

Trabajo Fin de Grado

Grado en Ingeniería de Organización Industrial

Desarrollo de un soporte informático en Python para modelos de optimización del mantenimiento en un horizonte temporal finito.

Autor: Beatriz M.^a Díaz Gandul

Tutor: Dr. Antonio Sánchez Herguedas

**Dpto. de Organización Industrial y Gestión de
Empresas I**

Escuela Técnica Superior de Ingeniería

Sevilla, 2020



Trabajo Fin de Grado
Grado en Ingeniería de Organización Industrial

**Desarrollo de un soporte informático en Python
para modelos de optimización del mantenimiento en
un horizonte temporal finito.**

Autor:

Beatriz M.^a Díaz Gandul

Tutor:

Dr. Antonio Sánchez Herguedas

Profesor titular

Dpto. de Organización Industrial y Gestión de Empresas I

Escuela Técnica Superior de Ingeniería

Universidad de Sevilla

Sevilla, 2020

Trabajo Fin de Grado: Desarrollo de un soporte informático en Python para modelos de optimización del mantenimiento en un horizonte temporal finito.

Autor: Beatriz M^a Díaz Gandul

Tutor: Dr. Antonio Sánchez Herguedas

El tribunal nombrado para juzgar el Proyecto arriba indicado, compuesto por los siguientes miembros:

Presidente:

Vocal/es:

Secretario:

Acuerdan otorgarle la calificación de:

El Secretario del Tribunal

Fecha:

A mi familia

A mis profesores

Agradecimientos

Me gustaría agradecer este proyecto a mi familia por confiar en mí y darme esta oportunidad.

A mis amigos por estar cerca o lejos, pero siempre a mi lado.

A mis compañeros y profesores por haberme acompañado en este camino.

Al Dr. D. Antonio Sánchez Herguedas por su ayuda y atención en la elaboración de este proyecto.

Y a todas las personas con las que me he cruzado estos cinco años que me han enseñado algo nuevo.

Beatriz M.^a Díaz Gandul

Alumna de la Escuela Técnica Superior de Ingeniería

Sevilla, 2020

Resumen

En el presente trabajo se avanza en el conocimiento de la resolución de un problema de optimización del mantenimiento, así como en su implantación y gestión en problemas reales.

Se trata de un problema de mantenimiento para un horizonte de tiempo finito de m pasos, donde se buscan los intervalos de mantenimiento preventivo óptimos. Se parte de un método de referencia en el que se modela el problema como un proceso Semi-Markoviano y se utiliza la técnica de la transformada z para resolver la ecuación recurrente del retorno (ingresos menos costes) promedio acumulado para un sistema que se inicia desde un estado *Operativo*.

La aportación al método creada en este trabajo, siguiendo dicha metodología, permite calcular el tiempo hasta el mantenimiento preventivo óptimo, inicializando el sistema adicionalmente desde los estados *Correctivo* y *Preventivo*.

En este proyecto se crea una herramienta de escritorio que permite hallar el momento óptimo en el que un sistema debe pasar a un estado donde se apliquen medidas de mantenimiento preventivo, de manera que se maximice el retorno medio acumulado en un horizonte de tiempo finito.

La herramienta se ha desarrollado a través del lenguaje de programación Python con el cual se ha creado una interfaz gráfica de fácil comprensión para el usuario final, gracias a la importación de la librería Tkinter. La aplicación permite comparar visualmente, a través de representaciones gráficas, los diferentes escenarios que puede presentar el sistema simplemente cambiando cualquiera de sus parámetros.

Tras demostrar el correcto funcionamiento del programa, la rapidez y facilidad en la obtención de resultados se revela como una ventaja adicional de la herramienta desarrollada, que puede favorecer la implantación de políticas de mantenimiento, en particular la planificación y gestión del mantenimiento preventivo.

Abstract

In the following project, progress is made in the understanding of the resolution of a maintenance optimization problem, as well as its implementation and management in real cases.

This maintenance problem is characterised for a finite time horizon model of m steps, where optimal preventive maintenance intervals are sought. Its resolution comes from a reference method where the problem has been modelled as a Semi-Markovian process and the z -transform technique is used for solving the recurring equation of the accumulated average return (income minus costs) for a system that starts from an Operational state.

The contribution to the method created in this work, following the mentioned methodology, allows the calculation of the optimal time for preventive maintenance, additionally initializing the system from the Corrective and Preventive states.

In the following project, a desktop tool has been created. It finds the optimal moment in which a system should change to a state where preventive maintenance measures should be applied. With the aim of maximizing the average accumulated return (income less costs) in a finite time horizon.

The tool has been developed through the Python programming language with which an easy-to-understand graphical interface has been created for the end user, thanks to the import of the Tkinter library. The application allows to visually compare, through graphic representations, the different scenarios that the system can present simply by changing any of its parameters.

After demonstrating the correct operation of the program, the speed and ease in obtaining results is revealed as an additional advantage of the developed tool, which can favor the implementation of maintenance policies, particularly the planning and management of preventive maintenance.

Agradecimientos	ix
Resumen	xi
Abstract	xiii
Índice	xv
Índice de Tablas	xix
Índice de Figuras	xxi
Notación	xxv
1 Introducción	1
1.1 Sumario sobre la estructura del proyecto	1
2 Antecedentes	3
2.1 Introducción a los Modelos de Optimización	3
2.1.1 Optimización	3
2.1.2 Sistema	3
2.1.3 Modelo de optimización	4
2.2 Introducción al Mantenimiento	4
2.2.1 Tipos de mantenimiento	5
2.2.1.1 Mantenimiento correctivo	5
2.2.1.2 Mantenimiento preventivo	5
2.2.2 Impactos del mantenimiento	5
2.2.2.1 Impacto económico del mantenimiento	5
2.2.2.2 Impacto medioambiental del mantenimiento	6
2.2.2.3 Impacto del mantenimiento en las personas	6
2.2.3 Implantación del mantenimiento	7
2.2.4 Planificación del mantenimiento preventivo	7
2.2.4.1 Objetivos	7
2.2.4.2 Estado del sistema	8
2.2.4.3 Horizonte de tiempo	8
2.2.4.4 Métodos	8
2.3 Introducción a Python y sus Características	9
3 Formalización del Problema	11
3.1 Descripción Inicial del Problema	11
3.2 Objetivo	11
3.3 Resolución	11
3.3.1 Modelo Semi-Markoviano	12
3.3.2 Costes e ingresos	13
3.3.3 Explicación de la resolución del sistema de ecuaciones en diferencias aplicando la transformada z	14
3.3.4 Aplicación de la transformada z al sistema de ecuaciones en diferencias del modelo Semi-Markoviano de 3 estados.	15
3.3.5 Obtención de la inversa de la transformada z	17
3.3.6 Optimización del retorno esperado al comenzar el sistema desde el estado <i>Operativo</i>	18
3.4 Aportación al Método	20
3.4.1 Inicialización del sistema en el estado <i>Correctivo</i>	20
3.4.2 Inicialización del sistema en el estado <i>Preventivo</i>	22
3.5 Descripción del Caso Práctico	23
3.5.1 Datos del problema	24

4	Manual de Usuario del Programa PM-3E	25
4.1	Interfaz de Usuario: Pantalla Principal	26
4.2	Datos Iniciales	28
4.3	Calculadora	29
4.3.1	Cálculo de tau óptimo	29
4.3.2	Cálculo del vector de retorno promedio acumulado, V	30
4.3.2.1	Opciones de cálculo para V	31
4.3.3	Cálculo del tiempo de operación, OT.	32
4.3.3.1	Opciones de cálculo para OT	33
4.3.4	Funciones de los botones	34
4.3.4.1	Botón Calcular	34
4.3.4.2	Botón Eliminar	34
4.3.4.3	Botón Guardar	35
4.4	Gráfico	37
4.4.1	Elecciones de representación	38
4.4.2	Funciones de los botones	40
4.4.2.1	Botón Representar	40
4.4.2.2	Botón Eliminar	40
4.4.2.3	Botón Guardar	40
4.5	Calculadora para Tau propio	42
4.6	Botones Ayuda	42
5	Resultados Experimentales	49
5.1	Datos Iniciales	49
5.2	Comprobación del Valor Óptimo de τ	50
5.2.1	Caso 1 (R1=45)	51
5.2.2	Caso 2 (R1=12)	51
5.2.3	Caso 3 (R1= 6)	52
5.2.4	Representación gráfica del valor óptimo de tau versus el número de transiciones	52
5.3	Comprobación del Primer Componente del Vector de Retorno Promedio Acumulado para pasos de transición pares, V1(m par)	53
5.3.1	Caso 1 (m=10)	54
5.3.2	Caso 2 (m=8)	55
5.3.3	Caso 3 (m=6)	55
5.4	Comprobación del Primer Componente del Vector de Retorno Promedio Acumulado para pasos de transición impares, V1(m impar)	56
5.4.1	Caso 1 (m=9)	56
5.4.2	Caso 2 (m=7)	57
5.4.3	Caso 3 (m=5)	58
5.5	Comprobación del Tiempo de Operación Promedio, AOT	59
5.6	Comprobación del Primer Vector de Retorno Acumulado para 10 pasos de transición	60
5.7	Comprobación del Valor del Tiempo de Funcionamiento Promedio Acumulado, AAOT	61
5.8	Comprobación del Valor del Retorno Promedio, V1/AAOT	62
6	Conclusiones	63
7	Trabajo Futuro	65
Anexo A:	Codificación en Python del programa PM-3E	67
A.1	Importación de Librerías	67
A.2	Creación de la Ventana Principal	67
A.3	Declaración de las Variables Principales	68
A.4	Declaración de Widgets	68
A.4.1	Datos iniciales: títulos	68
A.4.2	Datos iniciales: etiquetas y casillas de escritura	69
A.4.3	Botón V1, V2 o V3	70
A.4.4	Cálculo de tau óptimo	71
A.4.5	Cálculo de V	71
A.4.6	Cálculo de OT	72
A.4.7	Sección plantilla gráfico	72

A.4.8	Sección gráfico	72
A.4.9	Cálculo para tau propio	73
A.4.10	Creación botones ayuda	74
A.5	Posicionamiento de Widgets	74
A.5.1	Datos iniciales	74
A.5.2	Botones V1, V2 y V3	75
A.5.3	Cálculo de tau óptimo	75
A.5.4	Cálculo de V	76
A.5.5	Cálculo de OT	76
A.5.6	Representación gráfica	76
A.5.7	Cálculo para tau propio	77
A.5.8	Botones ayuda	77
A.6	Definición Botones Ayuda	77
A.6.1	Retornos	77
A.6.2	Datos iniciales	78
A.6.3	Cálculo de tau óptimo	79
A.6.4	Cálculo de V	79
A.6.5	Cálculo de OT	80
A.6.6	Guardar	81
A.6.7	Opciones representación gráfica	81
A.6.8	Representación gráfica	82
A.6.9	Cálculo para tau propio	82
A.7	Definición Botón Eliminar	83
A.7.1	Resultado de tau óptimo	83
A.7.2	Resultado de V	83
A.7.3	Resultado de OT	84
A.7.4	Gráfico	84
A.7.5	Resultado para tau propio	84
A.8	Función: Cálculo de Tau Óptimo	84
A.9	Función: Cálculo de V	86
A.10	Función: Cálculo de OT	90
A.11	Función: Guardar Resultados	91
A.12	Función: Guardar Gráfico	94
A.13	Función: Representación de Resultados	94
A.14	Función: Cálculo para Tau Propio	98
A.15	Cierre de la Aplicación	100
8	Referencias	101

Índice de Tablas

Tabla 2–1. Métodos aplicados a la planificación del mantenimiento preventivo. [28]	8
Tabla 3–1. Valores iniciales para las funciones de distribución (FD) [1].	24
Tabla 3–2. Datos iniciales para los retornos [1].	24
Tabla 5–1. Valores de los parámetros utilizados para el caso de estudio [1]	49
Tabla 5–2. Valores de los retornos utilizados para el caso de estudio [1].	50
Tabla 5–3. τ_0 para diferentes pasos y retornos de operación [1].	50
Tabla 5–4. τ_0 para $R1=45$ [1]	51
Tabla 5–5. τ_0 para $R1=12$ [1]	51
Tabla 5–6. τ_0 para $R1=6$ [1]	52
Tabla 5–7: Resultado del retorno promedio acumulado para 10 pasos de transición.	60
Tabla 5–8: Resultado para AAOT correspondiente a 10 pasos de transición [1]	61
Tabla 5–9: Resultados para $V1/AAOT$ correspondientes a 10 pasos de transición [1]	62

Índice de Figuras

Figura 2-1: Formato general del modelo de optimización.	4
Figura 2-2: Requisitos ecológicos del mantenimiento [19]. [Traducción propia]	6
Figura 3-1: Esquema cambio de fases del sistema. [Fuente: Elaboración propia].	12
Figura 3-2: Transición de un estado a otro. [Fuente: Elaboración propia].	12
Figura 3-3: Volquete. [38]	23
Figura 4-1: Pantalla principal de PM-3E.	26
Figura 4-2: Opciones para minimizar, maximizar y cerrar la aplicación.	26
Figura 4-3: Diferentes secciones de la herramienta PM-3E	27
Figura 4-4: Señalización botón “Ayuda”	27
Figura 4-5: Sección de la herramienta donde se deben introducir los datos iniciales.	28
Figura 4-6: Sección de la herramienta donde se muestran los resultados	29
Figura 4-7: Sección de la herramienta para el cálculo de tau óptimo.	29
Figura 4-8: Representación de cómo calcular tau óptimo.	30
Figura 4-9: Señalización del resultado de tau óptimo para un número de pasos determinado.	30
Figura 4-10: Sección de la herramienta para el cálculo de V.	30
Figura 4-11: Representación de cómo calcular V.	31
Figura 4-12: Señalización del resultado de V1 para un número de pasos determinado.	31
Figura 4-13: Señalización del resultado de V1 para un número de pasos determinado.	31
Figura 4-14: Señalización del resultado de V1 para un número de pasos determinado.	32
Figura 4-15: Sección de la herramienta para el cálculo del tiempo de operación y opciones disponibles de cálculo.	32
Figura 4-16: Representación de cómo calcular los tiempos de operación.	32
Figura 4-17: Señalización del resultado de AOT para un número de pasos determinado.	33
Figura 4-18: Señalización del resultado de AAOT para un número de pasos determinado.	33
Figura 4-19: Señalización del resultado de AR para un número de pasos determinado.	33
Figura 4-20: Señalización de los botones existentes en la herramienta para calcular resultados.	34
Figura 4-21: Representación de cómo eliminar un resultado.	34
Figura 4-22: Explicación de cómo guardar los resultados obtenidos.	35
Figura 4-23: Explicación de cómo guardar los resultados obtenidos.	35
Figura 4-24: Archivo creado al guardar los resultados a través de la herramienta.	36
Figura 4-25: Ejemplo de cómo se guardan los datos obtenidos a través del programa.	36
Figura 4-26: Sección de la herramienta para la representación gráfica de los resultados	37
Figura 4-27: Representación de cómo realizar una representación gráfica usando la herramienta.	37
Figura 4-28: Representación gráfica de tau frente a V1 creada a través de la herramienta.	38

Figura 4-29: Representación gráfica de los pasos de transición frente al tau óptimo creada a través de la herramienta.	38
Figura 4-30: Representación gráfica del número de pasos frente a V1, creada a través de la herramienta.	39
Figura 4-31: Representación gráfica del número de pasos frente a AAOT creada a través de la herramienta.	39
Figura 4-32: Representación gráfica del número de pasos frente a AR creada a través de la herramienta.	40
Figura 4-33: Explicación de cómo guardar la representación gráfica creada con la herramienta.	40
Figura 4-34: Pestaña para guardar el archivo creado.	41
Figura 4-35: Imagen creada a partir de la herramienta.	41
Figura 4-36: Sección de PM-3E, calculadora para tau propio	42
Figura 4-37: Opciones de cálculo para un tau propio dado por el usuario.	42
Figura 4-38: Ejemplo de uso de la Calculadora para Tau propio de la herramienta.	42
Figura 4-39: Señalización del botón ayuda de la herramienta	42
Figura 4-40: Nota explicativa emergente al clicar en un botón Ayuda asociado a “Datos Iniciales”.	43
Figura 4-41: Nota explicativa emergente al clicar en un botón Ayuda asociado a “Retornos”.	43
Figura 4-42: Nota explicativa emergente al clicar en un botón Ayuda, asociado al botón para guardar el archivo.	44
Figura 4-43: Nota explicativa emergente al clicar en un botón Ayuda asociado al cálculo de tau óptimo.	44
Figura 4-44: Nota explicativa emergente al clicar en un botón Ayuda asociado al cálculo del vector de retorno promedio acumulado.	45
Figura 4-45: Nota explicativa emergente al clicar en un botón Ayuda, asociado al cálculo de los tiempos de operación.	45
Figura 4-46: Nota explicativa emergente al clicar en un botón Ayuda, asociado a las opciones de representación gráfica que dispone el programa.	46
Figura 4-47: Nota explicativa emergente al clicar en un botón Ayuda asociado a la representación gráfica de los datos.	46
Figura 4-48: Nota explicativa emergente al clicar en un botón Ayuda asociado al cálculo de parámetros con el uso de un tau dado.	47
Figura 5-1: Datos iniciales. [Imagen tomada de la herramienta PM-3E].	50
Figura 5-2: τ_0 para 9 pasos de transición y R1=45. Resultado de 1-6 pasos. [Imagen tomada de la herramienta PM-3E].	51
Figura 5-3: τ_0 para 9 pasos de transición y R1=45. Resultado de 4-9 pasos. [Imagen tomada de la herramienta PM-3E].	51
Figura 5-4: τ_0 para 9 pasos de transición y R1=12. Resultado de 1-6 pasos. [Imagen tomada de la herramienta PM-3E].	51
Figura 5-5: τ_0 para 9 pasos de transición y R1=12. Resultado de 4-9 pasos. [Imagen tomada de la herramienta PM-3E].	51
Figura 5-6: τ_0 para 9 pasos de transición y R1=6.	52
Figura 5-7: τ_0 para 9 pasos de transición y R1=6.	52
Figura 5-8: Tiempo hasta la acción preventiva versus el número de transiciones [1].	52
Figura 5-9: Representación gráfica del valor óptimo de tau, para 9 pasos en tres casos distintos. [Imagen tomada de la herramienta PM-3E].	53
Figura 5-10: Primer componente del vector de retorno promedio versus tiempo para τ preventivo, para tres valores de transiciones pares diferentes [1].	53

Figura 5-11: Introducción de valor de $m=10$. [Imagen tomada de la herramienta PM-3E].	54
Figura 5-12: Selección para la representación gráfica del tiempo preventivo frente al vector de retorno promedio acumulado. [Imagen tomada de la herramienta PM-3E].	54
Figura 5-13 Primer componente del vector de retorno acumulado promedio versus tiempo para τ preventivo, para un valor de transición par ($m=10$). [Imagen tomada de la herramienta PM-3E].	54
Figura 5-14: Introducción de valor de $m=8$. [Imagen tomada de PM-3E].	55
Figura 5-15: Primer componente del vector de retorno acumulado promedio versus tiempo para τ preventivo, para dos valores de transiciones pares diferentes ($m=10$, línea naranja y $m=8$, línea roja). [Imagen tomada de PM-3E].	55
Figura 5-16: Introducción de valor de $m=6$. [Imagen tomada de PM-3E]	55
Figura 5-17: Representación gráfica del primer componente del vector retorno promedio acumulado para τ preventivo, para tres valores de transiciones pares diferentes ($m=10$, línea naranja; $m=8$, línea roja y $m=6$, línea marrón). [Imagen tomada de PM-3E].	55
Figura 5-18: Primer componente del vector de retorno acumulado promedio versus tiempo para τ preventivo, para tres valores de transiciones impares diferentes [1].	56
Figura 5-19: Introducción de valor de $m=9$. [Imagen tomada de PM-3E].	56
Figura 5-20: Selección para la representación gráfica del tiempo preventivo frente al vector de retorno promedio acumulado. [Imagen tomada de PM-3E].	56
Figura 5-21: Primer componente del vector de retorno acumulado promedio versus tiempo para τ preventivo, para un valor de transición impar ($m=9$). [Imagen tomada de PM-3E].	57
Figura 5-22: Introducción de valor de $m=7$. [Imagen tomada de PM-3E].	57
Figura 5-23: Primer componente del vector de retorno acumulado promedio versus tiempo para τ preventivo, para dos valores de transiciones impares diferentes ($m=9$, línea naranja y $m=7$, línea roja). [Imagen tomada de PM-3E].	57
Figura 5-24: Introducción de valor de $m=5$. [Imagen tomada de PM-3E].	58
Figura 5-25: Representación gráfica del primer componente del vector retorno promedio acumulado para τ preventivo, para tres valores de transiciones impares diferentes ($m=9$, línea naranja; $m=7$, línea roja y $m=5$, línea marrón). [Imagen tomada de PM-3E].	58
Figura 5-26: Cálculo de τ_0 para $m=10$. [Imagen tomada de PM-3E].	59
Figura 5-27: Cálculo de V_1 pasos de transición impares. [Imagen tomada de PM-3E].	59
Figura 5-28: Cálculo de AOT para 10 pasos de transición. [Imagen tomada de PM-3E].	59
Figura 5-29: Resultado de V_1 de $m=1$ hasta $m=6$ para 10 pasos de transición. [Imagen tomada de PM-3E].	60
Figura 5-30: Resultado de V_1 de $m=5$ hasta $m=10$ para 10 pasos de transición. [Imagen tomada de PM-3E].	60
Figura 5-31: Resultado de AAOT de $m=1$ hasta $m=6$ para 10 pasos de transición. [Imagen tomada de PM-3E].	61
Figura 5-32: Resultado de AAOT de $m=5$ hasta $m=10$ para 10 pasos de transición. [Imagen tomada de PM-3E].	61
Figura 5-33: Resultado de $V_1/AAOT$ de $m=1$ hasta $m=5$ para 10 pasos de transición. [Imagen de PM-3E].	62
Figura 5-34: Resultado de $V_1/AAOT$ de $m=5$ hasta $m=10$, para 10 pasos de transición. [Imagen de PM-3E].	62

Notación

SMP	Proceso Semi-Markoviano
PM-3E	Nombre del programa creado: Mantenimiento preventivo, sistema con 3 estados
E1	Estado <i>Operativo</i>
E2	Estado <i>Correctivo</i>
E3	Estado <i>Preventivo</i>
m	Paso de transición de un estado a otro
p_{ij}	Probabilidad de transición del estado i al estado j
P	Matriz de probabilidades de transición
F	Matriz de tiempos de permanencia
A	Tiempo promedio de permanencia en el estado <i>Operativo</i>
τ	Tau, tiempo de permanencia en el estado <i>Operativo</i> antes de pasar al <i>Preventivo</i>
τ_0	Tau óptimo, tiempo óptimo de permanencia en el estado <i>Operativo</i> antes de pasar al <i>Preventivo</i>
B	Tiempo promedio de permanencia en el estado <i>Correctivo</i>
C	Tiempo promedio de permanencia en el estado <i>Preventivo</i>
$F(t)$	Función de distribución acumulativa de fallos del equipo
$f(t)$	Función de densidad de la probabilidad de fallos del equipo
$G(t_c)$	Función de distribución del tiempo de permanencia bajo mantenimiento correctivo
$g(t_c)$	Función de densidad de la probabilidad del tiempo bajo mantenimiento correctivo
$H(t_p)$	Función de distribución del tiempo de permanencia bajo mantenimiento preventivo
$h(t_p)$	Función de densidad de la probabilidad del tiempo bajo mantenimiento preventivo
r_{ij}	Retorno al pasar del estado i al estado j
R1	Ingreso por unidad de tiempo que el sistema permanece en el estado <i>Operativo</i>
R2	Coste por unidad de tiempo que el sistema permanece en el estado <i>Correctivo</i>
R3	Coste por unidad de tiempo que el sistema permanece en el estado <i>Preventivo</i>
R12	Coste de transición del estado <i>Operativo</i> al <i>Correctivo</i>
R21	Coste de transición del estado <i>Correctivo</i> al <i>Operativo</i>
R13	Coste de transición del estado <i>Operativo</i> al <i>Preventivo</i>
R31	Coste de transición del estado <i>Preventivo</i> al <i>Operativo</i>
$v_i(m)$	Retorno acumulado medio
$V(m)$	Vector de retorno promedio acumulado
$Z[x]$	Función de la transformada z
AOT	Tiempo de operación promedio del sistema
AAOT	Tiempo de operación promedio acumulado
AR	Retorno acumulado
OT	Tiempo de operación

1 INTRODUCCIÓN

La mejor forma de predecir el futuro es crearlo.

