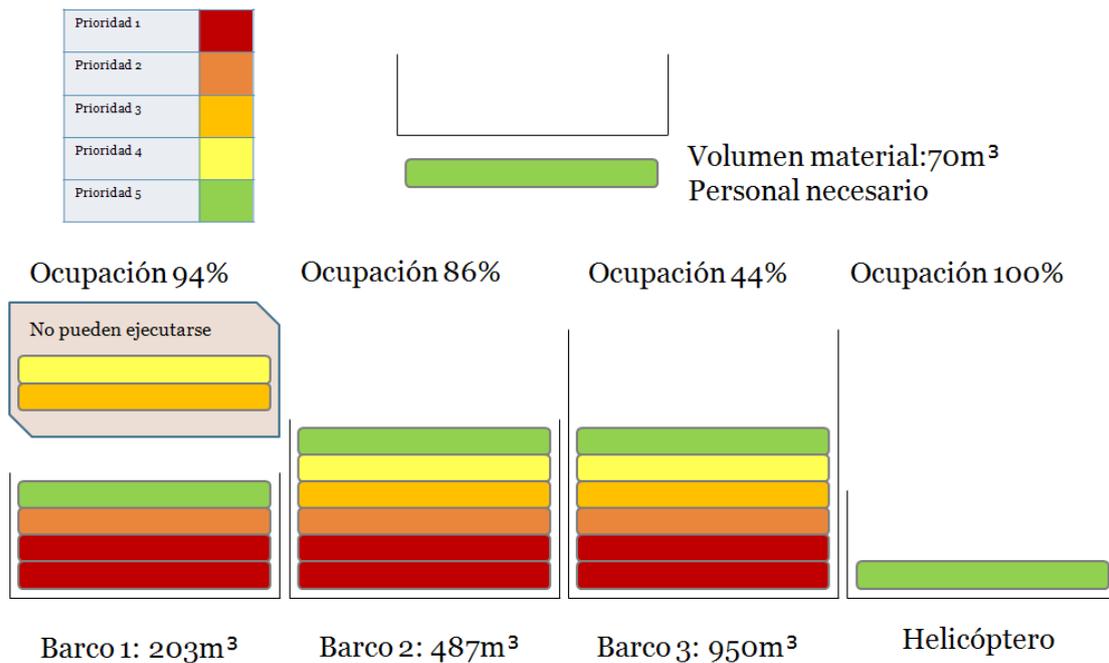


Figura 11: Sistema de agrupación de órdenes de trabajo

Como se puede ver, se trata de un procedimiento que se basa en los parámetros de las embarcaciones y de las órdenes susceptibles de agrupar para determinar si cada una de las opciones puede o no llevarla a cabo, y calculando en caso positivo el resto de parámetros que se utilizarán para decidir cuál de las opciones es más adecuada.

Este subsistema se utilizará también para proporcionar la información al experto operativo que tome la decisión de validar o cancelar los resultados de la decisión multiobjetivo, con el objetivo de proporcionar toda la información posible que pueda ayudar a tomar dicha decisión. De este modo, el experto puede ver las órdenes que han sido agrupadas y las que no, y las distintas opciones con las que ha trabajado el sistema.

4.3. Subsistema de decisión multiobjetivo de tareas de mantenimiento

Uno de los puntos clave a desarrollar dentro del sistema automático de gestión del mantenimiento es el subsistema de decisión multiobjetivo para la ejecución de órdenes de trabajo. Este subsistema trabaja conjuntamente con el anterior con el fin de proponer al experto de mantenimiento la mejor solución dentro de las posibilidades disponibles en cuanto

a tareas a ejecutar, tiempos disponibles para las reparaciones, embarcaciones, costes y demás factores utilizados a lo largo del sistema de gestión.

En este apartado se presentan las opciones analizadas para el desarrollo de este subsistema. Se han estudiado ampliamente las técnicas más empleadas dentro del campo de la decisión multiobjetivo, ya que se pretende que se analicen las agrupaciones en más de un aspecto, eliminando así la decisión básica en cuanto a costes, que puede resultar no ser la más adecuada para la empresa.

Los métodos que se proponen para este subsistema son los basados en restricciones, los basados en interfaz de preferencia y los basados en lógica borrosa y asignación de pesos.

Como hemos dicho anteriormente, se pretende analizar distintos factores para la toma de decisiones, mejorando así las opciones y logrando un resultado más preciso. Estos factores se han decidido en función al análisis de los requisitos del sistema de mantenimiento.

Al tratarse de metodologías genéricas de decisión multiobjetivo, proporcionan la posibilidad de ampliar los parámetros de análisis o cambiarlos sin afectar al funcionamiento del sistema.

Los parámetros que se han decidido utilizar para la toma de decisión del subsistema son los siguientes:

- Coste total de la ejecución del mantenimiento
- Porcentaje de ocupación de espacio de la embarcación
- Porcentaje de órdenes de prioridad 1 que se atienden con la agrupación
- Prioridad más restrictiva de las tareas que no pueden ser agrupadas

Estos parámetros se calculan para cada una de las agrupaciones disponibles, y se introducen como entrada para la metodología empleada para el subsistema de decisión multiobjetivo.

A continuación, se va a analizar detalladamente cada una de las metodologías mencionadas, consideradas las más adecuadas para el tipo de sistema en el que se quieren implementar. Finalmente, se presentarán las ventajas e inconvenientes de cada una de ellas, con el objetivo de justificar la metodología seleccionada para la implementación.

4.3.1. Método basado en restricciones

El primer método que se va a analizar en este apartado es el sistema de decisión basado en restricciones. Este método consiste en determinar un valor umbral para cada una de las variables mencionadas anteriormente, y dejar una de las variables, la considerada más importante, como parámetro de optimización.

La decisión de los valores de las restricciones es muy importante, ya que en ellas se basará el sistema de decisión multiobjetivo para seleccionar la opción más adecuada, y si los valores no están definidos correctamente se pueden tomar conclusiones totalmente equivocadas. Es por esto que se presenta la metodología y su funcionamiento, tomando unos valores estimativos, dejando la decisión del valor exacto de los rangos de las restricciones en manos de la empresa de mantenimiento.

Las restricciones consideradas para esta explicación son las siguientes:

- Ocupación de la embarcación superior al 50%
- Tareas Prioridad 1 atendidas superior al 80% de las tareas de prioridad 1 disponibles
- Resto de tareas sean de prioridad 3 o superior, o no queden tareas por agrupar

Basándonos en estas restricciones, queda como parámetro de optimización el coste total de la embarcación, incluyendo en este el coste del viaje y de todas las tareas agrupadas en cada opción.

El funcionamiento del sistema es el siguiente:

En primer lugar, se calculan los valores para los parámetros en cada una de las opciones resultantes de la agrupación, y se comprueba para cada una de ellas si se cumplen las restricciones establecidas por el sistema. Dentro de las opciones que cumplen las restricciones, se selecciona la opción de coste más reducido como solución óptima.

Para comprobar la metodología analizada, se desarrolló un simulador en la herramienta Matlab, cuyas bases se emplearon para la implementación posterior dentro del sistema de gestión de mantenimiento en el programa Labview. En la Fig. 12 podemos ver el simulador diseñado.

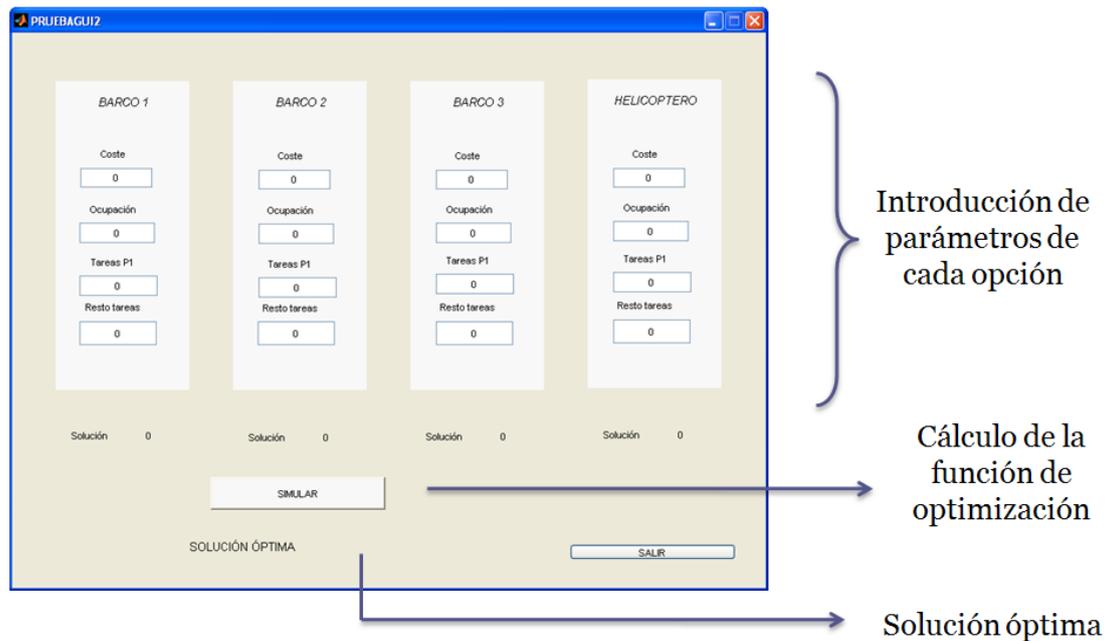


Figura 12: Interfaz Matlab para optimización

Como se puede ver, cada una de las posibles agrupaciones debe introducir en el sistema los valores calculados, y se simula cuál sería la opción más adecuada mediante la metodología basada en restricciones.

4.3.2. Método basado en interfaz de preferencias

El funcionamiento de esta metodología basada en interfaz de preferencias consiste en asignar un valor de etiqueta a cada uno de los parámetros analizados, y ordenas las agrupaciones en función de lo adecuadas que sean para el sistema.

Los parámetros que se emplearán para la optimización son los mismos que se han analizado en el apartado anterior: coste, ocupación, órdenes atendidas y prioridades.

En primer lugar, se deben definir los rangos de cada parámetro y las etiquetas que se asignarán a cada uno de ellos (Tabla 7).

Una vez más, estos valores son importantes debido a la criticidad del sistema y rangos o etiquetas inadecuadas darían lugar a un sistema con optimización incorrecta, por lo que se proponen unos valores estimados, totalmente configurables por la empresa y los expertos en

mantenimiento, sin que la modificación de estos límites afecte al correcto funcionamiento de la aplicación.

Tabla 7: Etiquetas y rangos para sistema de interfaz de preferencias

Objetivo	Rango		
	Deseable (D)	Tolerable(T)	Indeseable (I)
$J_1(\theta)$	$[a_{10}, a_{11}[$	$[a_{11}, a_{12}[$	$[a_{12}, a_{13}[$
$J_2(\theta)$	$[a_{20}, a_{21}[$	$[a_{21}, a_{22}[$	$[a_{22}, a_{23}[$
$J_3(\theta)$	$[a_{30}, a_{31}[$	$[a_{31}, a_{32}[$	$[a_{32}, a_{33}[$

Una vez analizados los parámetros para la aplicación deseada, se presentan los rangos estimados que se emplearán para la implementación del subsistema (Tabla 8).

Como se puede ver en la tabla, para este caso se han definido cinco etiquetas para los rangos de los parámetros. Las etiquetas hacen referencia a la adecuación del valor a las necesidades del parámetro en cuestión.

Tabla 8: Etiquetas y rangos para sistema de interfaz de preferencias

Función	MD	D	T	ND	I
Tareas P1	≥ 90	80~90	70~80	50~70	< 50
Ocupación	80~100	60~80	50~60	20~50	< 20
Coste	≥ 10000	7000~10000	3000~7000	1000~3000	< 1000
Resto tareas	0 ó 5	4	3	2	1

En este caso, las etiquetas se corresponden con los siguientes valores:

- Muy deseable
- Deseable
- Tolerable

- No Deseable
- Intolerable

Los valores asignados a los rangos establecidos son estimativos, y se pueden modificar fácilmente para adaptarse a los requisitos del sistema y las necesidades de la empresa contratante.

El procedimiento del subsistema es el siguiente:

Se analizan los valores calculados para los parámetros y se determina en cuál de los rangos se encuentran, asignándole la etiqueta correspondiente. Posteriormente se ordenan las opciones en función de las etiquetas, seleccionando así la más adecuada.

Esta organización se realiza partiendo de la base de que se pretende evitar los valores indeseables, y se trata de conseguir los más deseables.

Se presenta un ejemplo de decisión entre 3 casos, para analizar mejor la ordenación de las opciones y la decisión de solución óptima.

En la parte inferior de la tabla podemos ver el resumen de etiquetas de cada opción, para poder compararlas de forma más clara.

Tabla 9: Caso ejemplo con interfaz de preferencias

Función	Opción 1	Opción 2	Opción 3
Tareas P1	D	D	T
Ocupación	MD	D	T
Coste	I	T	MD
Resto tareas	MD	T	ND
	MD: 2 D: 1 T: 0 ND: 0 I: 1	MD: 0 D: 2 T: 2 ND: 0 I: 0	MD: 1 D: 0 T: 2 ND: 1 I: 0

Si se clasifican en función de su adecuación a los valores, en primer lugar se descartaría la opción 1, ya que tiene valor Indeseable y las demás no. Después, entre la opción 2 y 3, el sistema descartaría la 3, ya que tiene un valor no deseable, y la 2 no lo tiene. De este modo, ordenadas por etiquetas quedarían, de mejor a peor solución, la opción 2, después la 3 y por último la 1.

Para este caso, la implementación se realizaría mediante un interfaz muy similar al estudiado en el apartado anterior, en el que se introduce como entradas los valores de cada uno de los parámetros y automáticamente se calcula la etiqueta que se corresponde con cada uno de ellos. (Fig. 13)

Si se sigue el funcionamiento tal y como se ha explicado en este apartado, el sistema calcula la solución óptima entre las distintas posibilidades de agrupación de tareas de mantenimiento.

Con este ejemplo se pretende explicar que no se considera más adecuado seleccionar una opción que tiene muchos valores muy deseables, sino justo lo contrario, descartar las opciones cuyos valores no son convenientes y presentan desventaja para la empresa, tratando de elegir una opción que evite valores críticos muy negativos y se mantenga en término central.

OPCIÓN 1	OPCIÓN 2	OPCIÓN 3	OPCIÓN 4
BARCO 1	BARCO 2	BARCO 3	HELICOPTERO
Coste: 600	Coste: 1500	Coste: 2500	Coste: 4000
Ocupación: 100	Ocupación: 75	Ocupación: 50	Ocupación: 100
Tareas P1: 85	Tareas P1: 100	Tareas P1: 100	Tareas P1: 60
Resto tareas: 1	Resto tareas: 3	Resto tareas: 0	Resto tareas: 1
<i>Cálculo Rangos</i>			
Coste: MD	Coste: D	Coste: D	Coste: T
Ocupación: MD	Ocupación: D	Ocupación: T	Ocupación: MD
Tareas P1: D	Tareas P1: MD	Tareas P1: MD	Tareas P1: ND
Resto tareas: I	Resto tareas: T	Resto tareas: MD	Resto tareas: I
CALCULAR		SOLUCIÓN ÓPTIMA Opcion 3	
		SALIR	

Figura 13: Interfaz Matlab para optimización con interfaz de preferencias

4.3.3. Método basado en control inteligente (lógica fuzzy) y asignación de pesos

El tercer método analizado para la implementación de este subsistema consiste en la utilización conjunta de técnicas de lógica borrosa y una función de asignación de pesos para la valoración de cada uno de los parámetros empleados por el sistema de agrupación.

El funcionamiento consiste en asignar los rangos para cada parámetro, como se hacía en el apartado anterior, pero en lugar de clasificar con una etiqueta se asigna un valor numérico a partir de un sistema de lógica borrosa.

Posteriormente, los resultados de cada parámetro se combinan mediante una función matemática, que asigna a cada valor un peso para determinar su importancia frente al resto de parámetros.

Al igual que en las propuestas anteriores, todos los pesos y rangos de los parámetros pueden marcar la diferencia entre un sistema correcto o inapropiado, por lo que se utilizará un valor genérico que deberá ser ajustado por los expertos, pero la modificación de estos valores no afectará al funcionamiento del subsistema de decisión.

Tabla 10: Rangos de valores para optimización con metodología Fuzzy

Coste	Peso
0-10000€	0-10
Tareas P1	Peso
0-100%	0-10
Ocupación	Peso
0-100%	0-10
Resto tareas	Peso
Ninguna	10
P4,P5	8
P3,P4,P5	6
P2,P3,P4,P5	3
P1,P2,P3,P4,P5	1

La utilización de lógica fuzzy permite que los valores asignados a cada parámetro sean más precisos y se pueda tomar una decisión con factores más detallados, ya que dentro de un mismo rango puede haber muchos valores intermedios que con otras metodologías más genéricas pasarían desapercibidos.

Sin embargo, para el parámetro asociado a la prioridad del resto de tareas que quedan por resolver, no se consideraría adecuada esta metodología de lógica borrosa, ya que las prioridades son números enteros de uno a cinco y no se utilizan valores intermedios. Es por esto que se propone utilizar una asignación de pesos para este parámetro, y lógica fuzzy para el resto (Tabla 10).

La función de asignación de pesos distribuye un valor a cada parámetro. Para lograr un resultado significativo la suma de estos pesos debe resultar 1. Se presenta a continuación la función empleada para el subsistema:

$$\begin{aligned} \text{Solución óptima} = & 0'5x\text{RangoCoste} + 0'3x\text{TareasP1atendidas} \\ & + 0'1x\text{TareasPendientes} + 0'1x\text{OcupaciónBarco} \end{aligned}$$

Para la implementación del subsistema fuzzy se utiliza la herramienta “Fuzzy Toolbox” (Fig. 14) de Matlab. De este modo, se utiliza un sistema fuzzy de una entrada y una salida para cada uno de los parámetros (coste, tareas de prioridad 1 y ocupación del barco).

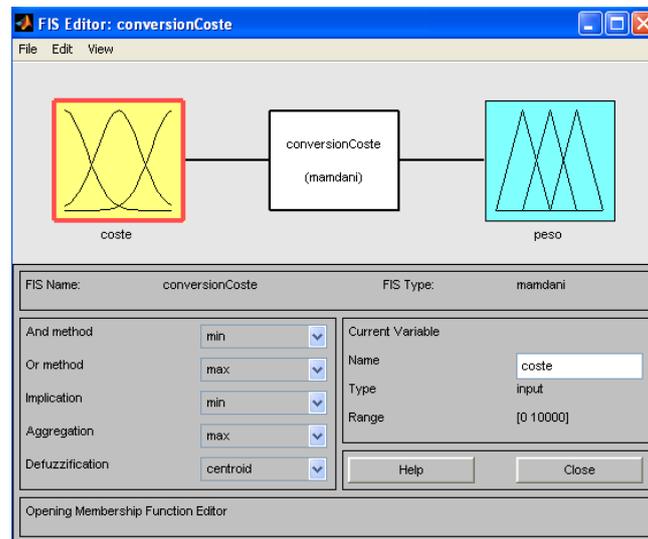


Figura 14: Interfaz “Fuzzy Toolbox”

Una vez definidas las entradas y salidas, esta herramienta permite utilizar el editor de reglas para determinar la relación que tienen entre ellas, y de este modo la forma de determinar el valor de salida en función del valor de la entrada.

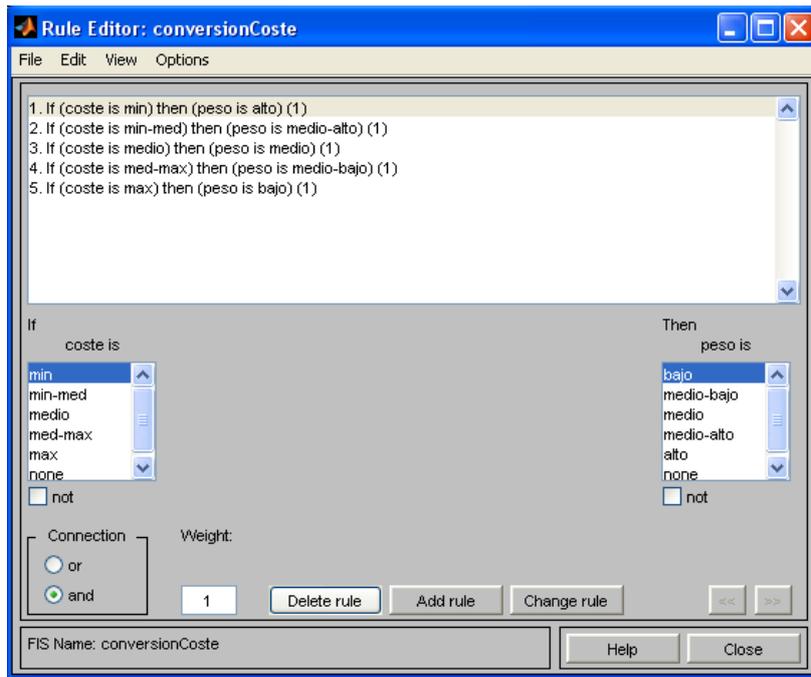


Figura 15: Editor reglas "Fuzzy Tollbox para parámetro Coste"

Posteriormente, se utiliza la herramienta Simulink para evaluar, según el valor del parámetro en cada instante, cuál sería la salida del sistema fuzzy y mediante la asignación de pesos dar un valor global a cada una de las posibles agrupaciones. Utilizando este valor se determina cuál de las opciones es la más óptima para el sistema.

El interfaz introduce los valores de cada embarcación en sus respectivas entradas, y a partir de la programación de cada barco, se calcula el valor de optimización de cada embarcación.

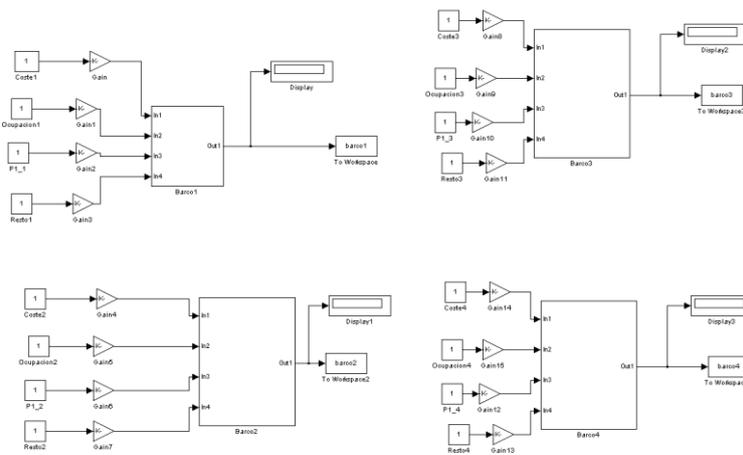


Figura 16: Esquema simulink para optimización

En la Fig. 16 se puede ver el esquema implementado en simulink para el subsistema de optimización mediante fuzzy y asignación de pesos de cada barco. La evaluación del resultado se realiza mediante Matlab, utilizando el interfaz implementado (Fig. 17).

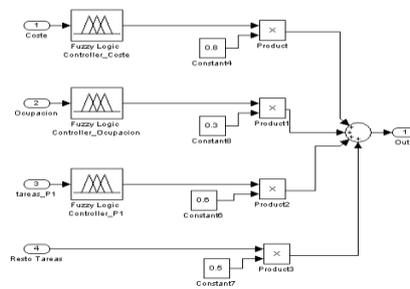


Figura 17: Sistema de optimización con metodología Fuzzy y asignación de pesos

4.3.4. Solución adoptada y parámetros de implementación

En los apartados anteriores se han podido analizar distintas metodologías propuestas para el sistema de decisión multiobjetivo. El funcionamiento de estas técnicas es muy distinto entre ellas, pero cumplen el objetivo común del subsistema: determinar entre las agrupaciones propuestas la más adecuada en función de diferentes parámetros.

En términos generales, se puede determinar que las dos primeras metodologías propuestas, basadas en restricciones e interfaz de preferencias respectivamente, son técnicas muy empleadas dentro del campo de la decisión multiobjetivo, mientras que la utilización de lógica borrosa y asignación de pesos surge del campo de la ingeniería.

Además, esta última metodología precisa conocimientos de lógica borrosa y programación en Matlab si se pretende hacer modificaciones sobre el sistema original. En cambio, las dos primeras opciones pueden ser fácilmente mejoradas o aumentadas sus prestaciones simplemente realizando cambios en los rangos de las variables, algo que puede realizar cualquier técnico sin conocimientos de programación, siendo los resultados igualmente correctos y favorables para el sistema de mantenimiento.

Por todo esto, se va a optar por la utilización conjunta de las dos primeras metodologías, dejando la lógica borrosa implementada de forma externa para su posible utilización si la empresa decidiera que fuera necesario.

El sistema se implementará de forma paralela con las técnicas de restricciones e interfaz de preferencias, evaluando por separado los resultados y en caso de obtener resultados distintos, determinando entre ellas la de mayor valoración.

En función de esta solución a adoptar, se implementará el sistema en la herramienta Labview. Aunque en principio se propuso el software Matlab para la implementación de los subsistemas, a lo largo del desarrollo de la tesina se ha optado por la utilización de Labview, ya que aunque se precisaba un desarrollo mayor, se lograba evitar la utilización de otra herramienta y se facilitaba así la comunicación entre programas y el funcionamiento general del sistema de mantenimiento.

5. ANÁLISIS COSTE RIESGO

A la hora de tomar decisiones dentro de un sistema de gestión integral del mantenimiento, existen gran cantidad de parámetros a tener en cuenta.

Una de las técnicas más empleadas y consultadas para tomar decisiones acerca de cuándo ejecutar una tarea de mantenimiento es el análisis coste riesgo. Este tipo de análisis son especialmente importantes en proyectos como el que se presenta, ya que tal y como se ha determinado previamente no se trabaja sólo con mantenimiento programado, sino que además se trabaja con futuros fallos y previsión temporal de cuándo ocurrirán.

Es por esto que la decisión de cuándo enviar la tarea de mantenimiento no se basa sólo en el calendario y los mantenimientos programados, sino que las técnicas que nos ocupan toman relevancia.

En este apartado se pretende analizar la metodología “Análisis Coste Riesgo” (ACR) y su aplicación al presente sistema.

Se realizará en primer lugar una descripción del funcionamiento de la metodología y los avances que supone su aplicación al sistema a diseñar.

Posteriormente, se analizarán los diversos software analizados para el desarrollo de la aplicación, analizando las ventajas e inconvenientes que cada uno de ellos supone.

Finalmente, se presenta la aplicación diseñada específicamente para la tesina, mostrando tanto su funcionamiento interno como el interfaz de usuario que ha sido creado.

5.1. Metodología.

Para tomar la decisión de permitir a una tarea de mantenimiento su agrupación o no con las demás tareas disponibles se deben considerar todos los factores asociados a la agrupación, analizados en apartados previos. Sin embargo, si se desea lograr una agrupación óptima de tareas, se debe tener en cuenta también el análisis coste-riesgo de la inclusión de dicha tarea en la agrupación.

Se presentan a continuación diversas metodologías para el desarrollo de técnicas de análisis coste-riesgo para el subsistema de agrupación.

A lo largo de los años, la gestión del mantenimiento se ha basado en disminuir frecuencias de fallos. Sin embargo, actualmente se utiliza en gran medida la técnica de Optimización Costo Riesgo (OCR), que explicaremos con más detalle en apartados posteriores.

Continuando con los análisis coste-riesgo más empleados, se puede concretar que los criterios más utilizados para seleccionar la frecuencia de aplicación de las distintas actividades o para tomar decisiones han sido los históricos de fallos y el tiempo promedio hasta el fallo (TPF). Se utilizan estos parámetros mediante una tabla comparativa de comportamiento del sistema estudiado, tratando de determinar el mejor momento para el mantenimiento.

En la Figura 18 podemos ver el registro de históricos empleado en este tipo de análisis.

Como podemos ver en la propia figura, a medida que se ha evolucionado en el sector, han surgido determinadas preguntas acerca de la idoneidad de esta técnica. Estas preguntas analizan si es suficiente con este tipo de análisis, si se deberían emplear más parámetros y si es la mejor opción disponible para optimizar las tareas de mantenimiento.

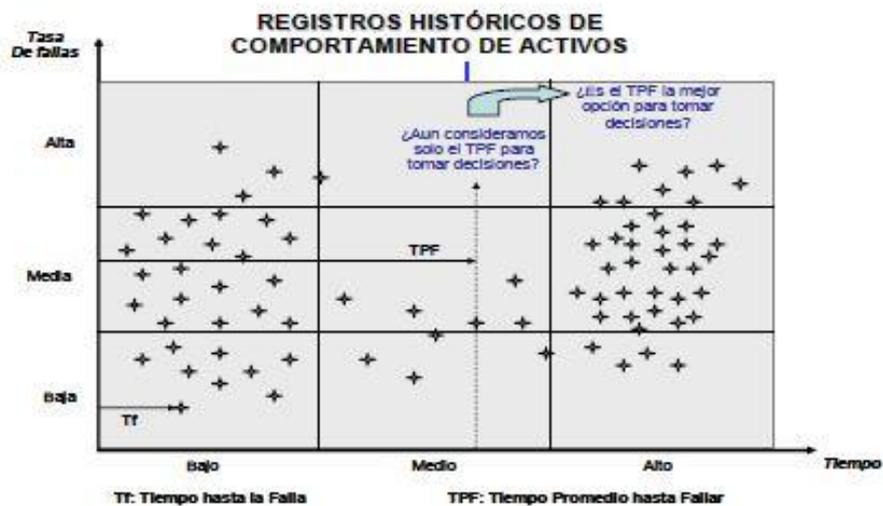


Figura 18: Análisis Coste Riesgo basado en TPF

Actualmente, las empresas no pueden basar sus decisiones únicamente en la frecuencia de la ocurrencia de los fallos, determinadas a partir de los históricos de fallos.

Es por esto que determinadas empresas han demostrado efectividad y beneficios dentro del mercado, utilizando como base para su metodología la siguiente premisa: “Preservar la función de los activos aplicando estrategias efectivas que consideren desde las fases iniciales la relación “Coste-Riesgo-Beneficio”, donde adicionalmente se están minimizando los riesgos asociados a las consecuencias que generan los distintos modos de fallas dentro del contexto operacional que se trabaja”.

De esta base surge la necesidad de mejorar las técnicas de análisis coste riesgo, dando lugar a la siguiente metodología planteada, la Optimización Coste-Riesgo (OCR).

5.2. Optimización Coste-Riesgo (OCR)

Por definición, la OCR es una metodología que permite lograr una combinación óptima entre los costos asociados al realizar una actividad, tomar decisiones y los logros (beneficios) esperados que dichos aspectos generan, considerando el riesgo que involucra la realización o no de tal actividad o inversión, incluyendo en la misma el hecho de disponer o no de los recursos para esta. Esta es una técnica o metodología de “Gerencia de Activos” que permite la toma de decisiones basada en el concepto de “Mínimo impacto total en el negocio”.

Un correcto sistema de OCR precisa un conocimiento exhaustivo de cada uno de los componentes y subsistemas, y de los parámetros que lo definen. Se necesita por lo tanto datos reales estadísticos, exactos y precisos, para lograr una representación objetiva del modelo a analizar.

Para analizar en mayor profundidad esta metodología, vamos a determinar su funcionamiento general, dejando para apartados posteriores un ejemplo práctico de aplicación.

La metodología se basa en determinar dos gráficas de tendencias a lo largo del tiempo de los siguientes parámetros:

- Costes de mantenimiento
- Riesgo de operación

Para la definición de estas curvas, como se ha analizado anteriormente, se precisan datos concretos de cada componente y subsistema, que sean capaces de representar el comportamiento de los mismos a lo largo del tiempo de una forma veraz.

Para este tipo de representaciones se emplean, generalmente, curvas de distribución exponencial. En caso de no disponer de suficientes datos, se pueden emplear distintos tipos de software de análisis que permiten utilizar la información disponible para estimar el comportamiento futuro de la distribución.



Figura 19: Optimización Coste Riesgo

En la figura 19 se puede ver la representación gráfica de la metodología que se presenta. La distribución roja corresponde al riesgo de no ejecución de la tarea de mantenimiento. El parámetro riesgo se define como la relación entre la criticidad del fallo y el valor de la consecuencia, y para su determinación se emplean los parámetros que se pueden ver en la figura:

- Costos por fallos e ineficiencia de equipos
- Reducción de vida útil
- Impacto en seguridad
- Impacto ambiental

Para la distribución del riesgo sería conveniente utilizar una distribución exponencial creciente, ya que el riesgo de fallo de cada componente aumenta con el paso del tiempo. La vida útil de cada componente, que determina el eje x de la tabla coste riesgo, debe ser obtenida a partir de los datos técnicos de la hoja de características correspondiente.

Para la distribución del coste de las tareas de mantenimiento, se emplea generalmente una distribución exponencial decreciente, ya que los costes asociados a una reparación prematura

son mayores que los generados por una intervención de mantenimiento a lo largo de la vida útil del componente.

A partir de las distribuciones de coste y riesgo se obtiene la curva de impacto total, mediante la agregación en cada instante de tiempo de los valores de las dos curvas.

Una vez se dispone de la curva de impacto total, se utiliza una función de optimización en busca del mínimo global de la distribución. Este mínimo determinará el momento óptimo para ejecutar la tarea de mantenimiento.