- Peter Drucker-

En el presente trabajo fin de grado se considera la importancia de la gestión del mantenimiento y su implantación para garantizar el buen funcionamiento de instalaciones, sistemas y equipos, de tal forma que se logren buenas condiciones de operación con la máxima disponibilidad a un costo general optimizado. Sin embargo, la mayoría de los análisis y métodos sugeridos en la literatura se realizaron basándose en cálculos matemáticos modelados en horizontes de tiempo infinitos, en lugar de centrarse en la solución de problemas reales en la industria con horizontes de tiempo finitos.

El objetivo de este proyecto es la creación de una herramienta de escritorio diseñada en Python, a la que se le ha denominado **PM-3E**. El nombre hace referencia al Mantenimiento Preventivo para un sistema con 3 posibles Estados. Su finalidad es el cálculo del tiempo óptimo hasta la intervención por mantenimiento preventivo para un sistema, adoptando un enfoque de procesos Semi-Markovianos y aplicando la técnica de la transformada z. El método utilizado es un método exacto y directo, descrito por Sánchez Herguedas A. [1], donde se lleva a cabo la política de mantenimiento que proporciona el retorno acumulado promedio máximo (ingresos-costos) del sistema, para una serie de pasos de transición m. El programa incluye, además, una aportación propia al método desarrollada en este proyecto.

Para ver el funcionamiento y utilidad de la herramienta creada, se tomaron los datos del caso práctico del artículo de referencia [1]. En éste se realiza un estudio real y se implementa el método anteriormente descrito a un motor Diesel de alta potencia. Finalmente, se obtuvieron los resultados para el caso real usando el programa y se compararon éstos con los del artículo de referencia para verificar su funcionamiento y utilidad.

1.1 Sumario sobre la estructura del proyecto

El documento se desarrolla en 7 capítulos y un anexo, como se explica brevemente:

En el capítulo 1, “Introducción”, se revela una breve descripción del proyecto que se va a presentar y cómo se distribuyen cada uno de los capítulos que lo contienen.

En el capítulo 2, “Antecedentes”, en primer lugar, se hace una introducción al concepto de optimización y los modelos matemáticos. En segundo lugar, se explica el significado de mantenimiento, además de los impactos que supone su mala gestión, los tipos que existen, el significado de implantación del mantenimiento y los elementos a tener en cuenta para la planificación del mantenimiento preventivo. Por último, se presenta una descripción del lenguaje de programación Python y las características que lo definen.

En el capítulo 3, “Formalización del problema”, se explica el problema y su objetivo. Después, se explica toda la metodología seguida para su resolución. Finalmente, se presenta una aportación propia al método y se define el caso práctico.

En el capítulo 4, “Manual de usuario del programa PM-3E” se puede encontrar un manual de uso para la herramienta creada. En él se explican cada uno de los elementos que aparecen, los parámetros que se pueden calcular a través de ella y su funcionamiento.

En el capítulo 5, “Resultados Experimentales”, se revelan los resultados obtenidos para el caso práctico a través de la herramienta y se comparan con los resultados del artículo de referencia. De esta manera, se verifica el correcto funcionamiento del programa.

En el capítulo 6, “Conclusiones”, se exponen las conclusiones derivadas a partir del análisis y el desarrollo de este proyecto.

Capítulo 7, “Trabajo Futuro”, en él se manifiestan algunas mejoras que se podrían aplicar al programa creado para que estuviera más completo y cumpliera con todos los requisitos del posible usuario final.

Por último, se incluye el anexo denominado “Codificación en Python”. En él se refleja todo el código realizado para la creación del programa y se incluye la explicación de ciertas funciones interesantes contenidas en Python que han sido utilizadas.

2 ANTECEDENTES

*El pasado muere, el presente vive, el recuerdo
queda y la vida sigue.*

- Anónimo -

Como se ha explicado en la introducción, la herramienta creada a través de Python consiste brevemente en el cálculo del instante óptimo para el cual un sistema debe pasar al mantenimiento preventivo a través del uso de un modelo Semi-Markoviano y aplicando la técnica de la transformada z. Luego en este apartado, se han querido introducir 3 elementos:

- En primer lugar, una breve introducción sobre el concepto de optimización y la creación de un modelo de programación matemático.
- En segundo lugar, en qué consiste el mantenimiento, los impactos que supone y los tipos de mantenimiento que se desarrollan dentro de la herramienta. Además de comentar el significado de la implantación del mantenimiento en una empresa y los aspectos a tener en cuenta para poner en marcha la planificación del mantenimiento preventivo.
- Finalmente, puesto que la herramienta se ha desarrollado a través de Python, se realizará una breve introducción a este lenguaje de programación y sus características.

2.1 Introducción a los Modelos de Optimización

Se toman decisiones constantemente para dirigir las acciones llevadas a cabo. La mayor parte de estas decisiones se basan en la intuición. Sin embargo, cuando se plantean disyuntivas de mayor repercusión, ya sean económicas, sociales u ambientales, se considera crucial realizar un previo estudio de las posibles alternativas existentes para un mismo problema [2]. Por tanto, cuanto más importancia tenga la consecuencia, mayor atención se deberá poner al estudio. No obstante, para una mayor dificultad existen instrumentos matemáticos, entre ellos los modelos, que permiten un mejor análisis de la situación y facilitan la resolución del conflicto [3].

2.1.1 Optimización

Cuando se estudian las posibles alternativas existentes para la resolución de una misma acción y se busca la mejor manera de realizar dicha actividad, se denomina optimizar. Asimismo, optimizar se define dentro del campo de las matemáticas como *determinar los valores de las variables que intervienen en un proceso o sistema para que el resultado obtenido sea el mejor posible*.

En otras palabras, se habla de un problema de optimización cuando se busca dar solución a una actividad que se encuentra sujeta a unas normas/especificaciones y participan una serie de elementos y recursos. Igualmente, existen muchas formas de cumplir con esta actividad. Sin embargo, el problema especifica un criterio de valoración con el que se pretende obtener el modo más eficiente de realizar la actividad.

2.1.2 Sistema

Un sistema se presenta como un entorno organizado cuya gestión puede plantearse y resolverse mediante técnicas de optimización. Sobre este entorno se plantea una gestión determinada de sus recursos, cuyo objetivo es realizar una serie de actividades con un criterio de eficiencia.

Para resolver el sistema se debe modelar el problema. Modelar consiste en convertir la descripción del funcionamiento del sistema en un lenguaje matemático utilizable por las técnicas de optimización y programación matemática. Por consiguiente, un modelo corresponde con la representación simplificada de la realidad del problema mediante ecuaciones, funciones o fórmulas matemáticas.

2.1.3 Modelo de optimización

El modelo creado a partir del sistema se denomina Modelo de Programación Matemática o Modelo de Optimización, modelo que finalmente se resolverá para obtener la solución a la gestión del sistema [4].

Un Modelo de Optimización está formado por los siguientes componentes principales:

- **Actividades de decisión:** Son acciones simples de valor indeterminado que se producen en el sistema. Definen las variables principales del modelo y se encuentran asociadas a los elementos del sistema.
- **Elementos/Recursos:** Los elementos son todos los actores/entidades que participan en el sistema. Son de naturaleza diversa, tangibles o intangibles, desde personas, herramientas, lugares, tiempo, etc. y pueden estar sometidos a cualquier proceso.
- **Conjunto de restricciones:** Son todas las acciones definidas, normas determinadas y especificaciones que se deben cumplir en un sistema. La restricción es una expresión matemática mientras que la especificación es una característica que debe cumplir el modelo y que generará una o varias restricciones.
- **Cálculos:** Son funciones que se realizan sobre las variables del sistema para obtener el valor de un parámetro relevante al mismo.
- **Función objetivo:** Se corresponde con los criterios de eficiencia, la dirección de optimización y guía la resolución del sistema.

El **formato general** de un modelo orientado a optimización está formado por:

$$\begin{aligned} & \text{Min } f(x) \\ & \text{sujeto a} \\ & G_i(x) \approx b \quad i=1..m \end{aligned}$$

Figura 2-1: Formato general del modelo de optimización.

Donde:

- x : Vector de variables.
- \approx : Signo de las restricciones [\leq ; $=$; \geq]
- b : Vector de términos independientes
- $G_i(x) \approx b$: Conjunto de m restricciones, $i=1..m$
- $f(x)$: Función objetivo
- **Min:** El objetivo del problema se considera de forma genérica en modo de minimizar una función. Cualquier expresión de maximizar una función se puede convertir en minimizar por una sencilla transformación: $\text{Max } f(x) = \text{Min} - f(x)$

El área de las matemáticas que se centra en el estudio de problemas de optimización es la Investigación Operativa.

2.2 Introducción al Mantenimiento

El mantenimiento se considera el procedimiento que tiene como objetivo tratar un bien determinado, ya sea físico o virtual, de modo que el paso del tiempo, el uso o el cambio de circunstancias externas, no afecten su función o su estado.

Asimismo, se define mantenimiento como una combinación de acciones técnicas y administrativas, incluidas actividades de supervisión, que aseguren que un sistema o equipo se encuentra en un estado de funcionamiento correcto o requerido [5] [6].

Las acciones que se relacionan con el mantenimiento de un sistema son aquellas tales como: reparación, reemplazo, revisión, inspección, servicio, ajuste, prueba, medición y detección de fallos [7] [8].

Todo ello para evitar el deterioro del sistema. De igual modo, si se ejecuta un mantenimiento efectivo se pueden reducir las consecuencias que se producen tras el fallo de un sistema [9]. Las consecuencias se refieren a los impactos, ya sea económicos, medioambientales o en las personas, que conllevarían una mala gestión del mantenimiento.

2.2.1 Tipos de mantenimiento

Dependiendo de los planes para llevar a cabo acciones de mantenimiento, se pueden diferenciar distintos tipos. En concreto para este proyecto se van a tener en cuenta dos:

2.2.1.1 Mantenimiento correctivo

Se pone en marcha en el momento en el que el sistema tiene un fallo de funcionamiento y no trabaja de la manera requerida. Acciones de mantenimiento, como reparación o reemplazo, son puestas en marcha [10]. No obstante, esta política conduce a fallos graves del sistema que implican altos costos de reparación y reemplazo. El mantenimiento correctivo puede ser inmediato o diferido [11]:

- **Mantenimiento inmediato:** Aquel que se realiza sin demora después de que se ha detectado una falla para evitar consecuencias inaceptables.
- **Mantenimiento diferido:** Es aquel que no se ejecuta inmediatamente después de una detección de falla y se retrasa de acuerdo con las reglas de mantenimiento dadas.

2.2.1.2 Mantenimiento preventivo

Las acciones de mantenimiento de un sistema no se efectúan en el momento del fallo, sino que se planifican y se realizan después de un período de tiempo específico para reducir la probabilidad de que éste falle [12]. Conjuntamente, al implantar acciones de mantenimiento preventivo, se aumenta la vida útil de un sistema [13]. Dentro del mantenimiento preventivo se pueden encontrar dos tipos:

- **Mantenimiento predeterminado:** Es de carácter preventivo programado, realizado de acuerdo con los intervalos de tiempo, pero sin investigación previa de la condición del sistema.
- **Mantenimiento basado en condiciones:** Su carácter preventivo está basado en el desempeño y/o monitoreo de parámetros y pronósticos derivados del análisis de la degradación de los equipos.

2.2.2 Impactos del mantenimiento

La gestión del mantenimiento nunca se ha considerado un elemento de interés, no obstante, es importante tenerlo en cuenta ya que afecta al volumen y los costos de producción, el rendimiento de los activos, la disponibilidad del equipo y la calidad del producto final. También afecta a la salud y a la seguridad de las personas [14], el entorno natural y el bienestar social [15].

2.2.2.1 Impacto económico del mantenimiento

Los impactos económicos relacionados con el mantenimiento suelen ser los más investigados en la actualidad. Se relacionan directa e indirectamente con el costo, el tiempo de inactividad, el desglose, el bajo rendimiento, el tiempo de espera, los defectos, el inventario adicional y el transporte adicional, todo aquello que afecte la calidad del producto y la productividad de la planta [16]. En el área de producción de una empresa un mantenimiento inadecuado puede conducir a una baja productividad y, por consiguiente, a la pérdida de oportunidades de mercado y menores ganancias. En cambio, un plan de mantenimiento adecuado proporcionaría un inventario adicional de repuestos y materiales necesarios para mitigar los efectos de las averías y reparar productos defectuosos causados por el mal funcionamiento del sistema. Este trabajo se centrará en el impacto económico del mantenimiento. A pesar de ello, seguidamente, se explicarán otros dos impactos de igual importancia.

2.2.2.2 Impacto medioambiental del mantenimiento

Pese a que los impactos medioambientales relacionados con el mantenimiento no son de menor importancia, no se han realizado demasiados estudios sobre el tema. Se ha destacado la necesidad de considerar estos impactos en las estrategias de mantenimiento [17], debido a que si se activan prácticas de mantenimiento mal definidas, se podrían generar muchos problemas medioambientales, como emisiones peligrosas, desperdicio de producción debido al mal funcionamiento del sistema, uso ineficiente de energía, consumo ineficaz de recursos y/o desperdicio de materiales almacenados [18].

En la Figura 2-2 se muestra una clasificación de los requisitos ecológicos del mantenimiento donde se incluye la compatibilidad ambiental, la eficiencia energética y los riesgos para la salud y la seguridad humana.

De todos los requisitos, los que se exhiben en rectángulos sombreados en la figura se pueden cumplir si se tienen en cuenta en la etapa de diseño. El resto de los requisitos son los que se pueden atender durante la fase de operación de mantenimiento [19].

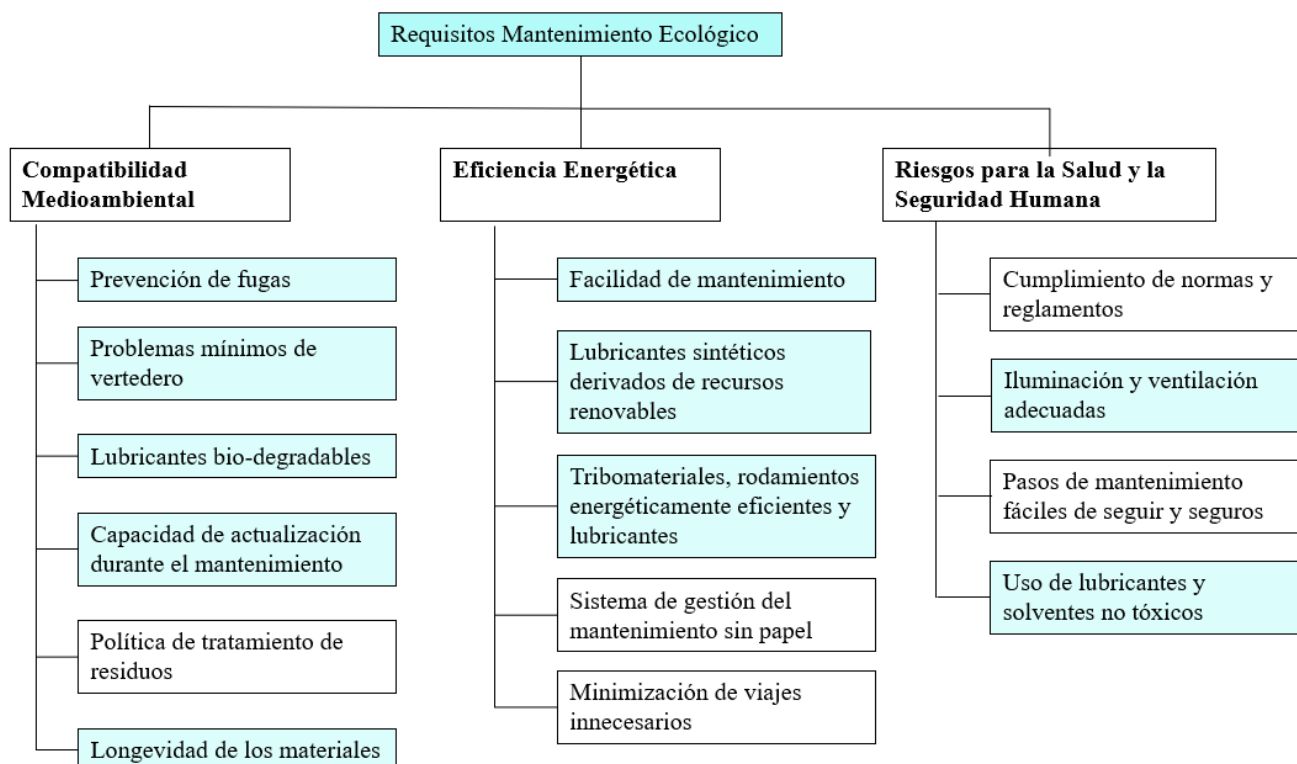


Figura 2-2: Requisitos ecológicos del mantenimiento [19]. [Traducción propia]

Los tribomateriales hacen referencia al desarrollo de nuevos materiales antifricción y de elevada resistencia al desgaste, más baratos y de menor peso [20]

En conclusión, si se quieren poner en marcha estrategias de mantenimiento medioambientalmente proactivas se recomienda considerar estos requisitos ecológicos [21].

2.2.2.3 Impacto del mantenimiento en las personas

El principal impacto social relacionado con el mantenimiento se refiere a la salud y a la seguridad humana. No sólo la de los trabajadores directamente involucrados en las tareas de mantenimiento, sino también la de todos los empleados e incluso la de los clientes. Una mala gestión del mismo causaría condiciones de trabajo inseguras, accidentes, situaciones peligrosas y productos manufacturados de baja calidad [22]. Conjuntamente, si se mira a largo plazo, los efectos peligrosos acumulativos de los contaminantes debido a la ausencia de actividades de mantenimiento pueden influir en el cambio climático y causar posteriores impactos en la salud humana [23].

2.2.3 Implantación del mantenimiento

La implantación del mantenimiento consiste en el seguimiento de unas políticas específicas que definan los planes de acción a realizar, para proporcionar directrices y pautas que gestionen las acciones de mantenimiento adicionales requeridas por un sistema [24].

Tras describir los impactos del mantenimiento, se puede apreciar lo que supone una mala gestión de éste y lo importante que es implantar políticas de mantenimiento.

El estudio de este proyecto se centra en el impacto económico. Por ello, la gestión del mantenimiento debe ser una actividad que se relacione con: asegurar el buen funcionamiento de las instalaciones, sistemas y equipos, lograr buenas condiciones de operación con la máxima disponibilidad, y todo ello con el menor coste posible. Por lo tanto, no solo se deben optimizar los costes en actividades comerciales o productivas, sino también en actividades relacionadas con el mantenimiento para obtener un nivel satisfactorio de productividad que mantenga a la empresa competitiva en los mercados.

Para que los gerentes puedan implantar políticas de mantenimiento, su planificación y control de las actividades derivadas, necesitan información sobre el desempeño del propio sistema. En este sentido, la Norma Europea EN15341: 2007, define el rendimiento del mantenimiento como un resultado de actividades complejas, que pueden evaluarse mediante indicadores apropiados para medir tanto los resultados reales como los esperados [25].

En resumen, para poder crear e implantar políticas de mantenimiento es necesario un estudio previo del sistema, al igual que la creación de herramientas que permitan estudiar la evolución de los resultados, tanto reales como esperados.

2.2.4 Planificación del mantenimiento preventivo

El objetivo de la aplicación que se desarrolla en este proyecto es la planificación del mantenimiento preventivo de un sistema en un periodo de tiempo finito, teniendo en cuenta sus impactos económicos. No todas las planificaciones del mantenimiento preventivo se realizan de la misma manera. Por ello, es necesario realizar una investigación previa que se puede resumir en cuatro facetas:

- Objetivos por los que se realiza la planificación del mantenimiento preventivo.
- Descripción del estado del sistema, su importancia y sus funciones.
- Horizonte de tiempo para el cual se realiza la planificación.
- Métodos que ayudan a determinar las mejores soluciones para los problemas destacados.

2.2.4.1 Objetivos

Los objetivos por los que se realiza la planificación de mantenimiento preventivo pueden ser [26] [27]:

- Fijación de los intervalos de mantenimiento preventivo óptimos: Cálculo del momento óptimo para llevar a cabo acciones de mantenimiento preventivas como el replazo de un sistema o un componente antes de que falle.
- Programación adecuada del trabajo: se busca secuenciar las acciones de mantenimiento preventivas dentro de un intervalo de tiempo.
- Asignación de tareas de mantenimiento preventivo: qué acciones se deben realizar para poner en práctica un correcto mantenimiento preventivo y cómo hacerlas.
- Determinación de la frecuencia de las inspecciones y el mantenimiento óptimo basado en la condición actual o futura de la maquinaria (CBM).
- Determinación de los recursos óptimos para cumplir con los requisitos de mantenimiento.
- Establecimiento del ciclo de vida económico de un equipo ante la disyuntiva de la reparación frente al replazo.