Este mínimo sería muy útil si se tratara sólo de mantenimiento programado, para determinar el momento de ejecución, pero si el objetivo del sistema es utilizar la agrupación de tareas para reducir costes, no se puede determinar un solo instante de acción. Es por esto que alrededor del valor determinado se asignan los intervalos de confianza, que aumentan el rango de posibilidades de ejecutar el mantenimiento.

Dentro del sistema de optimización diseñado para el proyecto se utiliza el parámetro Coste para evaluar la idoneidad de las distintas agrupaciones. Dentro de esta función de Coste se incluyen determinados conceptos, no sólo el coste del material y el personal, sino también los costes de enviar cada tipo de embarcación y por supuesto, las pérdidas por no producción.

Para completar este análisis de riesgos, sería recomendable incluir la optimización coste riesgo que se presenta dentro del funcionamiento general del sistema automático de mantenimiento y en el que el sistema de agrupación se basaría para determinar las opciones de asociación de tareas de mantenimiento. Para poder incluir este análisis dentro del sistema, se debería realizar el análisis de cada componente o subsistema y almacenarlo en bases de datos, permitiendo al sistema general consultar el valor mínimo y los intervalos inferior y superior y añadir este parámetro como una restricción más al sistema de agrupación.

Una vez conocido el funcionamiento general de la metodología de optimización coste-riesgo, se analizarán a continuación los sistemas software analizados para el desarrollo de esta técnica, tanto para su implementación como para un mejor tratamiento de los datos y las curvas características.

5.3. Análisis de Software para implementación

Durante el desarrollo del presente estudio de análisis coste riesgo se han analizado diferentes herramientas software, con el objetivo de mejorar las características de la metodología de optimización y facilitar la implementación de estas técnicas y su relación con el sistema de mantenimiento.

Existen gran cantidad de programas empleados por las empresas para la planificación del mantenimiento. Todos ellos están basados en tratamiento estadístico de datos y funciones de distribución de diversas características.

Se va a analizar este apartado desde dos puntos de vista. Por un lado, software recomendado para la implementación de la herramienta de optimización coste riesgo y por otro, programas que pueden mejorar las características de las distribuciones y que proporcionan opciones de análisis, toma de decisiones y simulaciones, como por ejemplo la simulación de Monte Carlo, ampliamente utilizada en el campo del mantenimiento.

En cuanto a las herramientas disponibles para mejorar las funciones de distribución empleadas, se presentarán a continuación los programas @Risk y Crystall_Ball, dos herramientas muy potentes para este tipo de estudios y que además presentan la ventaja de ser complementos para Excel, manteniendo así el número mínimo de programas a emplear.

Se analizará también la gran complejidad de estos sistemas y el hecho de que están diseñados para gran número de estudios y simulaciones que no resultan de utilidad para el presente proyecto.

En cuanto a la implementación del sistema de optimización coste riesgo, la herramienta más adecuada sería el programa Excel.

Esto es debido tanto a las amplias posibilidades que ofrece Excel en cuanto a análisis matemático y distribuciones, como a la posibilidad de trabajar con un software ya empleado para otros subsistemas del proceso, lo que supone no añadir nuevos programas y facilitar de este modo las comunicaciones entre subsistemas.

Además, se presentará la herramienta desarrollada para el análisis Coste Riesgo en Excel, como muestra de la facilidad de uso y fácil comprensión del funcionamiento.

5.4. Software @Risk y Crystall_Ball

En este apartado se analizarán los software @Risk y Crystall_Ball, que como se ha determinado son muy útiles para un tratamiento más avanzado de los datos y los tipos de distribuciones y para realizar estudios comparativos, como la simulación Monte Carlo, ampliamente utilizada en sistemas de Mantenimiento.

El objetivo del análisis de estas herramientas es otorgar una visión general de las posibilidades comerciales y los programas que pueden resultar de utilidad para mejorar el sistema.

Ambos software son herramientas muy potentes y completas, que precisan un amplio conocimiento en este tipo de análisis y simulaciones. De lo contrario, no se podría trabajar con los datos de forma adecuada. Los programas no son de funcionamiento amigable y es de gran complejidad interpretar los resultados obtenidos.

La mayor ventaja que presentan estos dos programas, entre todos los posibles programas comerciales existentes, consiste en que ambos funcionan como complementos a Excel, y permiten de este modo trabajar todo el sistema sobre la misma herramienta y no añadir más programas a la lista de software necesario, y facilita también la comunicación entre sistemas y el tratamiento de datos.

Por otro lado, aparte de la dificultad de uso de los programas, hay que considerar que se trata de software de pago, por lo que sería necesario analizar en profundidad las dos herramientas antes de determinar la inversión que supone su compra. Para poder analizar mejor las características, se ha trabajado con las versiones de prueba que están disponibles en las respectivas páginas web de los productos.

En primer lugar, vamos a repasar las características del programa @Risk. Una vez instalado el software en el equipo, al abrirlo se abre también Excel, y @Risk aparece como una ventana en el menú de Opciones.

Para realizar este análisis, se trabajó con los programas de prueba que proporciona @Risk. Se dispone de casi 50 programas de prueba en diversos temas, como por ejemplo:

- Calidad
- Estimación de Costos
- Finanzas
- Análisis de estrés

- Búsqueda objetivo
- Análisis PERT
- Teorema límite central
- Six Sigma

Como podemos ver, disponer de esta herramienta ampliaría el horizonte de posibilidades para la optimización, no restringiendo el uso a optimización coste riesgo, y en caso de seguir utilizando esta técnica, con los datos adecuados y sabiendo cómo utilizar las distribuciones, se obtendrían análisis muy específicos.

El menú de trabajo con este programa está dividido en tres submenús: Modelo, Simulación y Resultados.



Figura 20: Componentes del submenú "Modelo" de la herramienta @Risk

En la figura 20 podemos ver el submenú Modelo. En este apartado se realiza la introducción de datos y definición de distribuciones, que pueden ser de los tipos que vemos en la figura 21.

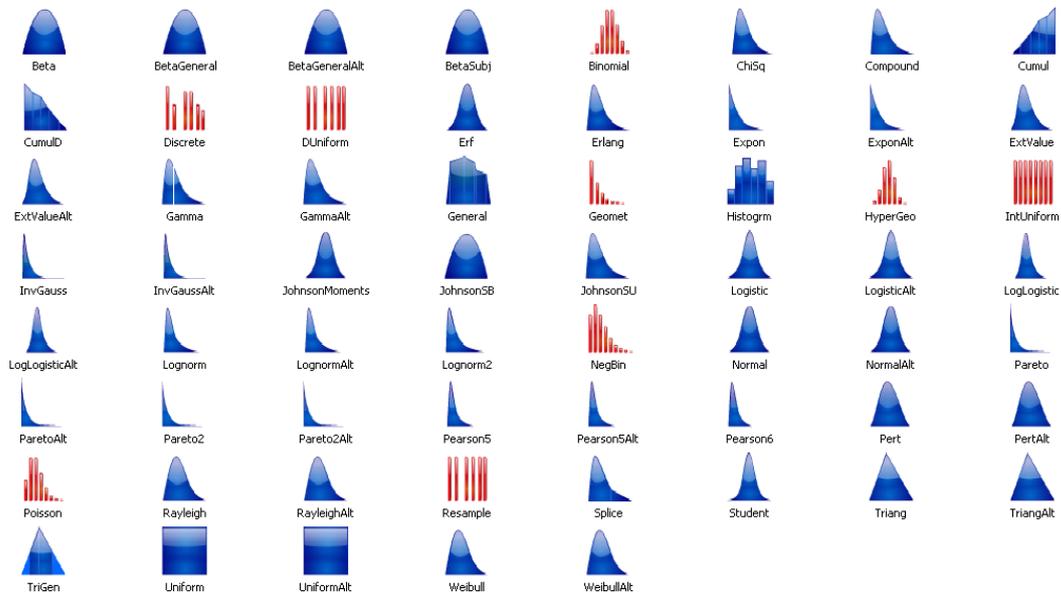


Figura 21: Tipos de distribuciones disponibles en la herramienta @Risk

También se pueden añadir distintos tipos de funciones para realizar las distribuciones, en la figura 22 vemos algunas de ellas, las más utilizadas. Al introducir estas funciones, se requerirá

determinar los valores que definen su estructura, que dependerán en cada caso del tipo de función utilizada.

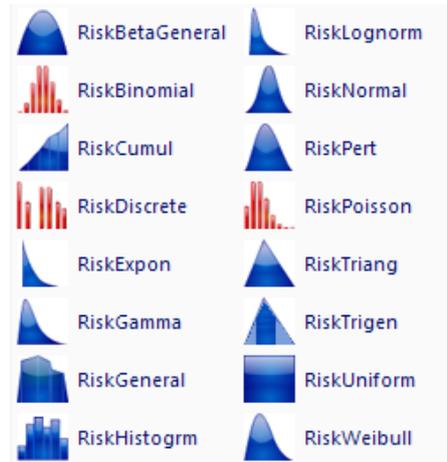


Figura 22: Funciones específicas de distribuciones

También encontramos en este submenú la posibilidad de ajustar una distribución a datos establecidos, y el artista de distribución, a partir del cual se puede realizar una distribución y ajustar sus parámetros de forma gráfica, y determinar en qué medida se ajustan los distintos tipos de distribuciones a la gráfica diseñada.

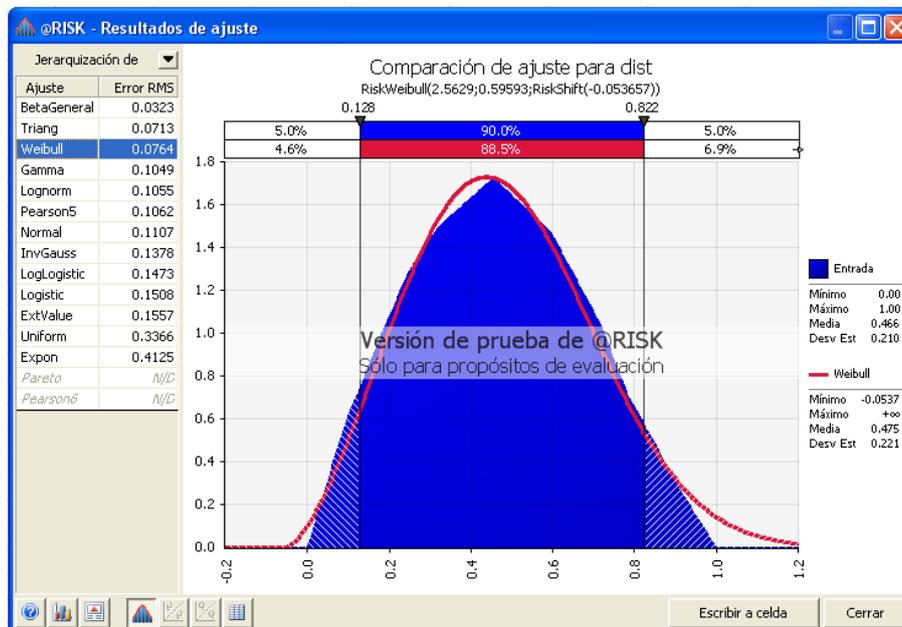


Figura 23: Ejemplo de ajuste de distribución

Por último, los submenús de Simulación y resultados nos permiten, una vez se dispone de la distribución en la celda deseada, realizar una simulación del comportamiento de la distribución y ver los resultados.

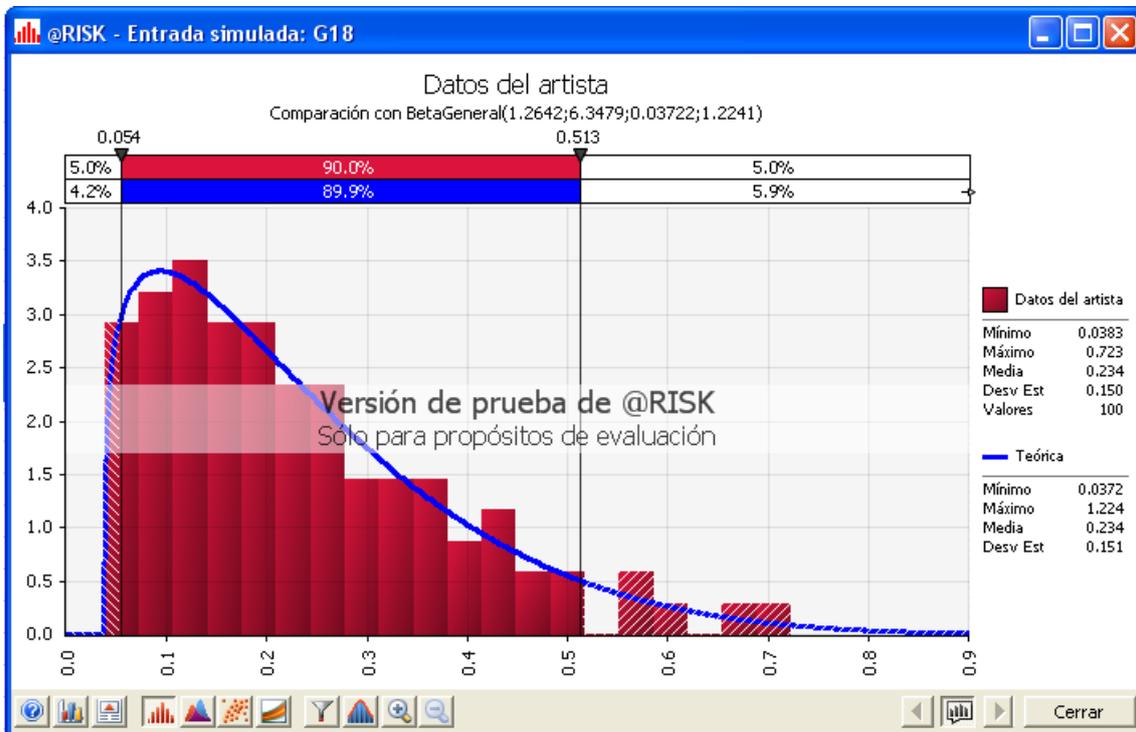


Figura 24: Ejemplo resultados de simulación

Como se ha podido ver, es un software muy completo, pero también con unas aplicaciones muy estadísticas, que precisan amplios conocimientos en estas técnicas para poder utilizar todas las posibilidades de la herramienta.

Veamos a continuación el programa Crystall_Ball, también complemento para Excel, por lo que se procede de igual manera a su instalación y aparece en el menú opciones.

Como se puede ver en la figura 25, la distribución de submenús es muy similar a @Risk, ya que se trabaja con una parte de definición de parámetros, otra de simulación y una última de análisis de resultados.



Figura 25: Componentes de la herramienta Crystall_Ball

Las opciones de distribuciones son más reducidas en este software, debido a que este software es más específico para toma de decisiones y para estimación de variables.

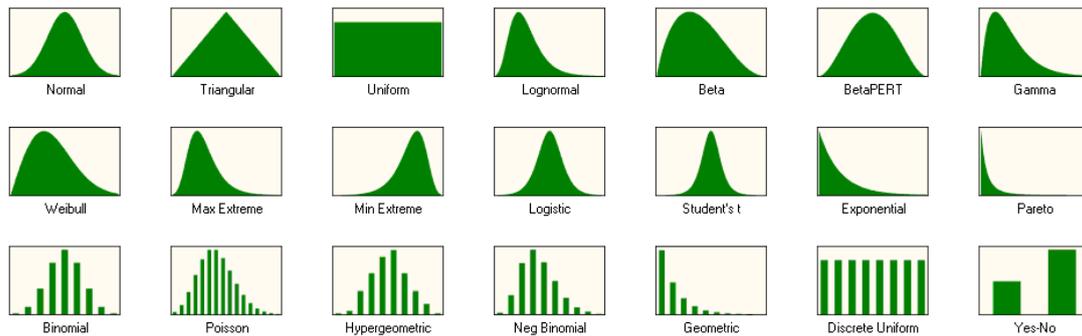


Figura 26: Tipos de distribuciones de la herramienta Crystall_Ball

Las opciones de herramientas de simulación son más amplias, permitiendo decidir el tipo de simulación. Esta es una de las diferencias entre los software, @Risk trabaja ampliamente con la estructura de los datos, pero deja menor libertad de simulación, mientras que Crystall_Ball no emplea tantos tipos de distribuciones para ajustar las variables, pero realiza distintas simulaciones.



Figura 27: Tipos de simulaciones disponibles

En la figura 28 podemos ver un programa ejemplo de estimación de costos, partiendo de la introducción de datos e intervalos de confianza y viendo los resultados de la simulación de probabilidad y frecuencia.

Tal y como se ha podido analizar en este apartado, se han estudiado estas dos opciones de software por las ventajas que implica que se trate de complementos de Excel y las grandes posibilidades en cuanto a estudios estadísticos y simulaciones. Sin embargo, se ha

determinado también que su uso es complejo y requiere amplios conocimientos de las técnicas a emplear.

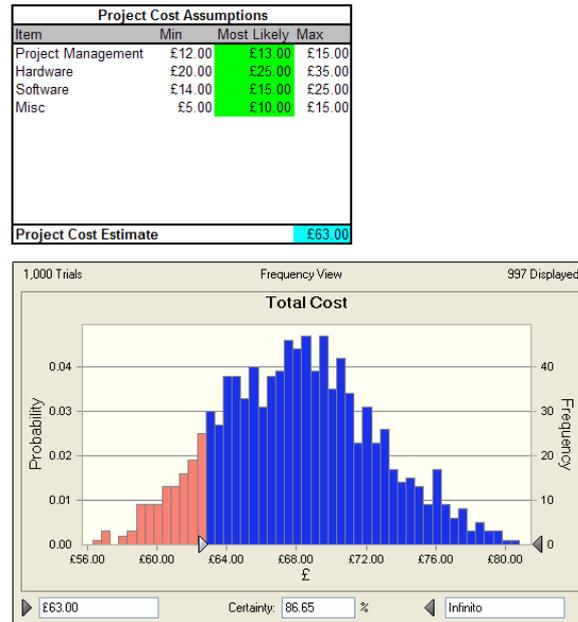


Figura 28: Ejemplo estimación de Costes

La mayor desventaja de estas herramientas se basa en que son programas sobrecargados de metodologías diversas, que no son de utilidad para el proyecto, y en caso de optar por el uso de alguna de estas herramientas se dispondría de un programa muy completo de opciones innecesarias. Teniendo en cuenta el tipo de análisis empleado para la optimización coste riesgo y las características de las distribuciones, estos softwares están sobredimensionados para el sistema a diseñar, por lo que se propone una herramienta específica para la aplicación.

5.5. Implementación sistema de optimización coste riesgo en Excel.

Tal y como se ha determinado anteriormente, la solución más apropiada para la implementación de un sistema de optimización Coste Riesgo sería la utilización de Excel. Son conocidas las altas capacidades de Excel en cuanto al trabajo con fórmulas, funciones, gráficas y bases de datos.

Este programa ya se emplea para diversos subsistemas del programa de gestión de mantenimiento desarrollado para el sistema, y se han creado librerías específicas para su comunicación con Labview, en caso de realizarse una implementación online, aunque idealmente se piensa en realizar el análisis de forma externa. La implementación del software de optimización coste riesgo se ha realizado mediante el uso de macros en Visual Basic, haciendo posible de este modo la automatización del proceso.

Para facilitar el uso de la herramienta, se ha desarrollado un interfaz de usuario en el que se introducirán los datos y será el encargado de trabajar sobre Excel realizando los cálculos necesarios, para finalmente mostrar al operador los resultados obtenidos. De este modo, se consigue que el usuario introduzca los datos y obtenga resultados sin tener que aprender a desarrollar la metodología y sin tener conocimientos del programa Excel, tanto a nivel usuario como programador.

A continuación, se va a presentar la herramienta desarrollada en Visual Basic, analizando su funcionamiento a nivel usuario, y realizando un ejemplo a modo de demostración.

El interfaz principal recibe el nombre de interfaz "OPTIMIZACIÓN COSTE RIESGO" (figura 29).

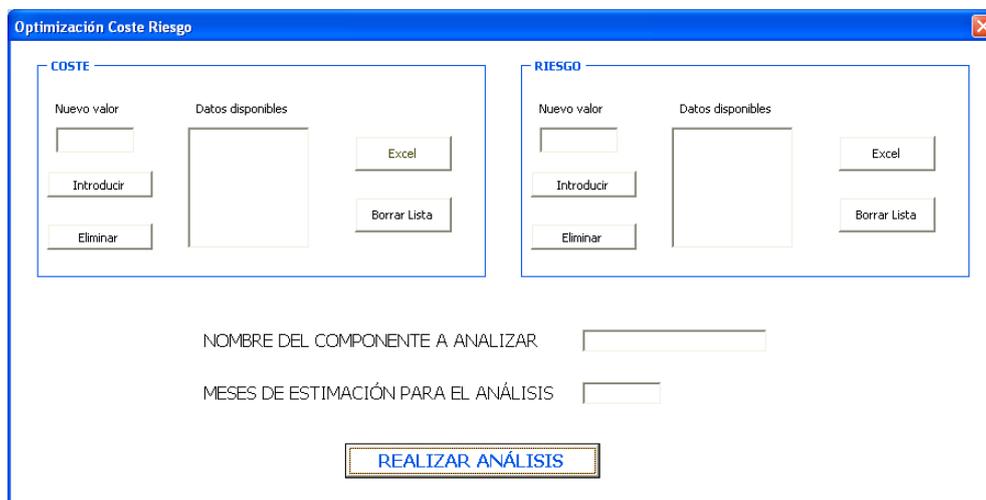


Figura 29: Interfaz "Optimización Coste Riesgo"

Como podemos ver en la figura 29, es en el interfaz principal donde se realiza la introducción de datos por parte del usuario. Podemos ver dos zonas separadas, la parte izquierda se corresponde con los datos de Costes y la parte derecha con los de Riesgos.

En cada bloque de datos, se dispone de una lista en la que se pueden ver los valores añadidos a la curva a utilizar. Estos valores se añaden desde el cuadro de datos de la esquina superior

izquierda de cada bloque, y pueden ser eliminados, bien uno valor independiente mediante el botón “eliminar” o bien todos los valores mediante el botón “Borrar lista”. El botón “Excel” se utiliza para introducir los datos en la plantilla de Excel, una vez se dispone de la información en la lista. En caso de no introducir las variables, la simulación se realizará con los valores previamente almacenados.

El sistema está preparado para que el número de datos a introducir no sea fijo, pensando en que la información disponible no es la misma en todos los casos. Por supuesto, a mayor información mejores resultados se obtendrán, y la optimización será más precisa.

En la parte inferior, se introduce el nombre del componente sobre el que se realiza la optimización, y los meses de estimación que se pretenden analizar.

Por último, se dispone del botón “Realizar análisis”. Al pulsar sobre él, se inicializa el sistema y la plantilla de Excel, a través de la macro implementada y los valores introducidos con el interfaz, comienza el análisis Coste-Riesgo, de acuerdo a la metodología de optimización que se ha presentado en el apartado anterior.

Una vez se han finalizado los cálculos internos del programa, se abre en una ventana nueva el interfaz “RESULTADOS” (figura 30).

En este interfaz se mostrará al usuario la gráfica resultado del análisis Coste Riesgo, donde se representa la estimación de las distribuciones de Coste y Riesgo, y la función de Optimización resultado del análisis.

En la parte inferior, se representa el valor numérico del mes más apropiado para la ejecución del mantenimiento, y se dispone también de dos cuadros de texto, donde el operario deberá introducir los intervalos inferior y superior, necesarios para el sistema de agrupación de tareas de mantenimiento.

El botón “Guardar Resultados” introduce en la plantilla Excel “BD Resultados”, al hacer click sobre él, el valor de punto óptimo, el nombre del componente, y los intervalos determinados por el usuario acorde a los resultados de la gráfica.

En la figura 30 podemos ver una simulación realizada para 10 meses, a partir de 5 datos.

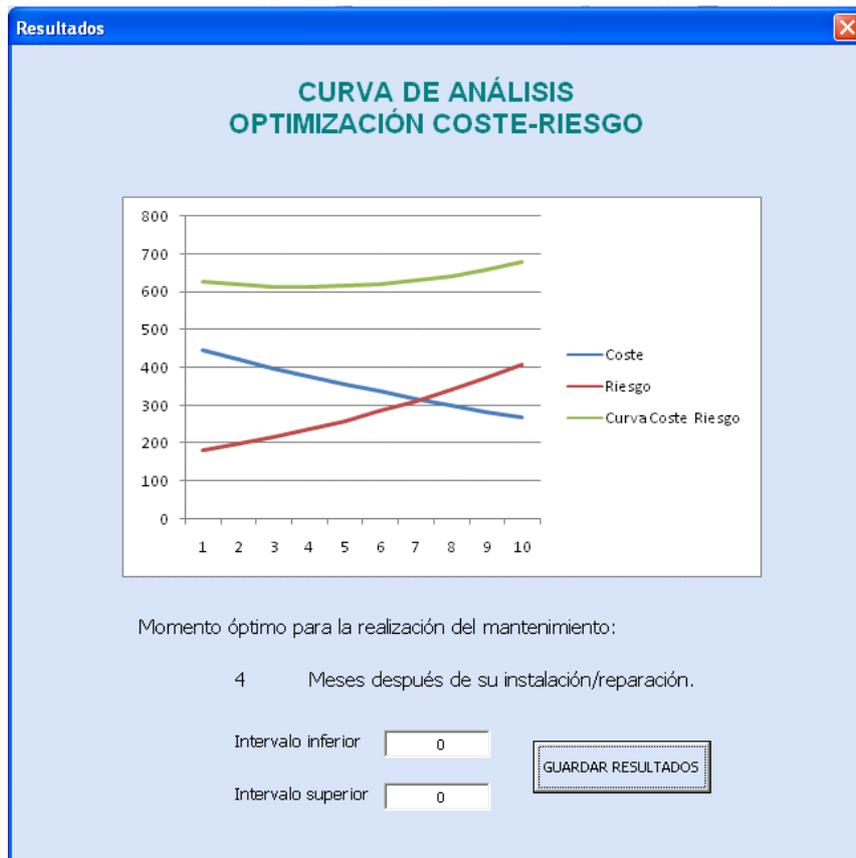


Figura 30: Interfaz "Resultados"

Se va a explicar a continuación, a grandes rasgos, el funcionamiento interno de la macro implementada para la herramienta de optimización.

En primer lugar, se introducen los datos del interfaz en la plantilla de cálculo "optimización", y se habilita ésta para su uso automático desde VisualBasic.

A continuación, con los datos introducidos se realiza la representación gráfica, a partir de la cual se introduce la estimación de línea de tendencia y se obtiene la fórmula que representa la evolución temporal de los valores mediante una distribución exponencial, del tipo:

$$y = Ae^{Bx}$$

Donde y es el parámetro calculado, y x es el valor temporal. Los valores A y B determinan la estructura de la distribución.

Una vez calculada la línea de tendencia, se extraen los valores A y B, para permitir que Excel cree la estimación de todos los meses determinados de forma automática. Este procedimiento se realiza paralelamente para los valores de Coste y Riesgo.

A partir de la estimación de valores mediante la distribución obtenida, se calcula la curva Optimización Coste Riesgo, y se introducen en una misma gráfica las tres distribuciones, para su posterior representación en el interfaz de Resultados.

Conocida la curva de optimización, la macro utiliza una función de búsqueda de mínimos y determina el valor mínimo global de la distribución, y el mes correspondiente.

Finalmente, se muestran los resultados en el interfaz correspondiente, permitiendo al usuario determinar los rangos de los intervalos y guardar los resultados, tal y como se ha explicado anteriormente.

5.6. Conclusiones Optimización Coste Riesgo

A lo largo del presente apartado se ha tratado de analizar la metodología empleada para los sistemas de optimización coste riesgo, estudiando en primer lugar el funcionamiento de este tipo de sistemas y a continuación las posibilidades de implementación.

Entre estas posibilidades, se han presentado dos software comerciales de nivel muy avanzado, que precisamente por los conocimientos que requiere y la gran cantidad de herramientas innecesarias que presentan, no se consideran apropiados para la implementación.

Por otro lado, se ha presentado un software de análisis coste riesgo desarrollado específicamente para la tesina, que si bien no dispone de tantos niveles de análisis y simulaciones, sí permite un funcionamiento más sencillo y comprensible.

El análisis, como ya se ha dicho, no está incluido dentro de la estructura del sistema general de mantenimiento, ya que se precisa un estudio individual de cada componente basado en parámetros técnicos y costes, que se realizaría fuera del funcionamiento del sistema pero cuyos resultados se utilizarían como parámetro para el subsistema de agrupación de tareas.

El sistema implementado está preparado especialmente para facilitar su inclusión en el sistema, una vez realizados los análisis necesarios, ya que los datos de punto óptimo e intervalos inferior y superior admisibles se almacenan en una base de datos. Su inclusión en el sistema general se realizaría a partir del sistema de agrupación de tareas de mantenimiento, incluyendo como restricción que se consultara si el valor temporal se encuentra entre los intervalos determinados al realizar el estudio.

6. SISTEMA GENERAL DE MANTENIMIENTO

A lo largo de este apartado se va a presentar la implementación de la Red de Petri Coloreada Adaptativa (en adelante, RdPCA) obtenida a partir de los estudios realizados para la tesina.

En apartados anteriores, se determinó una estructura para la implementación de Redes de Petri Coloreadas (RdPC) en Labview, que requiere para su implementación el desarrollo de nuevas librerías, tanto para la propia RdPC como para la comunicación entre Excel y Labview.

Por otro lado, se presentará en este apartado del informe toda la programación requerida en Labview para la implementación de la RdPCA en Labview, así como el funcionamiento y activación de todos sus estados y transiciones. Se va presentar por lo tanto todo el funcionamiento de la red, además de la programación y modo de uso de los subsistemas implementados.

6.1. Estructura RdPCA

En el apartado 3 se analizó la estructura de programación requerida para la implementación de la red de Petri Coloreada en Labview, a partir del estudio de las características del sistema de gestión de mantenimiento que se pretende implementar para trabajar conjuntamente con el software Maximo, se determinó que la red diseñada debía trabajar de forma totalmente automática, a partir de la información obtenida de Maximo y proporcionando en su fase final un apoyo al mencionado programa.

Se planteó también el problema que supone para la implementación de la herramienta no conocer el número de órdenes de trabajo operativas en cada momento, lo que supone una necesidad de creación de estructura variable, y un sistema adaptativo a estos cambios, ya que continuamente se generarán órdenes nuevas y se eliminarán del sistema las ya finalizadas.

Como conclusión, se determinó la creación de una nueva metodología de implementación de RdPC, capaz de añadir y borrar nuevas estructuras a su configuración inicial, impidiendo que estos cambios afectaran al funcionamiento del sistema.

Es por esto que los requisitos para la nueva estructura eran los siguientes:

- Red adaptativa a cambios de estructura.
- Todos los parámetros son genéricos, no dependen del tamaño de la red.

El procedimiento de funcionamiento de la red consiste en definir una estructura inicial que servirá de base para cada orden de trabajo. De este modo, cada nueva orden añade al sistema una copia de la estructura inicial, utilizando como marca de color para la subred el código identificativo de la orden de trabajo introducido.

De este modo, se trabaja con diversas redes de Petri duplicadas pero independientes entre sí. Esto supone una gran ventaja frente a métodos tradicionales de redes de Petri: se pueden generar tantas órdenes de mantenimiento como sean necesarias, ya que todo el sistema es automático y funciona en función del identificador de la orden.

Para definir los diferentes estados de la RdPC se utiliza la estructura de la tabla 11, definiendo estos parámetros para cada una de las transiciones asociadas. En ella, podemos diferenciar origen, número de transición, marcado inicial, destino, y marcado final. Origen y destino hacen referencia al lugar de la RdPC del que se extrae y deposita la marca al activarse la transición, que viene determinado por el número de transición. Mediante el marcado inicial y final determinamos la marca de color que debe haber en el lugar de origen para que se produzca el disparo, y el color de la marca en el lugar de destino.

Tabla 11: Definición original RdPC en Labview

Origen	Número transición	Marcado inicial	Destino	Marcado final
--------	-------------------	-----------------	---------	---------------

Para un correcto funcionamiento de la red, la implementación en Labview precisa que se cumplan los siguientes requisitos:

- Estructura de la red acorde con definición original (Tabla 11)
- Vector de Entradas y Vector de transiciones correctamente definidos
- Vector de activación de transiciones

Para facilitar la adaptabilidad de la red, tanto la estructura de la red como los vectores de entradas y transiciones se han implementado en Excel, ya que es una forma de mantener la

información y trabajar dinámicamente sobre ella, sin riesgos de pérdida de datos y permitiendo acceder a ella de forma externa al programa, en caso de necesitar realizar alguna consulta.

A la definición original se le anexará la definición de vectores de entradas y transiciones. Por otro lado, en una plantilla distinta, se dispondrá de un vector de booleanos de transiciones, que se utilizará para trabajar sobre él cuando se realice el disparo de las transiciones.

Se ha implementado también en Excel una plantilla con la información relativa a la orden, con los parámetros necesarios para un buen funcionamiento de la red, y que se analizará en el apartado 6.5.2.

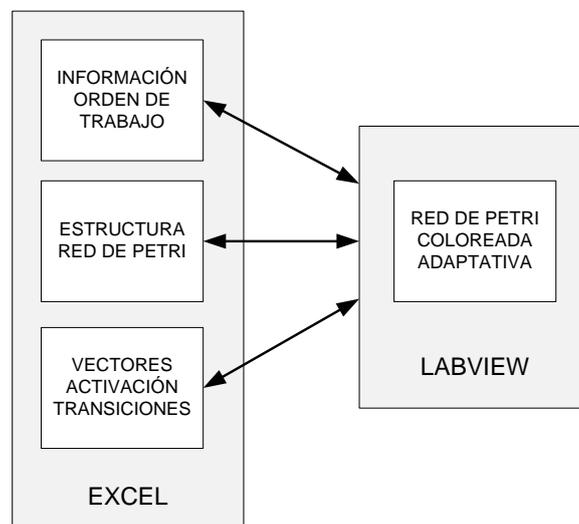


Figura 31: Esquema distribución por programas

Las librerías necesarias para la implementación de esta comunicación entre Excel y Labview se explicarán en detalle en los apartados posteriores.