2.2.4.2 Estado del sistema

Un sistema está compuesto por un conjunto de componentes interconectados dispuestos para ejecutar un proceso. Es necesario hacer un estudio del sistema cuando éste está funcionando con normalidad, cuando está funcionando con un fallo y cuando deja de funcionar completamente, de esta manera se pueden planificar medidas de prevención de acuerdo al estado del sistema. La planificación de las acciones a emprender será distinta si se trata de un sistema formado por una sola unidad o un sistema complejo.

- **Sistema único:** Aquel que consta de uno o varios componentes, pero donde la única acción de mantenimiento posible es renovar todo el sistema, mediante un reemplazo completo o mediante una reparación.
- **Sistema múltiple/complejo:** Está compuesto por sistemas que disponen de varios componentes y que forman una unidad. En este caso se pueden producir muchos modos de fallo, donde un solo componente de un sistema puede haber fallado sin afectar a los componentes de otro sistema. Éste puede ser reemplazado o reparado sin necesidad de sustituir el sistema múltiple al completo.

2.2.4.3 Horizonte de tiempo

Otro punto a tener en cuenta dentro de la planificación del mantenimiento preventivo es el horizonte de tiempo para el cual se realiza el estudio, que puede ser finito o infinito.

- Se habla de horizonte de **tiempo finito**, cuando la planificación se llevará a cabo para en un periodo de tiempo marcado por un instante de inicio y un punto final.
- Mientras que el horizonte de **tiempo infinito**, tiene en cuenta el instante de inicio, pero se desarrolla el problema con una situación de tiempo estacionaria.

2.2.4.4 Métodos

La planificación del mantenimiento preventivo puede realizarse con la ayuda de métodos que permitan resolver los problemas que han causado el colapso de un sistema.

A medida que un sistema se vuelve más complicado, se necesita una planificación de mantenimiento preventivo más sofisticada para resolver los problemas de mantenimiento relacionados con el rendimiento del sistema. Por lo tanto, los métodos adecuados que pueden lidiar con los problemas de mantenimiento dinámicamente, deben considerarse como parte del proceso de desarrollo para una planificación óptima del mantenimiento preventivo.

El método es la descripción de cualquier procedimiento a seguir para determinar un plan de mantenimiento preventivo óptimo bajo ciertos requisitos o restricciones de mantenimiento.

El propósito de establecer un método y resolverlo es facilitar la toma de decisiones respecto a la planificación del mantenimiento preventivo. Decisiones basadas en análisis y resultados precisos y eficientes. Varios métodos han sido aplicados por investigadores con respecto a la planificación del mantenimiento preventivo. Éstos se pueden clasificar en estas seis categorías:

Tabla 2–1. Métodos aplicados a la planificación del mantenimiento preventivo. [28]

Métodos	Clasificación	Descripción
Lógica difusa, Algoritmo Genético (GA), Red Neuronal, Cadena de Márkov, Red Bayesiana, Algoritmo Heurístico.	Inteligencia Artificial (AI)	La teoría y el desarrollo de sistemas informáticos que puedan realizar tareas y permitan que los sistemas realicen funciones que normalmente requerirían inteligencia humana [29]
SIMAN, Monte Carlo, Witness	Simulación	Una técnica computable que tiene la capacidad de analizar, diseñar y operar sistemas complejos para una mejor comprensión sin afectar al sistema real [30].

Métodos	Clasificación	Descripción
Programación lineal, Programación no lineal, Programación lineal entera (ILP), Programación dinámica, Programación lineal entera mixta (MILP), Distribución Weibull, Modelo de riesgo proporcional (PHM)	Formulación matemática	Representación matemática para tomar decisiones sobre la mejor asignación posible de recursos escasos [31].
Matriz de coeficientes de similitud	Matriz de formación	Una matriz rectangular que abarca números, símbolos u otros objetos matemáticos para los que se definen operaciones como la suma y la multiplicación [32].
Análisis de modo y efecto de fallo (FMEA), Efecto de modo de fallo y análisis crítico (FMECA)	Análisis crítico	Estudios detallados para identificar el mecanismo de fallo, la criticidad y cómo afectarían a la condición y a la vida útil de un sistema [33].
PROMETHEE, Proceso de red analítica (ANP), Proceso jerárquico analítico (AHP)	Multicriterio	Método de decisión que analiza varias alternativas al considerar todos los criterios en conflicto y los juicios de un decisor [34].

2.3 Introducción a Python y sus Características

Python es un lenguaje de programación. Un lenguaje de programación consiste en un lenguaje informático diseñado para expresar órdenes e instrucciones precisas, que deben ser puestas en marcha por un ordenador. Este lenguaje está compuesto por una serie de símbolos, reglas sintácticas y semánticas que definen la estructura de la lengua.

Se trata de un lenguaje de programación multiparadigma, en otras palabras, soporta programación funcional, programación imperativa y programación orientada a objetos.

Igualmente, Python es multiplataforma. Puede usarse para crear programas en diferentes sistemas operativos como Linux, Mac OS y Windows [35].

A diferencia de la mayoría de los lenguajes de programación, Python dispone de reglas de estilos a fin de poder escribir un código fuente más legible y de manera estandarizada. En definitiva, presenta un lenguaje sencillo a la vez que elegante y flexible. El código fuente hace referencia al conjunto de instrucciones y órdenes lógicas compuestas de algoritmos. Se encuentra escrito en un determinado lenguaje de programación y debe ser interpretado o compilado para permitir la ejecución del programa informático [36].

Otra característica, de igual importancia, es que Python se ejecuta sin necesidad de ser procesado por el compilador, se detectan los errores y se comprueban las variables en el tiempo de ejecución.

Por último, cabe destacar que Python contiene una gran cantidad de librerías, tipos de datos y funciones incorporadas en el propio lenguaje, que ayudan a la realización de muchas tareas comunes sin necesidad de tener que programarlas desde cero [37].

3 FORMALIZACIÓN DEL PROBLEMA

Una persona inteligente resuelve un problema.

Una persona sabia lo evita.

- Albert Einstein -

En el presente capítulo se expondrá el problema que se aborda en el proyecto, el objetivo y la metodología para su resolución. Además de una aportación propia al método y la descripción del caso práctico que se ha utilizado de referencia para la realización del programa.

3.1 Descripción Inicial del Problema

Se dispone de un sistema que se encuentra en un estado de funcionamiento correcto generando ingresos. Sin embargo, cada cierto tiempo el sistema presenta un fallo y éste debe ser reparado con unos costes. Una vez que el sistema es reparado vuelve a funcionar de manera correcta. No obstante, se quiere implantar un periodo donde el equipo pase por un mantenimiento preventivo de tal forma que se reduzca la probabilidad de que falle. Se busca una planificación del mantenimiento preventivo para un horizonte de tiempo finito, basada en los costes e ingresos que supone la implantación de este nuevo periodo, es decir, teniendo en cuenta los costes de reparación, sustitución, repuestos, herramientas, mano de obra, tiempo de inactividad, etc.

3.2 Objetivo

El objetivo del problema consiste en determinar el momento óptimo en el que un sistema, que se encuentra funcionando correctamente, debe pasar a un estado donde se apliquen medidas de mantenimiento preventivo, de manera que se maximice el retorno (ingresos menos costes) promedio acumulado en un horizonte de tiempo finito

3.3 Resolución

Para llevar a cabo la resolución de este problema de mantenimiento, se ha elegido un proceso Semi-Markoviano (SMP) como el formato estocástico para modelar el mismo. También existen unos costes e ingresos asociados a cada estado del sistema que se acumulan en cada transición del proceso, hecho que permite calcular el retorno acumulado promedio en m transiciones.

El modelo da origen a un sistema de ecuaciones, que pueden expresarse como un sistema de ecuaciones en diferencias.

Para resolver el sistema de ecuaciones en diferencias para el retorno acumulado promedio en m transiciones, se utiliza la técnica de la z transformada.

Una vez calculado el retorno acumulado promedio, a partir de su derivada, respecto del tiempo hasta el preventivo igualada a cero, se ha podido obtener el momento óptimo para realizar el mantenimiento preventivo.

Toda la resolución del método fue tomada de acuerdo a Sánchez-Herguedas [1].

3.3.1 Modelo Semi-Markoviano

Para este caso concreto, se define un modelo Semi-Markoviano de 3 estados, es decir, el sistema en cuestión se puede encontrar en:

E1: Estado *Operativo*. Momento en el que el sistema se encuentra funcionando correctamente.

E2: Estado *Correctivo*. Hace referencia a cuando el sistema está siendo reparado debido a un fallo.

E3: Estado *Preventivo*. Momento en el que el sistema se encuentra en un estado donde se están realizando acciones de mantenimiento preventivas.

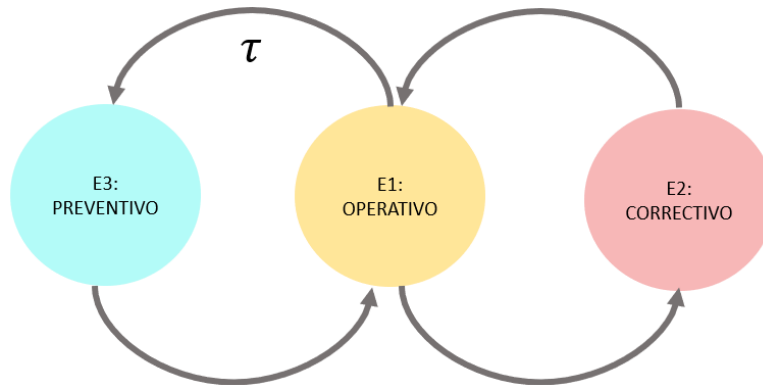


Figura 3-1: Esquema cambio de fases del sistema. [Fuente: Elaboración propia].

Hasta ahora las transiciones entre los estados tienen lugar en tiempos aleatorios. Cada cambio de estado corresponde a un paso de transición m , con $m = 1,2,3\dots$. Cada transición entre estados tiene asociada una probabilidad p , siendo p_{ij} la probabilidad de que el equipo pase del estado i al estado j .

En el instante inicial, el equipo se encuentra en el estado *Operativo*. Si tras un tiempo τ , el equipo no falla, el sistema hace una transición hacia el estado *Preventivo*, en el que se realiza la tarea de mantenimiento. Una vez terminada la tarea preventiva el sistema volvería al estado *Operativo* con probabilidad 1. Si antes de haber alcanzado el tiempo τ se produce el fallo, el sistema pasaría directamente del estado *Operativo* al *Correctivo*. El sistema compuesto por los tres estados evoluciona durante el tiempo que se estime oportuno (tiempo finito) mediante el mecanismo de ir saltando de un estado a otro de acuerdo con ciertas probabilidades de transición, formando un proceso de Márkov homogéneo.

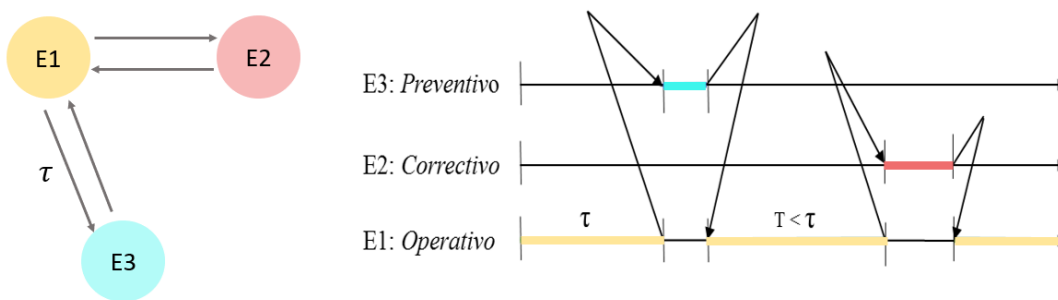


Figura 3-2: Transición de un estado a otro. [Fuente: Elaboración propia].

Se define P como la matriz de probabilidades de transición de un estado a otro. De igual importancia, F es la matriz de tiempos de permanencia en un estado. Donde F_{ij} es el tiempo medio del sistema en el estado i antes de pasar al estado j .

$$P = \begin{pmatrix} 0 & F(\tau) & 1 - F(\tau) \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix} ; F = \begin{pmatrix} 0 & A & \tau \\ B & 0 & 0 \\ C & 0 & 0 \end{pmatrix} \tag{3-1}$$

Siendo A el tiempo promedio que el sistema permanece en el estado *Operativo* antes de que ocurra la falla en el intervalo $(0, \tau)$. El tiempo de falla en ese intervalo es una variable aleatoria truncada. Por otra parte, τ es el tiempo que el sistema permanece en el estado *Operativo* antes de pasar al *Preventivo*. Finalmente, B es el tiempo promedio que el sistema permanece en el estado *Correctivo* y C el tiempo promedio que el sistema se encuentra en el estado *Preventivo* antes de pasar al *Correctivo*. Ninguno de estos dos últimos está sujeto a truncamiento. Los componentes de cada uno de ellos se expresan mediante estas ecuaciones:

$$A = \frac{1}{F(\tau)} \int_0^{\tau} t \cdot f(t) dt; \quad B = \int_0^{\infty} t_c g(t_c) dt_c; \quad C = \int_0^{\infty} t_p h(t_p) dt_p \quad (3-2)$$

Siendo los parámetros utilizados:

- $F(t)$ es la función de distribución acumulativa de fallos del equipo, $f(t)$ su función de densidad de probabilidad.
- $G(t_c)$ es la función de distribución del tiempo que el equipo permanece bajo mantenimiento correctivo y $g(t_c)$ su función de densidad de probabilidad.
- $H(t_p)$ es la función de distribución del tiempo que el equipo permanece bajo mantenimiento preventivo y $h(t_p)$ su función de densidad de probabilidad.

3.3.2 Costes e ingresos

Cuando el equipo se encuentra en el estado *Operativo* funcionará produciendo unos ingresos. En el momento que pasa al estado *Correctivo*, tras un fallo, acumula unos costes de mano de obra de reparación, de repuesto y de indisponibilidad. Y si el sistema no falla y pasa al estado *Preventivo* en el tiempo τ , se generan de igual modo, unos costes asociados a los repuestos, a la mano de obra y a la inactividad del sistema.

Ya sea un ingreso o un coste, este valor económico se ha denominado retorno. Siendo los ingresos, retornos positivos y, por el contrario, los costes, retornos negativos. Los retornos se denotan con una r_i , siendo i el estado en el que se encuentra el equipo.

Los cambios de estado también conllevan unos retornos, y se expresan de esta manera: r_{ij} ; siendo i el estado en el que se encuentra el equipo y j el estado al que pasaría.

Los retornos que se presentan son (costes e ingresos):

- R_1 , ingreso por unidad de tiempo que el sistema permanece en el estado 1 (E_1 : *Operativo*).
- R_2 , coste por unidad de tiempo que el sistema permanece en el estado 2 (E_2 : *Correctivo*).
- R_3 , coste por unidad de tiempo que el sistema permanece en el estado 3 (E_3 : *Preventivo*).
- R_{12} , coste de transición desde el estado 1 al estado 2.
- R_{21} , coste de transición desde el estado 2 al 1.
- R_{13} , coste de transición desde el estado 1 al 3.
- R_{31} , coste de transición desde el estado 3 al 1.

Partiendo de F , Eq.(3-1), se obtiene que la matriz de retornos R , donde $r_{ij} = R_i + R_{ij}$, es:

$$R = \begin{pmatrix} 0 & r_{12} & r_{13} \\ r_{21} & 0 & 0 \\ r_{31} & 0 & 0 \end{pmatrix} = \begin{pmatrix} 0 & R_1 A + R_{12} & R_1 \tau + R_{13} \\ R_2 B + R_{21} & 0 & 0 \\ R_3 C + R_{31} & 0 & 0 \end{pmatrix}. \quad (3-3)$$

A medida que el sistema va cambiando de estado, éste acumulará retornos, constituyendo el retorno acumulado en m transiciones o cambios de estado. El valor del retorno acumulado es un valor aleatorio ya que es posible determinar el estado inicial del equipo, pero es impredecible determinar el siguiente y los que siguen. De esta forma, el sistema puede evolucionar de muchas maneras en m transiciones. Ésta es la razón por la que interesa maximizar el retorno acumulado medio, y no el retorno acumulado en sí [1].

El retorno esperado en un solo paso se puede formular como:

$$v_i(1) = \sum_{j=1}^n r_{ij}(1) p_{ij} \quad (3-4)$$

Y el valor esperado del retorno de los $m - 1$ pasos restantes se puede formular como:

$$E(v_j(m - 1)) = \sum_{k=1}^n v_{jk}(m - 1) p_{jk} \quad (3-5)$$

Luego, el retorno esperado total del sistema en m pasos consiste en:

$$v_i(m) = v_i(1) + E(v_j(m - 1)) = v_i(1) + \sum_{k=1}^n v_{jk}(m - 1) p_{jk} \quad (3-6)$$

Si se define el vector $V(m) = (v_1(m), v_2(m), \dots, v_n(m))^t$, donde el superíndice t indica transposición en el sentido matricial, la ecuación anterior podría escribirse como una ecuación matricial en diferencias:

$$V(m) = V(1) + P V(m - 1) \quad (3-7)$$

Donde P es la matriz de probabilidades de transición. Siendo esta ecuación la que proporciona el retorno esperado en m pasos desde cualquier posible estado de partida i .

3.3.3 Explicación de la resolución del sistema de ecuaciones en diferencias aplicando la transformada z

Este es el momento en el que, para poder resolver el sistema de ecuaciones en diferencias descrito anteriormente, Eq.(3-7), como se anunció al principio, es necesario aplicar la técnica de la transformada z .

La transformada z es una función de variable compleja que se define como la siguiente serie de Laurent:

$$\mathcal{Z}[x(m)] = \sum_{n=0}^{\infty} x(n)z^{-n}, \quad z \in \mathbb{C} \setminus \{0\}. \quad (3-8)$$

La transformada z del vector es otro vector cuyas componentes son las transformadas z de los componentes. Es conveniente reescribir la ecuación en diferencias con el índice aumentado en una unidad, es decir: $V(m + 1) = V(1) + P V(m)$, y aplicarle la transformada z a ambos miembros de la igualdad. Para ello se tiene en cuenta la linealidad de la transformada y el hecho de que la matriz P es constante (es decir, no depende de m) [1]:

$$\mathcal{Z}[V(m + 1)] = \mathcal{Z}[V(1)] + P \mathcal{Z}[V(m)] \quad (3-9)$$

Usando las propiedades de la transformada z se alcanza:

$$z(Z[V(m)] - V(0)) = V(1) \frac{z}{z-1} + P Z[V(m)] \quad (3-10)$$

Multiplicando por la inversa de la matriz $I - z^{-1}P$ (que es regular), dividiendo por z y reordenando los términos, se obtiene:

$$Z[V(m)] = \frac{1}{z-1} (I - z^{-1}P)^{-1}V(1) + (I - z^{-1}P)^{-1}V(0) \quad (3-11)$$

El vector de retornos esperados inicial $V(0)$ es el vector nulo, ya que no hay ningún retorno anterior al instante inicial en que comienza la evolución de la cadena. Por lo tanto:

$$[V(m)] = \frac{1}{z-1} (I - z^{-1}P)^{-1}V(1) \quad (3-12)$$

Ahora, solamente queda invertir esta transformada para obtener los retornos medios en m pasos $V(m)$.

3.3.4 Aplicación de la transformada z al sistema de ecuaciones en diferencias del modelo Semi-Markoviano de 3 estados.

Una vez explicado el desarrollo de las ecuaciones para la transformada z , se aplica al modelo de tres estados presentado anteriormente para obtener el valor de los retornos medios en m pasos, $V(m)$.

En primer lugar, teniendo en cuenta que $V(0) = 0$, porque no hay retorno antes del primer paso, y simplificando $F(\tau) = p$, el vector $V(1)$, según la ecuación, Eq.(3-4), sería:

$$\begin{pmatrix} v_1(1) \\ v_2(1) \\ v_3(1) \end{pmatrix} = \begin{pmatrix} r_{12}p + r_{13}(1-p) \\ r_{21} \\ r_{31} \end{pmatrix} \quad (3-13)$$

Si se desarrolla la matriz $(I - z^{-1}P)^{-1}$:

$$(I - z^{-1}P)^{-1} = \frac{1}{z^2 - 1} \begin{pmatrix} z^2 & zp & z(1-p) \\ z & z^2 + p - 1 & 1-p \\ z & p & z^2 - p \end{pmatrix} \quad (3-14)$$

El resultando es:

$$Z[V(m)] = \frac{1}{(z-1)^2(z+1)} \begin{pmatrix} (r_{12}p + r_{13}(1-p))z^2 + (r_{21}p + r_{31}(1-p))z \\ (r_{12}p + r_{13}(1-p))z + r_{21}z^2 + (r_{31} - r_{21})(1-p) \\ (r_{12}p + r_{13}(1-p))z + r_{31}z^2 + (r_{21} - r_{31})p \end{pmatrix} \quad (3-15)$$

Las tres componentes de la Eq.(3-15), son combinaciones lineales de funciones racionales en z .

$$\frac{z^2}{(z-1)^2(z+1)} \quad \frac{z}{(z-1)^2(z+1)} \quad \frac{1}{(z-1)^2(z+1)} \quad (3-16)$$

Si se descomponen en fracciones simples:

$$\begin{aligned} \frac{z^2}{(z-1)^2(z+1)} &= \frac{1}{4} \left[\frac{1}{z+1} + \frac{3}{z-1} + \frac{2}{(z-1)^2} \right] \\ \frac{z}{(z-1)^2(z+1)} &= \frac{1}{4} \left[\frac{1}{z+1} + \frac{3}{z-1} + \frac{2}{(z-1)^2} \right] \\ \frac{1}{(z-1)^2(z+1)} &= \frac{1}{4} \left[\frac{1}{z+1} - \frac{1}{z-1} + \frac{2}{(z-1)^2} \right] \end{aligned} \quad (3-17)$$

Si se introducen las próximas notaciones para reducir el tamaño de las ecuaciones:

$$\begin{aligned} a &= r_{12}p + r_{13}(1-p), & b_1 &= r_{21}p + r_{31}(1-p), & b_2 &= r_{21}, \\ b_3 &= (r_{31} - r_{21})(1-p), & b_4 &= r_{31}, & b_5 &= (r_{21} - r_{31})p. \end{aligned} \quad (3-18)$$

Los tres componentes de la z transformada de $\mathbf{V}(\mathbf{m})$ serían:

$$\begin{aligned} Z[v_1(m)] &= \frac{1}{4} \left[\frac{a-b_1}{z+1} + \frac{3a+b_1}{z-1} + \frac{2(a+b_1)}{(z-1)^2} \right] \\ Z[v_2(m)] &= \frac{1}{4} \left[-\frac{a-b_2-b_3}{z+1} + \frac{a+3b_2-b_3}{z-1} + \frac{2(a+b_2+b_3)}{(z-1)^2} \right] \\ Z[v_3(m)] &= \frac{1}{4} \left[-\frac{a-b_4-b_5}{z+1} + \frac{a+3b_4-b_5}{z-1} + \frac{2(a+b_4+b_5)}{(z-1)^2} \right] \end{aligned} \quad (3-19)$$

Para cada una de ellas se puede encontrar su desarrollo en serie de Laurent y a partir de ahí, buscar sus transformadas inversas para obtener los retornos medios en \mathbf{m} pasos, $\mathbf{V}(\mathbf{m})$.

El desarrollo en serie de Laurent de cada una de las fracciones simples sería:

$$\begin{aligned} \frac{1}{z+1} &= \frac{1}{z} \cdot \frac{1}{1 - (-1/z)} = \frac{1}{z} \sum_{n=0}^{\infty} \left(\frac{-1}{z}\right)^n = \sum_{n=0}^{\infty} \frac{(-1)^n}{z^{n+1}} \\ \frac{1}{z-1} &= \frac{1}{z} \cdot \frac{1}{1 - (1/z)} = \frac{1}{z} \sum_{n=0}^{\infty} \frac{1}{z^n} = \sum_{n=0}^{\infty} \frac{1}{z^{n+1}} \\ \frac{1}{(z-1)^2} &= -\frac{d}{dz} \left(\frac{1}{z-1} \right) = -\frac{d}{dz} \sum_{n=0}^{\infty} \frac{1}{z^{n+1}} = -\sum_{n=0}^{\infty} \frac{d}{dz} \left(\frac{1}{z^{n+1}} \right) = \sum_{n=0}^{\infty} \frac{n+1}{z^{n+2}} \end{aligned} \quad (3-20)$$

Siendo la región de convergencia de estas series de Laurent $|z| > 1$, la misma para las tres.

El desarrollo en serie de Laurent completo sería:

Para el primer componente:

$$\begin{aligned} \mathcal{Z}[v_1(m)] &= \frac{1}{4} \left[\sum_{n=0}^{\infty} \frac{(-1)^n (a - b_1)}{z^{n+1}} + \sum_{n=0}^{\infty} \frac{3a + b_1}{z^{n+1}} + \sum_{n=0}^{\infty} \frac{2(a + b_1)(n + 1)}{z^{n+2}} \right] = \quad (3-21) \\ &= \frac{1}{4} \left[\frac{a - b_1}{z} + \sum_{n=0}^{\infty} \frac{(-1)^{n+1} (a - b_1)}{z^{n+2}} + \frac{3a + b_1}{z} + \sum_{n=0}^{\infty} \frac{3a + b_1}{z^{n+2}} + \sum_{n=0}^{\infty} \frac{2(a + b_1)(n + 1)}{z^{n+2}} \right] = \\ &= \frac{a}{z} + \frac{1}{4} \sum_{n=0}^{\infty} \frac{(5 + 2n + (-1)^{n+1})a + (3 + 2n - (-1)^{n+1})b_1}{z^{n+2}} \end{aligned}$$

Para el segundo componente:

$$\begin{aligned} \mathcal{Z}[v_2(m)] &= \frac{b_2}{z} + \quad (3-22) \\ &= \frac{1}{4} \sum_{n=0}^{\infty} \frac{(3 + 2n - (-1)^{n+1})a + (5 + 2n + (-1)^{n+1})b_2 + (1 + 2n + (-1)^{n+1})b_3}{z^{n+2}} \end{aligned}$$

Para el tercer componente:

$$\begin{aligned} \mathcal{Z}[v_3(m)] &= \frac{b_4}{z} + \quad (3-23) \\ &= \frac{1}{4} \sum_{n=0}^{\infty} \frac{(3 + 2n - (-1)^{n+1})a + (5 + 2n + (-1)^{n+1})b_4 + (1 + 2n + (-1)^{n+1})b_5}{z^{n+2}} \end{aligned}$$

Una vez que se tienen estas ecuaciones, se puede buscar sus transformadas inversas para obtener los retornos medios en m pasos, $V(m)$.

3.3.5 Obtención de la inversa de la transformada z

En esta ocasión, el sistema parte del estado *Operativo*, por lo que en este apartado se mostrará la inversa de la transformada z únicamente del primer componente de $V(m)$:

La inversa de la transformada z para el primer componente de $V(m)$, $v_1(m)$, se presenta con la siguiente ecuación:

$$v_1(m) = \frac{1}{4} [(5 + 2n + (-1)^{n+1})a + (3 + 2n - (-1)^{n+1})b_1] \quad (3-24)$$

Sustituyendo m por $n + 2$ quedaría:

$$\begin{aligned} v_1(0) &= 0 \\ v_1(1) &= a, \\ v_1(m) &= \frac{1}{4} [(2m + 1 + (-1)^{m-1})a + (2m - 1 - (-1)^{m-1})b_1] \\ &\text{for } m = 2, 3, 4, \dots \end{aligned} \quad (3-25)$$

Seguidamente sustituyendo a y b_1 por sus valores expresados en las Eq. (3–18):

$$\begin{aligned} v_1(0) &= 0 \\ v_1(1) &= r_{12}p + r_{13}(1-p), \\ v_1(m) &= \frac{1}{4}[(2m+1+(-1)^{m-1})(r_{12}p + r_{13}(1-p)) + (2m-1-(-1)^{m-1})(r_{21}p + r_{31}(1-p))] \\ &\quad \text{for } m = 2,3,4, \dots \end{aligned} \quad (3-26)$$

El próximo paso es sustituir los valores de r_{12} , r_{13} , r_{21} , r_{31} , usando la Eq. (3–3):

$$\begin{aligned} v_1(0) &= 0 \\ v_1(1) &= (R_1A + R_{12}) * F(\tau) + (R_1\tau + R_{13}) * (1 - F(\tau)) \\ v_1(m) &= \frac{1}{4}[(2m+1+(-1)^{m-1})((R_1A + R_{12}) * F(\tau) + (R_1\tau + R_{13}) * (1 - F(\tau))) \\ &\quad + (2m-1-(-1)^{m-1}) * ((R_2B + R_{21}) * F(\tau) + (R_3C + R_{31}) * (1 - F(\tau)))] \end{aligned} \quad (3-27)$$

Sustituyendo los valores de A , B y C de las ecuaciones, Eq. (3–2), se obtiene la ecuación para $v_1(m)$:

$$\begin{aligned} v_1(m) &= \frac{1}{4} \left[(2m+1+(-1)^{m-1}) \left(R_1 \int_0^\tau t f(t) dt + R_{12} F(\tau) + (R_1\tau + R_{13})(1 - F(\tau)) \right) \right. \\ &\quad + (2m-1-(-1)^{m-1}) \left(\left(R_2 \int_0^\infty t_c g(t_c) dt_c + R_{21} \right) F(\tau) \right. \\ &\quad \left. \left. + \left(R_3 \int_0^\infty t_p h(t_p) dt_p + R_{31} \right) (1 - F(\tau)) \right) \right] \end{aligned} \quad (3-28)$$

3.3.6 Optimización del retorno esperado al comenzar el sistema desde el estado *Operativo*

Como se comentó al inicio, el objetivo de este problema es la obtención del τ óptimo, que maximice el retorno promedio acumulado. En otras palabras, averiguar el instante en el que el sistema debe pasar al estado de mantenimiento *Preventivo*, de manera que se obtenga el máximo valor para la diferencia entre los ingresos y los gastos que supone la implantación de éste.

Para ello se parte de la ecuación resultante de $v_1(m)$ del apartado anterior, Eq. (3–28), y se deriva con respecto a τ , se iguala a cero y se obtiene el valor óptimo de τ .

En primer lugar, se realiza la derivada de $v_1(m)$ respecto a τ .

$$\begin{aligned} \frac{dv_1(m)}{d\tau} &= \frac{1}{4} \left[f(\tau) \left((2m+1+(-1)^{m-1})(R_{12} - R_{13}) \right. \right. \\ &\quad \left. \left. + (2m-1-(-1)^{m-1})(R_2B + R_{21} - R_3C - R_{31}) \right) \right] \\ &\quad + (2m+1+(-1)^{m-1}) \frac{R_1}{4} (1 - F(\tau)). \end{aligned} \quad (3-29)$$

Puesto que dentro de las ecuaciones de A , B y C (Eq.(3-2)), la que contiene τ es la ecuación para A , ésta ha sido la única que se ha sustituido dentro de la ecuación principal de $v_1(m)$. La sustitución de las ecuaciones para B y C no han sido necesarias para la resolución de la derivada respecto de τ , ya que no contienen este elemento.

El siguiente paso es sustituir $F(\tau)$ y $f(\tau)$ con sus ecuaciones. Para este caso, se ha tomado la distribución de Weibull de tres parámetros, ya que fue la que mejor se ajustaba a la distribución existente en el caso práctico realizado en el estudio [1].

$$f(t) = \frac{\alpha}{\beta} \left(\frac{t-\gamma}{\beta} \right)^{\alpha-1} e^{-\left(\frac{t-\gamma}{\beta}\right)^\alpha}, \quad F(t) = 1 - e^{-\left(\frac{t-\gamma}{\beta}\right)^\alpha}, \quad \alpha, \beta, \gamma > 0; \quad (3-30)$$

Al sustituir queda:

$$\frac{dv_1(m)}{d\tau} = \frac{e^{-\left(\frac{\tau}{\beta}\right)^\alpha}}{4} (2m+1 + (-1)^{m-1}) \cdot \left[\frac{\alpha}{\beta} \left(\frac{\tau}{\beta} \right)^{\alpha-1} \left(R_{12} - R_{13} + \frac{2m-1 - (-1)^{m-1}}{2m+1 + (-1)^{m-1}} (R_2B + R_{21} - R_3C - R_{31}) \right) + R_1 \right] \quad (3-31)$$

En segundo lugar, se iguala a cero la ecuación:

$$\frac{\alpha}{\beta} \left(\frac{\tau}{\beta} \right)^{\alpha-1} \left(R_{12} - R_{13} + \frac{2m-1 - (-1)^{m-1}}{2m+1 + (-1)^{m-1}} (R_2B + R_{21} - R_3C - R_{31}) \right) + R_1 = 0 \quad (3-32)$$

Finalmente, se despeja el parámetro τ :

$$\tau^{\alpha-1} = \frac{\beta^\alpha}{\alpha} \cdot \frac{-R_1}{R_{12} - R_{13} + \frac{2m-1 - (-1)^{m-1}}{2m+1 + (-1)^{m-1}} (R_2B + R_{21} - R_3C - R_{31})} \quad (3-33)$$

Con esta ecuación resultante, se puede calcular el instante óptimo en el que el sistema debe pasar al estado de mantenimiento *Preventivo*, maximizando el retorno promedio acumulado comenzando en el estado *Operativo*.

3.4 Aportación al Método

En el método anteriormente descrito [1] el valor de τ óptimo únicamente se calculaba para un sistema que parte del estado *Operativo*. No obstante, la aportación de este trabajo al método desarrollado consiste en calcular de igual forma τ óptimo, pero inicializando el sistema en el estado *Correctivo* y el estado *Preventivo*. Luego, se va a calcular τ a partir del segundo y tercer componente de $\mathbf{V}(\mathbf{m})$, $\mathbf{v}_2(\mathbf{m})$ y $\mathbf{v}_3(\mathbf{m})$, es decir, inicializando el sistema en el estado *Correctivo* y después desde el estado *Preventivo*.

3.4.1 Inicialización del sistema en el estado *Correctivo*

Para ello se toma la ecuación de $\mathcal{Z}[\mathbf{v}_2(\mathbf{m})]$, Eq. (3-22), y se realiza la inversa de la transformada z:

$$v_2(m) = \frac{1}{4} [(3 + 2n - (-1)^{n+1})a + (5 + 2n + (-1)^{n+1})b_2 + (1 + 2n + (-1)^{n+1})b_3] \quad (3-34)$$

Se sustituye \mathbf{m} por $n + 2$:

$$v_2(0) = 0 \quad (3-35)$$

$$v_2(1) = b_2$$

$$v_2(m) = \frac{1}{4} [(-1 + 2m - (-1)^{m-1})a + (1 + 2m + (-1)^{m-1})b_2 + (2m - 3 + (-1)^{m-1})b_3]$$

Para $m = 2, 3, 4, \dots$

Sustituyendo los valores \mathbf{a} , \mathbf{b}_2 , \mathbf{b}_3 por las ecuaciones, Eq. (3-18), los valores de \mathbf{r}_{12} , \mathbf{r}_{13} , \mathbf{r}_{21} , \mathbf{r}_{31} , usando la Eq. (3-3), y posteriormente sustituyendo \mathbf{A} , \mathbf{B} y \mathbf{C} por sus ecuaciones, Eq. (3-2), se obtiene:

$$v_2(m) = \frac{1}{4} \left[(-1 + 2m - (-1)^{m-1}) * \left(R_1 \int_0^\tau tf(t)dt + R_{12}F(\tau) + (R_1\tau + R_{13})(1 - F(\tau)) \right) \right. \quad (3-36)$$

$$+ (1 + 2m + (-1)^{m-1}) * (R_2 * \left(\int_0^\infty t_c g(t_c)dt_c \right) + R_{21})$$

$$+ (2m - 3 + (-1)^{m-1}) * ((R_3 \left(\int_0^\infty t_p h(t_p)dt_p \right) + R_{31})$$

$$\left. - (R_2 \left(\int_0^\infty t_c g(t_c)dt_c \right) + R_{21})) * (1 - F(\tau)) \right]$$

El próximo paso sería realizar la derivada de $\mathbf{v}_2(\mathbf{m})$ respecto de τ . Como se puede apreciar en la ecuación anterior, Eq. (3-36), las ecuaciones sustituidas para \mathbf{B} y \mathbf{C} no contienen el parámetro τ , luego para simplificar la ecuación, estos se mostrarán sin sustituir:

$$v_2(m) = \frac{1}{4} \left[(-1 + 2m - (-1)^{m-1}) * \left(R_1 \int_0^\tau tf(t)dt + R_{12}F(\tau) + (R_1\tau + R_{13})(1 - F(\tau)) \right) \right. \quad (3-37)$$

$$+ (1 + 2m + (-1)^{m-1}) * (R_2B + R_{21}) + (2m - 3 + (-1)^{m-1}) * ((R_3C$$

$$+ R_{31}) - (R_2B + R_{21})) * (1 - F(\tau)) \left. \right]$$

Se realiza la derivada:

$$\begin{aligned} \frac{dv_2(m)}{d\tau} = \frac{1}{4} & \left[(-1 + 2m - (-1)^{m-1})(R_1\tau f(\tau) + R_{12}f(\tau) + R_1 - R_1F(\tau) - R_1\tau f(\tau) \right. \\ & - R_{13}f(\tau)) \\ & + (1 + 2m - (-1)^{m-1})(R_2B + R_{21}) * 0 \\ & \left. + (2m - 3 + (-1)^{m-1})(R_3C + R_{31} - R_2B - R_{21})(-f(\tau)) \right] \end{aligned} \quad (3-38)$$

Al simplificar la derivada se obtiene:

$$\begin{aligned} \frac{dv_2(m)}{d\tau} = \frac{1}{4} & \left[f(\tau)((-1 + 2m - (-1)^{m-1})(R_{12} - R_{13}) \right. \\ & \left. + (2m - 3 + (-1)^{m-1})(R_2B + R_{21} - R_3C - R_{31})) \right] + (-1 + 2m - (-1)^{m-1}) \\ & \cdot \frac{R_1}{4} (1 - F(\tau)). \end{aligned} \quad (3-39)$$

Se sustituyen las ecuaciones de $f(t)$ y $F(t)$, Eq. (3-30):

$$\begin{aligned} \frac{dv_2(m)}{d\tau} = \frac{e^{-\left(\frac{\tau}{\beta}\right)^\alpha}}{4} & (-1 + 2m - (-1)^{m-1}) \\ & \cdot \left[\frac{\alpha}{\beta} \left(\frac{\tau}{\beta}\right)^{\alpha-1} \left(R_{12} - R_{13} + \frac{2m - 3 + (-1)^{m-1}}{-1 + 2m - (-1)^{m-1}} (R_2B + R_{21} - R_3C - R_{31}) \right) \right. \\ & \left. + R_1 \right] \end{aligned} \quad (3-40)$$

Se iguala la derivada a cero:

$$\frac{\alpha}{\beta} \left(\frac{\tau}{\beta}\right)^{\alpha-1} \left(R_{12} - R_{13} + \frac{2m - 3 + (-1)^{m-1}}{-1 + 2m - (-1)^{m-1}} (R_2B + R_{21} - R_3C - R_{31}) \right) + R_1 = 0 \quad (3-41)$$

Finalmente se despeja el valor de τ :

$$\tau^{\alpha-1} = \frac{\beta^\alpha}{\alpha} \cdot \frac{-R_1}{R_{12} - R_{13} + \frac{2m - 3 + (-1)^{m-1}}{-1 + 2m - (-1)^{m-1}} (R_2B + R_{21} - R_3C - R_{31})} \quad (3-42)$$

Con esta ecuación se obtiene el instante óptimo en el que el sistema debe pasar al estado *Preventivo* después de m transiciones, habiendo comenzado desde el estado *Correctivo*.

3.4.2 Inicialización del sistema en el estado *Preventivo*

En el caso de querer calcular τ a partir del tercer componente de $\mathbf{V}(\mathbf{m})$, $v_3(\mathbf{m})$, se realizarían exactamente los mismos pasos. Si se aprecian las ecuaciones para $\mathcal{Z}[v_2(\mathbf{m})]$ y $\mathcal{Z}[v_3(\mathbf{m})]$, ambas son iguales, con la única diferencia que \mathbf{b}_2 y \mathbf{b}_3 son sustituidos por \mathbf{b}_4 y \mathbf{b}_5 .

Se realiza la inversa de $\mathcal{Z}[v_3(\mathbf{m})]$:

$$v_3(m) = \frac{1}{4} [(-1 + 2m - (-1)^{m-1})a + (1 + 2m + (-1)^{m-1})b_4 + (2m - 3 + (-1)^{m-1})b_5] \quad (3-43)$$

Se sustituye $m = n+2$:

$$v_3(0) = 0 \quad (3-44)$$

$$v_3(1) = b_4$$

$$v_3(m) = \frac{1}{4} [(-1 + 2m - (-1)^{m-1})a + (1 + 2m + (-1)^{m-1})b_4 + (2m - 3 + (-1)^{m-1})b_5]$$

$$m = 2, 3, 4, \dots$$

Sustituyendo los valores a , \mathbf{b}_4 , \mathbf{b}_5 por las ecuaciones, Eq.(3-18), los valores de \mathbf{r}_{12} , \mathbf{r}_{13} , \mathbf{r}_{21} , \mathbf{r}_{31} , usando la Eq.(3-3), y posteriormente sustituyendo A , Eq. (3-2):

$$v_3(m) = \frac{1}{4} \left[(-1 + 2m - (-1)^{m-1}) * \left(R_1 \int_0^\tau tf(t)dt + R_{12}F(\tau) + (R_1\tau + R_{13})(1 - F(\tau)) \right) \right. \\ \left. + (1 + 2m + (-1)^{m-1}) * (R_3C + R_{31}) + (2m - 3 + (-1)^{m-1}) * ((R_2B + R_{21}) - (R_3C + R_{31})) * F(\tau) \right] \quad (3-45)$$

Se realiza la derivada respecto de τ :

$$\frac{dv_3(m)}{d\tau} = \frac{1}{4} [(-1 + 2m - (-1)^{m-1})(R_1\tau f(\tau) + R_{12}f(\tau) + R_1 - R_1F(\tau) - R_1\tau f(\tau) - R_{13}f(\tau)) \\ + (1 + 2m - (-1)^{m-1})((R_3C + R_{31})) * 0 \\ + (2m - 3 + (-1)^{m-1})(R_2B + R_{21} - R_3C - R_{31})(f(\tau))] \\ = \frac{1}{4} [f(\tau)((-1 + 2m - (-1)^{m-1})(R_{12} - R_{13}) \\ + (2m - 3 + (-1)^{m-1})(R_2B + R_{21} - R_3C - R_{31}))] \\ + (-1 + 2m - (-1)^{m-1}) \frac{R_1}{4} (1 - F(\tau)). \quad (3-46)$$

Si se aprecia el resultado de la derivada para $v_3(\mathbf{m})$, es exactamente igual que para $v_2(\mathbf{m})$, luego el valor de τ va a ser el mismo:

$$\tau^{\alpha-1} = \frac{\beta^\alpha}{\alpha} \cdot \frac{-R_1}{R_{12} - R_{13} + \frac{2m - 3 + (-1)^{m-1}}{-1 + 2m - (-1)^{m-1}} (R_2B + R_{21} - R_3C - R_{31})} \quad (3-47)$$

Con esta ecuación se obtiene el instante óptimo en el que el sistema debe pasar al estado *Preventivo* después de m transiciones, habiendo comenzado desde el estado *Preventivo*.

3.5 Descripción del Caso Práctico

Una vez descrito el problema general, el objetivo, su modelaje, y su resolución, se detallará el caso práctico que se ha utilizado para la comprobación de la herramienta creada, objetivo de este trabajo. El caso práctico se ha tomado del artículo de referencia [1].

En la descripción del problema, se habla de un sistema. En este caso, el equipo protagonista para al cual se realiza el estudio es un motor Diesel de 2 tiempos, 16 cilindros en V y de 1800KW, instalado en los camiones de una empresa minera en el sur de España.



Figura 3-3: Volquete. [38]

El motor del camión se encuentra funcionando correctamente hasta que ocurre un fallo. El modo de fallo se encuentra en el ventilador de admisión de aire posicionado dentro del motor. El fallo se debe a una fuga a través del retenedor de aceite del soplador [1].