La Red de Petri Coloreada determinada como estructura básica de la red es la que se puede observar en la figura 8 (apartado 3.2).

De acuerdo a esta Red de Petri Coloreada, y mediante la definición original de la estructura para la implementación, se ha definido la estructura básica para la Red de Petri Coloreada Adaptativa.

En la Tabla 12 vemos esta estructura para una orden de trabajo con indicador OT1, color de la marca de la red.

Tabla 12: Estructura básica de RdPC en Excel

	ORIGEN	Nº TRANS	DESTINO	MARCADO INICIAL	MARCADO FINAL	TRANSICIÓN	ENTRADAS
OT1	0	0	1		OT1	NuevaOT1	NuevaOT1
	0	1	4		OT1	NuevaOT1	NuevaOT1
	1	2	2	OT1	OT1	EquipoOT1	EquipoOT1
	2	3	3	OT1	OT1	prio1OT1	prio1OT1
	3	4	2	OT1	OT1	SolicitudEOT1	SolicitudEOT1
	2	5	7	OT1	OT1a	EquipoDispOT1	EquipoDispOT1
	4	6	5	OT1	OT1	MaterialOT1	MaterialOT1
	5	7	6	OT1	OT1	prio1OT1	prio1OT1
	6	8	5	OT1	OT1	SolicitudMOT1	SolicitudMOT1
	5	9	7	OT1	OT1b	MatDispOT1	MatDispOT1
	1	10	7	OT1	OT1a	NNEOT1	NNEOT1
	4	11	7	OT1	OT1b	NNMOT1	NNMOT1
	7	12	8	OT1a	OT1	MeteoOT1	MeteoOT1
	7	12	8	OT1b		MeteoOT1	MeteoOT1
	8	13	9	OT1	OT1	OpcionesOT1	OpcionesOT1
	9	14	10	OT1	OT1	OptimoOT1	OptimoOT1
	10	15	11	OT1	OT1	MeteoOT1	MeteoOT1
	10	16	8	OT1	OT1	CambioPrioOT1	CambioPrioOT1
	11	17	8	OT1	OT1	NegativaOT1	NegativaOT1
11	18	12	OT1	OT1	ValidaOT1	ValidaOT1	
12	19	12	OT1		BorradoOT1	BorradoOT1	

Viendo esta distribución, se aprecia que el orden de las columnas difiere de la definición original (Tabla 11).

Este cambio de orden se realiza en Excel para facilitar la posterior adquisición de datos del mismo formato, si bien el subprograma de adquisición reorganiza las columnas para seguir la distribución original.

Una vez conocida la estructura de la Red de Petri a implementar, se analizarán en los posteriores apartados las librerías necesarias para su desarrollo, y para la comunicación entre los dos programas, necesaria para el funcionamiento de la red global y también para la consulta de los parámetros de cada orden de trabajo necesarios en determinados subsistemas.

6.2. Librería RdPCA

Como ya se ha determinado a lo largo de la tesina, la herramienta seleccionada para la implementación y desarrollo de la Red de Petri Coloreada Adaptativa es el programa Labview.

Sin embargo, no se disponía de una herramienta específica para la implementación de esta técnica en el programa Labview, por lo que ha sido necesario desarrollar una librería con determinados procedimientos.

Los componentes que forman esta librería son los siguientes:



Figura 32: Componentes librería RdPCA



Este icono hace referencia a un programa de conversión entre los formatos “matriz de marcado” y “string”. Es un bloque muy utilizado con RdP para poder trabajar con los datos como matriz y también como strings, para el marcado en los lugares de la RdP. La programación interna de este bloque es la siguiente:

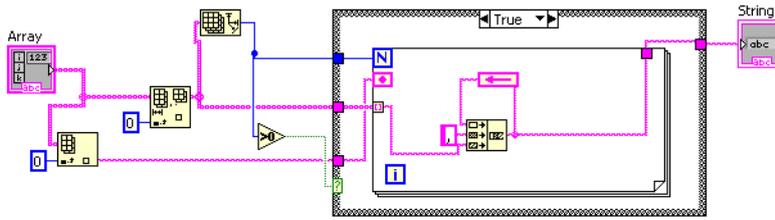


Figura 33: Programación bloque AM2STR.vi

Str2AM.vi



Este subprograma realiza el proceso contrario al anterior, convirtiendo los datos de formato “string” a formato “matriz de marcado”.

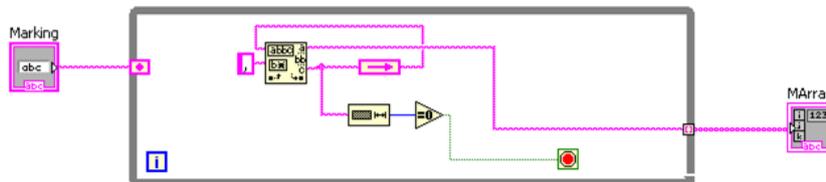


Figura 34: Programación bloque STR2Am.vi

input2transitions.vi



Este subprograma se encarga de construir el vector booleano de disparo de transiciones a partir de los vectores de entradas y transiciones y el vector booleano de valor de las entradas. Utiliza como entradas los vectores de entradas y transiciones y el vector booleano de entradas.

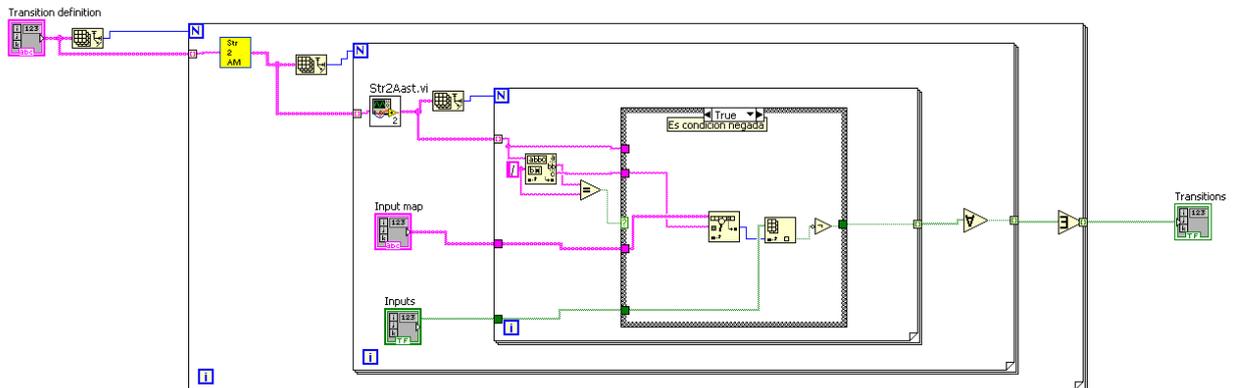


Figura 35: Programación bloque input2transitions.vi

Dentro de este subprograma, vemos el bloque “Str2Aast.vi”.



Este bloque se encarga de convertir el marcado en formato “string” a una matriz de marcado.

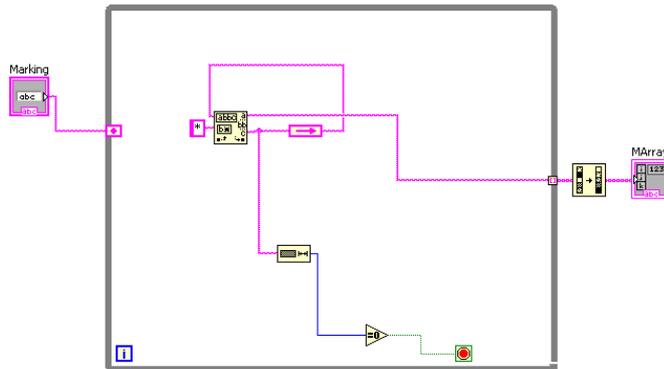


Figura 36: Programación bloque STR2Aast.vi



Este subprograma es el encargado de realizar la actualización del marcado de los lugares de la RdP, a partir del marcado anterior y el disparo producido en las transiciones, comprobando si están activas las transiciones en el momento de la ejecución del subsistema y asignado el marcado correspondiente al lugar indicado por la estructura de la RdPC.

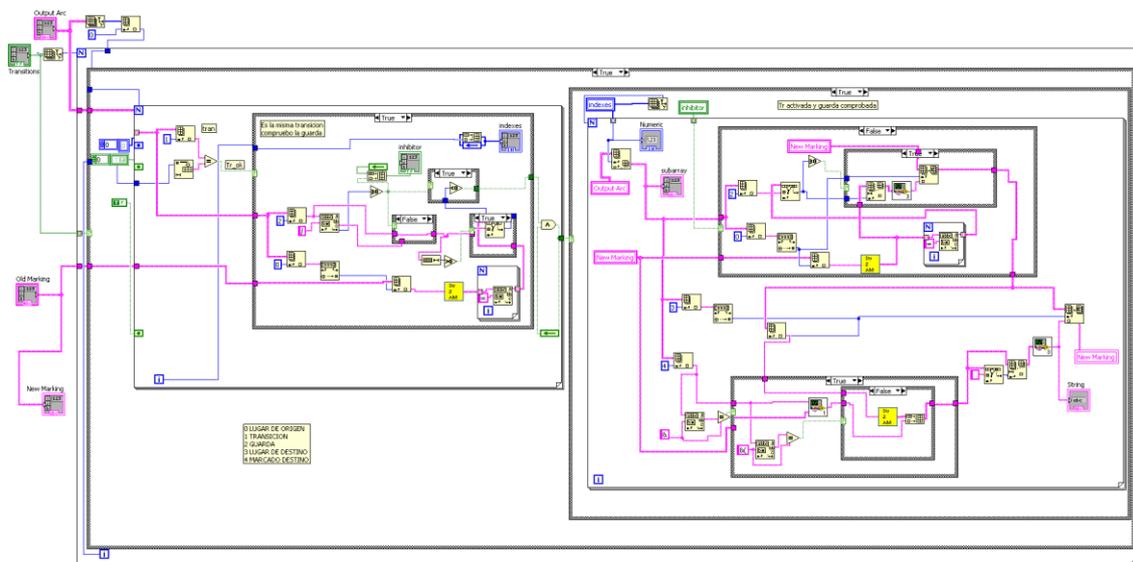


Figura 37: Programación bloque updateCPN_cont.vi

6.3. Librería comunicación LABVIEW-EXCEL

Previamente se ha analizado la estructura de las Redes de Petri para su implementación en Labview, determinando que se requería realizar la definición de la Red y los vectores de entradas, transiciones y disparo de transiciones en el programa Excel.

Debido a este factor, se ha trabajado ampliamente sobre la estructura de comunicación de estas dos herramientas.

Se va a presentar esta librería analizando en primer lugar las estructuras básicas de primer nivel que se emplean, analizando su funcionamiento. Posteriormente, se presentarán determinados subprogramas desarrollados utilizando estas herramientas básicas.

6.3.1. Estructuras básicas

En primer lugar, se presentan los subprogramas que permiten la apertura y cierre de la comunicación.



Figura 38: Bloques de comunicación Excel-Labview

Estos bloques abren y cierran el libro de trabajo de Excel (Workbook), la hoja sobre la que se va a trabajar (Worksheet) y la aplicación (Application), Deben ser abiertos en el orden: Aplicación -> Libro -> Hoja y cerrados en orden inverso.

Estos bloques deben estar presentes en cualquier diseño de aplicación con Excel, ya que sin ellos no es posible determinar sobre que archivo se trabaja o la página y por lo tanto la

referencia a las celdas no está bien definida. Esta estructura sigue en gran medida la programación en VisualBasic, pero trabajando con programación orientada a objetos.

Es por esto que es muy útil el siguiente bloque:

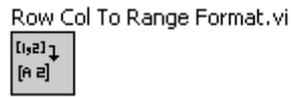


Figura 39: Bloque de conversión de formato de celdas

De este modo, podemos convertir el formato de referencia de celdas a partir de [Fila, Columna] utilizado en las matrices en Labview al formato Range, empleado en Excel.

A continuación, se van a analizar los bloques de subprogramas utilizados para obtener valores de Excel desde Labview.

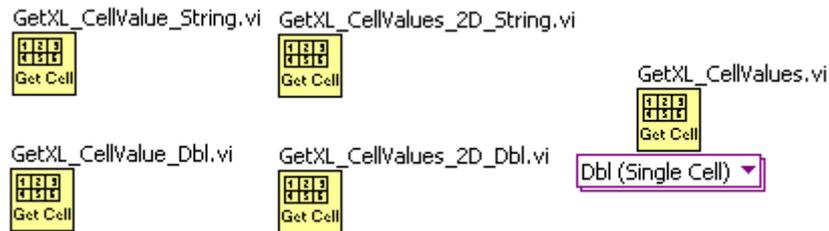


Figura 40: Bloques de lectura de datos

Como se puede ver en la figura, disponemos de 4 bloques individuales y un bloque con selector. Los bloques individuales se emplean para realizar adquisición de datos de los siguientes formatos:

- String (una celda)
- String 2D (más de una celda)
- Double (una celda)
- Double 2D (más de una celda)

Además de estos bloques, se dispone también de un bloque con selector, que permite utilizar el bloque genérico y mediante el desplegable elegir el tipo de dato que va a adquirirse. El funcionamiento interno es el siguiente:

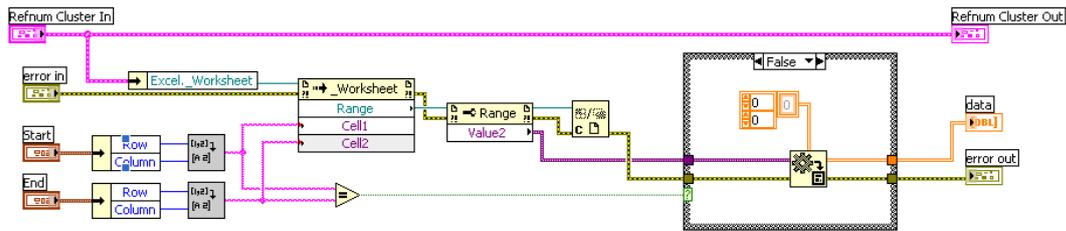


Figura 41: Programación bloque GetXL_CellValues.vi

Se introducen las celdas de inicio y fin de lectura, se extrae el valor y se le da el formato indicado en el selector.

En cuanto a la forma de escribir valores en Excel, el procedimiento es similar al de lectura, pero en este caso se introduce como una de las entradas el valor a guardar en Excel. Se dispone de las siguientes herramientas:

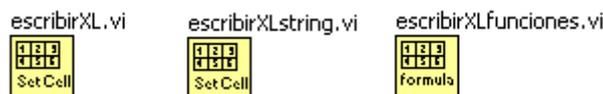


Figura 42: Bloques de escritura de datos

Los dos primeros bloques hacen referencia a la introducción de datos tipo “double” y “string”. Se introduce como entrada los referentes de celdas en formato [Fila, Columna] y el parámetro a escribir. En caso de escribir más de un valor, se procederá introduciendo el bloque en un bucle y cambiando la posición de la celda.

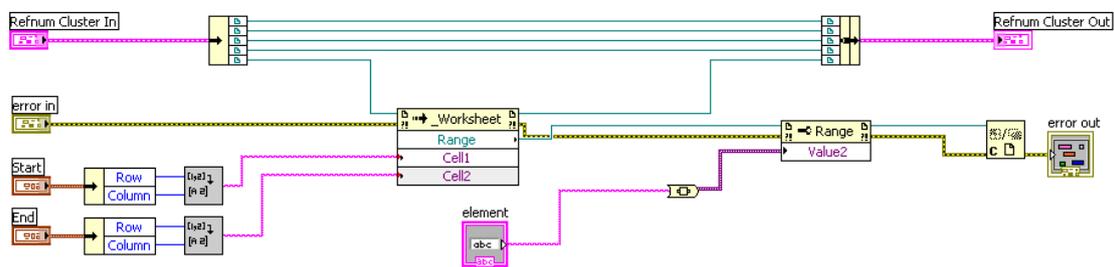


Figura 43: Programación bloques de escritura de datos

A partir del subprograma para escribir funciones se puede introducir una ecuación en modo fórmula, distinto a la introducción de datos.

Además de obtener datos y escribir nuevos valores, también se dispone de la posibilidad de marcar un rango de celdas para borrar sus valores, mediante el siguiente subprograma:



Figura 44: Bloque de borrado de datos

Por último, se presentan los subprogramas creados para guardar una hoja de trabajo con nuevo nombre, y para guardar los cambios realizados en una hoja.

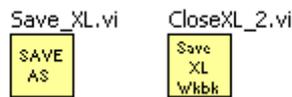


Figura 45: Bloques de guardado de datos

Este segundo bloque, “CloseXL_2.vi”, cierra el libro de trabajo, guardando previamente las modificaciones.

Se presenta a continuación un ejemplo de aplicación completa, a modo de demostración de uso de las diferentes herramientas presentadas en este apartado.

Se trata de una aplicación para la obtención de un valor de determinada celda en Excel.

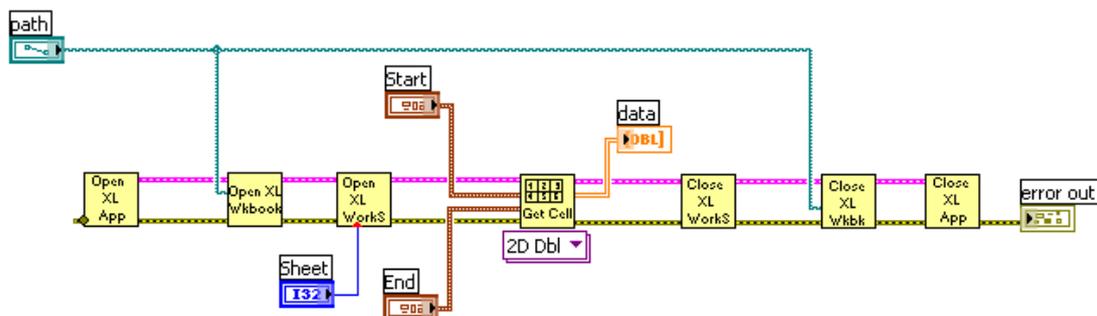


Figura 46: Ejemplo de uso de librería de comunicación Excel-Labview

El término “Path” hace referencia al directorio del archivo con el que se va a trabajar. El procedimiento de este programa es el siguiente:

- Se abre la aplicación
- Se abre el libro de trabajo marcado por el “Path”
- Se abre la hoja de trabajo

- Se extrae el valor de la celda, marcado por los indicadores “Start” y “End”, y se almacena en “data”
- Se cierra la hoja de trabajo
- Se cierra el libro de trabajo
- Se cierra la aplicación

En la siguiente figura podemos ver el interfaz gráfico para el usuario del ejemplo anterior, a través del cual se introducen los parámetros de ruta de acceso (“Path”), los indicadores de “Start” y “End” para la lectura de datos, en formato [Fila, Columna], y la hoja de trabajo (“Sheet”). A través del componente “data” se observaría por pantalla los datos extraídos del archivo Excel.

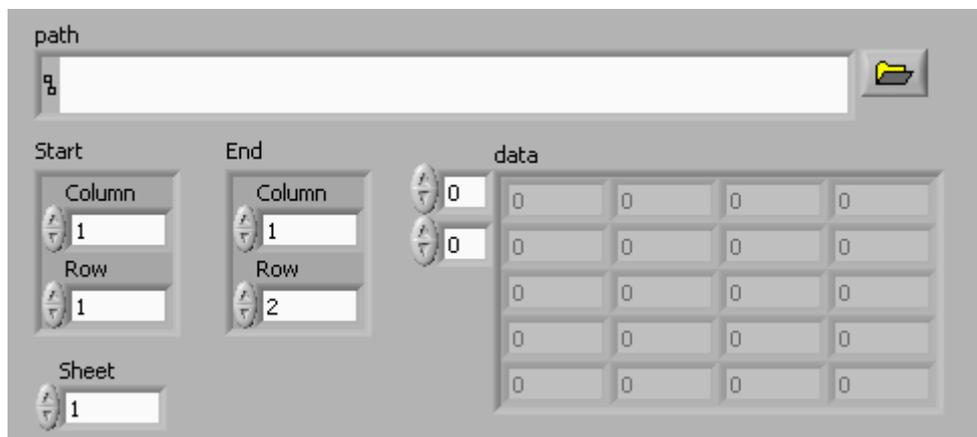


Figura 47: Interfaz gráfica del ejemplo planteado

Mediante este ejemplo se han mostrado los bloques principales desarrollados para la librería de comunicación Labview-Excel, que tal y como se verá a lo largo de la explicación del sistema desarrollado, se utiliza en gran cantidad de situaciones y facilita la implementación, y la futura comunicación con Maximo.

6.4. Estructura funcionamiento RdPCA

A lo largo del desarrollo de este apartado, y los sub-apartados que lo conforman, se va a presentar el desarrollo e implementación de la aplicación de Gestión de Mantenimiento en Labview, utilizando la estructura definida para Redes de Petri Coloreadas Adaptativas (RdPCA) y a partir de las librerías desarrolladas para la tesina.

El funcionamiento de RdPCA en Labview comienza con la declaración de variables y la asignación de valores iniciales y estructura. Posteriormente, dentro de un bucle repetido de forma temporal, se repite el procedimiento establecido dentro de dicho bucle.

Se va a proceder de la siguiente manera:

- Se analizarán las variables a utilizar y su inicialización
- Se explicará la metodología utilizada para la actualización de variables
- Se mostrará el funcionamiento de la actualización de marcado de la RdPCA

De este modo, se pretende analizar la estructura general de funcionamiento de la Red implementada, dejando para apartados posteriores el análisis detallado de los subsistemas que forman la aplicación.

6.4.1. Inicialización variables

La inicialización de variables se realiza de forma externa al bucle temporal de trabajo.

Se inicializarán en este punto todas las variables locales, gráficas y parámetros que se empleen a lo largo de la evolución del programa. Si esta inicialización no está bien definida pueden ocurrir pérdidas de datos del tipo:

- Valores que no se mantienen de una iteración a otra
- Vectores que pierden su formato (tamaño, valor, etc)
- Incorrecta asignación de datos durante la ejecución

Es por esto que para la asignación de valores en la inicialización, cada parámetro debe mantener su formato.

Además de estos conceptos básicos de la programación en Labview, en la fase de inicialización de variables del programa implementado se incluye el bloque de la figura 48.

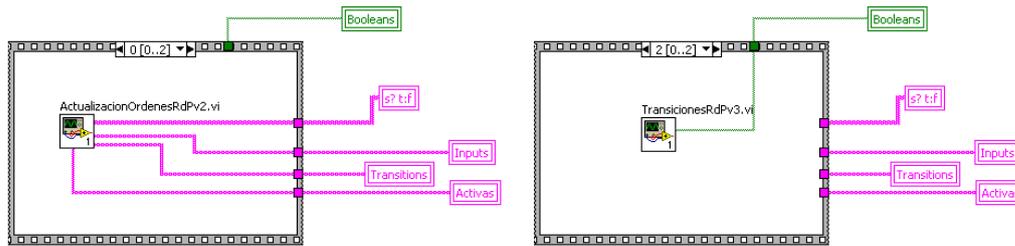


Figura 48: Inicialización estructuras RdPCA

Se trata de una estructura de programación en secuencias, formada por tres ventanas. En la imagen podemos ver la primera y la tercera. El contenido de la segunda es simplemente un retardo temporal, para dar al sistema tiempo suficiente de cerrar correctamente la aplicación Excel, y abrir la siguiente, y evitar problemas de malas inicializaciones.

La primera secuencia realiza una lectura de las estructuras activas de la RdPCA en Excel, a partir del subprograma “ActualizacionOrdenesRdPv2.vi” La tercera estructura utiliza el subprograma “TransicionesRdPv3.vi” para cargar el vector de booleanos correspondiente al disparo de las transiciones sobre el que trabajará posteriormente la aplicación. A continuación, se van a analizar estos dos subprogramas en detalle.

6.4.1.1. Inicialización estructura RdPCA

En la figura 49 se puede apreciar el conjunto completo de programación del subprograma “ActualizacionOrdenesRdPv2.vi”, remarcando 5 subdivisiones con el objeto de analizarlas por separado.

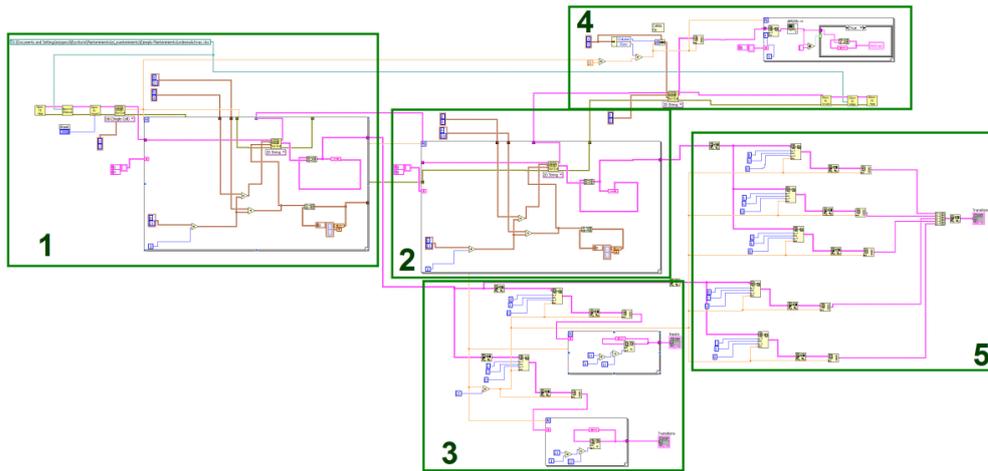


Figura 49: Programación bloque ActualizacionOrdenesRdPv2.vi con subdivisiones

En la subdivisión 1 se realiza la apertura de la aplicación Excel determinada en el “path”. Seguidamente, el primer bloque de lectura adquiere de Excel el número de órdenes declaradas en el archivo, lo que se utilizará como índice de referencia para los distintos bucles del programa.

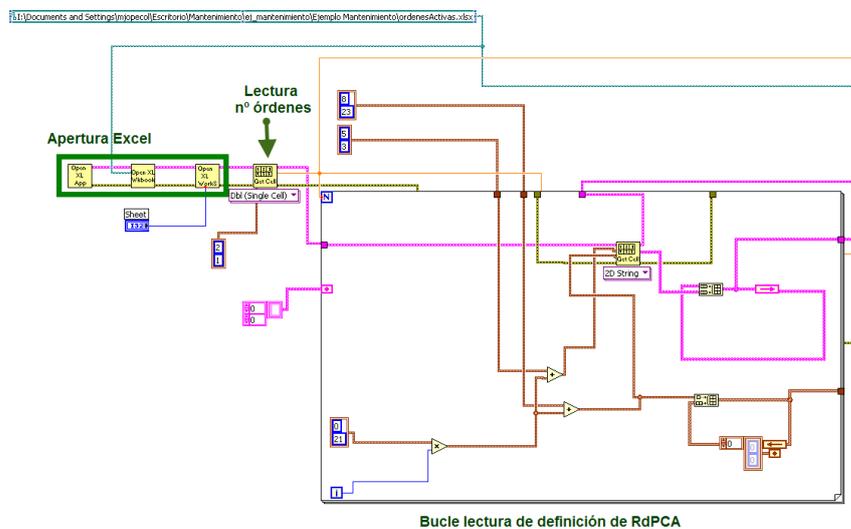


Figura 50: Programación subdivisión 1 bloque ActualizacionOrdenesRdPv2.vi

Después, se realiza el bucle de lectura, que realiza el mismo procedimiento tantas veces como órdenes de trabajo estén declaradas. Este bucle lee la definición de la estructura original para cada orden, y las almacena en la matriz de definición de estructura de la RdPCA de Labview, de las columnas tipo texto (Columnas 4 a 7 de Tabla 12).

En el bucle de lectura de la subdivisión 2 se realiza el mismo procedimiento, pero en este caso la lectura es de las 3 columnas numéricas de la definición de la red (Columnas 1 a 3 de la Tabla 12).

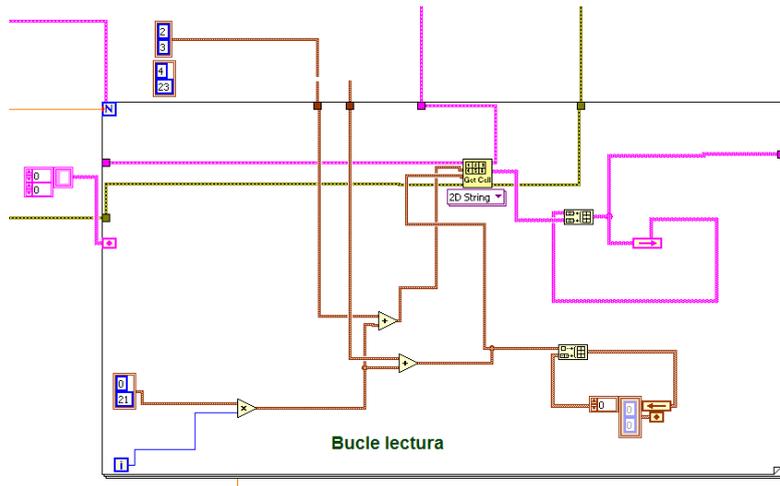


Figura 51: Programación subdivisión 2 bloque ActualizacionOrdenesRdPv2.vi

Se va a analizar a continuación la subdivisión 3. En la figura 52 se puede apreciar que se trata de dos estructuras de programación que repiten el mismo procedimiento. En este caso, se trata de separar de la matriz obtenida en la subdivisión 1 los vectores de Entradas y Transiciones y guardarlos en la variable correspondiente.

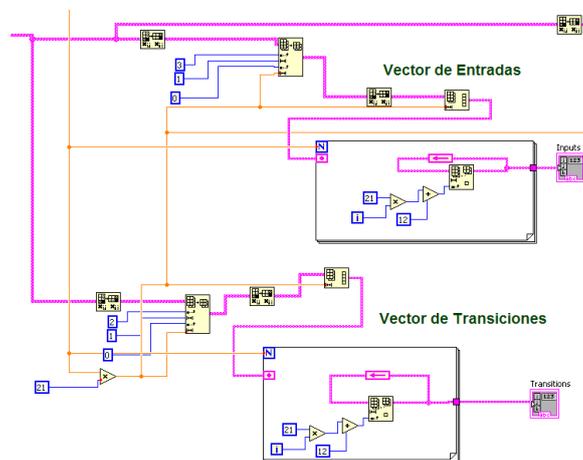


Figura 52: Programación subdivisión 3 bloque ActualizacionOrdenesRdPv2.vi

En la subdivisión 4 se extrae el primer vector del archivo Excel, donde están declaradas las órdenes de trabajo, y se obtienen los identificadores de todas las órdenes.

De este modo, se almacenan en un vector de órdenes activas, que se empleará en diversos subsistemas a lo largo de la tesina.

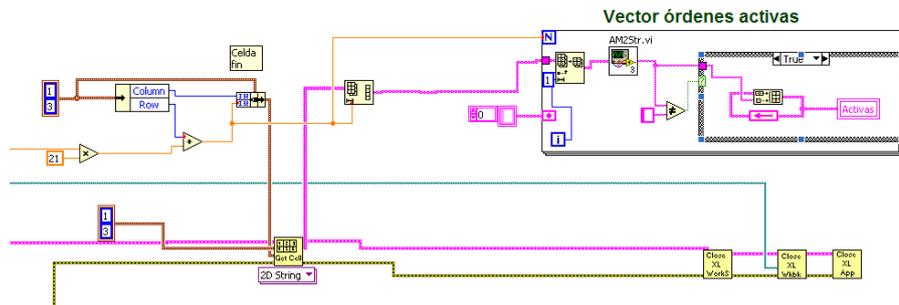


Figura 53: Programación subdivisión 4 bloque ActualizacionOrdenesRdPv2.vi

Por último, se va a analizar la subdivisión 5. En esta parte del subprograma, se separan los distintos vectores que dan forma a la definición de la estructura de la RdPCA, para dar formato a la matriz de definición. Este formato debe seguir la estructura de la Tabla 11.