Cuando el soplador falla, el motor registra una condición crítica repentina (alta presión del cárter). Al detectarse el fallo en el motor, la unidad de control electrónico (ECU) lo registra, luego los técnicos de mantenimiento diagnostican el problema y lo reparan. Este procedimiento puede tomar alrededor de 15 turnos de inactividad del sistema promedio: 120 horas de indisponibilidad.

La compañía tiene la opción de realizar un reemplazo preventivo de los sopladores viejos con elementos remanufacturados, después de τ horas de operación. El procedimiento de mantenimiento preventivo reduce el tiempo de inactividad a un período promedio de 6 turnos, 48 horas, con dos técnicos trabajando. El problema es que los estados del sistema pueden variar dentro de un número finito de períodos de tiempo, lo que afecta a los ingresos por unidad de tiempo cuando el sistema se encuentra en estado *Operativo*. La administración de la mina no está segura de los beneficios de implementar la tarea preventiva y quiere saber cómo esta medida afectará a su economía y a la planificación de los trabajadores.

Este dilema se puede resolver con el modelo presentado, de tal forma que se calcule el tiempo que debe permanecer el equipo en el estado *Operativo* antes de pasar al estado de mantenimiento *Preventivo*, maximizando el retorno promedio acumulado.

De igual importancia, tras el estudio realizado de los datos recogidos y la construcción de la función de distribución a partir de los tiempos de fallo, se concretó que ésta se ajusta a una función Weibull de tres parámetros [1].

$$\alpha=3,50, \beta=2.500, \gamma=0$$

3.5.1 Datos del problema

Antes de comenzar el cálculo del tiempo óptimo de permanencia en el estado *Operativo*, es necesario conocer los valores iniciales de ciertos parámetros especificados en la Tabla 3–1:

Tabla 3–1. Valores iniciales para las funciones de distribución (FD) [1].

FD de Fallos		FD Tiempos E. Correctivo		FD Tiempos E. Preventivo	
Weibull (α, β)		Normal (μ_1, σ_1)		Normal (μ_2, σ_2)	
α	β	μ_1	σ_1	μ_2	σ_2
3.50	2,500	120	2	48	1

A demás, es necesario saber los ingresos y los costes asociados a cada estado en el que se encuentra el sistema.

Los valores de los retornos de este caso práctico son éstos:

- R1, ingreso por unidad de tiempo que el sistema permanece en el estado 1. (E1: *Operativo*): 6 €/hora.
- R2, coste por unidad de tiempo que el sistema permanece en el estado 2. (E2: *Correctivo*): 20€/hora.
- R3, coste por unidad de tiempo que el sistema permanece en el estado 3. (E3: *Preventivo*): 20€/hora.
- R12, coste de transición del estado *Operativo* al *Correctivo*: 500€.
- R21, coste de transición del estado *Correctivo* al *Operativo*: 4100€.
- R13, coste de transición del estado *Operativo* al *Preventivo*: 1€.
- R31, coste de transición del estado *Preventivo* al *Operativo*: 290€.

La tabla resumen de los retornos mencionados es:

Tabla 3–2. Datos iniciales para los retornos [1].

Retornos Operación			Retorno Correctivo		Retorno Preventivo	
R ₁ (€/hora)	R ₁₂ (€)	R ₁₃ (€)	R ₂ (€/hora)	R ₂₁ (€)	R ₃ (€/hora)	R ₃₁ (€)
6.0	-500	-1.0	-20	-4100	-20	-290

Una vez que se tienen todos los datos iniciales, se puede resolver el problema utilizando el modelo descrito.

En el capítulo 5 se muestran los resultados obtenidos para el tiempo óptimo en el que el sistema debe pasar al mantenimiento *Preventivo*, utilizando la herramienta creada para este proyecto. Además, se incluye la verificación de los resultados, mediante la comparativa de éstos con los que se revelan en el artículo de referencia [1].

4 MANUAL DE USUARIO DEL PROGRAMA PM-3E

Lo dibujé en sólo unos segundos, pero me ha costado 34 años aprender a dibujarlo en sólo unos segundos.

- Paula Scher -

En el presente capítulo se explica el funcionamiento de la herramienta **PM-3E** y los pasos que debe seguir el usuario para poder obtener los resultados deseados rápidamente con el fin de facilitar la toma de decisiones futuras.

El programa **PM-3E** es una herramienta con la cual se puede hallar el tiempo de permanencia de un sistema en el estado *Operativo* antes de pasar al *Preventivo*, maximizando el retorno (ingreso menos coste) promedio acumulado en un horizonte finito de m pasos. Del mismo modo, a partir del tiempo óptimo obtenido el programa permite el cálculo del retorno promedio. Por otro lado, la herramienta está diseñada para un sistema que se puede encontrar en 3 estados diferentes, donde los estados posibles del sistema son:

- E1: Estado *Operativo*.
- E2: Estado *Correctivo*.
- E3: Estado *Preventivo*.

Por otro lado, la función de distribución de fallos que representa el tiempo de permanencia del sistema en el estado *Correctivo*, en este caso [1] se corresponde con la función Weibull de tres parámetros.

Esta herramienta se ha creado con Python 3.7 y, gracias a la importación de la librería Tkinter, se ha desarrollado una pestaña gráfica que facilita la interacción con el usuario.

La creación de este programa facilita, agiliza y compara visualmente, a través de representaciones gráficas, los diferentes escenarios que puede presentar el sistema simplemente cambiando cualquier parámetro en la distribución de Weibull o los valores de los retornos en cada estado [1].

Las principales funciones de la herramienta son:

- Cálculo del tiempo óptimo hasta la intervención de mantenimiento preventivo (Tau óptimo, τ).
- Cálculo del vector de retorno promedio acumulado (V).
- Cálculo del tiempo de operación promedio (Average Operating Time, AOT).
- Cálculo del tiempo de operación promedio acumulado (Average Accumulated Operating Time, AAOT).
- Cálculo del retorno promedio (Average Return, AR).
- Creación de un archivo de texto con los resultados obtenidos.
- Representación gráfica:
 - Número de pasos de transición frente al Tau óptimo.
 - Número de pasos de transición frente al vector de retorno promedio acumulado.
 - Tau frente al vector de retorno promedio acumulado.
 - Número de pasos de transición frente al tiempo de operación promedio acumulado.
 - Número de pasos de transición frente al retorno promedio.
- Cálculo de V, AOT, AAOT o AR para un tiempo dado.

Seguidamente, se explicará y se mostrará la interfaz de usuario, cada una de las partes que la componen y su respectivo funcionamiento.

Todo el código desarrollado para la creación de este programa se recoge en el Anexo A: Codificación en Python del programa PM-3E.

4.1 Interfaz de Usuario: Pantalla Principal

Al ejecutar el programa se abre la pantalla principal, con la cual el usuario podrá interactuar para calcular los diferentes valores mencionados.

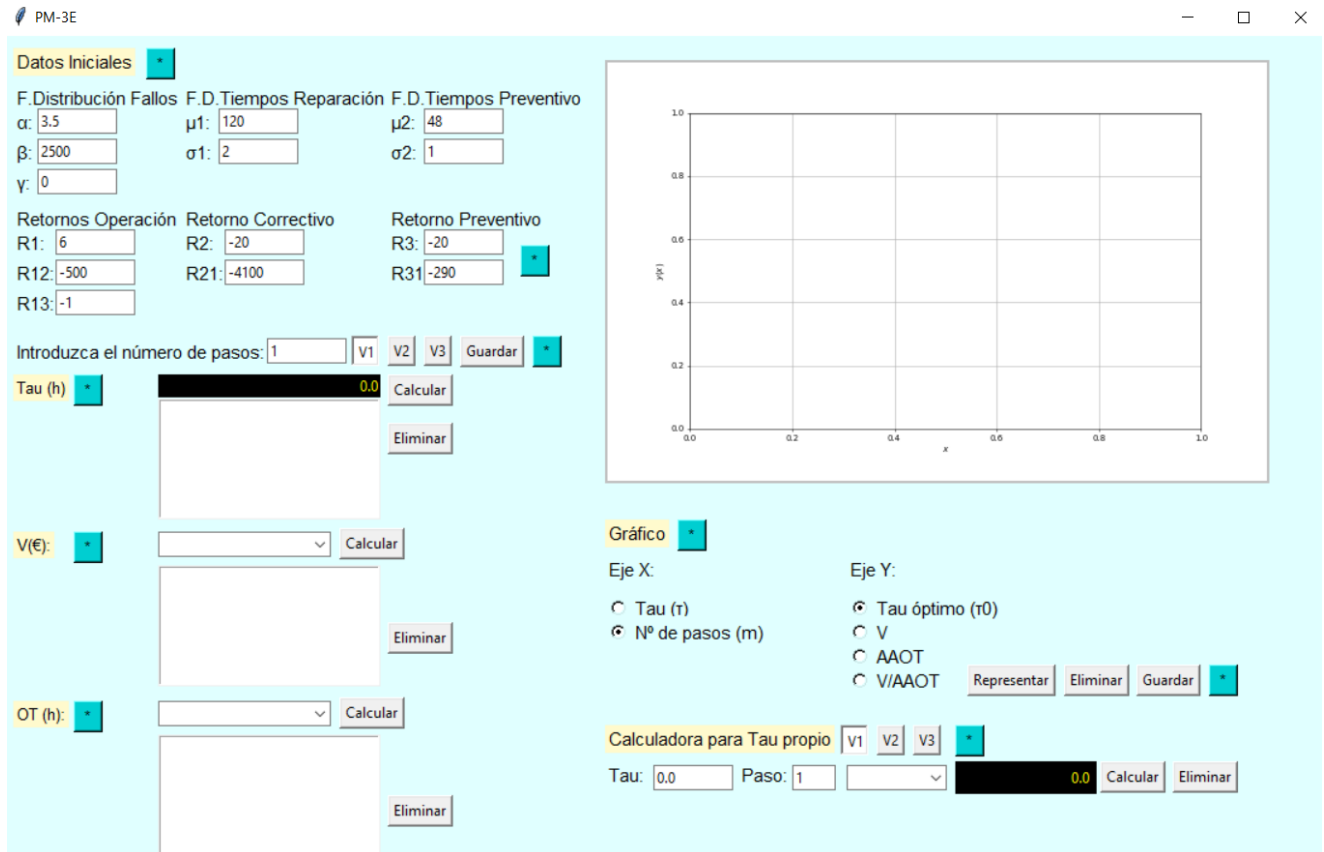


Figura 4-1: Pantalla principal de PM-3E.

En primer lugar, la pantalla principal se puede minimizar, maximizar y mover según las necesidades del usuario. Igualmente se puede cerrar cuando el usuario haya terminado de usarla.



Figura 4-2: Opciones para minimizar, maximizar y cerrar la aplicación.

Puesto que todas las funcionalidades de la herramienta se muestran en una única pantalla, ésta se ha estructurado de tal forma que se puedan distinguir cuatro secciones principales.

1. Datos Iniciales
2. Calculadora
3. Gráfico
4. Calculadora para Tau propio

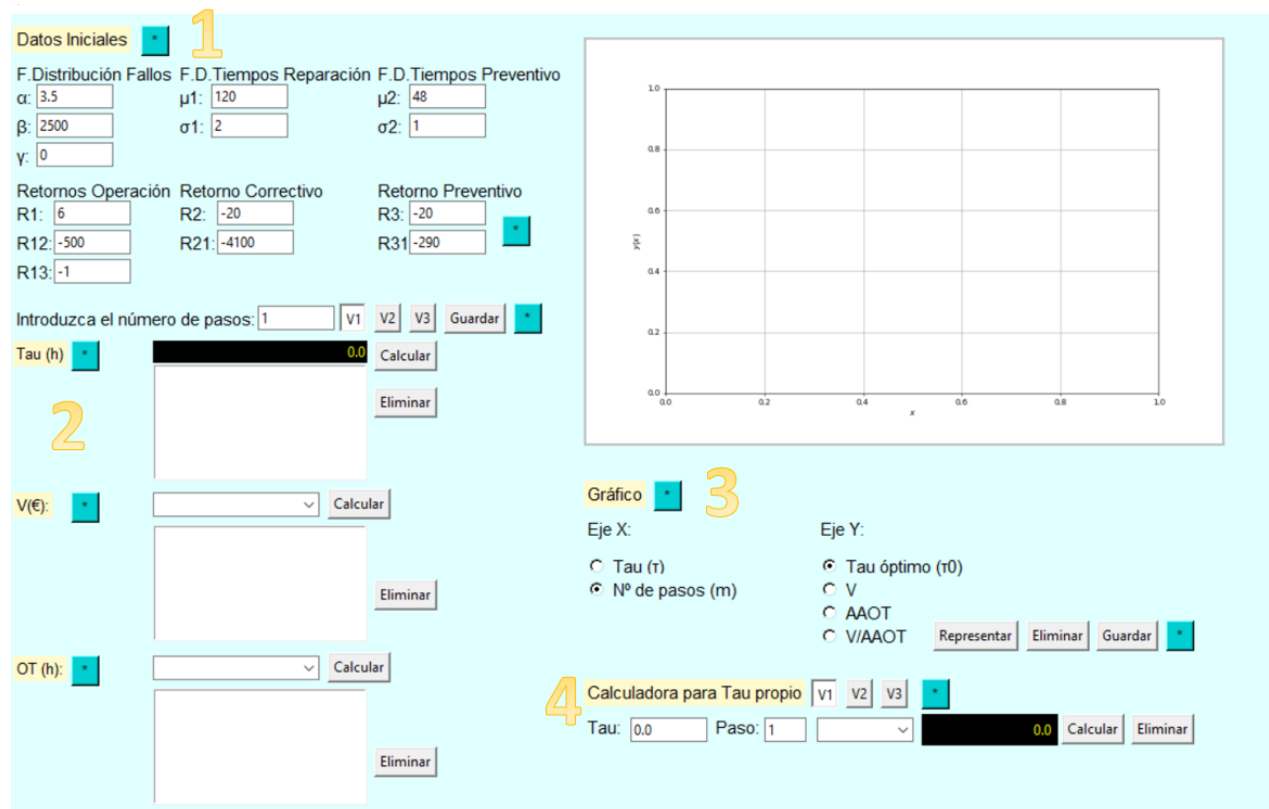


Figura 4-3: Diferentes secciones de la herramienta PM-3E

Adicionalmente, cada una de las secciones viene acompañada de uno o varios botones “Ayuda”.



Figura 4-4: Señalización botón “Ayuda”

4.2 Datos Iniciales

En esta sección se encuentran 14 casillas con las que el usuario puede interactuar, cada una de ellas viene acompañada de una etiqueta que indica el parámetro al que corresponde. El usuario puede modificar cualquiera de estas casillas introduciendo los valores adecuados a su caso. Si se desea introducir un valor decimal, se deberá usar “.” no “,” para marcar la parte decimal, en caso contrario el programa no realizará ningún cálculo.

Datos Iniciales		
F.D. Distribución Fallos	F.D. Tiempos Reparación	F.D. Tiempos Preventivo
α : 3.5	μ_1 : 120	μ_2 : 48
β : 2500	σ_1 : 2	σ_2 : 1
γ : 0		
Retornos Operación	Retorno Correctivo	Retorno Preventivo
R1: 6	R2: -20	R3: -20
R12: -500	R21: -4100	R31: -290
R13: -1		

Figura 4-5: Sección de la herramienta donde se deben introducir los datos iniciales.

El abrir el programa aparecen los datos que se reflejan en la Figura 4-5, para que el usuario pueda usarlos como ejemplo y verificar el funcionamiento de la herramienta. Igualmente, el usuario puede clicar en cualquiera de las casillas, borrar su contenido e incluir el valor que desee.

Es una herramienta diseñada para un sistema que pasa por tres estados diferentes:

- E1: Estado *Operativo*. Aquel en el que el sistema se encuentra funcionando correctamente.
- E2: Estado *Correctivo*. Momento en el que el sistema está siendo reparado debido a un fallo en el mismo.
- E3: Estado *Preventivo*. Cuando el sistema se encuentra en un estado donde se están realizando acciones de mantenimiento preventivas.

El tiempo de permanencia del sistema en cada estado para este caso, sigue las siguientes distribuciones:

- Función de distribución de Fallos: Función Weibull de tres parámetros (α , β , γ) para E1.
- Función de distribución Tiempos de Reparación: Normal (μ_1 , σ_1) para E2.
- Función de distribución Tiempos Preventivo: Normal (μ_2 , σ_2) para E3.

Por otro lado, cuando el sistema se encuentra en un estado o cambia a otro, genera ingresos y gastos. Éstos se representan a través de los retornos, que se escribirán con signo negativo si es un gasto.

- R1, ingreso por unidad de tiempo que el sistema permanece en el estado 1 (E1: *Operativo*)
- R2, costo por unidad de tiempo que el sistema permanece en el estado 2 (E2: *Correctivo*)
- R3, costo por unidad de tiempo que el sistema permanece en el estado 3 (E3: *Preventivo*).
- R12, costo de transición del estado *Operativo* al *Correctivo*.
- R21, costo de transición del estado *Correctivo* al *Operativo*.
- R13, costo de transición del estado *Operativo* al *Preventivo*.
- R31, costo de transición del estado *Preventivo* al *Operativo*.

4.3 Calculadora

En esta sección se calculan cada uno de los resultados. Cada vez que el sistema cambia de estado corresponde a un paso de transición m . A medida que el sistema cambia de estado, acumula ingresos y costes.

Se encuentra una primera casilla donde el usuario debe introducir el número de pasos de transición que desee para su estudio. Conjuntamente, existen tres botones, “V1”, “V2” y “V3”. Se deberá seleccionar “V1” si el sistema se inicia en el estado *Operativo*, “V2” si el sistema se inicia en el estado *Correctivo* y “V3” si se inicia en el estado *Preventivo*.

The screenshot shows a light blue interface with the following elements:

- Top: "Introduzca el número de pasos:" followed by an input field containing "1", and three buttons labeled "V1", "V2", and "V3". A yellow arrow points to a "Calcular" button to the right.
- Section 1: Labeled "1" in yellow. It has a "Tau (h)" label with a dropdown arrow, a display showing "0.0", a "Calcular" button, and an "Eliminar" button.
- Section 2: Labeled "2" in yellow. It has a "V(€)" label with a dropdown arrow, a "Calcular" button, and an "Eliminar" button.
- Section 3: Labeled "3" in yellow. It has an "OT (h)" label with a dropdown arrow, a "Calcular" button, and an "Eliminar" button.

Figura 4-6: Sección de la herramienta donde se muestran los resultados

A continuación, se encuentran 3 apartados dentro de esta misma sección. Cada uno revela el cálculo de un elemento.

4.3.1 Cálculo de tau óptimo

El tau óptimo indica el instante en horas en el que el sistema debe pasar al estado de mantenimiento preventivo, de modo que se maximice el retorno promedio acumulado (Ingresos-Gastos).

Para ello el usuario únicamente, tras introducir el número de pasos y señalar el estado inicial del sistema, debe pulsar el botón que dice “Calcular”.

The screenshot shows the calculator interface with the following elements:

- Top: "Introduzca el número de pasos:" followed by an input field containing "10", and three buttons labeled "V1", "V2", and "V3". A "Guardar" button and a dropdown arrow are also present.
- Section 1: Labeled "Tau (h)" with a dropdown arrow, a display showing "0.0", a "Calcular" button, and an "Eliminar" button. A yellow arrow points to the "Calcular" button.
- Overlay: A text box with the message "Pulsar Calcular para ejecutar la función para el cálculo de tau óptimo" in blue text.

Figura 4-7: Sección de la herramienta para el cálculo de tau óptimo.

De esta manera se ejecutará la función adecuada y el resultado de tau óptimo para el paso introducido aparecerá en la casilla negra conjunta a la etiqueta “Tau. (h)”.

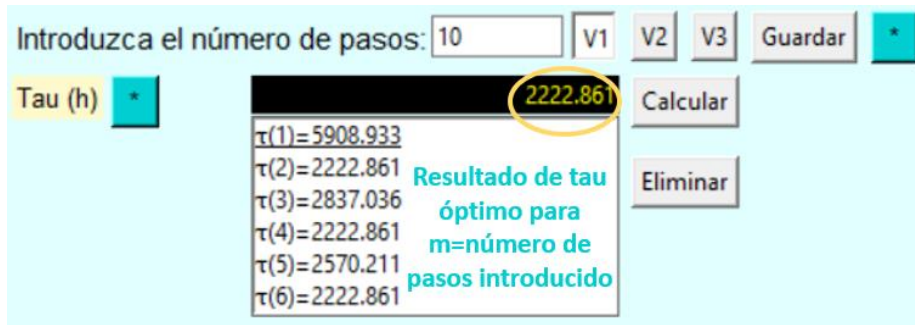


Figura 4-8: Representación de cómo calcular tau óptimo.

Al mismo tiempo, aparecerá en el recuadro que se encuentra justo debajo, el valor de tau óptimo de cada uno de los pasos hasta llegar al paso introducido.

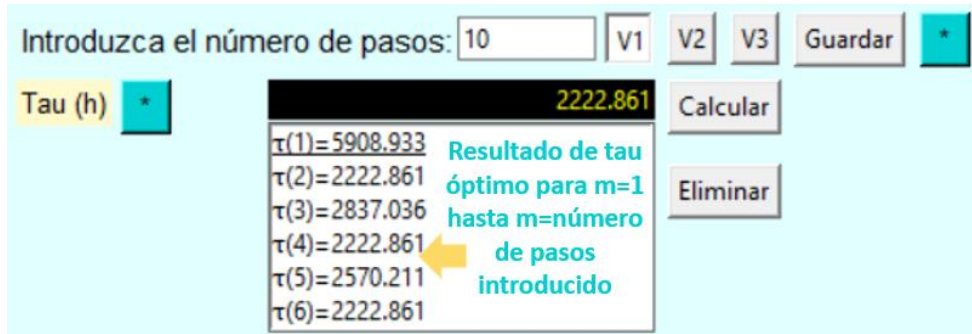


Figura 4-9: Señalización del resultado de tau óptimo para un número de pasos determinado.

4.3.2 Cálculo del vector de retorno promedio acumulado, V

En esta sección se incluye una pestaña donde se exponen varias opciones (PAR, IMPAR, TODO). Todas corresponden al cálculo del vector de retorno promedio acumulado en euros para el número de pasos introducido. De igual forma, si se ha seleccionado “V1” al inicio, se calculará el primer vector de retorno promedio, en el caso de “V2” el segundo vector y para “V3” el tercer vector.