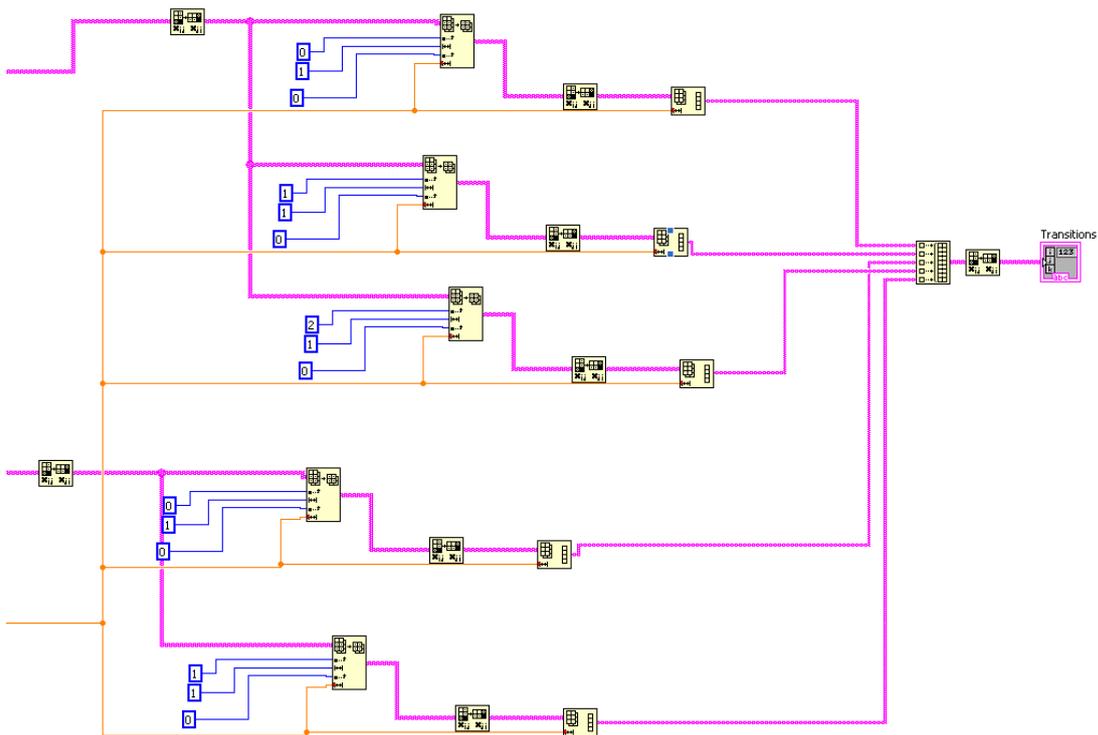


Figura 54: Programación subdivisión 5 bloque ActualizacionOrdenesRdPv2.vi

6.4.1.2. Inicialización vector de disparo de transiciones

El subprograma “TransicionesRdPv3.vi” se encarga de obtener de Excel el vector de booleanos que utilizará el programa para trabajar el disparo de las Transiciones.

Dentro de la estructura de programación, se pueden ver conceptos comunes a los analizados en el apartado anterior. Se puede apreciar que las estructuras de apertura y cierre de Excel siguen el mismo procedimiento, por lo que de ahora en adelante no se explicarán. Lo mismo ocurre para el parámetro de lectura de número de órdenes activas.

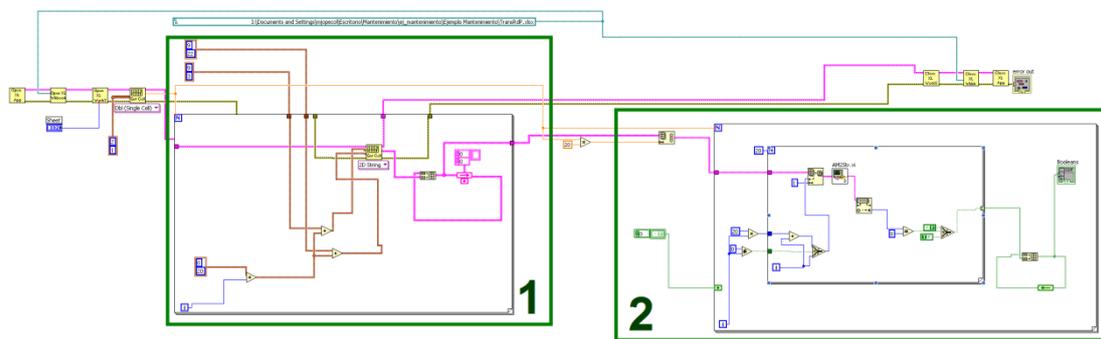


Figura 55: Programación subdivisión 1 bloque TransicionesRdPv3.vi

La subdivisión 1 realiza la lectura del vector de definición de transiciones en Excel, y lo almacena en un vector de Labview, procediendo de igual manera que la subdivisión 2 del apartado anterior.

En la subdivisión 2, se realiza la conversión de cada parámetro del vector de transiciones a formato Booleano, comprobando si el valor es igual o distinto a 0. De nuevo, esta estructura está situada dentro de un bucle que se repite tantas veces como órdenes declaradas existan en ese instante.

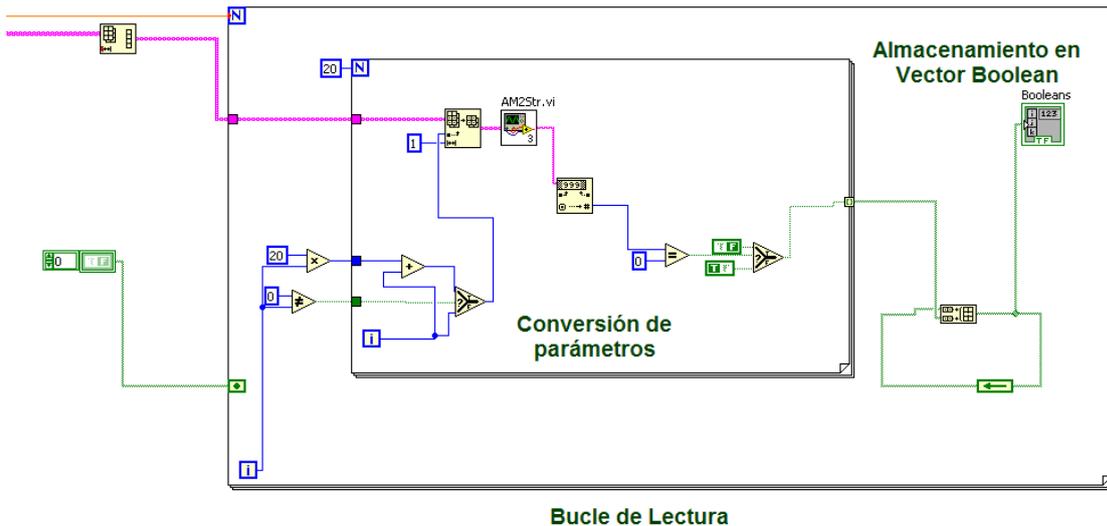


Figura 56: Programación subdivisión 2 bloque TransicionesRdPv3.vi

Una vez convertidos los valores a formato Boolean, se almacenan en el vector correspondiente. El resto del sistema trabajará sobre este vector.

6.4.2. Funcionamiento mediante estructura de secuencias

Tal y como se ha analizado previamente, el funcionamiento del programa procede inicializando las variables y después ejecuta la secuencia temporal, que es la encargada de la ejecución del programa.

Dentro de esta secuencia temporal, además de los diversos sistemas que estudiaremos más adelante, se utiliza una estructura de secuencias, para asegurar que el procedimiento sigue un orden determinado. El orden de procedimiento es el siguiente:

- Incremento de valor para los temporizadores
- Consulta de órdenes nuevas
- Actualización de Órdenes de trabajo y estructura RdPCA
- Actualización de Transiciones
- Obtención de vector booleano de transiciones
- Comprobación Deadlines y actualización de prioridades de Tareas.
- Actualización del marcado de los lugares de la RdPCA
- Ejecución de acciones asociadas al marcado

La estructura de secuencias debe seguir este orden para que todas las variables tengan el valor adecuado a la hora de ser utilizadas, y evitar problemas de reemplazo de valores o actualizaciones tardías.

6.4.2.1. Actualización estructura de red

La actualización de estructura de definición de la RdPCA se realiza en la secuencia 1 de la estructura de secuencias definida en el apartado anterior.

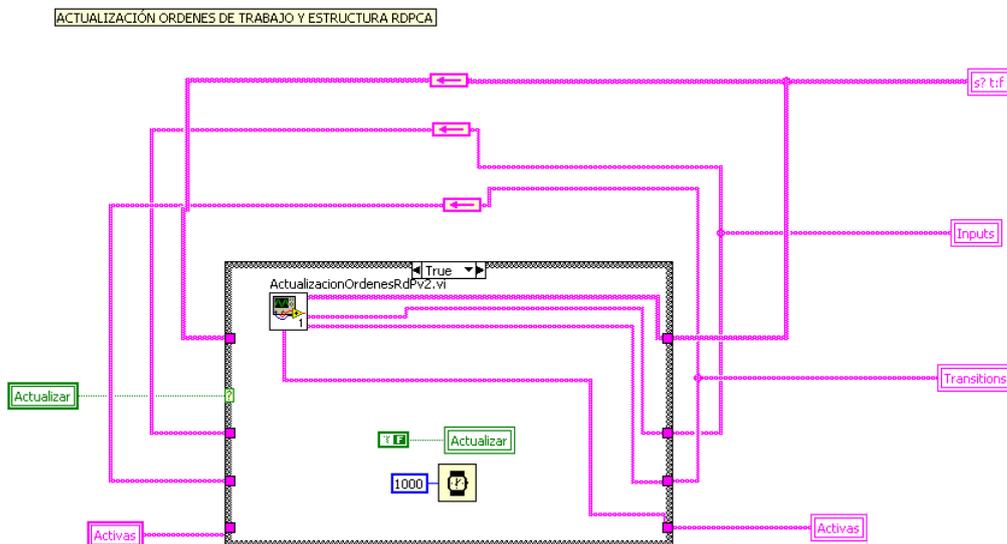


Figura 57: Estructura actualización órdenes de trabajo y estructura RdPCA

Como se puede ver en la figura 57, se trata de una estructura de trabajo "Case Structure", que se emplea para realizar un comportamiento en caso de que la variable de activación sea verdadera y otro comportamiento en caso de falsa.

En este caso, la variable de activación se asigna a la variable local "Actualizar", el cual se activa en caso de introducirse nuevas órdenes de trabajo en el sistema, o eliminar las órdenes finalizadas.

En caso de que dicha variable sea negativa, los vectores y matrices permanecen con los valores previos establecidos. En caso de ser positivo, se vuelve a leer del archivo Excel los parámetros.

En la figura se puede ver que el bloque de programación utilizado es el subprograma “ActualizacionOrdenesRdPv2.vi”, analizado en el apartado 6.4.1 de esta tesina. La realimentación se emplea para utilizar el valor actual de la variable en la próxima iteración.

6.4.2.2. Actualización de valores de entradas

La actualización del vector de valores booleanos de las entradas se realiza en la secuencia 1 de la estructura de secuencias previamente definida.

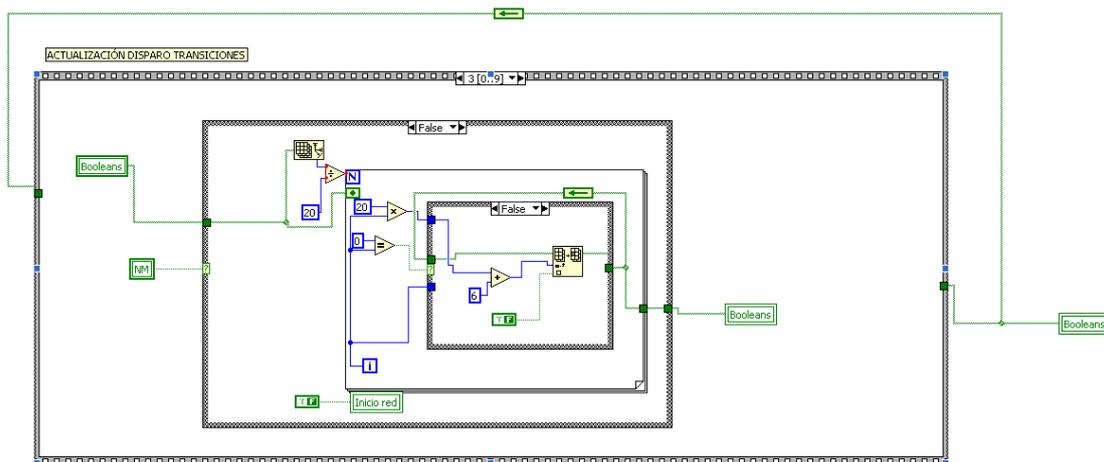


Figura 58: Estructura actualización valores entradas

Este programa utiliza de nuevo una estructura de secuencias. Esto es debido a que en cada secuencia se va a modificar, en caso necesario, el vector de booleans, por lo que activar y desactivar entradas sin el orden adecuado puede hacer que el programa no funcione correctamente.

De este modo, se trabajará con tantas secuencias como transiciones se empleen en el sistema, mas una secuencia para activar la red completa. El vector de booleans de las entradas se utilizará como entrada en casa secuencia, y antes de pasar a la secuencia siguiente se almacenarán los cambios en la variable. De este modo se permite también que aunque se

trabaje con distintas secuencias, sea posible el disparo de transiciones distintas sin que unos cambios sobrescriban otros.

Vamos a analizar los distintos tipos de secuencias por separado. En primer lugar, se va a analizar el caso en que se ha añadido una nueva orden de trabajo a la estructura, lo que implica que el vector de booleanos de entradas deba actualizarse también. En este caso, se hace uso del bloque “TransicionesRdPv3.vi”.

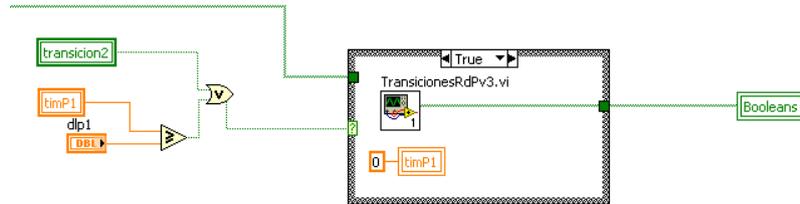


Figura 59: Activación de actualización de valores de entradas

A continuación, se analizará la estructura de programación implementada para iniciar la red de cada una de las órdenes activas en ese momento.

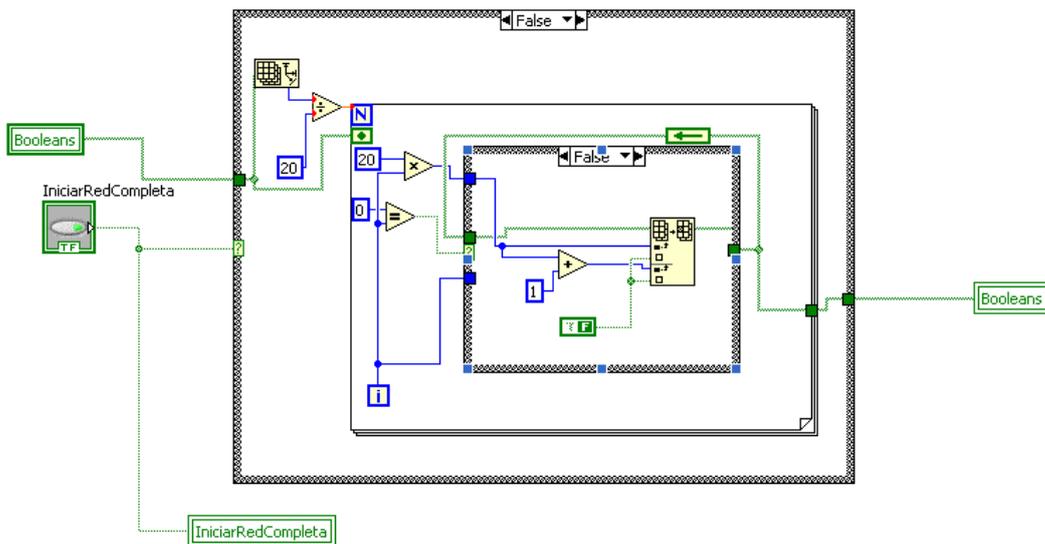


Figura 60: Activación Inicio completo de la RdPCA

La activación de esta estructura Case se realiza mediante la variable local “IniciarRedCompleta”.

Como se puede ver en la figura, se trata de un bucle for, necesario ya que se desea realizar el mismo procedimiento tantas veces como órdenes de trabajo existan en ese momento.

Dentro del bucle for, se puede ver otro bucle Case, en este caso empujado para determinar el índice de la orden de trabajo que se está activando en esa iteración del bucle for. El procedimiento dentro del Case consiste en activar las dos primeras entradas, encargadas de introducir la red en el sistema y empezar el marcado.

Otro caso dentro de la estructura consiste en activar la última orden añadida, permaneciendo las demás en su estado.

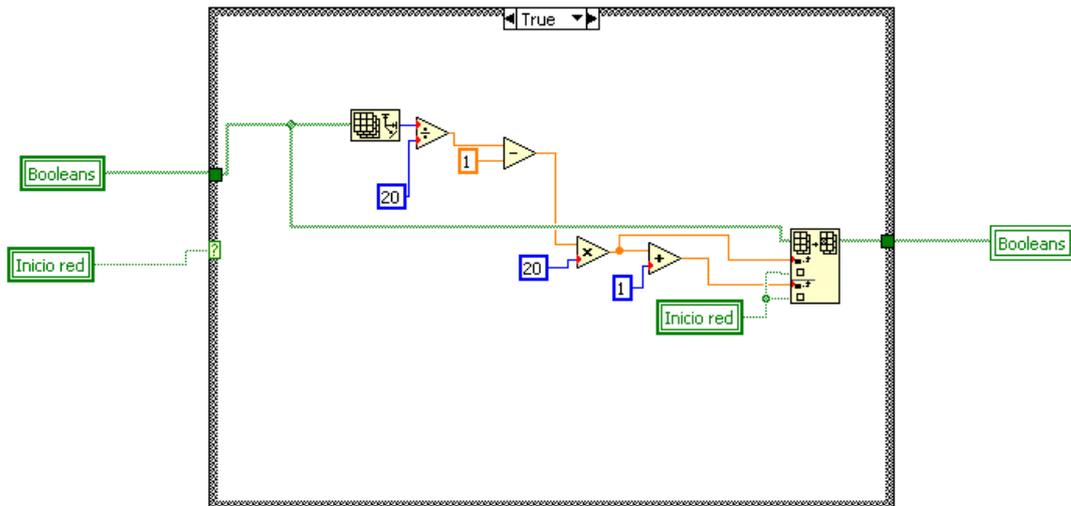


Figura 61: Activación subestructura específica de la RdPCA

En este caso, la variable de activación del Case es la variable local “Inicio Red”, que se activa automáticamente al introducir una nueva orden de mantenimiento en el sistema. Si esta variable es negativa, el vector de booleanos mantiene su valor, sin realizar ninguna modificación.

En caso de que la variable sea verdadera, el procedimiento consiste en activar las dos primeras entradas de la última subestructura de red dentro de la matriz de la RdPCA.

El resto de secuencia comparte la misma estructura de programación. Por esto, se va a utilizar uno de los casos como ejemplo, determinando que la diferencia entre los casos será la variable de activación y el índice de la entrada modificada en el vector de booleanos.

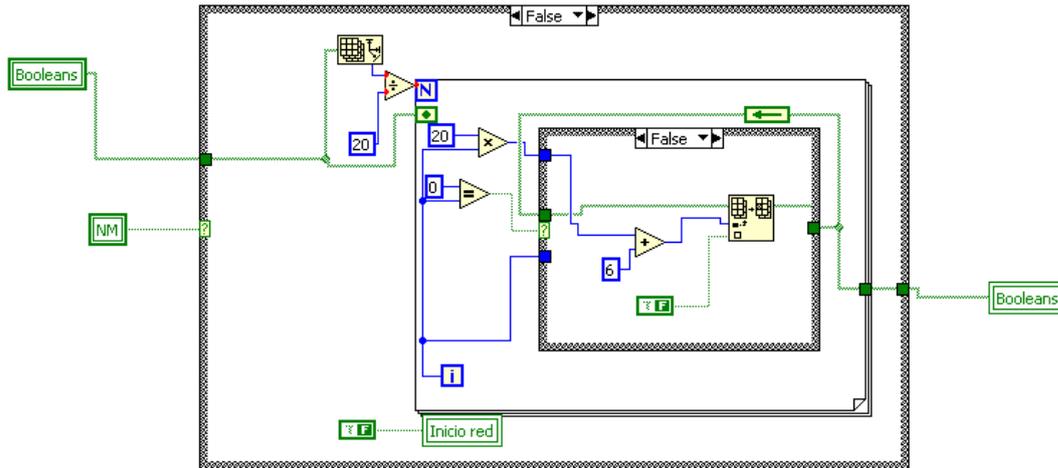


Figura 62: Desactivación de valor de determinada entrada

En este caso de ejemplo, la variable de activación estaría asignada a la variable local “NM”, que se activa tras consultar los requisitos de cada orden. El comportamiento en este caso es distinto si el valor es positivo o negativo.

En caso de ser negativo, tendría la estructura de la figura 62. De nuevo, se puede apreciar la estructura de programación tipo For, utilizada para determinar el índice de la orden con la que trabajar. En este caso, como no hay órdenes que necesiten material, se fuerza a valor falso la entrada correspondiente de todas las órdenes activas en el momento.

En caso de que alguna de las órdenes necesite material, la estructura sería de la forma visible en la figura 63. Se va a emplear una variable llamada “Indices”, que hace referencia a las órdenes que necesitan material.

En este caso, se activan tan sólo las transiciones correspondientes a los índices de la matriz “Indices”.

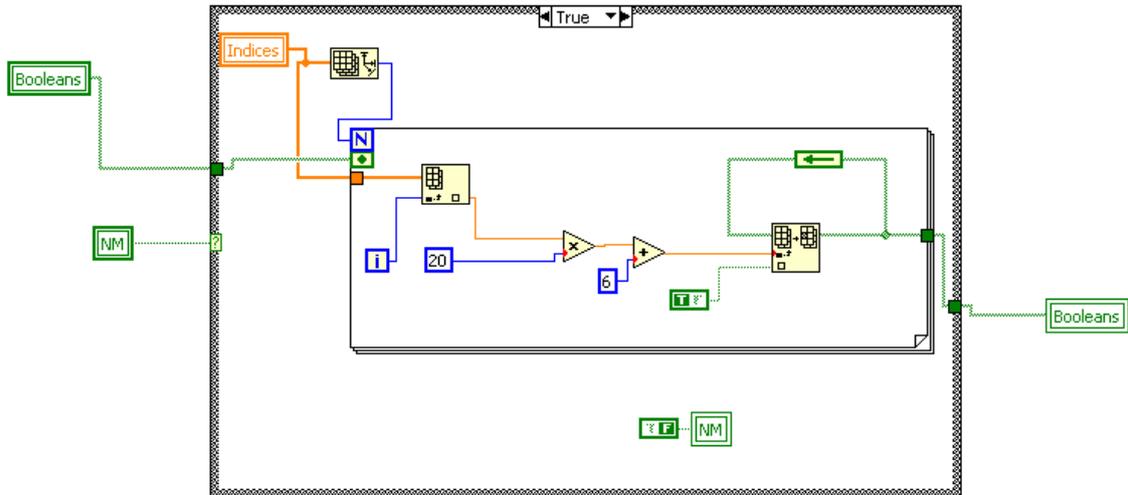


Figura 63: Activación de valor de determinada entrada

6.4.2.3. Comprobación deadlines y actualización de prioridades de tareas.

En cada iteración del programa, se incrementa el valor de los contadores, con la finalidad de poder comparar el valor actual con el valor límite establecido para cada tarea, el deadline. Esta comparación se realiza mediante el subsistema de comprobación de deadlines y actualización de prioridades (figura 64).

Este subsistema utiliza como entradas las matrices numéricas "ArrayTimers", "ArrayDLP" y "Prioridades". La primera de estas matrices contiene todos los temporizadores de las tareas activas, y la segunda contiene los respectivos deadlines.

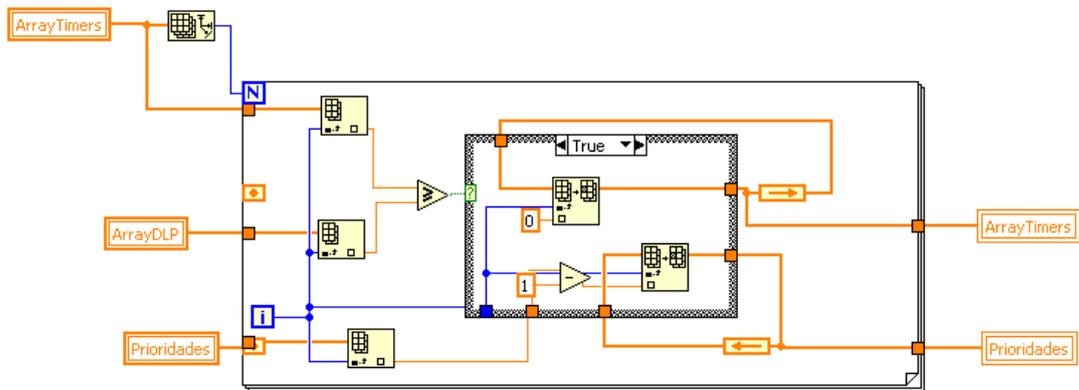


Figura 64: Estructura comprobación deadlines y actualización prioridades

A partir de un bucle For, se comparan estos valores para cada tarea, y en caso de que el temporizador alcance el tiempo máximo o deadline, la prioridad de la tarea sobre la que se está trabajando decrementa su valor.

Los valores de las prioridades de las tareas están disponibles en la matriz de “Prioridades”.

6.4.2.4. Actualización de marcado

La actualización del marcado de los lugares de la RdPCA está formada por dos procesos distintos.

Por un lado, en la ventana 3 de la estructura de secuencias utilizada, se utiliza el bloque de la librería “Input2Trans”. Como hemos visto en apartados anteriores, es mediante este bloque como el sistema obtiene el vector de disparo de las transiciones.

Para ello, emplea como entradas los vectores de Entradas y Transiciones, y el vector de booleanos de las Entradas activas, que ha sido modificado tal y como se ha analizado en el apartado 6.1.

A partir de estas tres entradas, el sistema genera el vector de booleanos de disparo de transiciones, que se transmitirá a la ventana 4 de la estructura de secuencias.

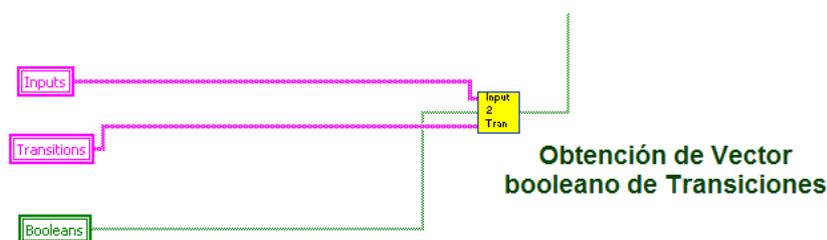


Figura 65: Estructura obtención de vector de transiciones

Desde la secuencia 4, y una vez está disponible el vector de disparo de transiciones, se realiza la actualización de marcado de los lugares de la RdPCA.

Para ello, se emplea el bloque “UpdateCPN_cont.vi”, previamente analizado en la librería de RdP. Este bloque utiliza como entradas la matriz con la definición de estructura de la red, el vector de booleanos de las transiciones y el marcado anterior.

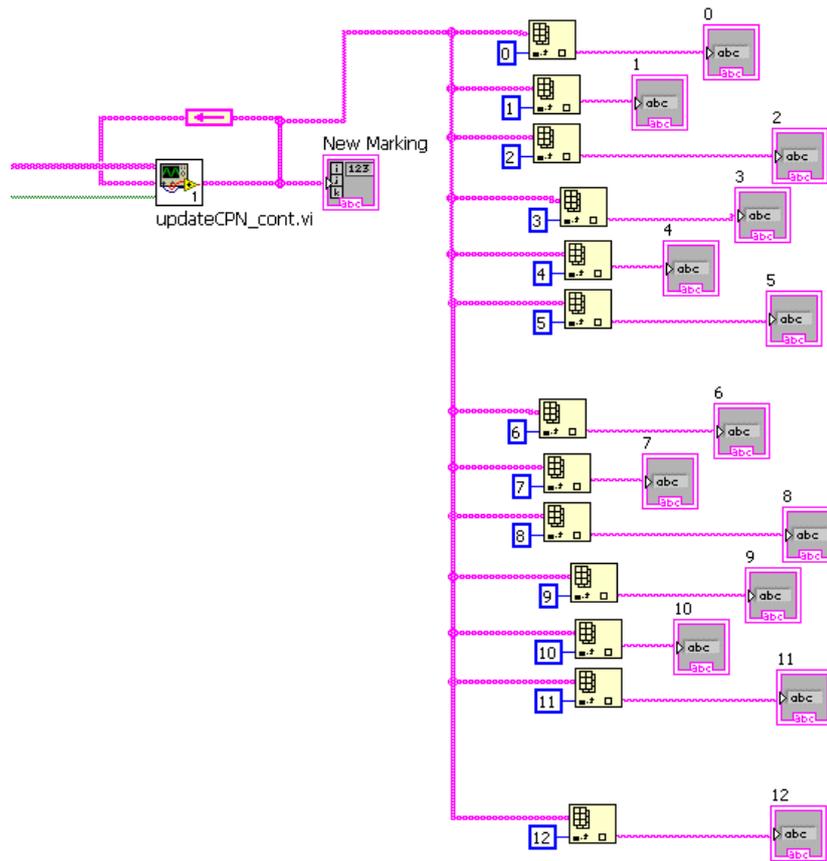


Figura 66: Estructura actualización de marcado

Una vez se dispone del nuevo marcado, se separa el valor de la marca de cada uno de los lugares de la RdPCA y se introduce en su estructura gráfica, para verlo desde el interfaz de usuario. Más adelante se analizarán determinados subsistemas que se realizarán en función de la marca que exista en cada lugar.

6.5. Subsistemas implementados para la RdPCA

En el apartado anterior se ha desarrollado la estructura de funcionamiento de la Red de Petri Coloreada Adaptativa, analizando la distribución temporal dentro del programa y las distintas secuencias llevadas a cabo para la implementación.

En este apartado se pretenden analizar los diversos subsistemas que se han desarrollado para el funcionamiento de la aplicación, cuya implementación se encuentra a lo largo del programa y las secuencias que lo forman.

6.5.1. Creación ordenes nuevas

La creación de órdenes nuevas es un apartado crítico de este sistema, ya que como se ha analizado en repetidas ocasiones el programa debe ser capaz de adaptar y redefinir la RdPCA, sin modificar los parámetros previos y sin que este cambio en la estructura de definición de la red afecte al funcionamiento normal de la aplicación.

A la hora de añadir una nueva Red a la estructura de la RdPCA, se deben realizar los siguientes procedimientos:

- Crear estructura completa de la red en el archivo Excel de definición de la RdPCA
- Añadir al vector de booleanos de las entradas los valores de la nueva red
- Habilitar la actualización de variables
- Iniciar la nueva red dentro del sistema actual
- Incorporar los parámetros de tiempo disponible y prioridad de tarea a las correspondientes matrices de datos

En la implementación del programa estos procedimientos se encuentran separados en dos subsistemas separados:

- Incorporación Timers y Prioridades
- Creación estructura red y vector entradas

En los siguientes apartados se estudiará el funcionamiento de estos subsistemas, así como la programación necesaria para su implementación.

6.5.1.1. Incorporación timers y prioridades

Se presenta a continuación el subsistema desarrollado para la incorporación de timers y prioridades.

Al crear una nueva orden de trabajo y añadir la estructura de una nueva red a la RdPCA, no se puede pasar por alto la necesidad de que las prioridades adapten su valor al paso del tiempo.

Es por esto que al añadir la orden, se debe activar también este subsistema, que utiliza la matriz de prioridades para añadir el valor de prioridad de la nueva tarea incorporada al sistema.

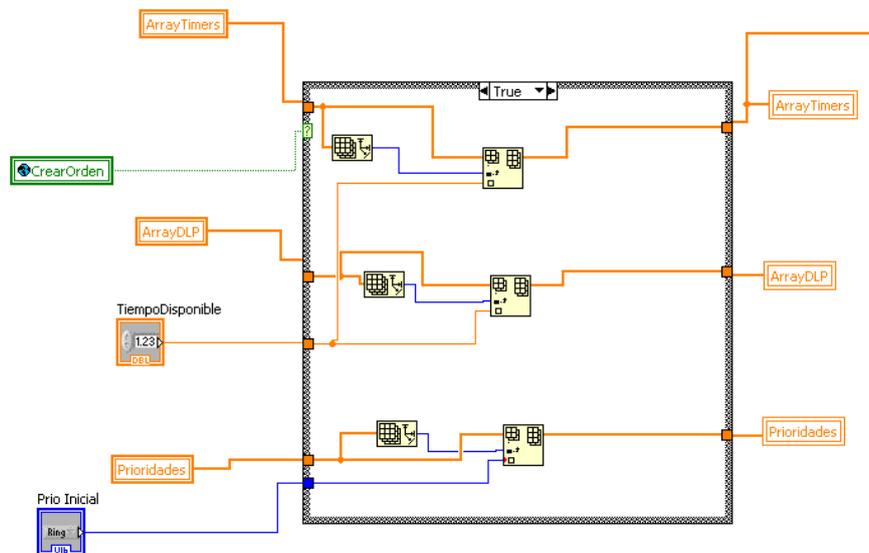


Figura 67: Estructura incorporación nuevos timers

Además, se utilizan dos matrices más: ArrayTimers y ArrayDLP. La matriz de ArrayTimers hace referencia al contador temporal incorporado en el sistema, que se empleará para analizar el paso del tiempo, y que se actualiza en cada iteración del programa.

La matriz ArrayDLP incluye un valor límite de evolución temporal de cada tarea. Esto significa que a cada tarea se le asigna un tiempo máximo de evolución temporal, tras el cual la prioridad de la tarea disminuye, aumentando la criticidad de la tarea. De este modo, se trabaja con prioridades que evolucionan en el tiempo, aumentando la prioridad según se acerca el tiempo máximo establecido para realizar la reparación.