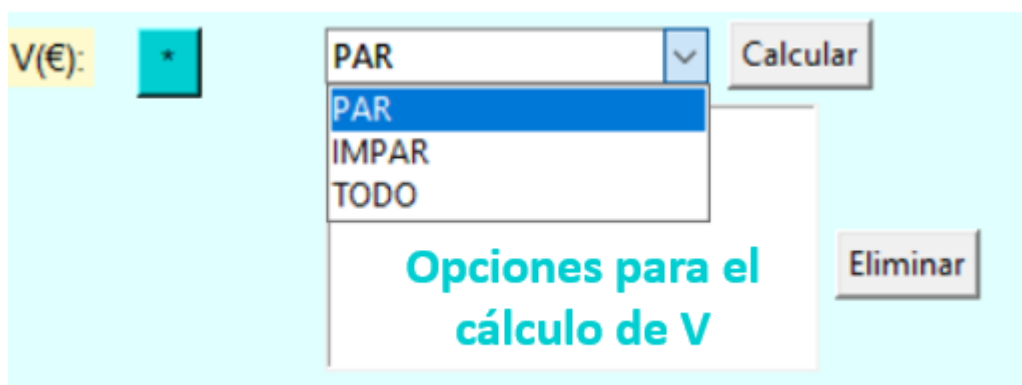


Figura 4-10: Sección de la herramienta para el cálculo de V.

Para poner en funcionamiento el cálculo de V, una vez seleccionada alguna de las opciones, el usuario debe pulsar el botón “Calcular” que se encuentra justo al lado del desplegable. Tras realizar esta acción el resultado aparecerá en el recuadro conjunto a esta sección.

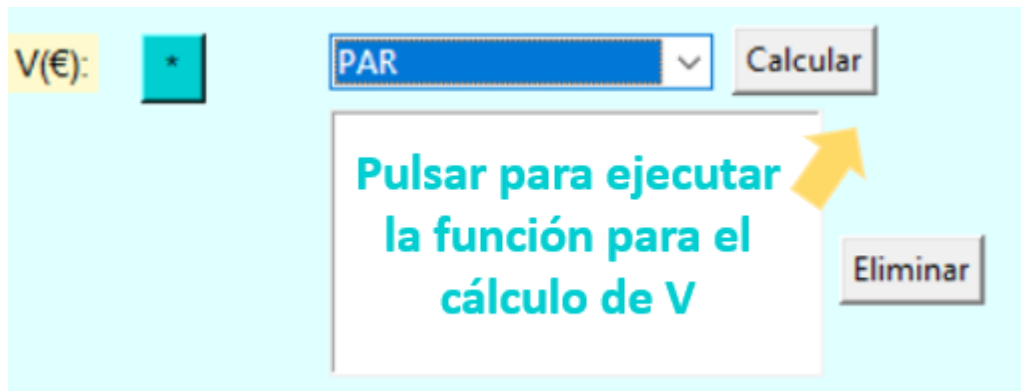


Figura 4-11: Representación de cómo calcular V.

4.3.2.1 Opciones de cálculo para V

PAR: Al seleccionar esta opción se calculará el valor de V tomando el valor de tau óptimo para el número de pasos introducido. Se mostrará el resultado únicamente para los pasos de transición pares desde 1 hasta el paso introducido al inicio (10 en este ejemplo).



Figura 4-12: Señalización del resultado de V1 para un número de pasos determinado.

El resultado se revela de esta manera: V (paso de transición) = Valor de V obtenido en ese paso para tau óptimo.

IMPAR: De igual modo, al seleccionar esta opción se calculará el valor de V tomando el valor de tau óptimo para el número de pasos introducido, y se mostrará el resultado únicamente para los pasos de transición impares desde 1 hasta el paso introducido.

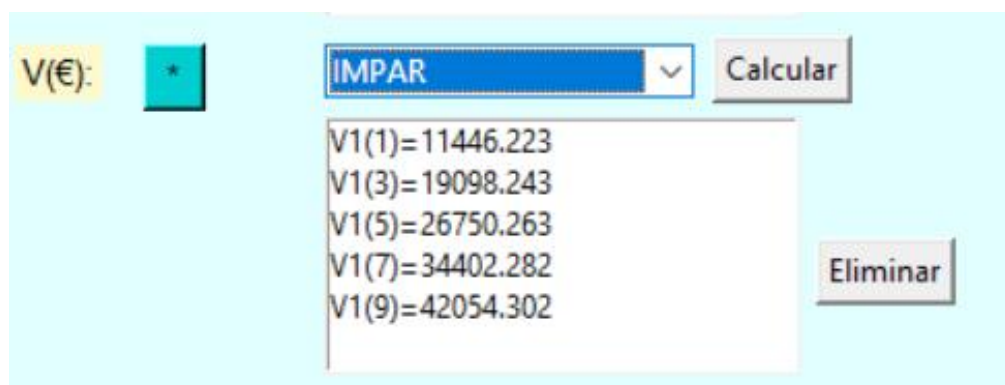


Figura 4-13: Señalización del resultado de V1 para un número de pasos determinado.

TODO: Seleccionando esta opción el usuario obtendrá el resultado de V para cada uno de los valores de m , desde 1 hasta el número de pasos de transición introducido, usando como en los casos anteriores el valor óptimo de tau.

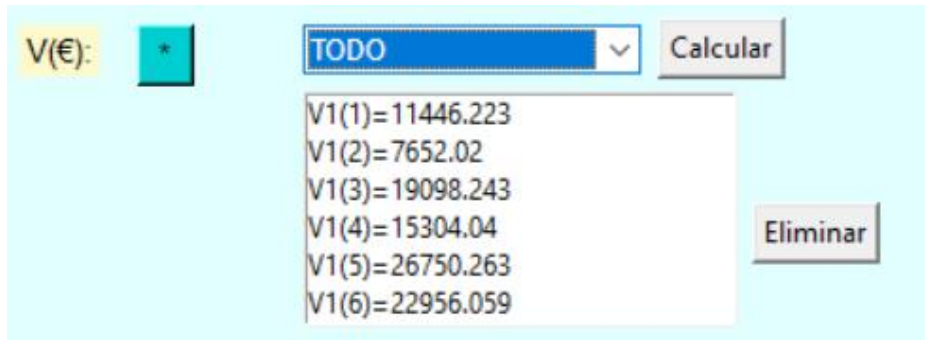


Figura 4-14: Señalización del resultado de V_1 para un número de pasos determinado.

4.3.3 Cálculo del tiempo de operación, OT.

Para poder poner en funcionamiento esta parte de la herramienta es necesario haber calculado previamente con la misma el valor de tau óptimo y el valor de V . Una vez se hayan calculado ambos parámetros, el usuario puede elegir de nuevo en un desplegable qué opción desea calcular.

- AOT: Tiempo de operación promedio.
- AAOT: Tiempo de operación promedio acumulado.
- $V/AAOT$: Retorno promedio del sistema.

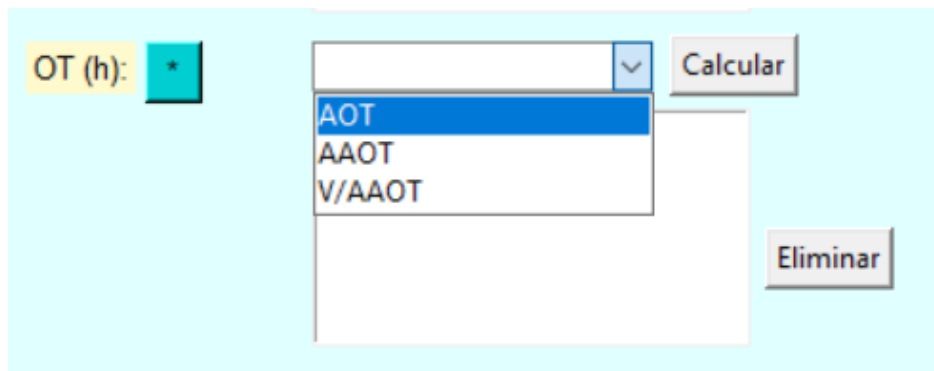


Figura 4-15: Sección de la herramienta para el cálculo del tiempo de operación y opciones disponibles de cálculo.

Una vez seleccionada una de las opciones, el usuario deberá pulsar el botón “Calcular” que se encuentra a la derecha del desplegable para poner en funcionamiento el cálculo.

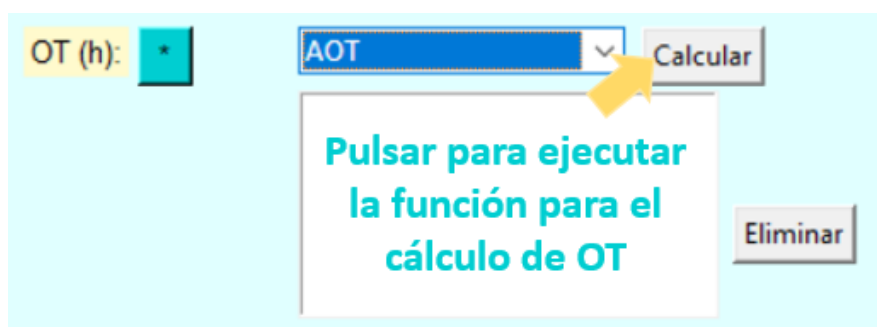


Figura 4-16: Representación de cómo calcular los tiempos de operación.

4.3.3.1 Opciones de cálculo para OT

AOT: Al marcar esta opción el usuario está indicando que quiere calcular el tiempo de operación promedio para el número de pasos de transición introducido al inicio.

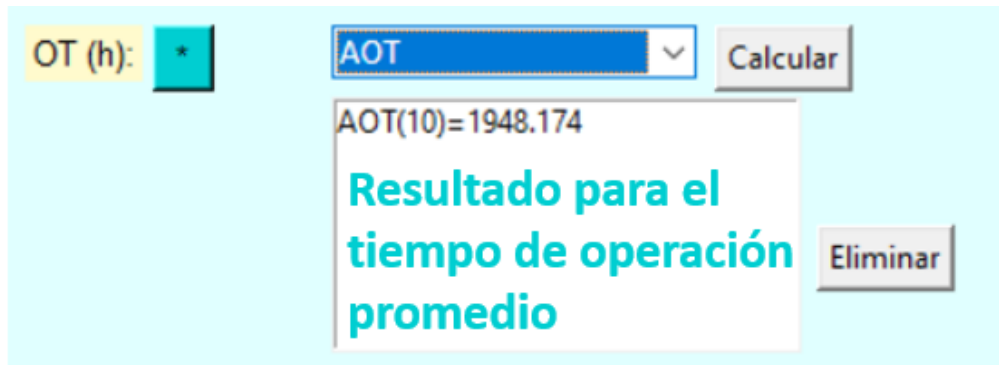


Figura 4-17: Señalización del resultado de AOT para un número de pasos determinado.

AAOT: Esta segunda opción dentro del desplegable, permite el cálculo del tiempo de operación promedio acumulado desde un paso de transición 1 hasta el paso introducido.

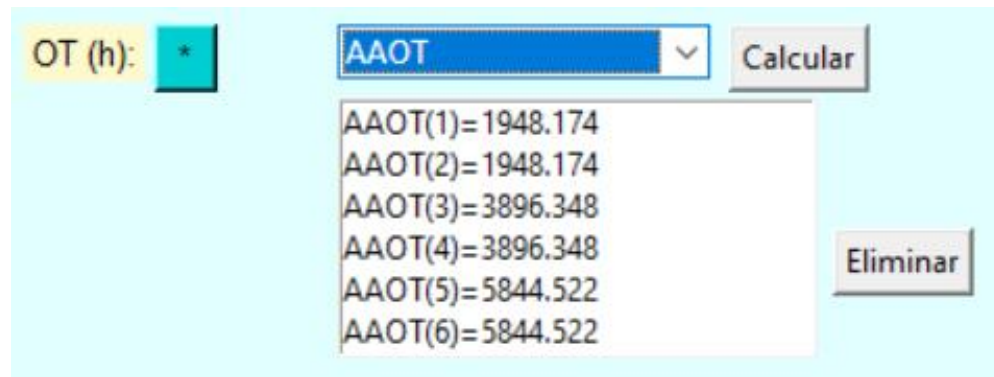


Figura 4-18: Señalización del resultado de AAOT para un número de pasos determinado.

V/AAOT: Al seleccionar esta opción la herramienta realizará la división entre el vector de retorno promedio acumulado y el tiempo de operación promedio acumulado. El resultado que se obtiene se denomina retorno promedio (AR), y al igual que en el caso anterior, el resultado que se obtiene es el valor de AR desde un paso de transición 1 hasta el paso de transición introducido. En este caso la unidad del resultado sería euro/hora.

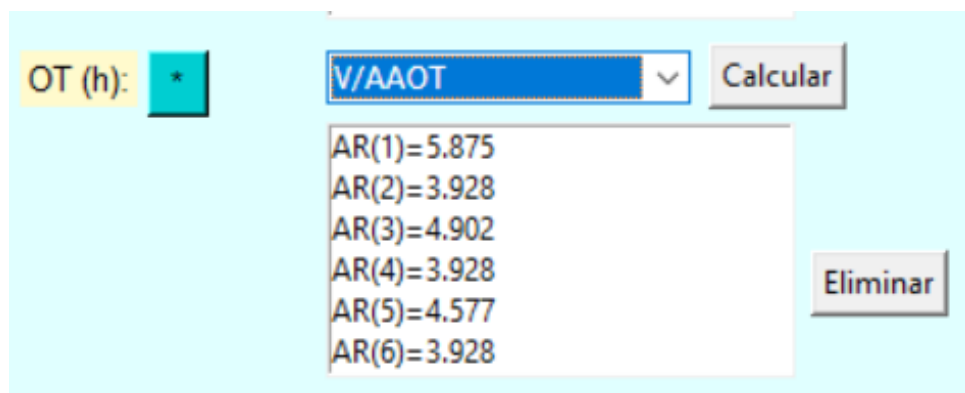


Figura 4-19: Señalización del resultado de AR para un número de pasos determinado.

4.3.4 Funciones de los botones

4.3.4.1 Botón Calcular

Esta sección contiene 3 botones “Calcular”. El usuario deberá pulsar uno u otro, dependiendo del valor que quiera obtener.

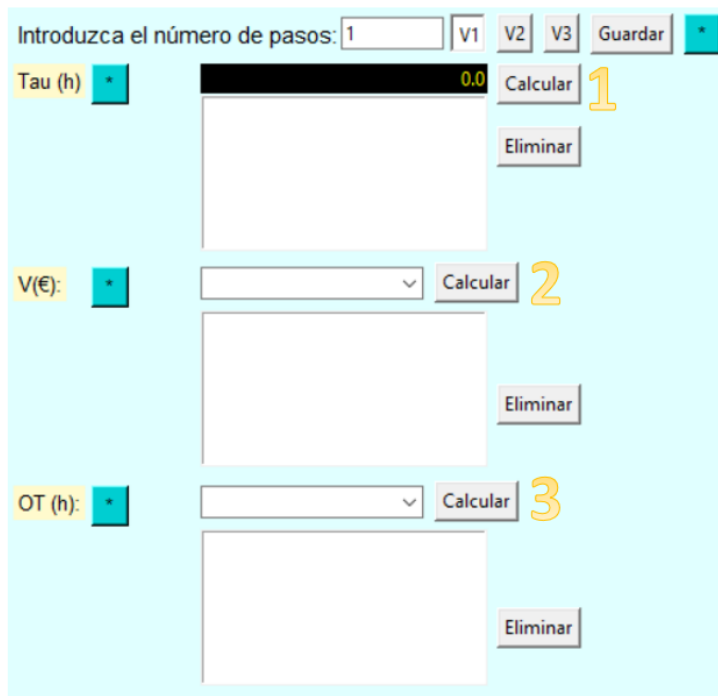


Figura 4-20: Señalización de los botones existentes en la herramienta para calcular resultados.

Funciones de cada uno de los botones “Calcular”:

- **Calcular (1):** Se pulsará para obtener el resultado del tau óptimo.
- **Calcular (2):** Se pulsará para obtener el resultado seleccionado para V.
- **Calcular (3):** Al pulsarlo el programa mostrará el resultado para la opción que el usuario haya elegido dentro de las opciones de OT.

4.3.4.2 Botón Eliminar

Igualmente, esta sección contiene tres botones “Eliminar”, cada uno sujeto a una sección dentro del apartado calculadora. Tras pulsarlo, se eliminará el contenido que se encuentra en el cuadro de resultados que tiene justo al lado. Siempre que el usuario desee calcular un nuevo elemento de una misma sección, deberá eliminar el resultado anterior. En caso contrario, si el recuadro contiene dos ejecuciones, el programa no podrá representar el resultado gráficamente.

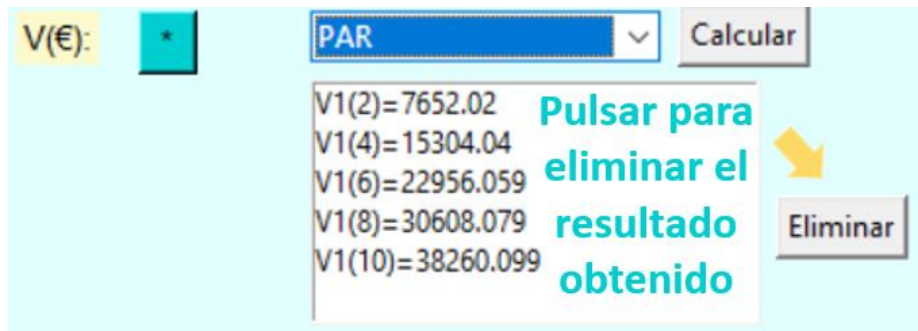


Figura 4-21: Representación de cómo eliminar un resultado.

4.3.4.3 Botón Guardar

En caso de que el usuario quiera guardar los resultados obtenidos, podrá hacerlo pulsando el botón “Guardar”. El programa guardará los datos iniciales introducidos y los resultados que se muestran en los recuadros (1, 2, 3).

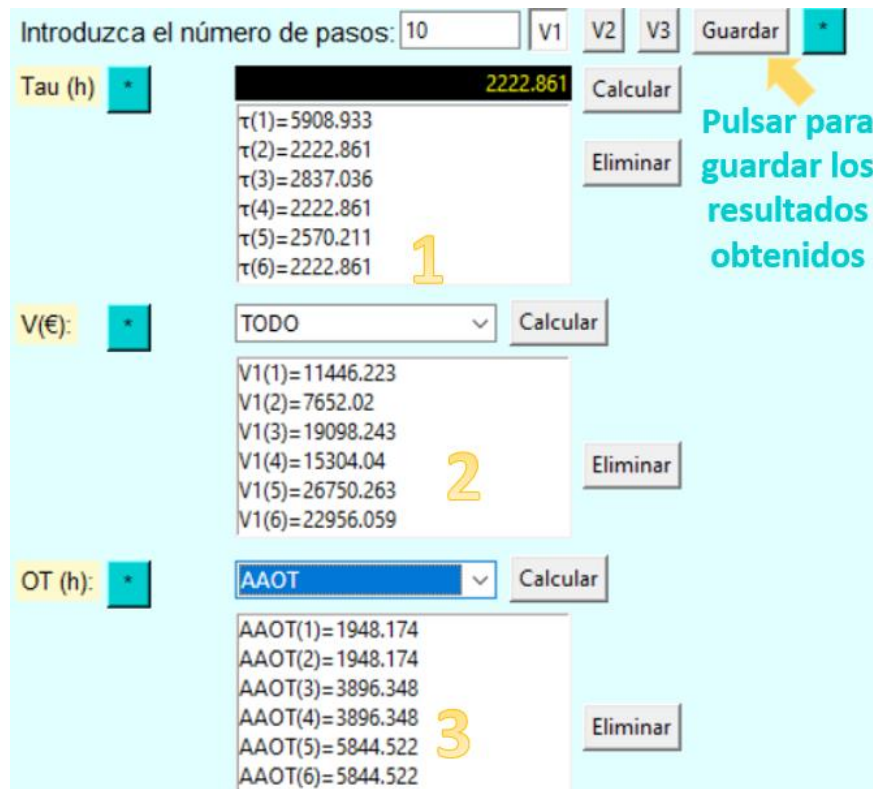


Figura 4-22: Explicación de cómo guardar los resultados obtenidos.

Acto seguido, tras pulsar el botón “Guardar”, aparecerá esta pantalla, donde el usuario deberá elegir el nombre del archivo y el lugar dónde lo quiera guardar:

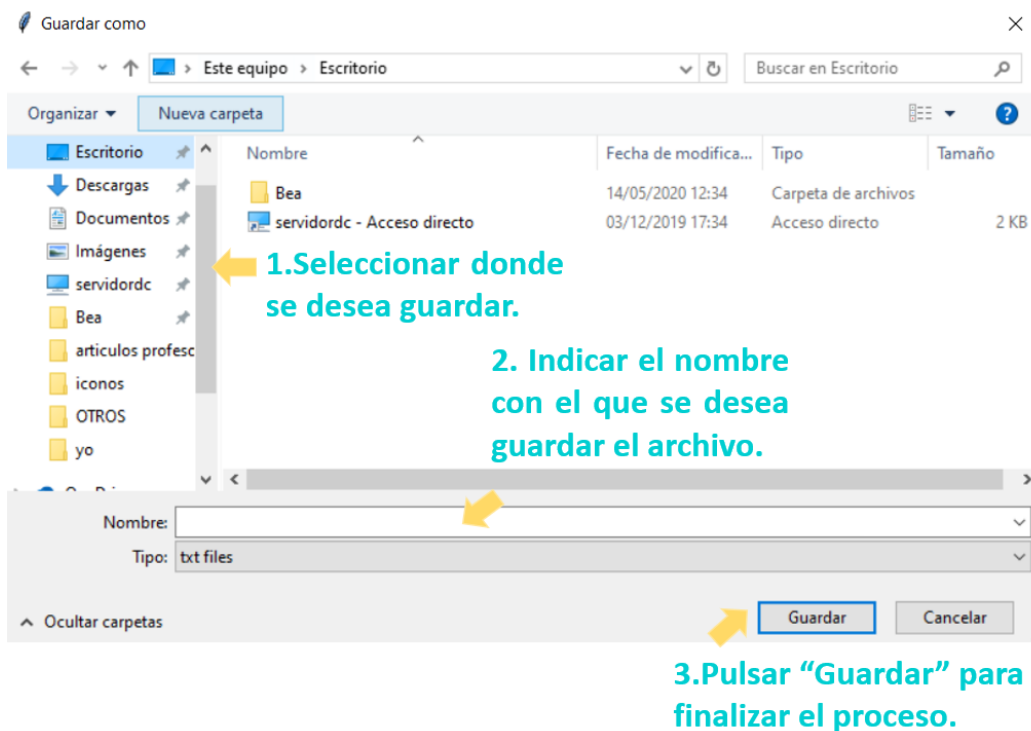


Figura 4-23: Explicación de cómo guardar los resultados obtenidos.