6.5.1.2. Crear orden y vector de entradas

El subsistema principal para la creación de nuevas tareas de mantenimiento dentro del sistema se encarga de crear la estructura de la nueva red y añadirla a la estructura de definición de la RdPCA actual, y de crear el correspondiente vector de booleanos de entradas.

Tal y como se puede ver en la figura 68, el parámetro de activación de este subsistema está asignado a la variable global CrearOrden, que podrá ser activado de forma externa al sistema de gestión de mantenimiento.

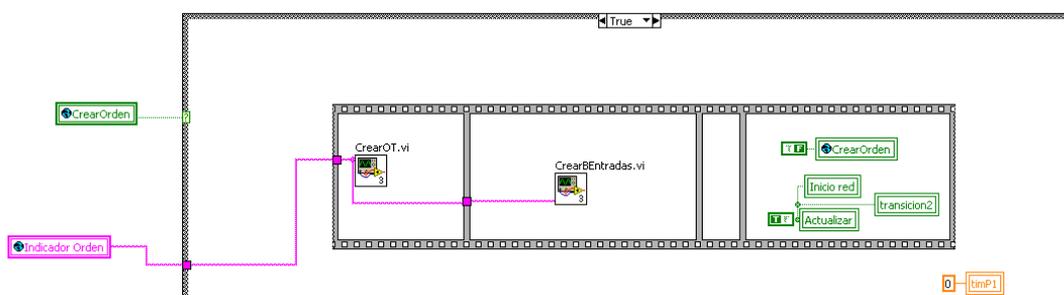


Figura 68: Estructura creación parámetros nueva orden

De este modo, cuando los distintos tipos de mantenimiento que se utilizan como entrada al sistema de gestión generan una nueva orden de trabajo, activan este valor, proporcionando al sistema un indicador de la orden creada, que se empleará como indicador de referencia de la orden a lo largo de todo el sistema.

Una vez está activo el subsistema de creación de órdenes de trabajo nuevas, se puede ver en la figura que se trabaja con una estructura de secuencias.

En la primera secuencia se crea la estructura de la nueva red a incorporar a la RdPCA, utilizando como marca de color el indicador de la nueva tarea de mantenimiento.

Para esta creación de una estructura totalmente nueva, se ha desarrollado el subprograma “CrearOT.vi”, que se analiza a continuación.

En la figura 69 se observa la estructura completa de programación de este subprograma. Dado su gran tamaño es complicado explicar el funcionamiento completo a partir de esta imagen,

por lo que se han remarcado las distintas subdivisiones que lo forman, para analizar la implementación.

La primera tarea del subsistema consiste en analizar el número de tareas activas, y determinar de este modo la posición donde comenzar la escritura, ya que como se ha analizado en otros apartados todo el sistema funciona con índices de referencia y así nos aseguramos que la nueva orden se incorpora en último lugar.

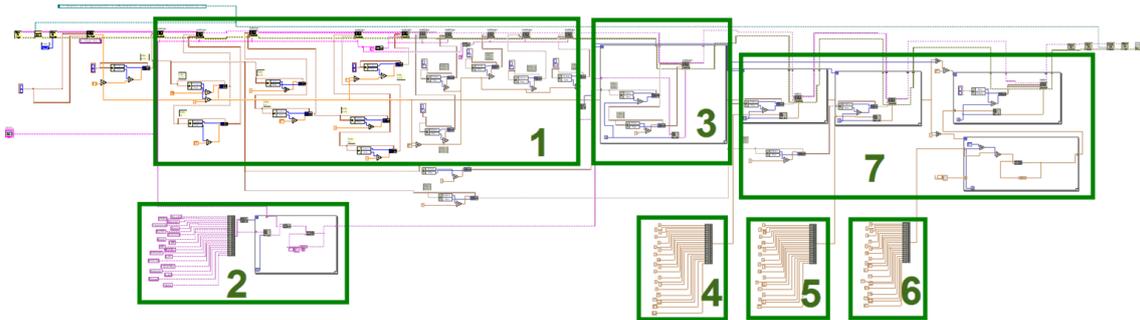


Figura 69: Programación bloque CrearOT.vi y subdivisiones

Este subprograma trabaja por un lado con los valores numéricos de la estructura, y por otro con los parámetros en formato texto. Se va a estudiar en primer lugar la estructura formato texto.

Tabla 13: Componentes string de definición RdPCA

MARCADO INICIAL	MARCADO FINAL	TRANSICIÓN	ENTRADAS
	OT1	NuevaOT1	NuevaOT1
	OT1	NuevaOT1	NuevaOT1
OT1	OT1	EquipoOT1	EquipoOT1
OT1	OT1	prio1OT1	prio1OT1
OT1	OT1	SolicitudEOT1	SolicitudEOT1
OT1	OT1a	EquipoDispOT1	EquipoDispOT1
OT1	OT1	MaterialOT1	MaterialOT1
OT1	OT1	prio1OT1	prio1OT1

OT1	OT1	SolicitudMOT1	SolicitudMOT1
OT1	OT1b	MatDispOT1	MatDispOT1
OT1	OT1a	NNEOT1	NNEOT1
OT1	OT1b	NNMOT1	NNMOT1
OT1a	OT1	MeteoOT1	MeteoOT1
OT1b		MeteoOT1	MeteoOT1
OT1	OT1	OpcionesOT1	OpcionesOT1
OT1	OT1	OptimoOT1	OptimoOT1
OT1	OT1	MeteoOT1	MeteoOT1
OT1	OT1	CambioPrioOT1	CambioPrioOT1
OT1	OT1	NegativaOT1	NegativaOT1
OT1	OT1	ValidaOT1	ValidaOT1
OT1		BorradoOT1	BorradoOT1

En la tabla 13 se puede ver parte de la composición de la estructura de definición de la RdPCA, tratando en este caso sólo la parte en formato texto. Las subdivisiones 1, 2 y 3 hacen posible la creación de esta parte de la estructura. La subestructura 1 realiza la creación de texto para el marcado inicial y final de los lugares de la RdPCA, a partir del identificador de la orden. Estos marcados se corresponden a las columnas 1 y 2 de la tabla 13.

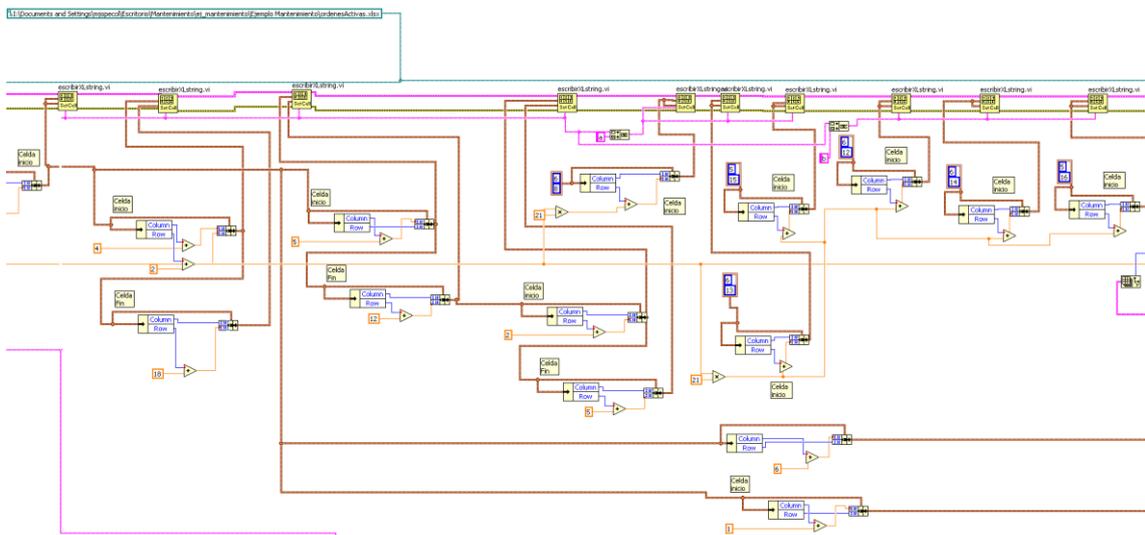


Figura 70: Subdivisión 1 bloque CrearOT.vi

La subdivisión 2 también utiliza el indicador de la orden para crear parámetros texto, aunque en este caso se encarga de crear los vectores de transiciones y entradas, en las columnas 3 y 4 de la tabla 13.

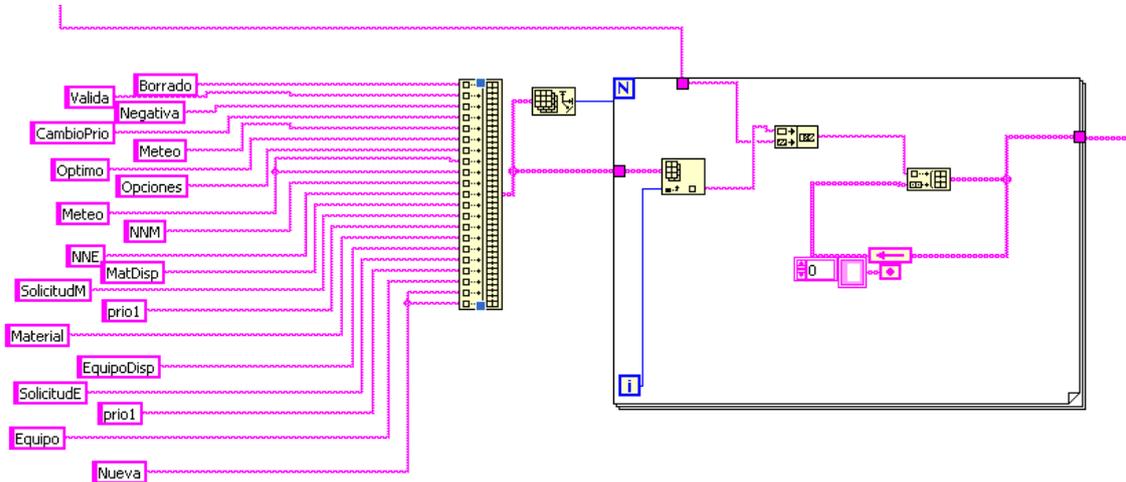


Figura 71: Subdivisión 2 bloque CrearOT.vi

Debe construir la matriz de texto correspondiente a cada valor, y añadir a todos ellos el indicador de la nueva orden. Por último, la subdivisión 3 escribe estos dos vectores en el archivo Excel, a partir de un bucle de escritura.

A continuación, se estudiarán la creación de los parámetros numéricos de la definición de estructura de la RdPCA.

Tabla 14: Componentes numéricos de definición RdPCA

ORIGEN	Nº TRANS	DESTINO
0	0	1
0	1	4
1	2	2
2	3	3
3	4	2
2	5	7

4	6	5
5	7	6
6	8	5
5	9	7
1	10	7
4	11	7
7	12	8
7	12	8
8	13	9
9	14	10
10	15	11
10	16	8
11	17	8
11	18	12
12	19	12

Como se puede ver en la tabla 14, se trata de los tres vectores correspondientes a lugar de origen, número de transición y lugar de destino. La creación de estos vectores se realiza a partir de las subdivisiones 4, 5 y 6 la figura 69, y se pueden ver en detalle en la figura 72.

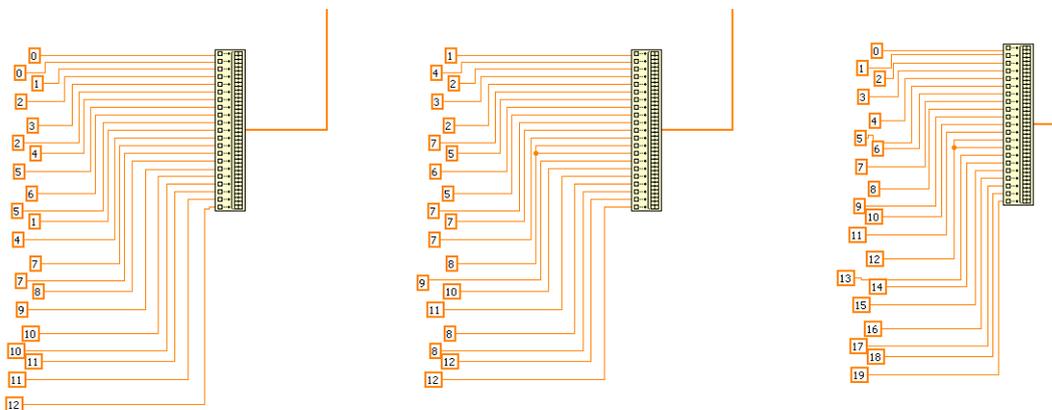


Figura 72: Subdivisiones 4, 5 y 6 bloque CrearOT.vi

En la figura 73 se puede ver la implementación de la subdivisión 7, cuya misión es escribir en las celdas correspondientes los vectores de Origen, Nº Transición y Destino, previamente obtenidos.

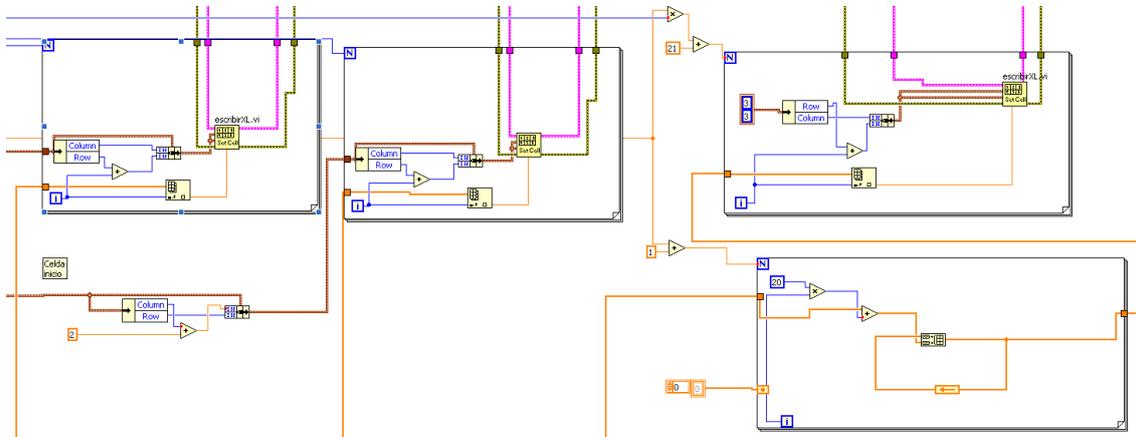


Figura 73: Subdivisión 7 bloque CrearOT.vi

Se concluye de este modo el análisis del subprograma “CrearOT.vi”. Se pasa a continuación a estudiar el subprograma “CrearBEntradas.vi”, utilizando en la segunda ventana de la estructura de secuencias.

Este subprograma es el encargado de crear, en el archivo Excel de valores de las entradas, el vector de booleanos de la nueva tarea de mantenimiento. Al igual que en el caso de creación de nueva tarea de mantenimiento, el primer punto a analizar es el número de órdenes activas en ese momento, para calcular la posición de escritura.

La estructura de este archivo Excel está formado por un indicador de la orden en la primera columna, seguido por un vector de valores de las entradas en la columna correspondiente. Es por esto que el subprograma tiene dos funciones de escritura, una de ellas introduce el indicador de la orden en el archivo, y la otra crea el vector de valores.

La programación de este subprograma puede verse en la figura 74.

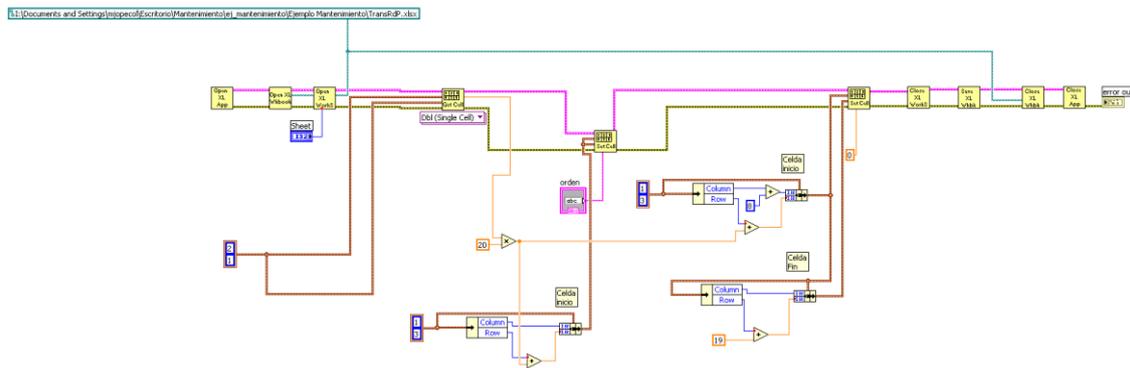


Figura 74: Programación bloque CrearBEntradas.vi

Analizando de nuevo la figura 68, vemos que se han analizado las primeras secuencias de la estructura. La última secuencia es la encargada de activar las variables de actualización para que todos los parámetros del sistema se adapten a la nueva estructura.

Se activa, además, la variable que hace posible el inicio de la nueva estructura dentro de la RdPCA, sin afectar al resto de partes de la definición ni al funcionamiento del sistema.

6.5.2. Información de tareas activas

A lo largo del sistema, es conveniente disponer de la información necesaria, relativa a cada una de las tareas activas.

Esta información se almacena en la matriz de información de las órdenes, llamada "InfoOT", y se emplea en diversos subsistemas posteriores para obtener la información relativa a cada parámetro.

Se dispondrá por lo tanto de una matriz de información, formado por todas las órdenes activas, que proporcionará la siguiente información:

- Nombre de la orden
- Prioridad
- Total de Personal
- Total de Material
- Coste/Hora

- Total horas tarea

En primer lugar, se define la matriz “InfoOT” como una matriz de 6 filas (una por parámetro) y tantas columnas como órdenes activas existan en el momento de la ejecución.

Se han creado diversos subsistemas para mantener actualizado y correcto el valor de los parámetros a estudiar, que se presentan a continuación.

Para la construcción del parámetro de Prioridad, se emplea el vector de prioridades creado al añadir una nueva orden.

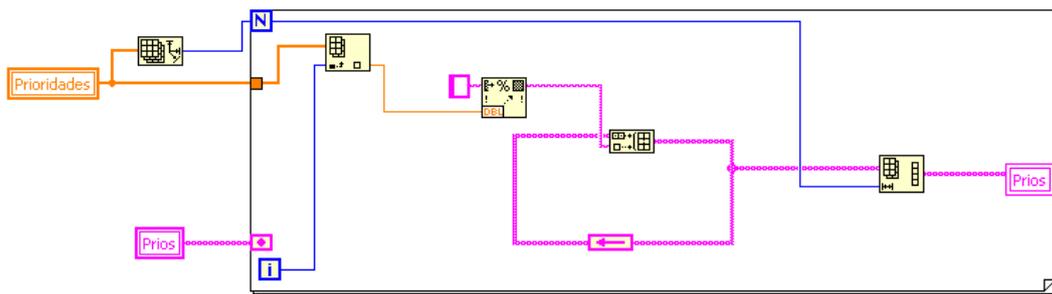


Figura 75: Actualización valores prioridades de matriz InfoOT

Este vector trata con valores numéricos, que deben ser convertidos para que la estructura de la matriz de información trabaje con el mismo tipo de datos en todos los parámetros. De otro modo, no se puede crear una sola matriz de información.

En la figura 75 se puede ver la estructura de programación del vector de prioridades, que se introducirá en la matriz de información.

Los parámetros de total de personal y material están concebidos como un valor total y no por separado ya que se dispone de la información detallada dentro de las respectivas matrices de información “PersonalOrden” y “MaterialOrden”.

Para la consulta de disponibilidades es adecuado disponer de la información separada por componentes, pero a la hora de calcular la agrupación sólo se precisa conocer el total para poder calcular el espacio que ocupa este material y la cantidad de personal que se precisarían en la embarcación.

En la figura 76 se puede ver la programación necesaria para el cálculo de personal. La implementación para material se realizaría de la misma manera.

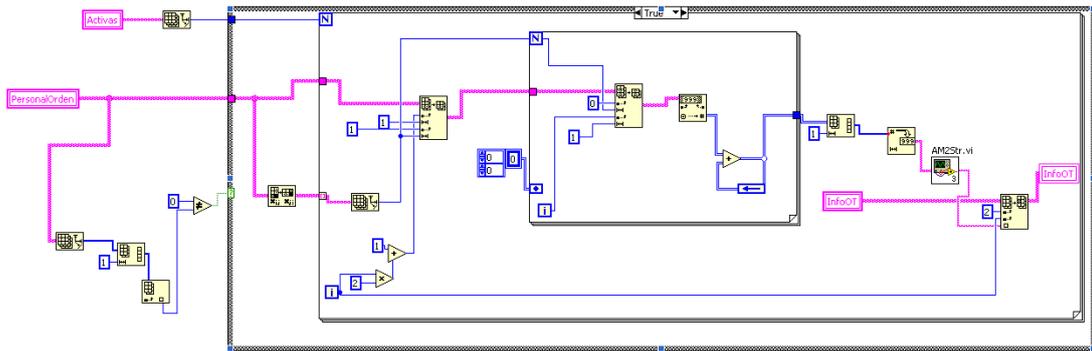


Figura 76: Actualización valor total de Personal de matriz InfoOT

Este subprograma calcula el total de personal necesario y sustituye el valor inicial de la matriz de información por el valor calculado.

El siguiente subprograma consiste en la actualización de valores de Coste/Hora de cada tarea.

A partir de la información de la plantilla de Excel se obtiene un valor estimado de coste/hora de ejecución de la tarea, obtenido a partir del software Maximo.

Mediante el subsistema de la figura 77 se modifica ese valor en la matriz de información de la orden de trabajo.

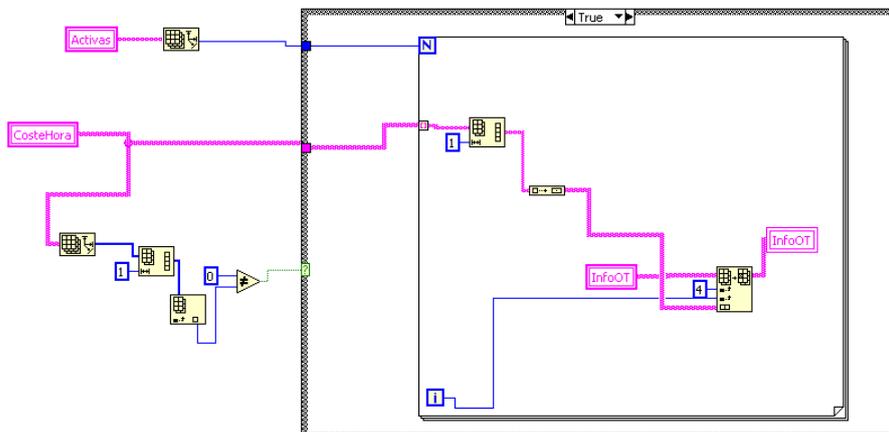


Figura 77: Actualización valores coste/hora de matriz InfoOT

De forma similar se procede para la actualización del total de horas de ejecución de la tarea.

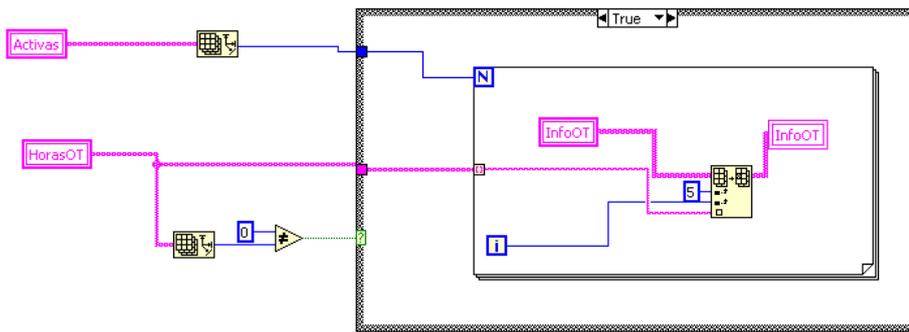


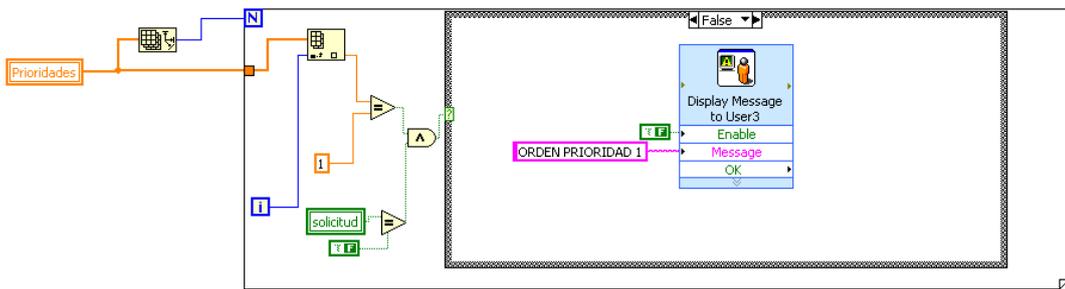
Figura 78: Actualización valores Horas de matriz InfoOT

Los parámetros actualizados de la matriz de Información de orden de trabajo se utilizarán como entrada para el subsistema de agrupación, que utilizará estos parámetros para calcular los parámetros de las diversas agrupaciones que se propongan.

6.5.3. Informe prioridad alta

En este apartado se va a analizar el subsistema encargado de avisar al usuario de la aplicación de que alguna de las tareas ha alcanzado la prioridad 1, valor más restrictivo.

En la figura 79 podemos ver el comportamiento del subsistema si se ha alcanzado la prioridad 1 (parte inferior) y si no se ha llegado a un valor tan crítico (parte superior).



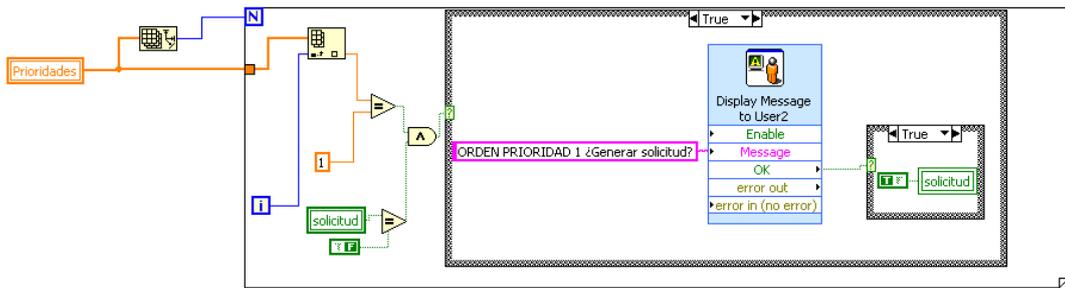


Figura 79: Estructura informe prioridad alta

Cuando se alcanza un valor crítico, la prioridad 1, significa que el tiempo disponible para realizar la reparación o arreglo está muy cerca, y esta tarea debe ser prioritaria en la agrupación.

Puede ocurrir que este valor de prioridad se active y la tarea no esté lista para agrupación, debido a la falta de material o personal necesario.

Es por esto que se utiliza un interfaz de aviso al usuario, que informa por pantalla de que alguna de las tareas ha alcanzado la prioridad uno, para que el usuario pueda acelerar el proceso de disponibilidad de estos parámetros en la medida de lo posible.

6.5.4. Consulta requerimientos

En el presente apartado se va a analizar el subsistema de consulta de requerimientos de la orden de trabajo.

De acuerdo al funcionamiento de la RdPCA en Labview, ampliamente analizado en apartados previos, la existencia de la marca de una orden de trabajo en cada estado activa la realización de determinados procedimientos.

En este caso, partiendo de los estados 1 y 4 de la Red de Petri de la figura 8 (apartado 3.2), las marcas en estos estados deben consultar la correspondiente información de orden de trabajo para determinar los requisitos de material y personal necesarios para la ejecución de la tarea.

El funcionamiento del subsistema implementado está activado por la presencia de marcas de color en los estados 1 ó 4.

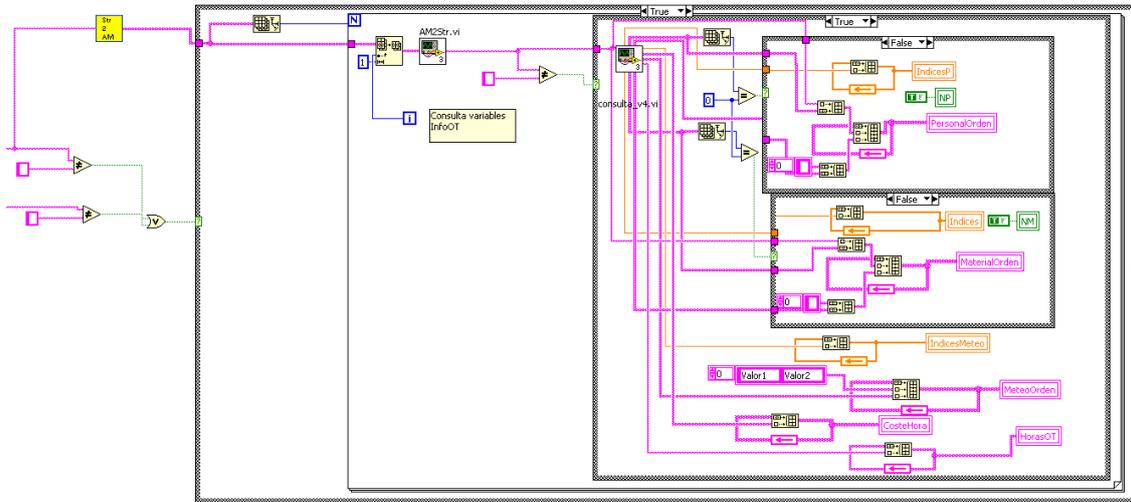


Figura 80: Estructura Consulta de requerimientos

El procedimiento consiste en extraer las marcas establecidas en los lugares para conocer las órdenes de las que se trata de extraer la información.

A continuación, el subsistema implementado utiliza el subprograma “consulta.vi”, que se analizará después. Con la información obtenida, se debe introducir en las matrices adecuadas para cada caso, junto con los índices de referencia de las órdenes. El funcionamiento del almacenamiento de datos es el siguiente:

- Material
 - Necesidad: Guardar índices y valores
 - No necesidad: Guardar índices
- Personal
 - Necesidad: Guardar índices y valores
 - No necesidad: Guardar índices
- Meteorología
 - Guardar índices
 - Guardar valores
- Costes/hora
 - Guardar índices
 - Guardar valores

- Horas de trabajo
 - Guardar índices
 - Guardar valores

Se va a analizar a continuación el subprograma desarrollado para la consulta de requisitos de cada orden de trabajo, “consulta.vi”.

El procedimiento de este subprograma consiste en abrir el Excel de información relativa a las órdenes de trabajo, y realizar una búsqueda para el indicador de orden que se le introduce como entrada.

Esta búsqueda consiste en analizar los parámetros analizados previamente y guardarlos en la variable adecuada.

En la figura 81 podemos ver la estructura total de programación del subprograma.

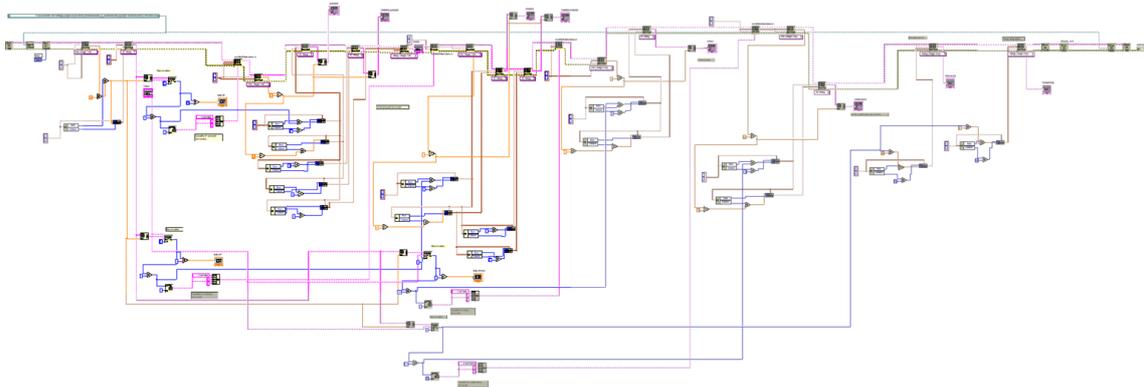


Figura 81: Programación bloque consulta.vi

Para analizar en detalle el procedimiento que realiza el subsistema, se puede ver en la figura 82 un recorte de código de la estructura de programación.

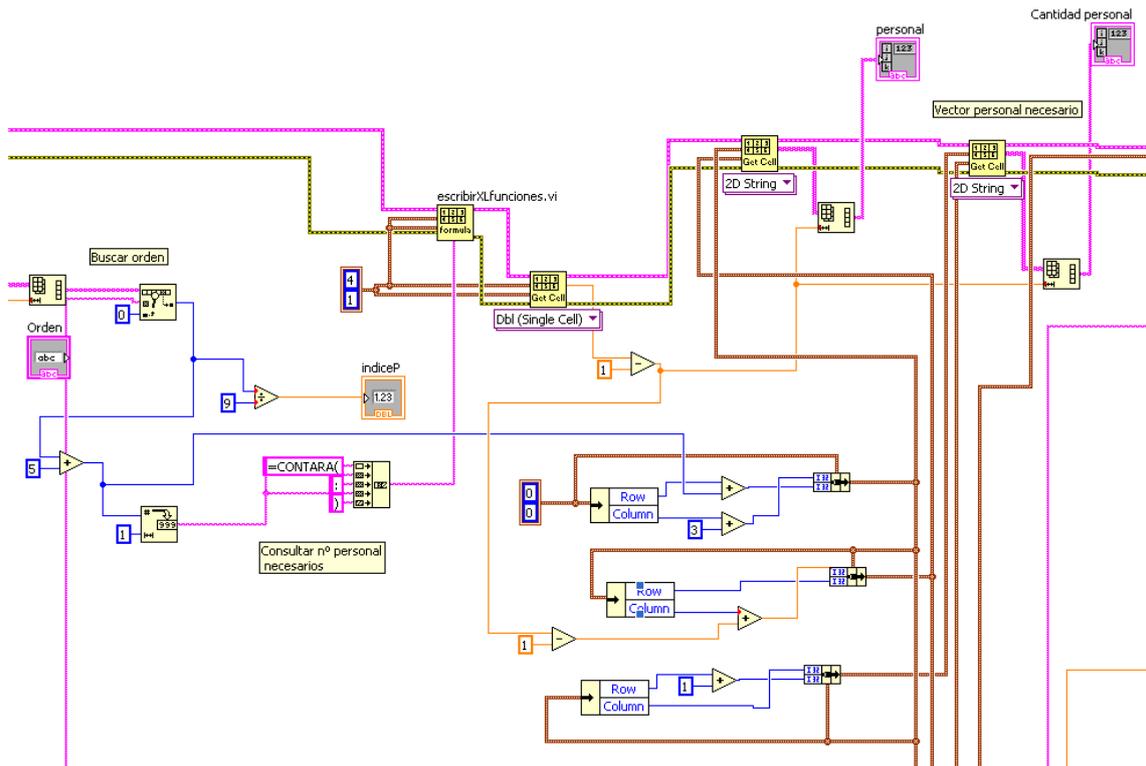


Figura 82: Detalle programación bloque consulta.vi

El procedimiento a realizar es el siguiente:

- A partir del indicador de la orden de trabajo, se busca el índice de esta dentro del archivo Excel
- Se introduce la fórmula que cuenta el número de elementos del vector de Personal necesario
- Se obtiene el resultado de la ejecución de la fórmula en Excel.
- Se utiliza este valor para saber el número de parámetros que se han de guardar
- Se obtiene el vector de Personal necesario
- Se obtiene la cantidad de Personal necesario

El procedimiento a seguir sigue la misma estructura para el resto de parámetros estudiados.

Todos los vectores obtenidos se definen como salidas del bloque de subprograma implementado, y se utilizarán en la estructura general de consulta de requisitos.

6.5.5. Consulta disponibilidad material y equipo

Una vez se ha consultado la necesidad de material y equipo, entre otros parámetros, y se dispone de la información correctamente almacenada en los vectores correspondientes, se pasa a consultar la disponibilidad de material y equipo necesarios.

El programa ha sido diseñado para realizar esta consulta sobre bases de datos implementadas en formato Excel, si bien no se dispone de las bases empleadas por la empresa. Si se trabaja con el mismo formato de archivos la adaptación a las bases de datos adecuada supondría un reajuste de la asignación de celdas utilizadas, pero el procedimiento se mantiene.

Por otro lado, si el trabajo con bases de datos desde Maximo lo permite, se puede habilitar el sistema para que reciba esta información al introducirse una nueva tarea de mantenimiento y se almacene en el archivo de información relativa a la orden.

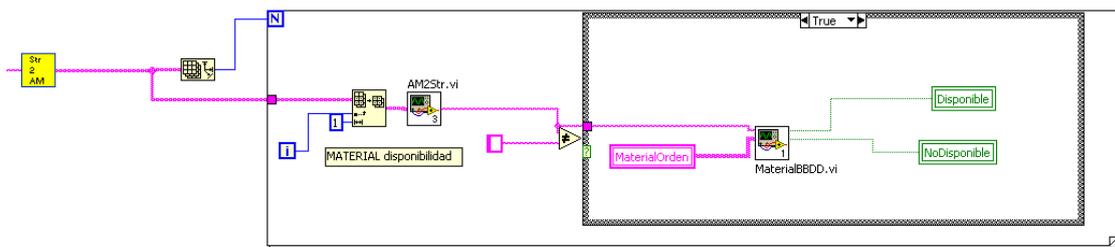


Figura 83: Estructura consulta disponibilidad material

En la figura 83 se puede ver el esquema de programación de este subsistema. En este caso, se activa el proceso cuando existe una marca de color en el estado 5 de la RdPCA.

Se utiliza el subprograma “MaterialBBDD.vi”, desarrollado específicamente para este proceso, para determinar si el material buscado está disponible o no. Esta disponibilidad se almacena en las variables Disponible y NoDisponible, booleanos inicializados a valor falso, que son las encargadas de activar el booleano de la entrada correspondiente para, en la próxima iteración, pasar a la siguiente etapa.

Las entradas al subprograma “MaterialBBDD.vi” son el nombre de la orden con la que se está trabajando, y la matriz de MaterialOrden, con la información relativa al material de todas las órdenes activas.

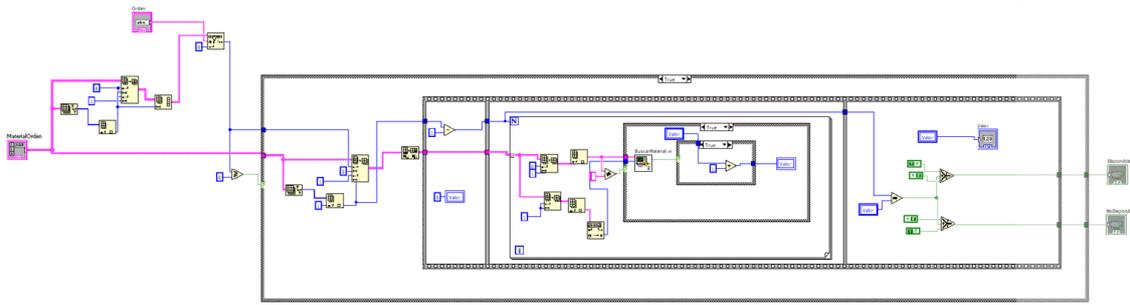


Figura 84: Programación bloque MaterialBDD.vi

En la figura 84 podemos ver la programación interna de este subprograma. A partir de la matriz de información de todas las órdenes, se busca la orden con la que se está trabajando y se extrae su información de la matriz.

Una vez se dispone de la información, se realiza un bucle for que hace la búsqueda de cada uno de los componentes, y la cantidad necesaria.

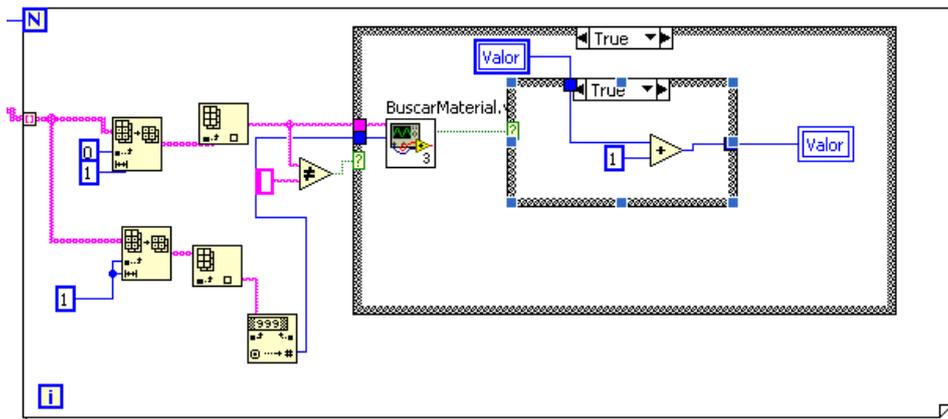


Figura 85: Detalle programación búsqueda de material

El primer paso dentro del bucle consiste en separar la matriz de información de la orden en dos parámetros: Nombre del material y Cantidad necesaria. Una vez obtenidos estos valores, se realiza la llamada al subprograma “BuscarMaterial.vi”.

Este subsistema nos dirá si ese material en concreto está disponible, por lo que sólo queda contar el número de materiales disponibles y compararlo con el número de materiales necesarios. Si todos los materiales están disponibles, se activará la variable de salida Disponible.

El uso de la estructura Case se utiliza como medida de seguridad, para certificar que sólo se realiza la búsqueda de material de órdenes que lo precisan, para evitar validar variables debido a búsquedas incorrectas.

En la figura 86 se puede ver la estructura interna del programa “BuscarMaterial.vi”.

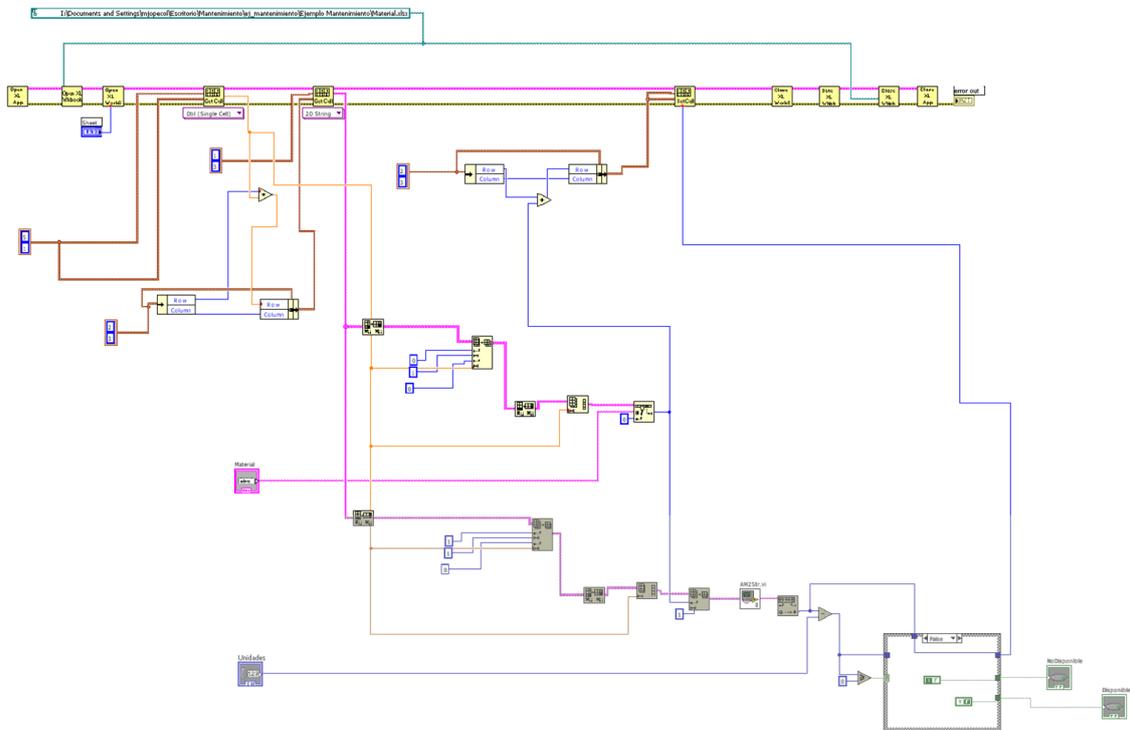


Figura 86: Programación bloque BuscarMaterial.vi

Este subprograma tiene como entradas el nombre del Material y la cantidad necesaria. El funcionamiento consiste en abrir el Excel de la Base de Datos de Material y buscar en la columna de nombres de material el componente que se busca. Una vez disponemos del índice del material buscado, se extrae del archivo la cantidad disponible en la base datos.

Para determinar si el componente está disponible o no, sólo falta comparar el valor necesario con el disponible, y si en la base de datos hay suficiente cantidad de material, se activa la variable Disponible.

El procedimiento para la búsqueda de Personal en la base de datos correspondiente seguiría la misma estructura de programación, teniendo en cuenta que las matrices de entrada serán las equivalentes a Personal y las variables de salida a activar harán referencia a la consulta de Personal.

También es importante utilizar los bloques de programación correspondientes al caso de búsqueda de Personal, que serán:

- “PersonalBBDD.vi”
- “BuscarPersonal.vi”.

Además, este subsistema debe considerar también el tiempo de trabajo que se precisa que el personal a consultar esté disponible.

La estructura de programación de este bloque sería la siguiente:

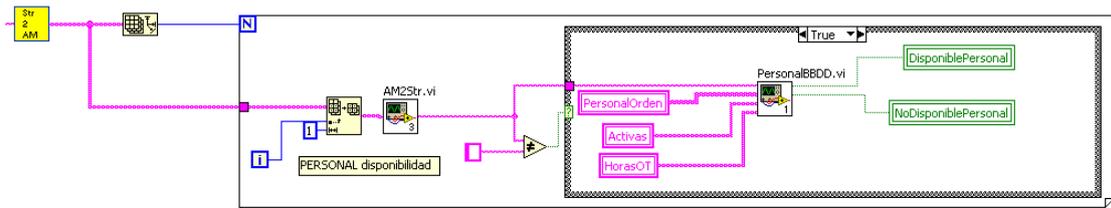


Figura 87: Estructura consulta disponibilidad material

Como se puede ver en la figura 87, además del nombre de la orden que se desea consultar y la información relativa al personal que necesita, se debe utilizar como entradas las matrices de órdenes activas y horas de trabajo de cada orden.

El subprograma “PersonalBBDD.vi” tiene una estructura de programación muy similar al de búsqueda de material, pero con la variación del tiempo de trabajo.

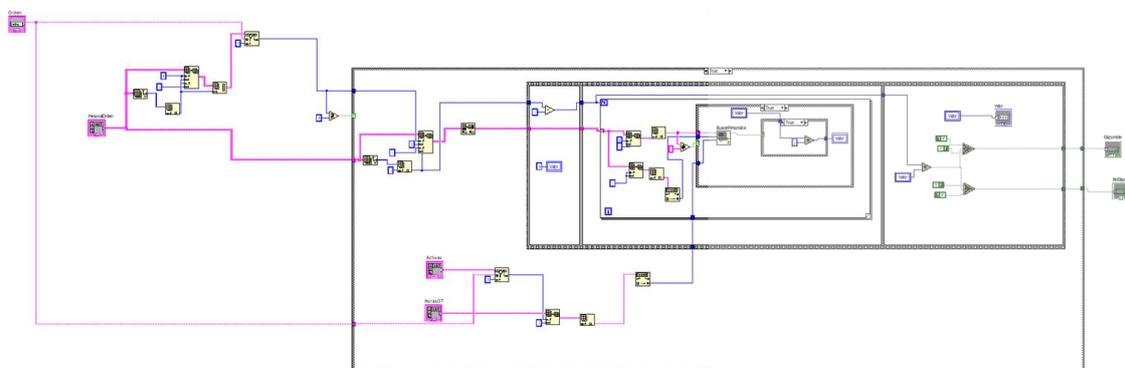


Figura 88: Programación bloque PersonalBBDD.vi

En la parte inferior de la figura 88 se puede ver la estructura de búsqueda de la orden dentro de las órdenes activas, para extraer el valor total de horas necesarias para su ejecución.

Este valor obtenido se introducirá como entrada al subsistema “BuscarPersonal.vi”, que se encargará de verificar su disponibilidad.

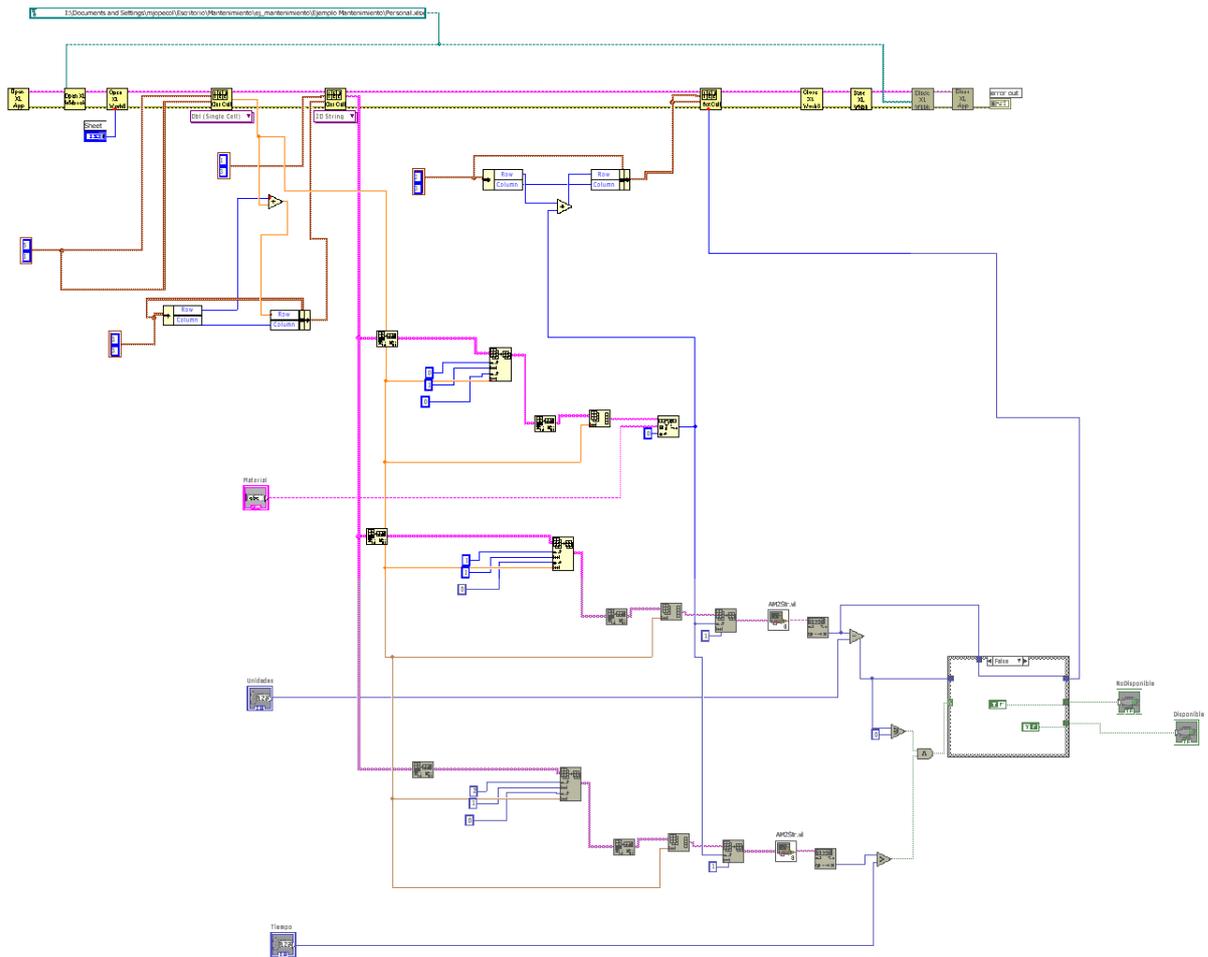


Figura 89: Programación bloque BuscarPersonal.vi

Dentro de la programación interna del bloque “BuscarPersonal.vi” (figura 89), se puede apreciar que en este caso se extraen tres columnas del archivo Excel de base de datos de personal. Estas tres columnas hacen referencia a la categoría del personal, las unidades necesarias y el tiempo disponible.

El funcionamiento sigue el mismo procedimiento que el ejemplo de búsqueda de material, localizando el personal necesario y extrayendo la cantidad y tiempo disponibles.

A través de la comparación de valores disponibles con valores requeridos se determina la disponibilidad del personal.

Tal y como se ha determinado, estas búsquedas en bases de datos son totalmente configurables a la estructura interna de la base de datos a emplear.

Para el programa desarrollado se ha utilizado un archivo Excel con un formato determinado, que se puede adaptar fácilmente a otros formatos.

6.5.6. Eliminación órdenes finalizadas

En numerosas ocasiones se ha hecho referencia a la importancia de actualizar la estructura de definición de la RdPCA al añadir nuevas órdenes, para que los cambios no afecten al correcto funcionamiento del sistema.

Sin embargo, también hay que tener en cuenta que la duración de las distintas tareas dentro de la RdPCA es variable, debido a la agrupación y ejecución de algunas tareas y otras no.

Es por esto que se ha diseñado también un subsistema de eliminación de órdenes que se ejecutará cuando los resultados de la optimización hayan sido aprobados por el operario correspondiente, y se de orden a Maximo de ejecutar las tareas.

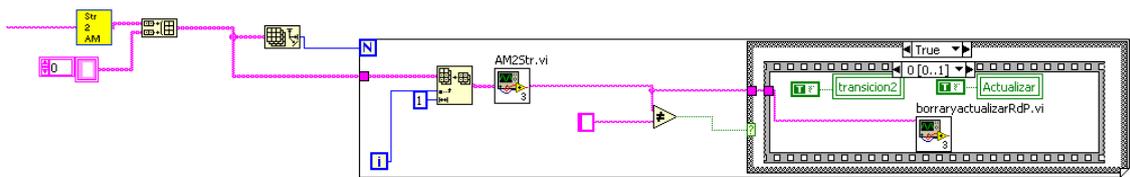


Figura 90: Estructura borrado de órdenes finalizadas

En la figura 90 se puede apreciar la programación requerida para el subsistema de eliminación de órdenes.

De forma similar a otros casos, el primer paso consiste en extraer del lugar de la red la marca identificativa de la orden que se desea eliminar.

A continuación, se trabaja con una estructura en ventanas de secuencia. En la imagen 151 se pueden ver las dos secuencias programadas.

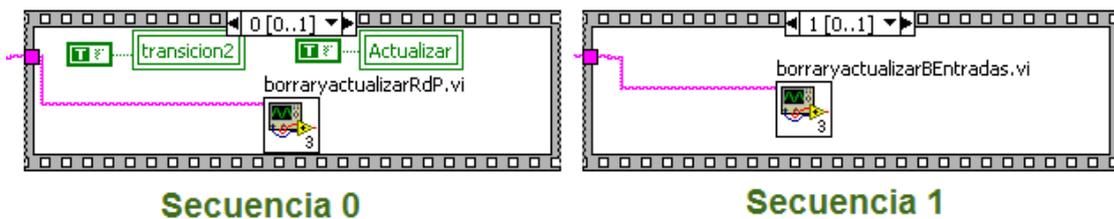


Figura 91: Secuencias de Estructura de borrado de órdenes.

Al igual que en los casos de creación de órdenes nuevas y actualización de variables, se trabaja por separado con la estructura de la red y con el vector de booleanos de valores de las entradas.

En la secuencia 0 se realiza el borrado de la estructura de la red a partir del subprograma “BorraryactualizarRdP.vi”.

En la secuencia 1 se realiza el borrado del vector booleano de entradas a partir del subprograma “BorraryactualizarBEntradas.vi”.

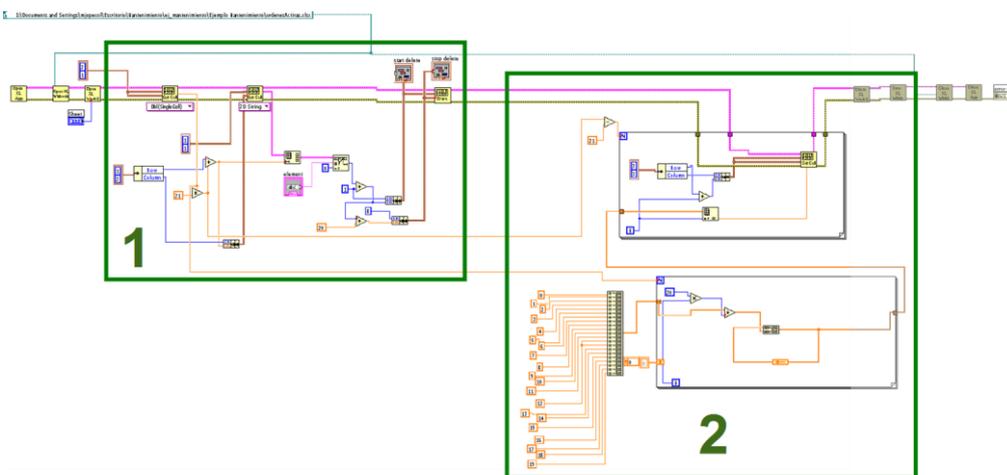


Figura 92: Programación bloque BorraryactualizarRdP.vi y subdivisiones

En la figura 92 se puede ver la programación interna del bloque “BorraryactualizarRdP.vi”. El programa está separado en dos divisiones, 1 y 2.

La división 1 se encarga de buscar la orden a eliminar dentro de la estructura de la RdPCA en Excel, y mediante el índice que la referencia definir las celdas de inicio y fin de borrado, que se realiza mediante el componente “BorrarXL.vi” de la librería de comunicación Excel-Labview.

La división 2 tiene como objetivo actualizar los parámetros de número de transiciones. Esta numeración no sería un problema si se borrara siempre la última red, pero al poder borrar cualquier orden de la estructura, se deben redefinir los parámetros de número de transición, para que el sistema funcione correctamente.

De acuerdo a la secuencia 0, una vez realizado el borrado de la red, se activa la variable “Actualización”, para que el sistema se adapte a la nueva estructura.

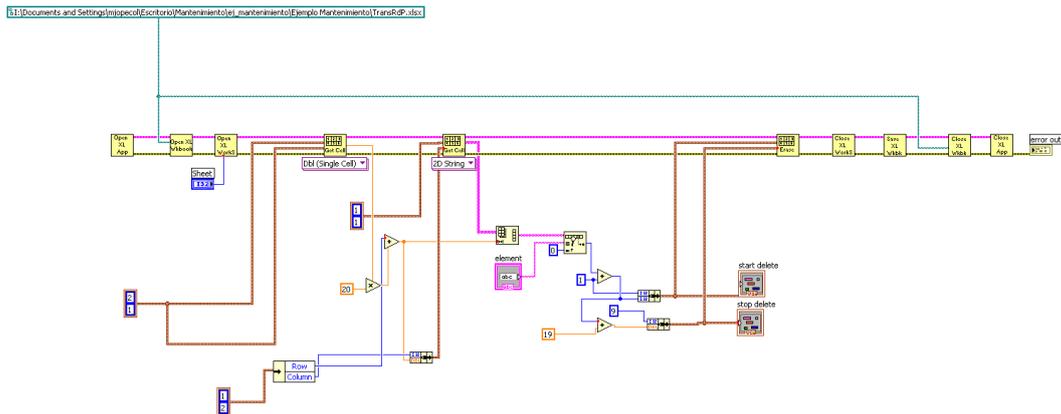


Figura 93: Programación bloque BorraryactualizarBEntradas.vi

Se va a analizar a continuación el programa “BorraryactualizarBEntradas.vi”.

En este caso, se busca el indicador de la orden en el archivo que contiene los valores booleanos de las entradas.

A partir del índice que referencia la orden dentro del archivo, se configuran las celdas de inicio y fin de borrado, y se procede a eliminar los valores de las entradas.

Para el caso de las entradas, al no haber ningún tipo de numeración no es necesaria la parte de actualización analizada en el ejemplo anterior.

6.6. Subsistema de Meteorología

El subsistema que se presenta es el encargado de determinar si la predicción meteorológica es adecuada para la ejecución de las tareas.

Para ello, cada tarea lleva asociada en su archivo de información de parámetros de la tarea, los valores meteorológicos adecuados para su ejecución. De este modo, se realiza una comparación entre los valores máximos permitidos por la tarea y los valores meteorológicos de la predicción para determinar si es el momento adecuado para llevar a cabo el mantenimiento o si, por el contrario, la predicción meteorológica prevé unas condiciones inapropiadas.

De acuerdo a los parámetros establecidos para el proyecto, el sistema de mantenimiento no genera la predicción meteorológica, que se realizará de forma externa. Es por esto que se ha preparado la aplicación especialmente para dicho fin, permitiendo que se introduzca la previsión meteorológica de los tres posteriores días para cada uno de los parámetros empleados, y trabajando con los parámetros introducidos.

La introducción y actualización de la previsión meteorológica se analizará en detalle a partir del entorno gráfico de usuario, en el apartado 7 de la presente tesina.

Siguiendo con la comparación de la previsión meteorológica, se va a analizar su funcionamiento.

Esta comprobación de parámetros debe realizarse en dos estados distintos de la Red de Petri Coloreada Adaptativa que define la estructura de funcionamiento del sistema.

En primer lugar, debe utilizarse este subsistema en el estado de órdenes en espera. La orden de trabajo llega a este estado tras comprobar las necesidades en cuanto a personal y material y su disponibilidad en caso necesario, y una vez que dichos requisitos han sido cumplidos. Es en este punto donde se determina la meteorología, con el fin de que únicamente las órdenes que pueden ser ejecutadas con una previsión a tres días pasen al subsistema de agrupación.

Cuando se realiza la consulta de requerimientos de cada orden de trabajo a partir de su archivo de información, se almacenan los parámetros de cada orden en el vector "IndicesMeteo", que se empleará para la comprobación.

En la figura 94 se puede ver la estructura general de programación del bloque de comprobación meteorológica.

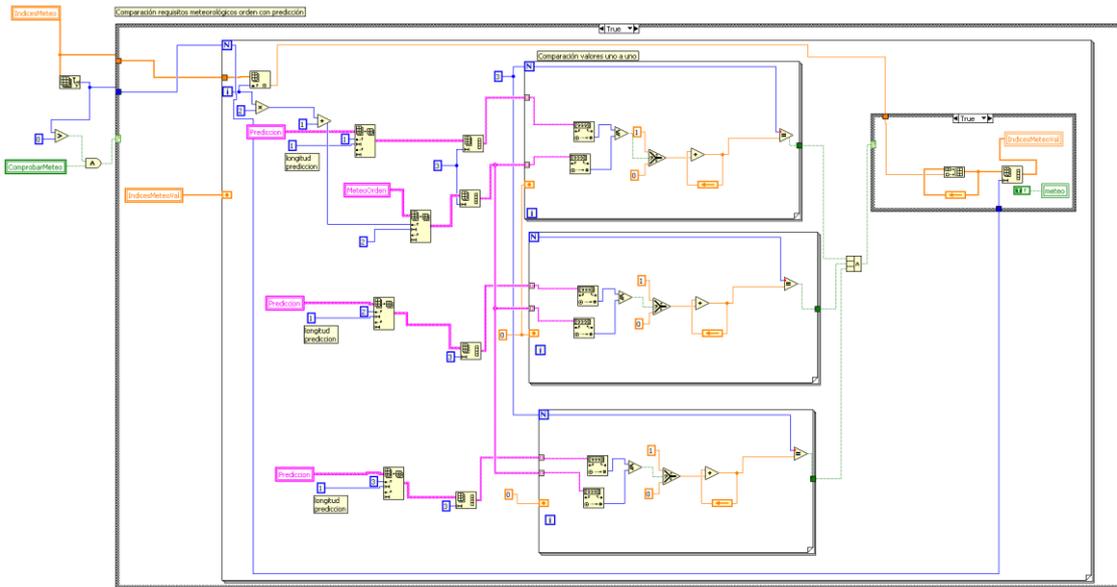


Figura 94: Bloque de programación del subsistema de Meteorología

La utilización de este subsistema viene marcada por la activación de la variable “ComprobarMeteo”, activada desde la Red de Petri cuando las distintas tareas llegan al estado de espera.

Como se puede ver en la figura 94, existen tres estructuras trabajando en paralelo, que se corresponden con la comprobación de los parámetros para cada uno de los tres días de horizonte de predicción.

En la figura 95 se puede ver el detalle de la comprobación meteorológica para uno de los días. A partir de esta figura se explicará el funcionamiento de dicha comprobación.

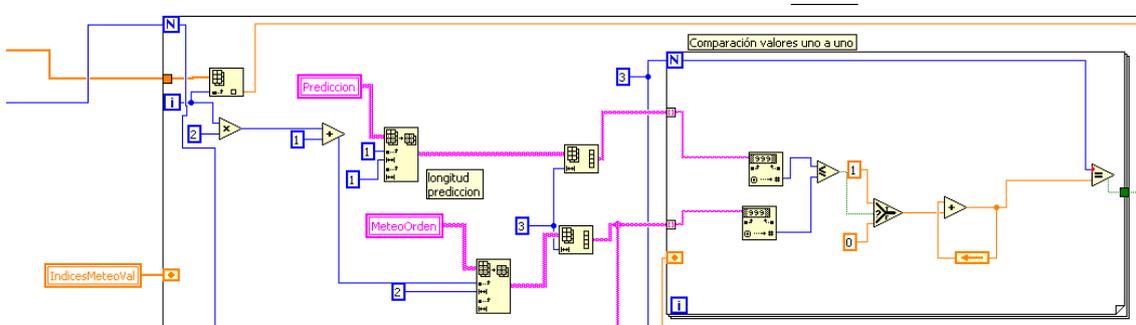


Figura 95: Detalle de comprobación meteorológica

El funcionamiento del subsistema consiste en comparar uno a uno los valores de la previsión meteorológica con los parámetros almacenados en el vector de meteorología de cada orden de trabajo.

El sistema está preparado para realizar tantas consultas como parámetros se hayan declarado en la predicción, por lo que si se desean añadir más términos meteorológicos a la comparación, tan sólo es necesario añadirlos en el vector de predicciones y en la información de las tareas.

Una vez realizadas las comparaciones de los parámetros, si se cumplen todos los requisitos establecidos por la orden de trabajo, se asigna un valor positivo a la comparación para ese día.

Como ya se ha dicho, se realiza la comparación en paralelo para los tres días que forman el horizonte de predicción, con sus valores correspondientes. Si la salida de las tres comparaciones es positiva, la orden de trabajo puede ser ejecutada en ese espacio temporal establecido, por lo que se activa el paso de la tarea analizada al subsistema de agrupación.