Una vez guardado, el documento que se habrá creado será semejante a éste:

```

prueba: Bloc de notas
Archivo Edición Formato Ver Ayuda
F.Distribución Fallos α:3.5; β:2500.0; γ:0.0
F.D.Tiempos Reparación μ1:120.0; σ1:2.0
F.D.Tiempos Preventivo μ2:48.0; σ2:1.0
Retornos Operación R1:6.0; R12:-500.0; R13:-1.0
Retorno Correctivo R2:-20.0; R21:-4100.0
Retorno Preventivo R3:-20.0; R31:-290.0

NºPasos:10.0
VALOR ÓPTIMO (h):2222.861
τ(1)=5908.933
τ(2)=2222.861
τ(3)=2837.036
τ(4)=2222.861
τ(5)=2570.211
τ(6)=2222.861
τ(7)=2465.659
τ(8)=2222.861
τ(9)=2409.612
τ(10)=2222.861

V1(€): TODO
V1(1)=11446.223
V1(2)=7652.02
V1(3)=19098.243
V1(4)=15304.04
V1(5)=26750.263
V1(6)=22956.059
V1(7)=34402.282
V1(8)=30608.079
V1(9)=42054.302
V1(10)=38260.099

Tiempo de Operación:AAOT
AAOT(1)=1948.174
AAOT(2)=1948.174
AAOT(3)=3896.348
AAOT(4)=3896.348
AAOT(5)=5844.522
AAOT(6)=5844.522
AAOT(7)=7792.696
AAOT(8)=7792.696
AAOT(9)=9740.87
AAOT(10)=9740.87
    
```

← Datos iniciales introducidos.

← Número de pasos introducido.

← Resultado: Cálculo de Tau óptimo.

← Resultado: Cálculo de V1 para la opción elegida.

← Resultado: Cálculo del OT para la opción elegida.

Figura 4-24: Archivo creado al guardar los resultados a través de la herramienta.

The screenshot shows the software interface with the following elements:

- Inputs:**
 - Introduzca el número de pasos: 10
 - Tau (h): 2222.861
 - V(€): TODO
 - OT (h): V/AAOT
- Buttons:** Calcular, Eliminar (for each section).
- Results:**
 - Tau (h):** τ(1)=5908.933, τ(2)=2222.861, τ(3)=2837.036, τ(4)=2222.861, τ(5)=2570.211, τ(6)=2222.861
 - V(€):** V1(1)=11446.223, V1(2)=7652.02, V1(3)=19098.243, V1(4)=15304.04, V1(5)=26750.263, V1(6)=22956.059
 - OT (h):** AR(1)=5.875, AR(2)=3.928, AR(3)=4.902, AR(4)=3.928, AR(5)=4.577, AR(6)=3.928
- Summary (top right):**
 - NºPasos:10.0
 - VALOR ÓPTIMO (h):2222.861
 - τ(1)=5908.933, τ(2)=2222.861, τ(3)=2837.036, τ(4)=2222.861, τ(5)=2570.211, τ(6)=2222.861, τ(7)=2465.659, τ(8)=2222.861, τ(9)=2409.612, τ(10)=2222.861
 - V(€): TODO
 - V1(1)=11446.223, V1(2)=7652.02, V1(3)=19098.243, V1(4)=15304.04, V1(5)=26750.263, V1(6)=22956.059, V1(7)=34402.282, V1(8)=30608.079, V1(9)=42054.302, V1(10)=38260.099
 - Tiempo de Operación:V/AAOT
 - AR(1)=5.875, AR(2)=3.928, AR(3)=4.902, AR(4)=3.928, AR(5)=4.577, AR(6)=3.928, AR(7)=4.415, AR(8)=3.928, AR(9)=4.317, AR(10)=3.928

Figura 4-25: Ejemplo de cómo se guardan los datos obtenidos a través del programa.

4.4 Gráfico

En este apartado de la herramienta se puede representar cada uno de los resultados que se han obtenido en la sección anterior. El usuario no podrá representar un valor que no haya calculado previamente.

Esta sección contiene en primer lugar una cuadrícula donde se representarán los resultados. Justo debajo se encuentran dos columnas. La primera, con una etiqueta que pone “Eje X”, donde se encuentran las dos opciones que permite el programa representar en dicho eje. La columna de al lado con la etiqueta “Eje Y” expone cada una de las cuatro opciones permitidas para representar en el eje Y. Finalmente, en la esquina izquierda se encuentran tres botones: “Representar”, “Eliminar” y “Guardar”.

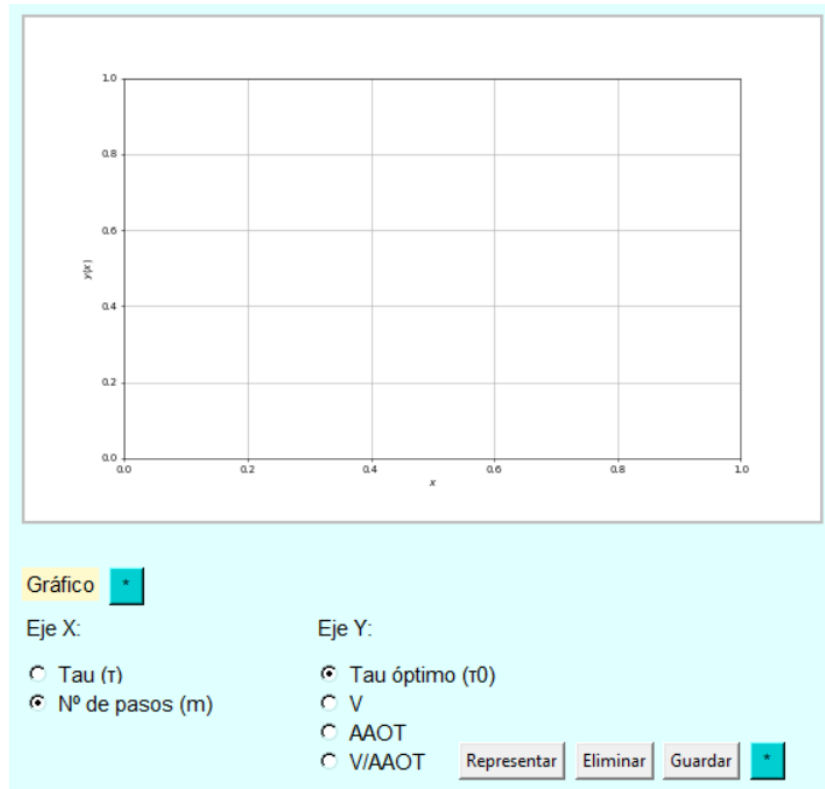


Figura 4-26: Sección de la herramienta para la representación gráfica de los resultados

El usuario deberá clicar en una de las opciones que se muestran debajo de la etiqueta “Eje X” y clicar en una de las opciones que se encuentran debajo de la etiqueta “Eje Y”.

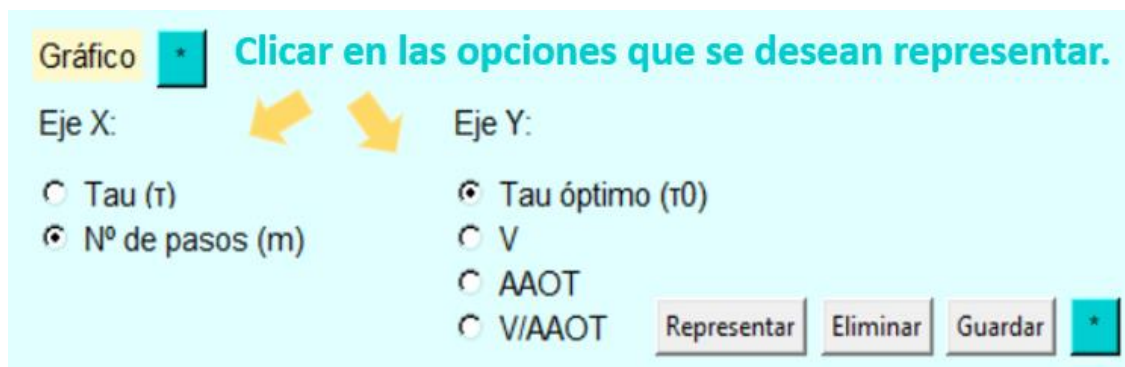


Figura 4-27: Representación de cómo realizar una representación gráfica usando la herramienta.

Es cierto que el usuario puede elegir la combinación que desee, pero si no está incluida en la próxima lista, al pulsar el botón de “Representar” no aparecerá nada nuevo en la cuadrícula superior.

4.4.1 Elecciones de representación

Eje X: Tau; Eje Y: V. Esta opción es la única que podrá ser representada sin necesidad de que haya calculado su valor en el apartado calculadora, ya que no existe opción para ello. Esta opción representa gráficamente una serie de valores de tau para un mismo paso y su respectivo valor de V. Y señala en la propia gráfica el valor de tau óptimo para ese paso de transición indicado.

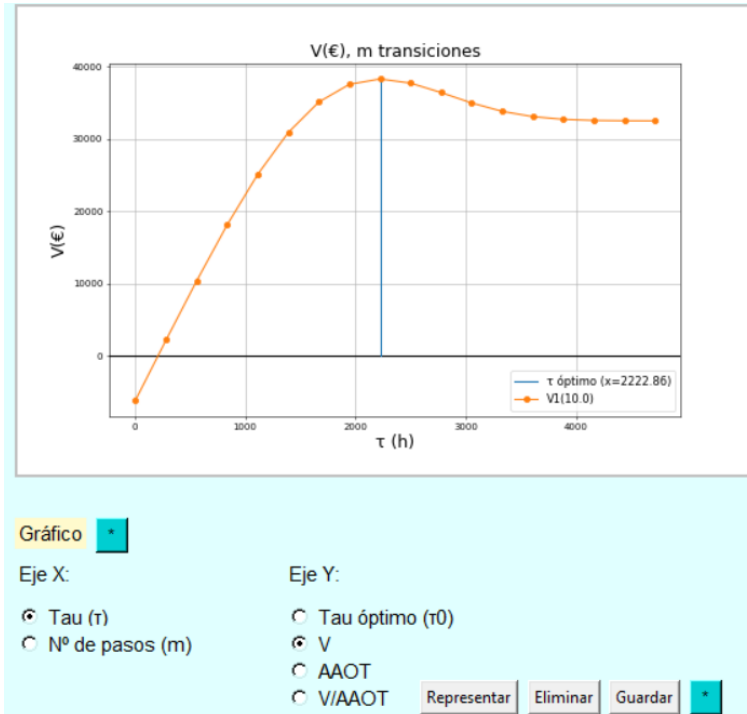


Figura 4-28: Representación gráfica de tau frente a V1 creada a través de la herramienta.

Eje X: N.º de pasos (m); Eje Y: Tau óptimo. En caso de que se haya calculado el tau óptimo en la sección de cálculos, se podrá representar su valor frente a los pasos de transición.



Figura 4-29: Representación gráfica de los pasos de transición frente al tau óptimo creada a través de la herramienta.

Eje X: N.º de pasos (m); Eje Y: V. Esta selección se representará siempre y cuando en la sección de cálculo de V se haya calculado alguna de las opciones que dispone: “Par, Impar, Todo”



Figura 4-30: Representación gráfica del número de pasos frente a V1, creada a través de la herramienta.

Eje X: N.º de pasos (m); Eje Y: AAOT. Para poder representar esta opción es necesario haber calculado previamente el valor de AAOT. En el recuadro de resultados de OT se debe mostrar el valor de AAOT para cada uno de los valores de m desde $m = 1$ hasta el paso de transición introducido.

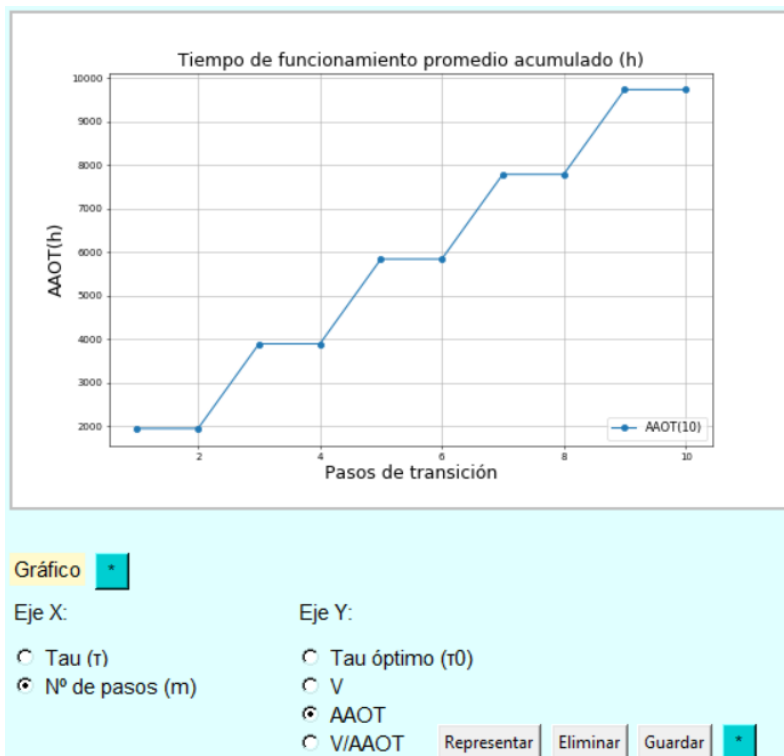


Figura 4-31: Representación gráfica del número de pasos frente a AAOT creada a través de la herramienta.

Eje X: N.º de pasos (m); Eje Y: V/AAOT. Asimismo, para poder representar esta opción es necesario haber calculado previamente en la sección de OT la opción “V/AAOT”.



Figura 4-32: Representación gráfica del número de pasos frente a AR creada a través de la herramienta.

4.4.2 Funciones de los botones

4.4.2.1 Botón Representar

Tras haber marcado la combinación deseada, el usuario debe pulsar este botón. Acto seguido se mostrará en la cuadrícula el gráfico elegido.

4.4.2.2 Botón Eliminar

Este botón se pulsará siempre que se haya hecho una representación y seguidamente se quiera representar una función diferente, es decir, una nueva combinación. Pese a que, tras pulsar el botón visualmente para el usuario no cambiará nada, tras volver a pulsar “Representar” desaparecerá la función anterior quedando en la cuadrícula únicamente la nueva. Si no se pulsa “Eliminar”, se representará una combinación de ambos gráficos.

4.4.2.3 Botón Guardar

El usuario deberá pulsar este botón si desea guardar la representación creada.

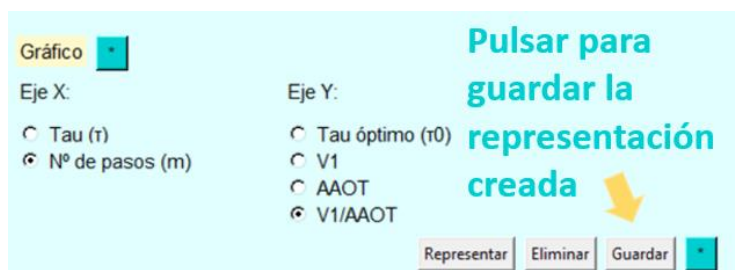


Figura 4-33: Explicación de cómo guardar la representación gráfica creada con la herramienta.

El programa abrirá esta pantalla:

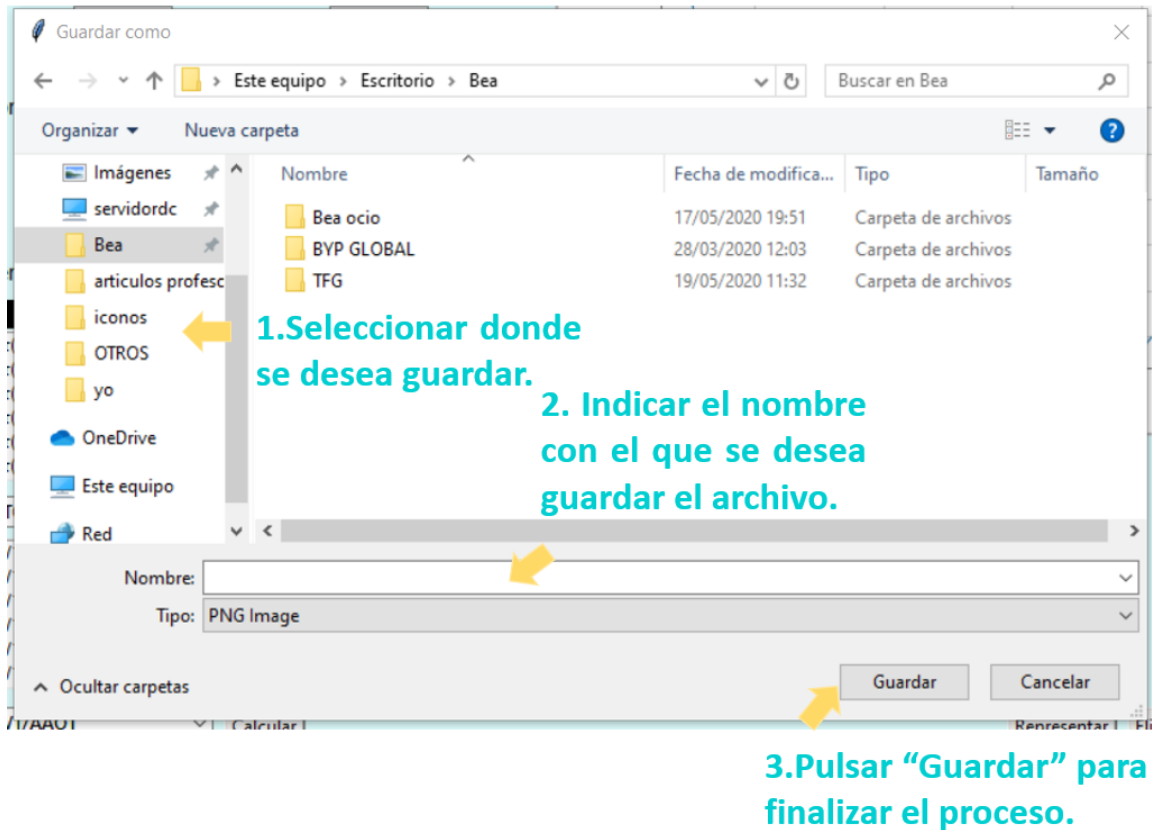


Figura 4-34: Pestaña para guardar el archivo creado.

En ella el usuario seleccionará el nombre y el lugar dónde desea guardar el archivo. Éste se guardará en formato png, es decir, como una imagen. Al abrir el archivo creado, se mostrará una imagen de la representación guardada.

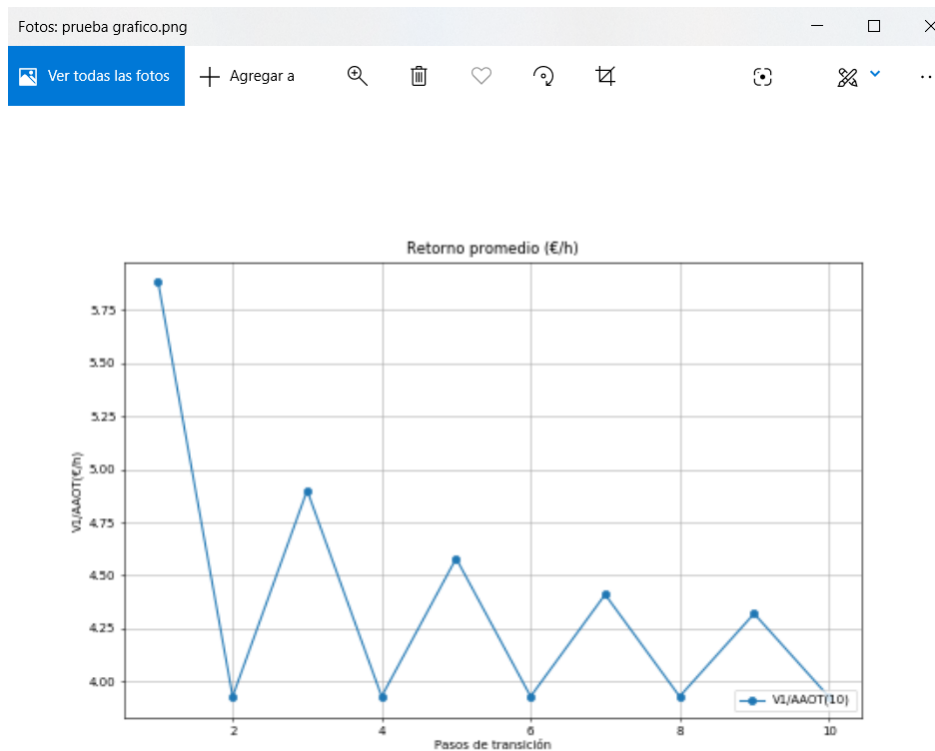


Figura 4-35: Imagen creada a partir de la herramienta.

4.5 Calculadora para Tau propio

En esta sección el usuario podrá introducir un valor de tau que desee, es decir, un momento dado (instante en horas) para realizar la intervención del mantenimiento preventivo y el paso de transición al que pertenece. Al igual que el estado inicial en el que se encontraba el sistema:

- V1: Estado inicial *Operativo*.
- V2: Estado inicial *Correctivo*.
- V3: Estado inicial *Preventivo*.

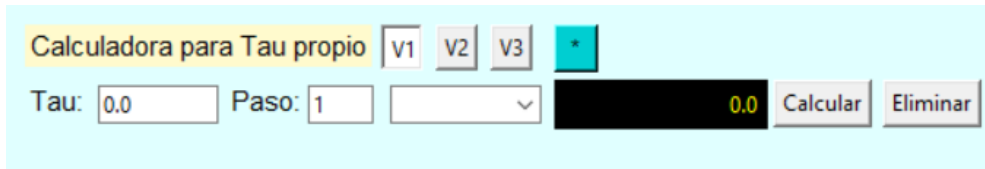


Figura 4-36: Sección de PM-3E, calculadora para tau propio

Acto seguido, el usuario deberá elegir la opción que desea calcular para dicho tau:

- V: Vector de retorno promedio acumulado.
- AOT: Tiempo de operación promedio.
- AAOT: Tiempo de operación promedio acumulado.
- V/AAOT: Retorno promedio.