La comparación de parámetros meteorológicos se realiza de una vez para todas las tareas que se encuentran en ese momento en el estado de espera, pasando al estado de agrupación de tareas tan sólo aquellas que cumplen los requisitos y quedando en el estado de espera las que no puedan ser ejecutadas.

La segunda utilización del subsistema de meteorología se realiza en el estado de revisión. Una vez se han calculado las agrupaciones y se ha realizado la optimización, seleccionando las tareas que se proponen como mejor solución de ejecución del mantenimiento, dichas tareas llegan al estado de revisión. El objetivo de este estado es comprobar que las condiciones de ejecución siguen siendo favorables para todas las tareas.

Esta revisión es importante ya que se trata de un sistema en ejecución continua, pero no inmediata, por lo que desde que se realiza la primera comprobación hasta la ejecución de la tarea pueden haber cambios en las predicciones meteorológicas que determinen una previsión desfavorable para la ejecución de alguna de las tareas.

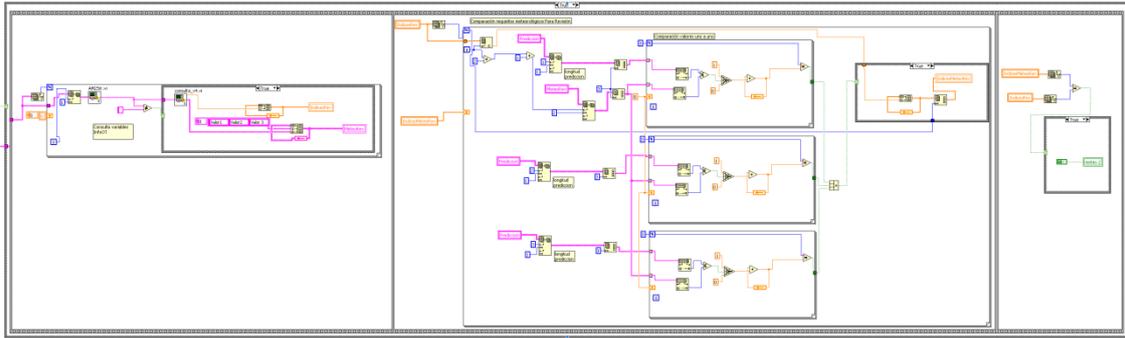


Figura 96: Estructura programación comprobación meteorológica en Revisión

Como se puede ver en la figura 96, este subsistema de comprobación meteorológica es distinto al anterior, incorporando una fase previa y otra posterior, que se analizan a continuación.

Se trata de una estructura de secuencias, que como ya se ha visto en casos anteriores, se utiliza para garantizar que el procedimiento sigue el orden adecuado y evitar pérdidas de variables o utilización de parámetros antes de ser actualizados. Se van a analizar las tres secuencias que forman este subsistema de forma separada.

En primer lugar, la secuencia primera se encarga de crear una nueva matriz de requisitos meteorológicos, ya que entre las dos comprobaciones pueden haber sido añadidas nuevas órdenes y de este modo se asegura que se trabaja exclusivamente con las órdenes de trabajo que han llegado al estado de revisión.

Para cada una de ellas, se utiliza de nuevo el bloque "consulta_v4.vi", que como se analizó en el informe anterior se emplea para extraer la información de la tarea del archivo Excel que contiene los parámetros de cada una de las tareas existentes en el sistema.

Con esta nueva consulta, se crea la matriz "MeteoRev", que se empleará en la secuencia posterior para la comparación de valores meteorológicos.

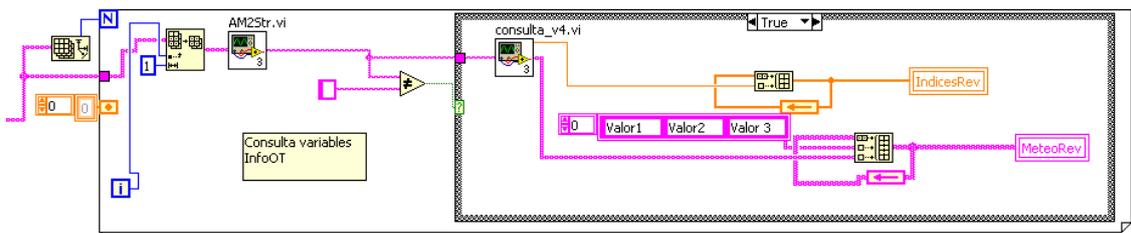


Figura 97: Primera secuencia comprobación meteorológica en Revisión

Una vez se dispone de las matrices de órdenes a comparar y sus valores máximos para cada parámetro meteorológico, se pasa a la siguiente secuencia de la estructura, cuya programación se puede ver en la figura 98.

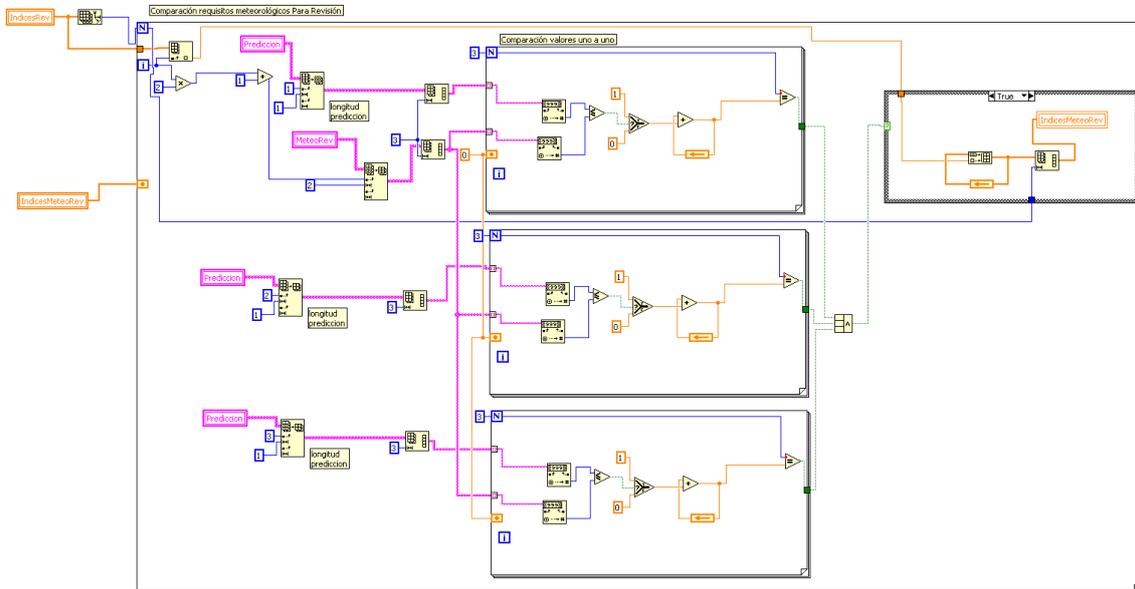


Figura 98: Segunda secuencia comprobación meteorológica en Revisión

En la segunda secuencia se realiza la comprobación de los valores de cada orden de trabajo con las previsiones meteorológicas con horizonte de predicción a tres días, al igual que en el caso de la comprobación en el estado de espera.

La diferencia respecto al subsistema presentado anteriormente se basa en las matrices que se emplean para la comprobación de parámetros.

La matriz de Predicción continúa siendo la misma, con sus correspondientes actualizaciones. Sin embargo, la matriz de valores de las tareas que se utilizará será la calculada en la primera secuencia, no la inicial obtenida de la consulta de requisitos.

Al igual que en el caso anterior, los órdenes que cumplen con todos los requisitos para los próximos tres días, se dan por validados, y sus indicadores se almacenan y pasan a la tercera secuencia.

Al tratarse de una fase de revisión, y estar trabajando con los resultados de la optimización, si una de las órdenes de trabajo no cumple con los requisitos meteorológicos, se anularía toda la agrupación y optimización, y las tareas volverían a la fase de espera.

Es por esto que se ha añadido la tercera secuencia a la estructura, que es la encargada de comprobar que todas las órdenes que han llegado al estado de revisión cumplen los requisitos.

Como se ha podido ver a lo largo del apartado, se ha analizado en detalle la programación requerida para los subsistemas de consulta de parámetros meteorológicos. En el apartado de análisis del entorno de usuario se mostrará cómo se crea la matriz de previsión meteorológica que se emplea en los dos subsistemas presentados en este apartado.

6.7. Subsistema de Agrupación de Tareas

La aplicación diseñada para este subsistema de agrupación de órdenes de mantenimiento parte de la base de que las posibles embarcaciones y sus parámetros son conocidos por la empresa de mantenimiento, en base al contrato establecido con la naviera correspondiente.

El funcionamiento, por lo tanto, consiste en analizar una por una las tareas que llegan a la fase de agrupación, tratar de agruparlas en las distintas embarcaciones disponibles y calcular los parámetros necesarios para la posterior fase de decisión multiobjetivo.

De este modo, una vez analizadas todas las tareas susceptibles de agrupación, se dispondrá de 4 posibles soluciones de mantenimiento, correspondientes a las 4 embarcaciones empleadas para este caso. Se ha diseñado el sistema de tal manera que la incorporación de más embarcaciones al sistema sólo supone ampliar ligeramente la estructura, con los nuevos parámetros de los barcos.

En la imagen 99 podemos ver la estructura esquemática de este procedimiento, determinando 4 posibles embarcaciones. Sus capacidades totales están basadas en datos reales de embarcaciones empleadas para trabajos en alta mar, y se incluye también la posibilidad de emplear helicóptero para tareas que no requieran material.

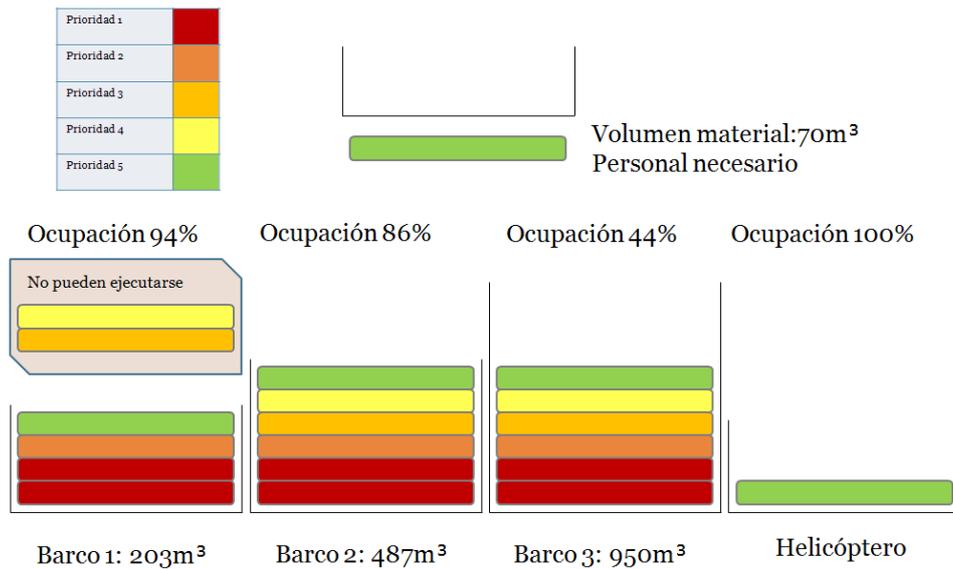


Figura 99: Estructura esquemática del subsistema de agrupación de tareas

Se va a analizar a continuación la estructura de programación del subsistema de agrupación de tareas.

El primer paso consiste en almacenar en la matriz "Agrupación" los órdenes de trabajo que han llegado a la fase de agrupación, para trabajar sólo con ellas, excluyendo al resto de tareas existentes dentro del sistema. Una vez obtenida la matriz, se activa la variable que inicia el subsistema de agrupación de tareas.

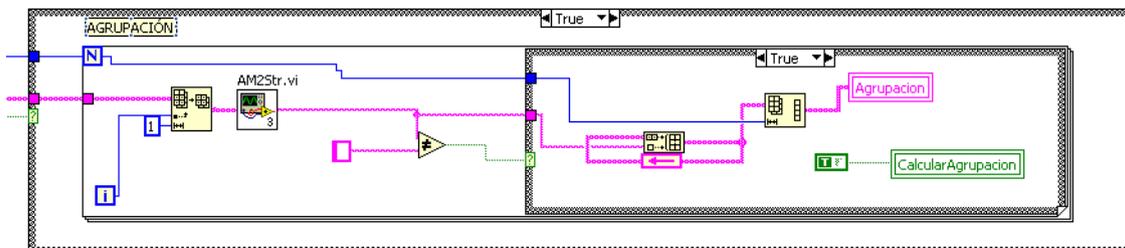


Figura 100: Inicialización subsistema de Agrupación

El subsistema de agrupación de tareas (figura 100) trabaja de nuevo con una estructura de secuencias. En este caso, la primera secuencia consiste en inicializar las variables correspondientes a cada una de las embarcaciones.

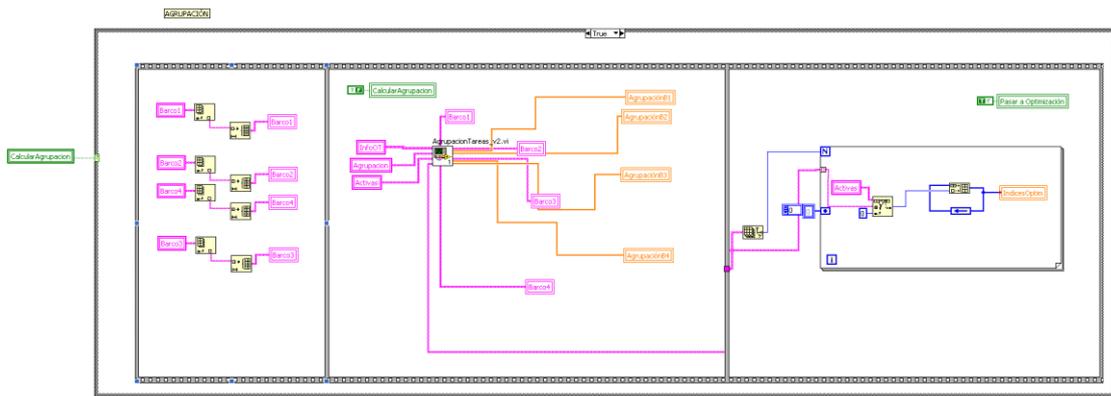


Figura 101: Subsistema Agrupación de tareas en opciones

En la segunda secuencia, se emplea el bloque “Agrupación_Tareas.vi”, cuya programación interna distribuye las órdenes en las diversas opciones, y calcula los parámetros necesarios para el sistema de optimización.

Finalmente, la última secuencia se encarga de habilitar el paso al siguiente lugar de la red, correspondiente a la fase de optimización, de todas aquellas tareas que han sido agrupadas en alguna de las posibles opciones, mientras que las que no se puedan agrupar por no cumplir los requisitos permanecen en el lugar actual.

Se va a analizar ahora el bloque “Agrupación_Tareas.vi”, diseñado específicamente para la agrupación en opciones de las tareas.

Este bloque trabaja con secuencias. En primer lugar, se realiza una inicialización de variables, para asegurarnos que cada vez que se emplea el subsistema se reinician los valores y evitar así fallos de cálculo. También en esta inicialización se asigna a los barcos la capacidad en cuanto a personal y material, pudiendo así ser modificada si fuera necesario.

En la segunda secuencia se realiza todo el cálculo de la agrupación en las distintas opciones disponibles. El primer paso de esta secuencia se basa en obtener una lista de tareas a agrupar ordenada por prioridades. Para esto se trabaja con los parámetros de información de cada orden, que serán necesarios para realizar la agrupación, y se reorganizan las columnas de mayor a menor prioridad.

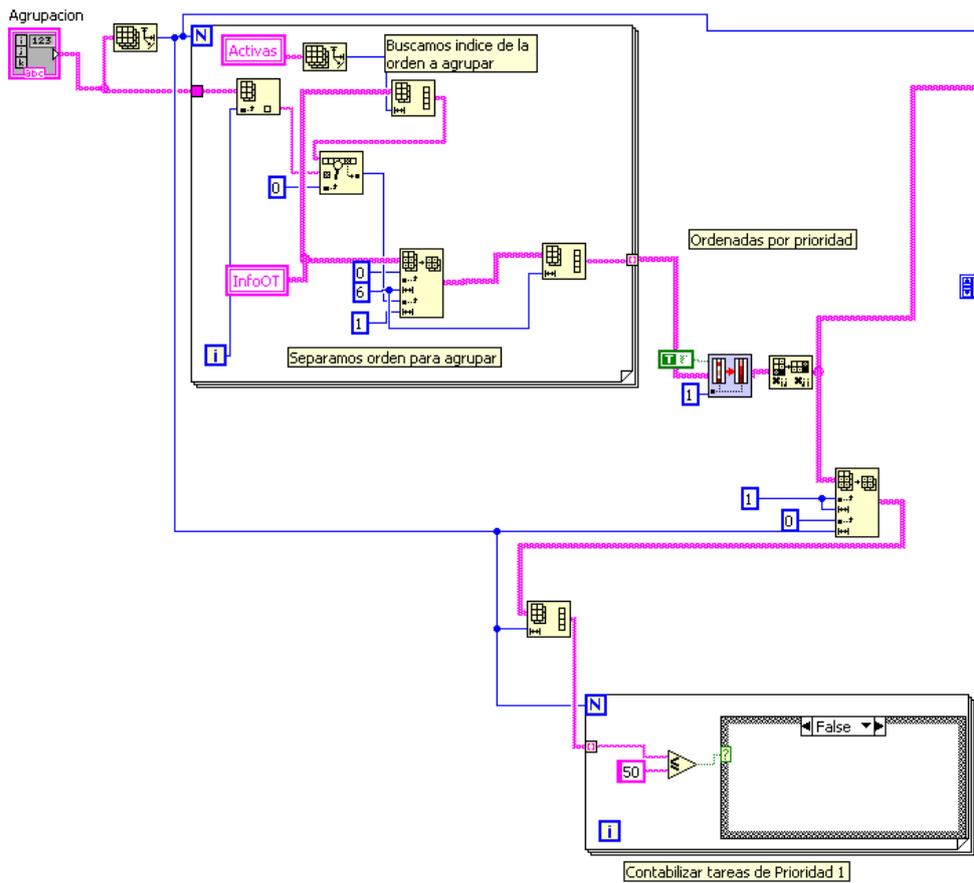


Figura 102: Reorganización de parámetros en función de la prioridad

Además, se contabiliza el número de tareas de prioridad alta con las que se va a trabajar en este cálculo de la agrupación. Este valor se empleará para calcular los parámetros de cada agrupación que hacen referencia a porcentajes.

Una vez se dispone de la información correctamente ordenada por valor de prioridad, se procede a realizar las agrupaciones.

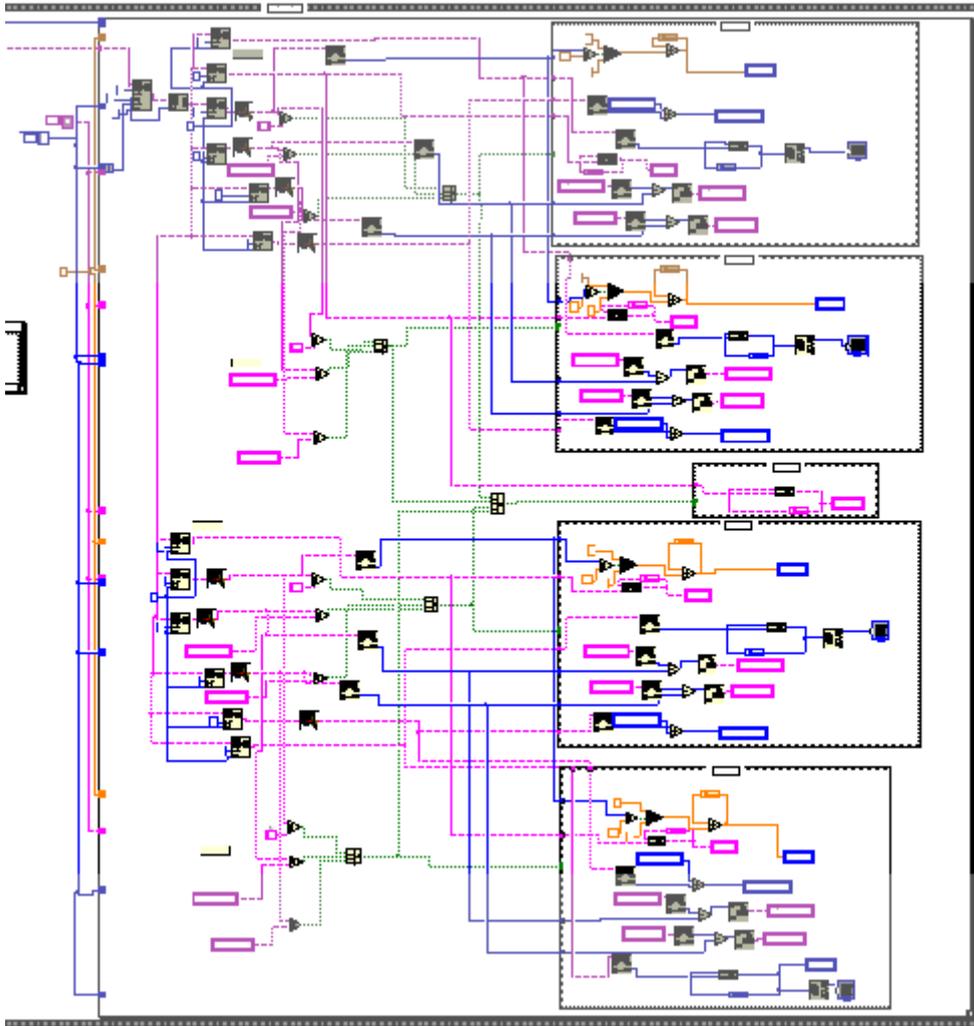


Figura 103: Cálculo de agrupaciones para las 4 embarcaciones

El procedimiento empleado para la agrupación es el siguiente:

- Se analizan las órdenes una a una, siguiendo el orden establecido de acuerdo a las prioridades
- De cada orden, se extrae los parámetros de prioridad, volumen de material, cantidad de personal, horas de trabajo y coste por hora.
- Se compara el valor de estos parámetros con las limitaciones de cada una de las 4 embarcaciones, y en caso de ser posible, se acepta para la agrupación en esa embarcación.
- Si se acepta en la agrupación, se introducen los valores de la orden para el cálculo de parámetros de la agrupación

Una vez analizadas todas las ordenes de trabajo, se pasa a la tercera secuencia del sistema.

En la tercera secuencia se emplean los valores obtenidos para cada embarcación para calcular los parámetros de cada agrupación que se precisan en el sistema de optimización para compararlas entre sí.

Estos parámetros son:

- Coste total de la agrupación
- Porcentaje de órdenes de prioridad alta que se incluyen en la agrupación
- Porcentaje de ocupación de la embarcación correspondiente
- Valor más restrictivo de prioridad de las órdenes no agrupadas

Al finalizar la ejecución de este subsistema, se dispone de 4 embarcaciones con las tareas agrupadas en cada una de ellas, y de 4 vectores de agrupación que incluyen los valores mencionados arriba.

Las órdenes que han sido agrupadas en alguna de las opciones pasan al estado de optimización, en el que se utilizarán los parámetros calculados para cada agrupación para determinar la mejor solución entre las opciones que se proponen.

6.8. Subsistema de Decisión Multiobjetivo

El subsistema de decisión multiobjetivo es uno de los más importantes a desarrollar en la tesina, junto con el sistema de agrupación de tareas. Estos dos sistemas aportan una novedad a los programas de gestión de mantenimiento comúnmente utilizados, ya que permiten calcular distintas agrupaciones en función de las embarcaciones y las tareas activas, y mediante la decisión multiobjetivo determinar cuál de estas agrupaciones resulta más interesante para el mantenimiento en función de los parámetros analizados.

Es por esto que se estudiaron diversas metodologías de optimización multiobjetivo, ya que son varios los parámetros a optimizar y utilizar tan sólo el coste como objetivo a reducir sería demasiado restrictivo, se deseaban considerar todos los parámetros.

Como resultado, el programa utiliza dos metodologías de decisión multiobjetivo, que se emplean de forma paralela:

- Basada en restricciones
- Por interfaz de preferencias

Se va a analizar a continuación la programación Labview necesaria para la implementación del subsistema de optimización multiobjetivo.

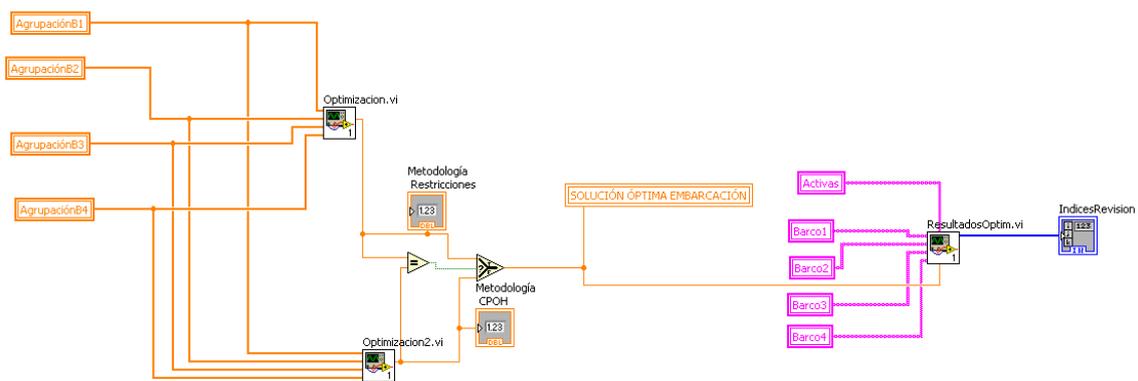


Figura 104: Subsistema de Optimización Multiobjetivo

Este subsistema utiliza como parámetros de entrada los valores calculados para las distintas agrupaciones, y tal y como se puede ver en la figura 104 emplea dos subprogramas en paralelo para el cálculo de la decisión multiobjetivo. Estos programas son “Optimizacion.vi” y “Decisión multiobjetivo2.vi”.

El primero de ellos hace referencia a la optimización basada en restricciones, y el segundo a la basada en interfaz de preferencias. Cada metodología dará como resultado un número de agrupación óptima. En caso de que los resultados difieran, se ha empleado como preferente la metodología basada en interfaz de preferencias, o CPOH.

Los programas “Optimizacion.vi” y “Optimización2.vi” emplean de forma interna scripts de Matlab, lo que permite emplear la tecnología desarrollada en dicho programa en las fases previas al desarrollo en Labview.

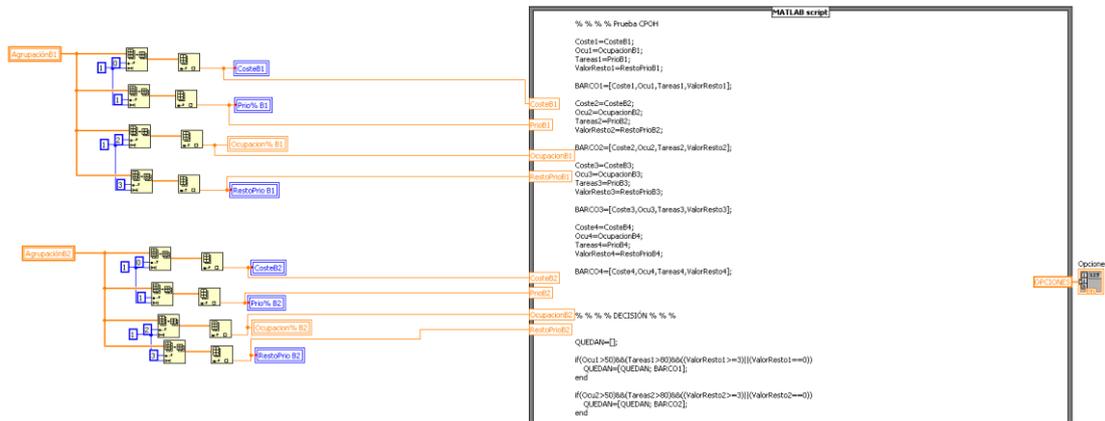


Figura 105: Detalle programación interna Optimización.vi

En la figura 105 podemos ver un detalle de la programación interna desarrollada para el subprograma “Optimización.vi”. Tal y como se puede apreciar, se extraen los valores de cada parámetro de la agrupación, y se introducen como entradas al script de Matlab, que tras calcular los términos necesarios determina cuál de las agrupaciones es más conveniente, y muestra de forma gráfica los resultados.

Una vez se dispone de un valor de embarcación óptima, se emplea el subprograma “ResultadosOptim.vi” para analizar las órdenes de trabajo agrupadas en dicha embarcación, y activar sus índices correspondientes, con el objetivo de permitir las órdenes agrupadas pasen a la fase de revisión.

Los resultados obtenidos mediante el programa de optimización multiobjetivo serán empleados posteriormente para la creación de informes para revisión del usuario, así como para crear el informe que ha de ser enviado a Maximo en caso de ser aceptada la decisión multiobjetivo.

6.9. Presentación de resultados para Aceptación de experto

Tras la fase de revisión, en la que se comprueba que los parámetros meteorológicos continúan siendo favorables para la ejecución del mantenimiento, las tareas agrupadas en la solución óptima se presentan al usuario, con el fin de que estudie las agrupaciones calculadas y la decisión multiobjetivo y determine si se acepta o rechaza la solución propuesta.

La programación necesaria para este informe por pantalla se puede ver en la figura 106.

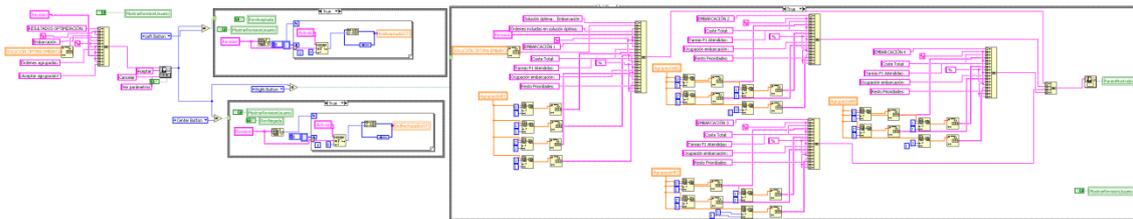


Figura 106: Creación de Informe por pantalla de usuario

De forma gráfica, a través de la pantalla del usuario se puede ver la siguiente información:

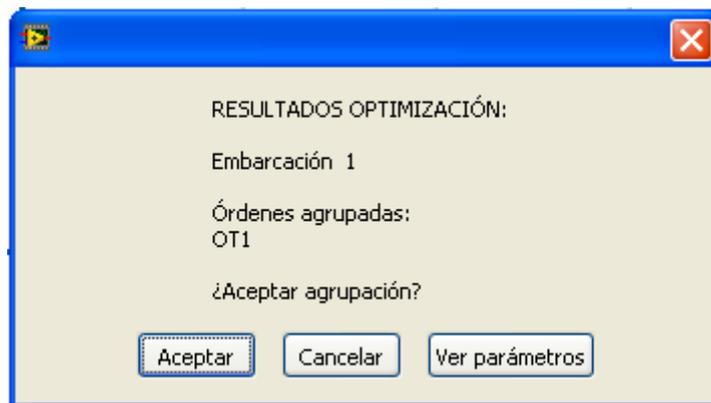


Figura 107: Informe por pantalla

Como se puede ver en la figura 108, el informe básico que se ofrece por pantalla presenta la embarcación óptima resultado de la optimización multiobjetivo, y las órdenes de trabajo incluidas en la agrupación.

Como opciones, permite al usuario determinar lo que se desea realizar con la agrupación propuesta: Aceptarla, cancelarla o ver los parámetros para estudiar los resultados en mayor profundidad.

En caso de aceptar la revisión, órdenes incluidas y los parámetros calculados pasan a la última fase del programa, donde se creará el informe para máximo que se estudiará en el siguiente apartado.

En caso de rechazar la agrupación, todas las tareas incluidas en esa opción volverán a la fase de agrupación, para iniciar el proceso de nuevo, incluyendo las nuevas órdenes de trabajo que hayan podido llegar a esta etapa durante el proceso.

La tercera opción permite al usuario estudiar los resultados de la agrupación y la decisión multiobjetivo.

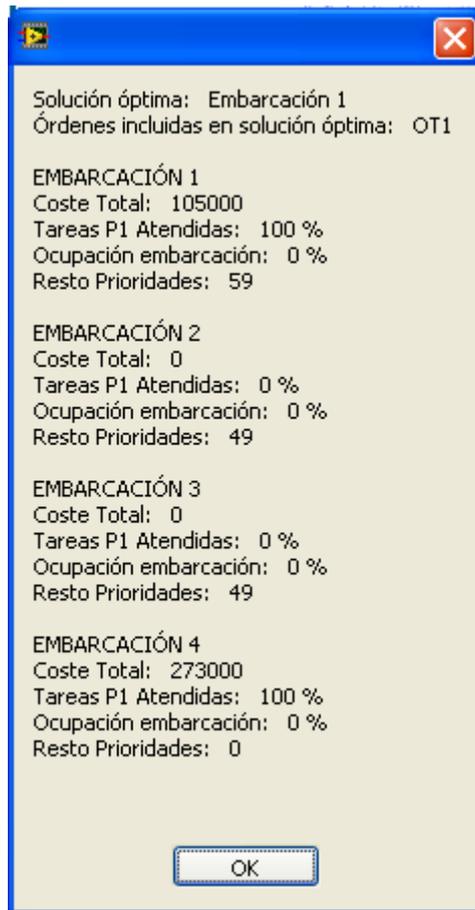


Figura 109: Opción "Ver Parámetros"

En este caso, la información que se muestra por pantalla incluye la embarcación óptima y las órdenes agrupadas, al igual que en el caso anterior, y además incluye los parámetros calculados para cada una de las agrupaciones propuestas, con el objetivo de que el experto determine si la decisión multiobjetivo es correcta, o en base a su experiencia considera que no es suficiente o alguna de las otras opciones puede ser recomendable también.

Los parámetros que se muestran de cada agrupación son los siguientes:

- Coste total de la agrupación
- Porcentaje de órdenes de prioridad alta que se incluyen en la agrupación
- Porcentaje de ocupación de la embarcación correspondiente
- Valor más restrictivo de prioridad de las órdenes no agrupadas

6.10. Creación de Informe para Maximo

El último estado de cada tarea dentro del sistema de gestión del mantenimiento consiste en crear el informe de los resultados obtenidos de la agrupación y decisión multiobjetivo.

Este informe se prepara en formato PDF, y se puede configurar su uso para incorporarlo como archivo adjunto a un e-mail, que se enviaría directamente al responsable de Maximo, cuando el programa esté configurado para enviar y recibir mails.

La programación necesaria para la creación del informe se puede ver en la figura 110.

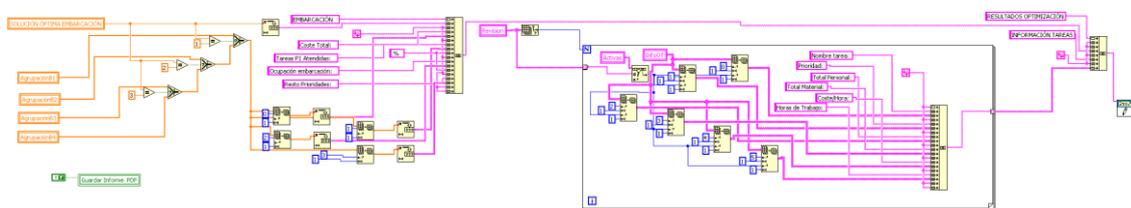


Figura 110: Creación informe Maximo

Una vez se ha generado el informe y se ha almacenado y enviado correctamente, la orden de trabajo debe ser eliminada completamente del sistema, borrando sus datos de los siguientes archivos Excel:

- OrdenesActivas
- TransRdP

Estos archivos incluyen toda la estructura de la Red de Petri Coloreada Adaptativa para cada orden de trabajo y el vector de valores de entradas.

Es por esto que la eliminación de una orden de trabajo específica supone una reorganización de los archivos que debe realizarse de forma automática, y toda la estructura de la red debe adecuarse a las nuevas dimensiones de trabajo mientras continua la ejecución del programa, sin afectar al resto de órdenes existentes en el momento del borrado.

La información del informe debe contener los siguientes parámetros:

- Resultados de Decisión multiobjetivo
- Embarcación seleccionada
- Coste total del mantenimiento

- Tareas de prioridad alta atendidas
- Ocupación de la embarcación
- Resto de prioridades
- Información de cada tarea agrupada
 - Nombre orden de trabajo
 - Prioridad
 - Total Personal
 - Total volumen material
 - Coste/hora
 - Total horas de trabajo

7. INTERFAZ DE USUARIO GMAO

En el presente apartado se va a analizar el interfaz gráfico desarrollado para la utilización del programa por parte del usuario experto.

Todo programa de gestión dispone de una programación interna de gran complejidad, que debe quedar oculta al usuario y mostrar tan sólo los campos de interés para la interacción con el programa y la representación de resultados.

Es por esto que se ha desarrollado un interfaz gráfico específico para la aplicación, con el fin de mostrar por pantalla la información relevante y el estado de las órdenes de trabajo y habilitar sólo los controles que puedan ser modificados por el experto, dejando de forma oculta y restringida el resto de parámetros y modificaciones.

En la figura 111 se puede ver el interfaz de usuario implementado.



Figura 111: Interfaz de usuario

Se va a analizar el interfaz implementado desde dos puntos de vista.

En primer lugar, se analizará la distribución estructural, diferenciando tres partes:

- Red de Petri Coloreada Adaptativa
- Información
- Botones de control

Por otro lado, se presentará la división de la pantalla por sectores, correspondientes a diversos subsistemas y representación de la información. Estos sectores son:

- Parámetros relativos a Personal y Material
- Predicción meteorológica
- Información de funcionamiento
- Agrupación
- Decisión multiobjetivo
- Envío de informes a Maximo

7.1. Distribución estructural

En este apartado se va a analizar la distribución de componentes a lo largo del interfaz de usuario, desde el punto de vista estructural.

Como se ha mencionado anteriormente, los tres campos a estudiar en este aspecto son:

- Red de Petri Coloreada Adaptativa (1)
- Información (2)
- Botones de control (3)

En la imagen 112 se puede ver la distribución estructural del interfaz de usuario, con la numeración correspondiente a cada sector.



Figura 112: Distribución estructural interfaz

En la subdivisión 1, que hace referencia a los estados de la red de Petri Coloreada Adaptativa, se pueden ver los diversos estados correspondientes a los lugares de la Red de Petri diseñada para la tesina y analizada en los apartados anteriores.

<p>ÓRDENES EN CONSULTA DE REQUISITOS</p> <p>OT1,OT5</p>	<p>ÓRDENES EN CONSULTA DE REQUISITOS</p> <p>OT1,OT5</p>
<p>CONSULTA DISPONIBILIDAD DE PERSONAL</p>	<p>CONSULTA DISPONIBILIDAD DE MATERIAL</p>
<p>ÓRDENES QUE PRECISAN SOLICITAR PERSONAL</p>	<p>ÓRDENES QUE PRECISAN SOLICITAR MATERIAL</p>

<p>ORDENES EN ESPERA</p> <p>REQUISITOS CUMPLIDOS</p>	<p>ESPERAR ACEPTACIÓN EXPERTO</p>
<p>AGRUPACIÓN</p>	<p>AGRUPACIÓN ACEPTADA</p> <p>PENDIENTE DE ENVÍO A MAXIMO</p>
<p>OPTIMIZACIÓN</p>	<p>Guardar Informe PDF</p> <p>OFF</p>
<p>REVISIÓN METEOROLÓGICA</p>	

Figura 113: Subdivisión Red de Petri

Como se puede ver en la figura, los recuadros representan los distintos lugares de la Red de Petri Coloreada. En este caso, se puede ver cómo dos órdenes de trabajo, OT1 y OT5, se encuentran en el estado de consulta de requisitos.

A pesar de dotar al programa de un interfaz con aspecto de programa de gestión, se ha querido mantener la estructura de una Red de Petri, para conocer el estado de las órdenes de trabajo en cada instante a primera vista.

De este modo se conoce perfectamente la situación de cada orden de trabajo, independientemente del subproceso en que se encuentre cada una de ellas y permitiendo mayor información al usuario.

Respecto a la subdivisión número 2, se trata de la parte gráfica de representación de parámetros internos de funcionamiento.

En la figura 114 se puede apreciar con más detalle los parámetros de información correspondientes a esta subdivisión.

PREDICCIÓN METEOROLÓGICA



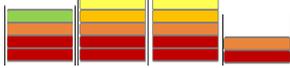
	Parámetro 1	Parámetro 2	Parámetro 3
Primer Día	12	10	4
Segundo Día	14	10	5
Tercer Día	13	12	6

INFORMACIÓN DE FUNCIONAMIENTO

Información Tareas Activas		
Nombre	OT1	OT5
Prioridad	49.000000	59.000000
Material	0	0
Personal	0	0
Horas	500	800
Coste/Hora	210	45

Meteorología Órdenes			
OT1	Parámetro1	Parámetro2	Parámetro3
2	2	2	2
OT5	Parámetro1	Parámetro2	Parámetro3
2	2	2	2

AGRUPACIÓN



OPTIMIZACIÓN

Metodología Restricciones	Metodología CPOH
0	1

SOLUCIÓN ÓPTIMA EMBARCACIÓN
 0

Figura 114: Subdivisión Información

Dentro de los parámetros informativos del sistema, se consideran los siguientes conceptos:

- Predicción Meteorológica
- Información de funcionamiento
- Resultados Agrupación
- Resultados Decisión multiobjetivo.

Sus características serán analizadas en detalle en el apartado de distribución por sectores temáticos.

IniciarRedCompleta	InicioRedIndividual	Ver Información Tareas	Ver Meteorología Tareas
<input type="button" value="OFF"/>	<input type="button" value="OFF"/>	<input type="button" value="OFF"/>	<input type="button" value="OFF"/>

Figura 2: Botones de cuadro de controles

Por último, encontramos los botones disponibles dentro de la subdivisión 3, correspondientes a los controles que puede utilizar el usuario.

El primer botón, `IniciarRedCompleta`, permite iniciar todas las órdenes activas definidas en el archivo de órdenes existentes. Se presenta como opción ante posibles paradas del sistema, para permitir reiniciar el funcionamiento.

El segundo botón, se utiliza para dar inicio a una orden de trabajo individual, situada en la última posición de las órdenes activas. Está pensada para poder actuar en caso de generarse una nueva orden de trabajo y por algún problema no poder ser iniciada automáticamente, poder así iniciar el proceso tan sólo para la nueva orden, sin afectar al funcionamiento de las demás órdenes activas.

Los dos últimos botones, `Ver Información Tareas` y `Ver Meteorologías Tareas`, se utilizan para habilitar la visualización de información en las tablas de información correspondientes.

7.2. Distribución por sectores temáticos

Tal y como se ha comentado previamente, este interfaz está dividido en diversos sectores, correspondientes al estado de cada tarea, la información que se está visualizando o los subsistemas activos.

Los sectores que se estudiarán en detalle a continuación, son los siguientes:

- Parámetros relativos a Personal y Material
- Predicción meteorológica
- Información de funcionamiento
- Agrupación
- Decisión multiobjetivo
- Envío de informes a Maximo

En primer lugar, se va a analizar el sector relacionado con los parámetros relativos a Personal y Material.

En las primeras etapas de la Red de Petri Coloreada Adaptativa, las órdenes de trabajo pasan por una fase de consulta de requerimientos, en la que el sistema recurre al archivo de información de las tareas activas y consulta determinados parámetros, entre ellos el Personal necesario y el Material, obteniendo valores de tipo, cantidad, etc.

A continuación, si todos los parámetros requeridos están disponibles para su uso, las tareas pasan directamente al estado de órdenes en espera, pero si no es así, se pasa a la etapa de espera de disponibilidad, de la que saldrán cuando se cumplan los requisitos.

Por último, si la prioridad de la tarea en cuestión aumenta hasta llegar al nivel crítico, el orden de trabajo pasa a un estado nuevo, en el que se avisa al usuario de la urgencia de solicitar el material o personal necesario.

Las etapas analizadas se realizan de forma paralela para material y personal por lo que este sector presenta la información por separado y en paralelo también.

Todo este proceso puede verse en la pantalla principal del interfaz de usuario diseñado. En la figura 116 se puede ver el sector del interfaz relativo a Material y Personal, identificando las etapas previamente mencionadas.

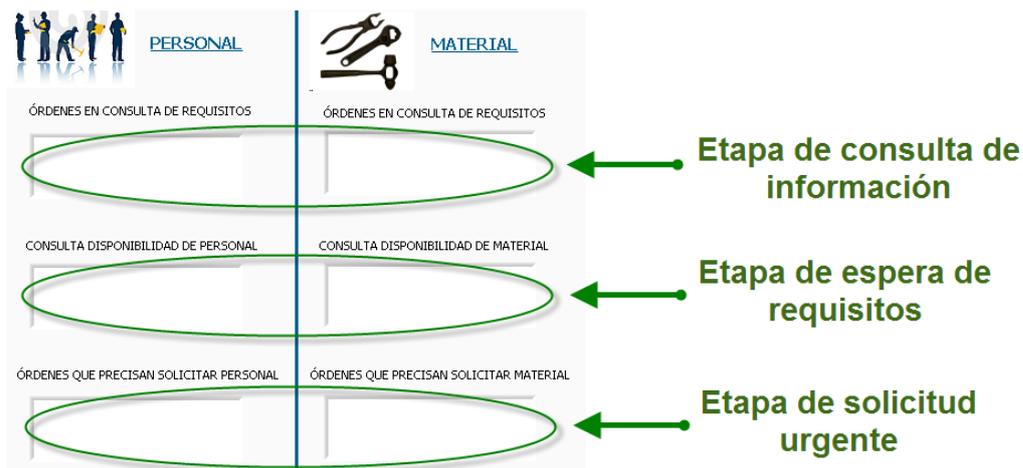
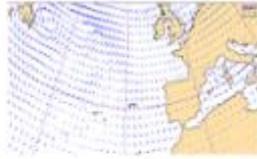


Figura 3: Diversas etapas del sector de información de Personal y Material

Respecto al sector de Predicción Meteorológica, se debe considerar este apartado como el espacio disponible para la introducción de la previsión meteorológica por parte del usuario, ya que la información de cada tarea en este campo se presenta en una división distinta.

En la figura 117 se puede observar la representación gráfica de la predicción en el interfaz de usuario.

PREDICCIÓN METEOROLÓGICA



	Parámetro 1	Parámetro 2	Parámetro 3
Primer Día	12	10	4
Segundo Día	14	10	5
Tercer Día	13	12	6

Guardar cambios
Predicción Meteorológica

OFF

Figura 4: Sector Predicción Meteorológica

Las casillas de introducción de datos se utilizan para almacenar los valores de la predicción en la matriz correspondiente, con la que trabaja el sistema para realizar las comprobaciones. Los valores se corresponden con los tres días que forman el horizonte de predicción, para los tres parámetros considerados.

Las modificaciones no son efectivas hasta que se utiliza el botón Guardar Cambios.

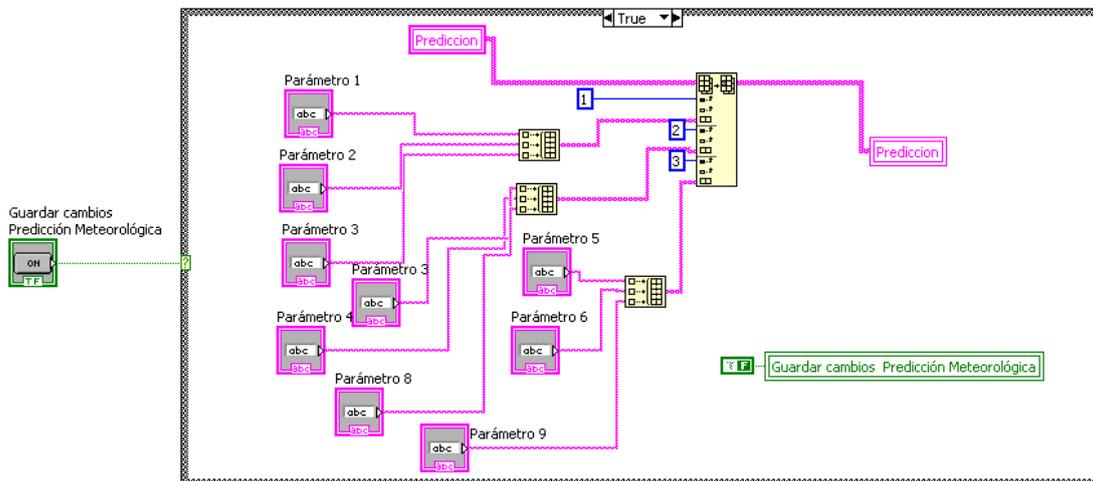


Figura 5: Construcción de la matriz de predicción meteorológica a partir de datos del interfaz

A continuación se va a analizar el sector del interfaz que hace referencia a la información de funcionamiento.

INFORMACIÓN DE FUNCIONAMIENTO

Información Tareas Activas

Nombre	OT1	OT5
Prioridad	49.000000	59.000000
Material	0	0
Personal	0	0
Horas	500	800
Coste/Hora	210	45

Meteorología Órdenes

OT1	Parámetro1	Parámetro2	Parámetro3
	2	2	2
OT5	Parámetro1	Parámetro2	Parámetro3
	2	2	2

Figura 6: Sector Información de funcionamiento

Tal y como se puede ver en la figura 119, dentro del sector de información nos encontramos con dos tablas de datos, que se analizarán por separado.

Por un lado, en la parte superior de la imagen se dispone de la información de las Tareas Activas. Como ya se ha dicho, la visualización de estos parámetros está disponible mientras el botón Ver Información Tareas del cuadro de controles permanezca activado.

Los parámetros de información de cada tarea representados son:

- Nombre de la tarea
- Nivel de Prioridad
- Volumen total de Material necesario
- Cantidad de Personal requerido
- Horas de trabajo
- Coste/Hora

El tamaño de la tabla dependerá de las tareas activas en el momento de la activación de la representación. La barra de deslizamiento situada en la parte inferior de la tabla permite al usuario visualizar toda la información representada.

Por otro lado, en la parte inferior de la imagen se encuentra la tabla de información meteorológica de las tareas.

En este caso, la información representada muestra el valor máximo permitido de cada parámetro que se establece para la correcta ejecución de la tarea, con el que se realiza la comparación con los valores de previsión meteorológica.

De igual manera que en el caso anterior, el tamaño de la información mostrada dependerá de las órdenes activas, por lo que se dispone de una barra de desplazamiento vertical para disponer de toda la información.

A continuación se analizará el sector de resultados de agrupación. Esta información no está representada directamente en pantalla, por lo que para acceder a ella se debe pulsar el botón Agrupación, representado por el esquemático del subsistema.



Figura 7: Botón de acceso a los resultados de Agrupación

Al presionar este botón, aparece una nueva pantalla en el entorno de usuario. Esta pantalla está diseñada tan sólo para mostrar la información de los resultados del subsistema de agrupación, sin salir del sistema general de mantenimiento. Es por esto que presenta una cierta transparencia, para permitir al usuario observar los resultados sin perder de vista el funcionamiento general del sistema.

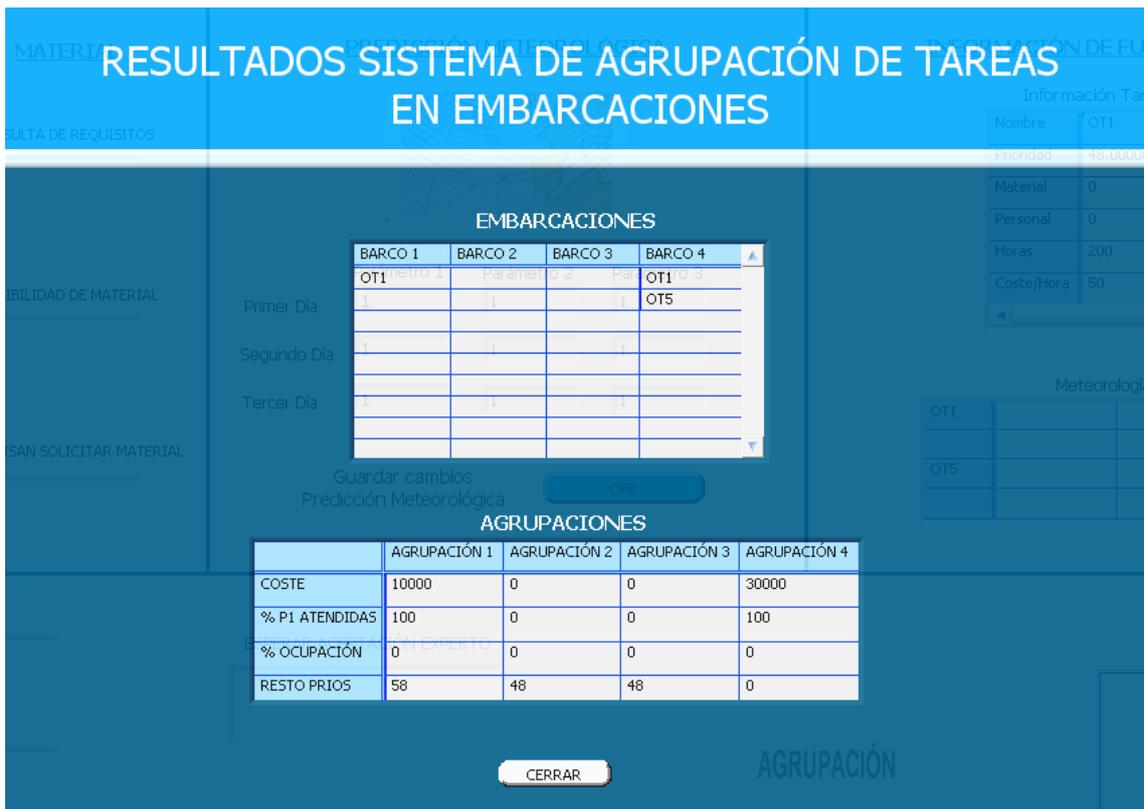


Figura 8: Ventana de resultados Agrupación

En la figura 121 se puede ver la ventana de resultados del subsistema de Agrupación de tareas. Tal y como se puede observar, se dispone de dos tablas de información.

En la parte superior se puede ver la información relativa a las embarcaciones analizadas. Durante la representación de datos, se muestra en la columna correspondiente a cada embarcación el nombre de las tareas que han sido agrupadas.

En la parte inferior de la ventana, se dispone de los parámetros calculados para cada una de las agrupaciones. Estos parámetros se estudiaron ampliamente en apartados anteriores, y tal y como se dijo se emplean para el subsistema de decisión multiobjetivo.

Para evitar errores, siempre hay que presionar el botón Cerrar para salir de la ventana.

La figura 122 representa el sector del interfaz gráfico donde se muestran al usuario los resultados del subsistema de Decisión multiobjetivo.

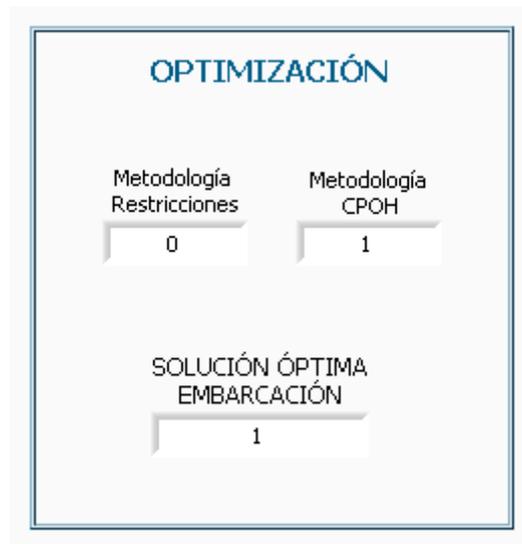


Figura 9: Sector Resultados de Optimización

En informes previos se determinó que para el subsistema de decisión multiobjetivo se utilizarían de forma paralela las metodologías basadas en restricciones y en interfaz de preferencias (CPOH).

También se determinó que en caso de divergencia en los resultados obtenidos, caso posible dada la diferente valoración de los parámetros que empelan las metodologías, se seleccionaría como solución preferente la calculada mediante interfaz de preferencias.

En la imagen se puede ver el resultado calculado mediante las dos técnicas mencionadas, y en la parte inferior el número de embarcación que finalmente se seleccionará como óptimo.

Por último, se va a analizar el sector de envío de información a Maximo. En la figura 123 se puede ver la estructura del interfaz de usuario de este sector.

En la parte superior se encuentra el estado de espera a la aceptación del experto, que tal y como se analiza en apartados previos de este informe ofrece al usuario una ventana de información con los resultados de la decisión multiobjetivo y permite ver la agrupación calculada. El paso de este estado al siguiente depende de si la solución propuesta es aceptada por el experto.

En ese caso, las órdenes de trabajo incluidas en la agrupación pasa al último estado de la Red de Petri, que también es el último estado del interfaz.

Durante este estado, al presionar el botón Guardar Informe PDF se genera el archivo de resultados de decisión multiobjetivo, que debe ser enviado a Maximo para asegurar su ejecución.



Figura 10: Sector resultados de Optimización

Una vez se ha generado y almacenado el informe, con los datos correspondientes a la agrupación seleccionada, las órdenes de trabajo incluidas en dicho informe se eliminan completamente del sistema y su estructura.

8. CONCLUSIONES

En el presente documento se plantearon como objetivos los siguientes puntos:

- Describir del sistema global de gestión del mantenimiento.
- Englobar el sistema en el diagrama general del proyecto (SISEDIFYM)
- Definir la relación entre el sistema de mantenimiento y los demás sistemas del proyecto.
- Analizar los diferentes subsistemas y describir la metodología aplicada en cada caso.

A lo largo de los diversos apartados, se ha podido analizar el sistema global a implementar, así como la herramienta seleccionada para ello, el programa Labview mediante la utilización de Redes de Petri Coloreadas Adaptativas.

Se han definido los distintos subsistemas incluidos en el programa global, presentando las posibilidades que existen para su implementación y la selección realizada en base a ventajas e inconvenientes de cada método, explicando cómo se realizará la implementación. Por otro lado, se han analizado también los problemas de comunicaciones entre subsistemas y también del sistema con el programa Máximo.

Se han analizado las técnicas empleadas para los análisis Coste Riesgo, muy utilizados para la toma de decisiones. En el caso de los sistemas de mantenimiento, estas técnicas son muy importantes ya que se utilizan para determinar el mejor momento para la realización de la tarea de mantenimiento. Si la metodología empleada además incorpora conceptos de decisión multiobjetivo, utilizando técnicas más precisas que el tiempo entre fallos, se otorgará al sistema de mantenimiento de una herramienta muy capacitada para la toma de decisiones.

Se han estudiado las herramientas disponibles en el mercado, en concreto los programas @Risk y Crystall_Ball que trabajan como complementos de Excel, analizando las ventajas que aportaría su utilización al sistema diseñado. Sin embargo, se ha determinado también que se trata de sistemas sobredimensionados con gran cantidad de aplicaciones que no resultan de utilidad para el proyecto, por lo que se ha presentado una herramienta de desarrollo propio con características de funcionamiento específicas para la aplicación de optimización Coste Riesgo que se quería diseñar.

Se ha presentado también la implementación del Sistema de Gestión del Mantenimiento desarrollada para la tesina.

El sistema de Gestión del Mantenimiento desarrollado a lo largo del proyecto EOLIA ofrece una nueva perspectiva a los sistemas tradicionales, ya que permite trabajar conjuntamente con una herramienta tan completa como Maximo y además aporta nuevos conceptos al mantenimiento.

Dentro de estas mejoras que ofrece el sistema, se distinguen los siguientes conceptos:

- Emplea la metodología de Redes de Petri Coloreadas Adaptativas desarrollada, que aporta una nueva visión del tratamiento de la información en este tipo de sistemas.
- Se trata de una estructura adaptativa que permite trabajar con tantas órdenes de trabajo como existan en cada momento, sin afectar a la estructura del sistema.
- El sistema es completamente automático, a excepción de los parámetros de consulta al usuario para la revisión de agrupaciones y envío de informes.
- Emplea sistemas de comprobación de parámetros meteorológicos, mediante la incorporación de previsiones con horizonte de predicción a tres días.
- Incorpora conceptos de prioridades variables, en función de la evolución temporal de las órdenes de trabajo dentro del sistema.
- Ofrece el cálculo automático de opciones de mantenimiento en función de los parámetros establecidos por las órdenes de trabajo y las posibilidades en cuanto a embarcaciones de mantenimiento.
- Utiliza las agrupaciones calculadas para obtener una solución óptima de ejecución, con el fin de mejorar las tareas y reducir el alto impacto del mantenimiento en instalaciones offshore.
- Para el cálculo de la solución óptima se emplean dos metodologías distintas de optimización multiobjetivo.

Este sistema ha sido implementado en la herramienta Labview, trabajando conjuntamente con el software Excel. Es por esto que se han presentados dos librerías específicamente desarrolladas para la tesina: una librería de comunicación entre los mencionados programas y otra librería de implementación propia de la Red de Petri Coloreada Adaptativa diseñada.

Se ha podido presentar también toda la estructura de funcionamiento de dicha red en el programa Labview, explicando el funcionamiento paso a paso del programa y viendo en detalle los diversos subsistemas implementados para la tesina.

Posteriormente, se ha presentado la programación interna de cada subsistema y su modo de uso, permitiendo de este modo una mayor comprensión del sistema, no sólo a nivel visual sino también a nivel de funcionamiento interno.

Por último, se ha presentado el interfaz gráfico de usuario, desarrollado para trabajar de forma independiente a Maximo y ser utilizado por un usuario experto, analizando las posibilidades de interacción con el programa y el análisis de resultados.

Queda de este modo completado el análisis del sistema de Gestión de Mantenimiento implementado para la presente tesina de máster.

9. REFERENCIAS

García, E., Morant, F., Correcher, A., Quiles E. "Seguimiento de Estado, Diagnóstico de Fallos y Mantenimiento Industrial: una visión integrada." Proceedings of the Novena Semana Tecnológica. 19-23 October 2009. La Habana. Cuba.

Pérez M., García E., Morant F., Correcher A. and Quiles E. "Optimal Maintenance system for offshore wind turbines". International Conference on Renewable Energies and Power Quality (ICREPQ'10).Granada (Spain), 23th to 25th March, 2010

M. García, M.A. Sanz and J. Del Pico, "Aplicación de técnicas de inteligencia artificial en el mantenimiento predictivo de aerogeneradores", in IV Jornada de Fiabilidad: Confiabilidad, Las Palmas de G.C., September 2002.

G.J.W. van Bussel "Reliability, availability and maintenance aspects of large-scale offshore wind farms, a concepts study", Delft University of Technology, The Netherlands, MAREC 2001 Conference Proceedings, pages 119 – 126.

G.J.W. Bussel van and A.R. Henderson, "State of the art and technology trends for offshore wind energy: operation and maintenance issues", Proceedings of offshore wind energy special topic conference 10–12 December, Brussels, Belgium, EWEA (2001), pp. 1–4.

L.W.M.M. Rademakers, H. Braam, T.S. Obdam, P. Frohböse, N. Kruse, "Assessment and optimization of operation and maintenance of offshore wind turbines", Proceedings of the European wind energy conference, 16–19 June, Madrid, Spain, EWEA (2003) 6 pp.

H.-J. Krokoszinski, "Efficiency and effectiveness of wind farms—keys to cost optimized operation and maintenance", Renewable Energy 28 (14)(2003), pp. 2165–2178.

J. Nilsson, L. Bertling. "Maintenance management of wind power systems using condition monitoring systems". IEEE Transactions on Energy Conversion, vol.22, No.1, March 2007.

F. Besnard, M. Patriksson, A. Strömberg, A. Wojciechoski, L. Bertling. "An Optimization Framework for opportunistic maintenance of offshore wind power system" IEEE Bucharest Power Tech Conference, June 28th –July 2nd, Bucharest, Romania.

L.W.M.M. Rademakers, H. Braam, T.S. Obdam. "Estimating costs of operation & maintenance for offshore wind farms" EWEC 2008, March 31st-April 3rd.

L.W.M.M. Rademakers, H. Braam, T.S. Obdam, R.P. v.d. Pieterman. "Operation and maintenance cost estimator to estimate the future o&m costs of offshore wind farms" European Offshore Wind 2009 Conference, Stockholm, 14-16 September, 2009

Lajara, José Rafael; Pelegrí, José "LabView: entorno gráfico de programación" ,Barcelona, España; Marcombo, 2007.

Bitter, Rick; Mohiuddin, Taqi; Nawrocki, Matt. "LabVIEW: Advanced programming Techniques" 2^o Edition. CRC Press. 2007.

Isermann, R. "Supervision, fault-detection and fault-diagnosis methods -- An introduction" *Control Engineering Practice*, , 5, pp 639 – 652, 1997.

P. Caselitz, , J. Giebhardt, M. Mevenkamp. "Development of a Fault Detection System for Wind Energy Converters". Proceedings of the EUWEC '96, Göteborg, pp. 1004 - 1007

P. Caselitz, J. Giebhardt, M. Mevenkamp, M. Reichardt, "Application of condition monitoring systems in wind energy converter". Proceedings of EWEC '97, Dublin, pp. 579 – 582.

P. Caselitz, J. Giebhardt, M. Mevenkamp, "On-line Fault Detection and Prediction in Wind Energy Converters",. Proceedings of EWEC '94, Thessaloniki,, pp. 623 – 627.

Hiroshi Shihara. "A study on Maintenance of LNG Cargo Lines using Risk-Based Maintenance System". Proceedings of 3rd PAAMES and AMEC 2008

Navarrete, Treto, Rodríguez, anreus, Lorenzo. "Gestión y Calidad del Mantenimiento".

Damien Coache. "Maintenance of offshore wind Turbines – An analysis of the effect of accessibility". MSc Thesis, Cranfield University. 2006.

Julia Nilsson. "Maintenance management of wind power systems". Master Thesis. 2006

Jezdimir Knezevic. "Mantenimiento". Monografías de Ingeniería de Sistemas.

John Dalsgaard Sorensen. "Optimal, risk-based operation and maintenance planning for offshore wind turbines".

M. Cruz García Fernández. "Sistema Inteligente de ayuda al mantenimiento de aerogeneradores"

Sammy Seifeddine. "Desarrollo y Optimización Efectiva de un Programa de Mantenimiento" Confiabilidad.Net

José Duran. "Aplicación de la Técnica "Optimización Costo-Riesgo" Confiabilidad.Net

D. León, I. Quintero, W. Zuluaga. "Crystall Ball". Universidad Nacional de Colombia. 2004