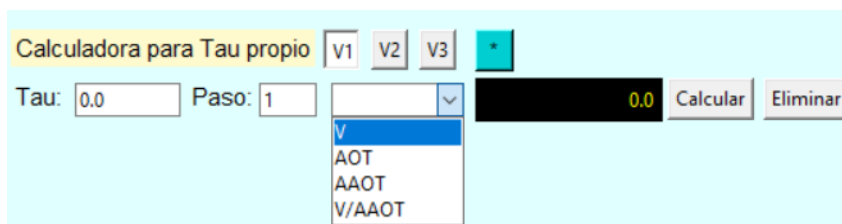


Figura 4-37: Opciones de cálculo para un tau propio dado por el usuario.

Una vez elegida la opción deberá pulsar “Calcular”.

Ejemplo de funcionamiento:



Figura 4-38: Ejemplo de uso de la Calculadora para Tau propio de la herramienta.

Antes de realizar un nuevo cálculo deberá pulsar “Eliminar”.

4.6 Botones Ayuda

Estos botones se presentan de este modo:



Figura 4-39: Señalización del botón ayuda de la herramienta

Se encuentran distribuidos a lo largo de toda la herramienta, y su función es aclarar las dudas que le pueden surgir al usuario al interactuar con la herramienta. El clicarlos aparece una ventana nueva.

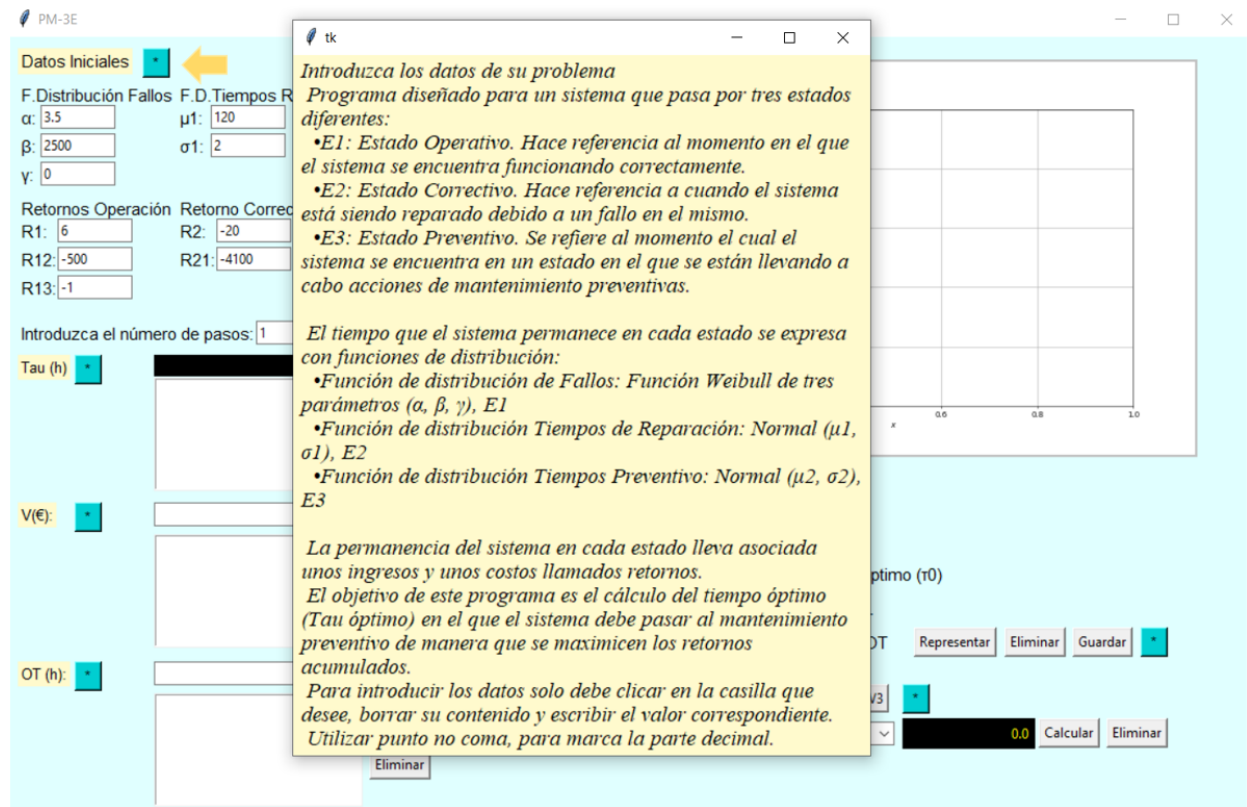


Figura 4-40: Nota explicativa emergente al clicar en un botón Ayuda asociado a “Datos Iniciales”.

Cada uno de los botones ayuda contiene información explicativa para el usuario. Las notas explicativas que podemos encontrar son éstas:

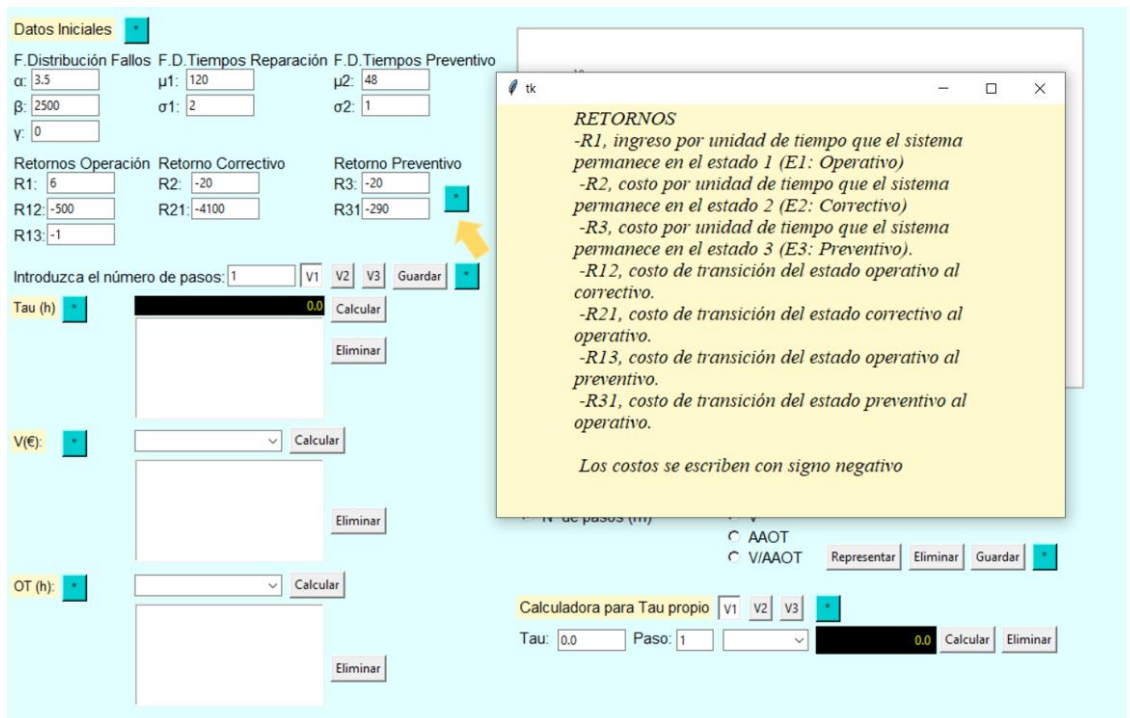


Figura 4-41: Nota explicativa emergente al clicar en un botón Ayuda asociado a “Retornos”.

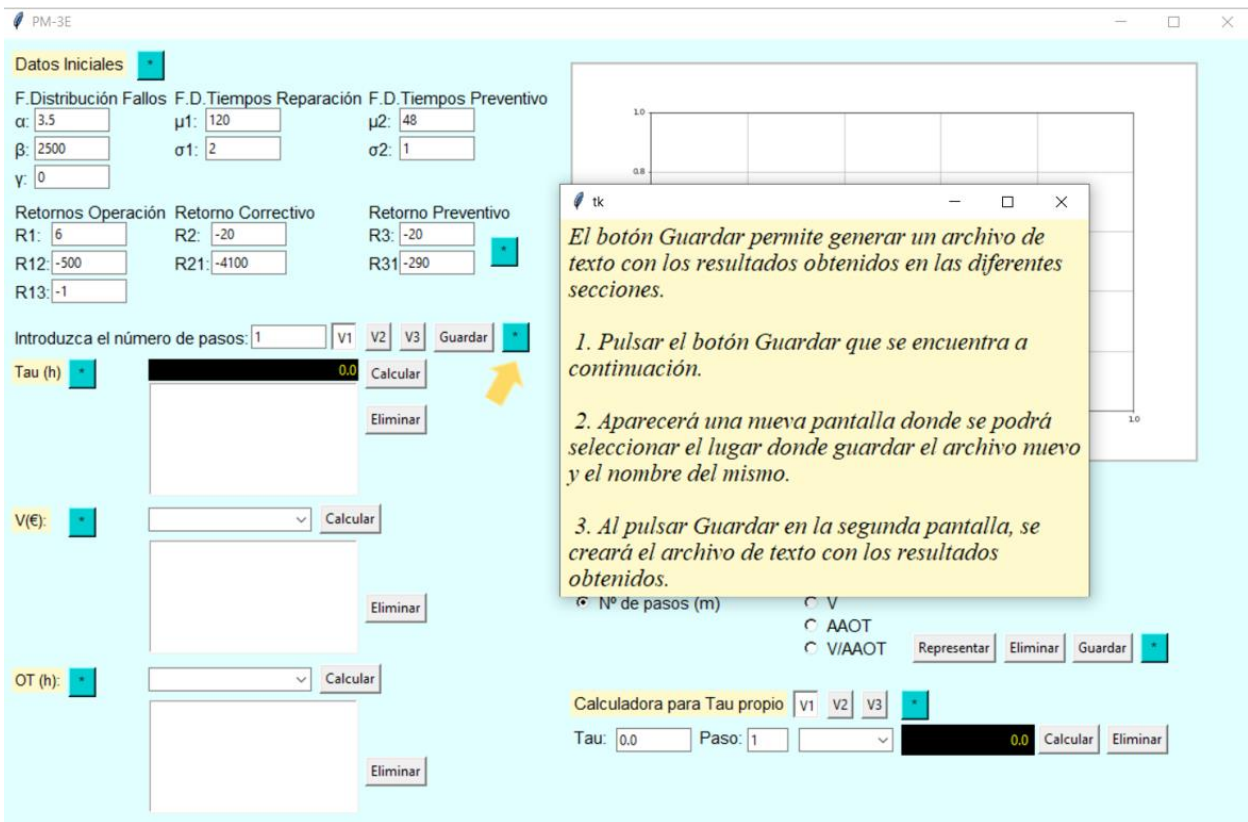


Figura 4-42: Nota explicativa emergente al clicar en un botón Ayuda, asociado al botón para guardar el archivo.

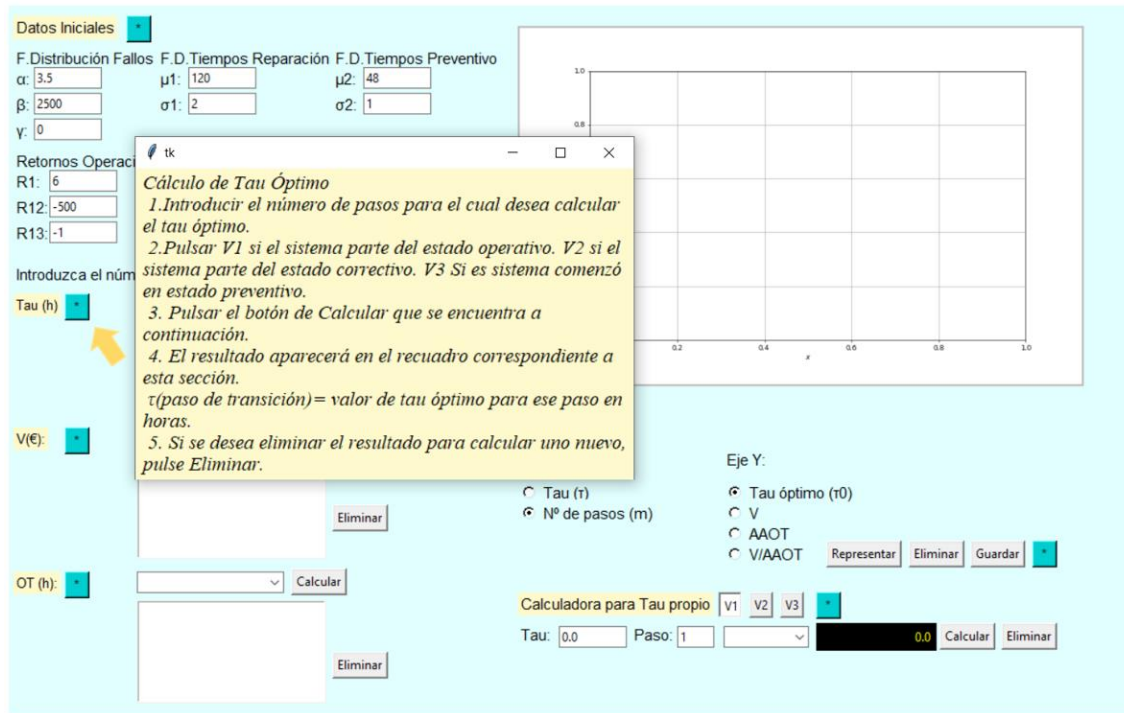


Figura 4-43: Nota explicativa emergente al clicar en un botón Ayuda asociado al cálculo de tau óptimo.

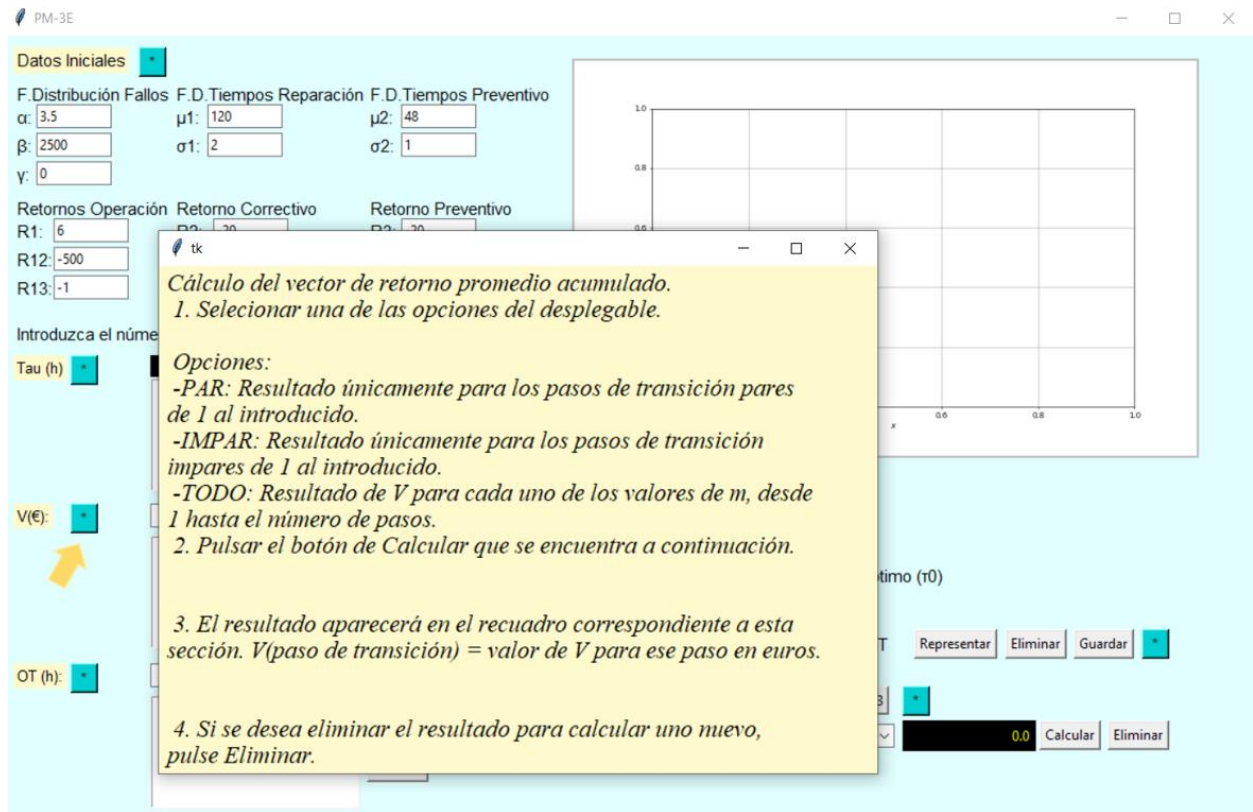


Figura 4-44: Nota explicativa emergente al clicar en un botón Ayuda asociado al cálculo del vector de retorno promedio acumulado.

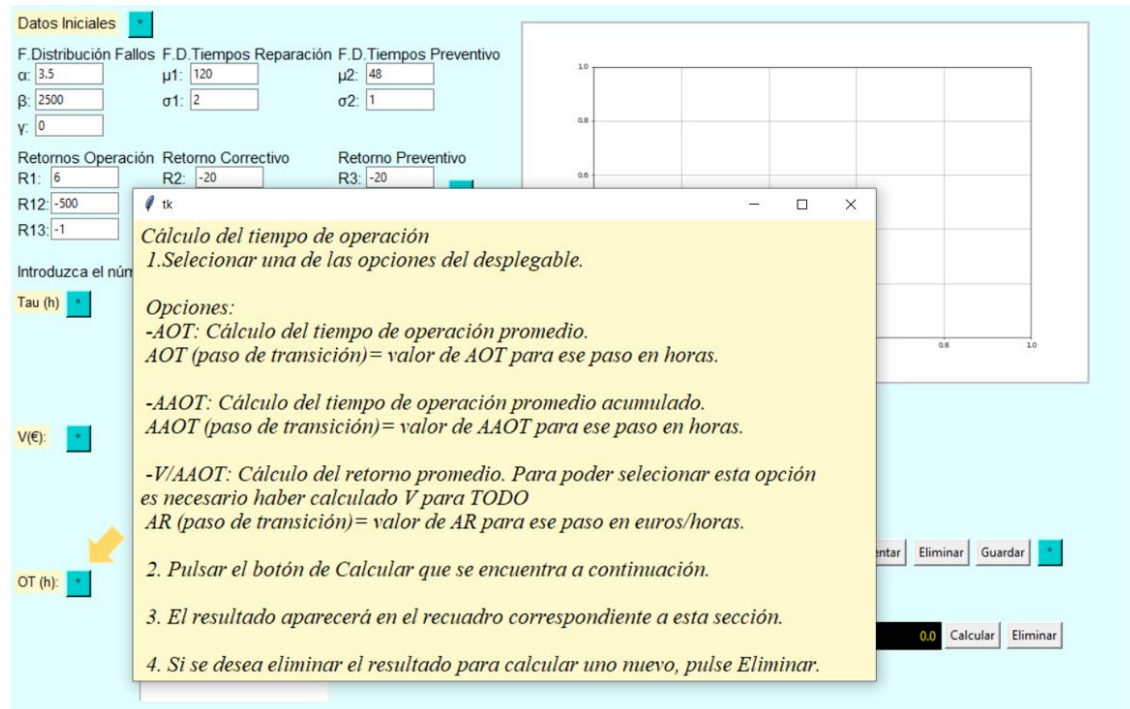


Figura 4-45: Nota explicativa emergente al clicar en un botón Ayuda, asociado al cálculo de los tiempos de operación.

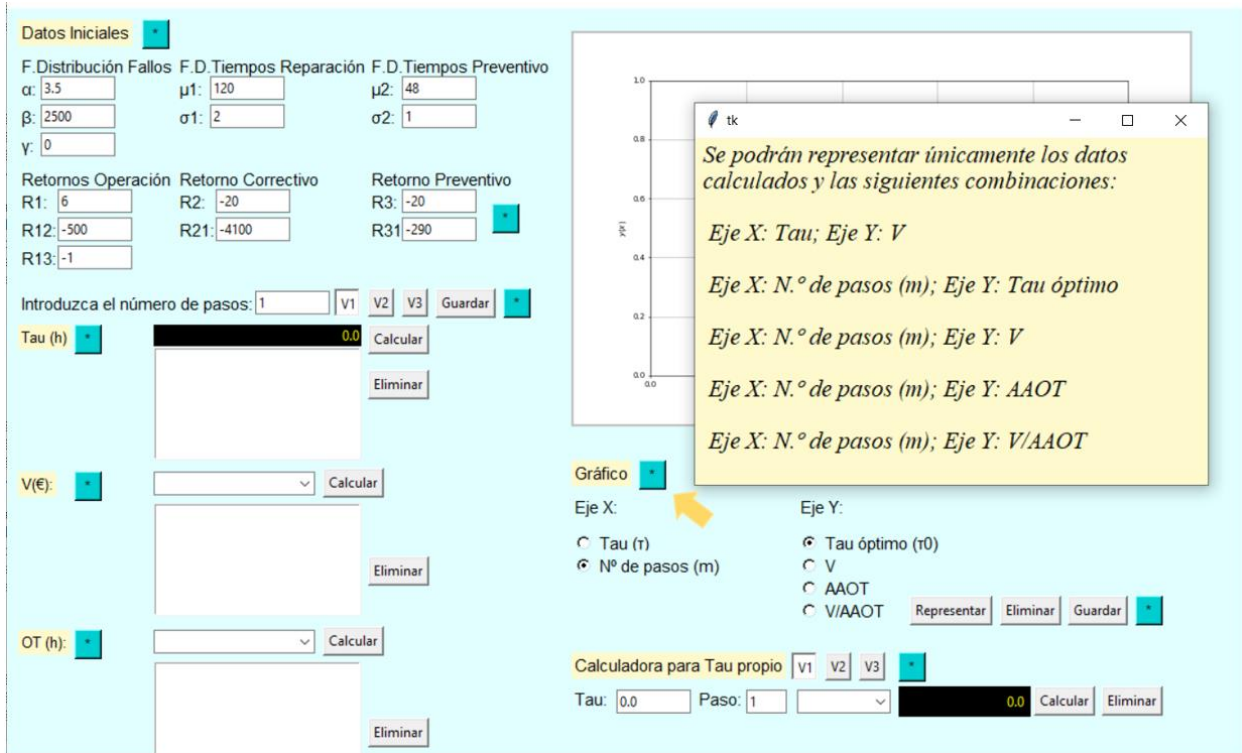


Figura 4-46: Nota explicativa emergente al clicar en un botón Ayuda, asociado a las opciones de representación gráfica que dispone el programa.

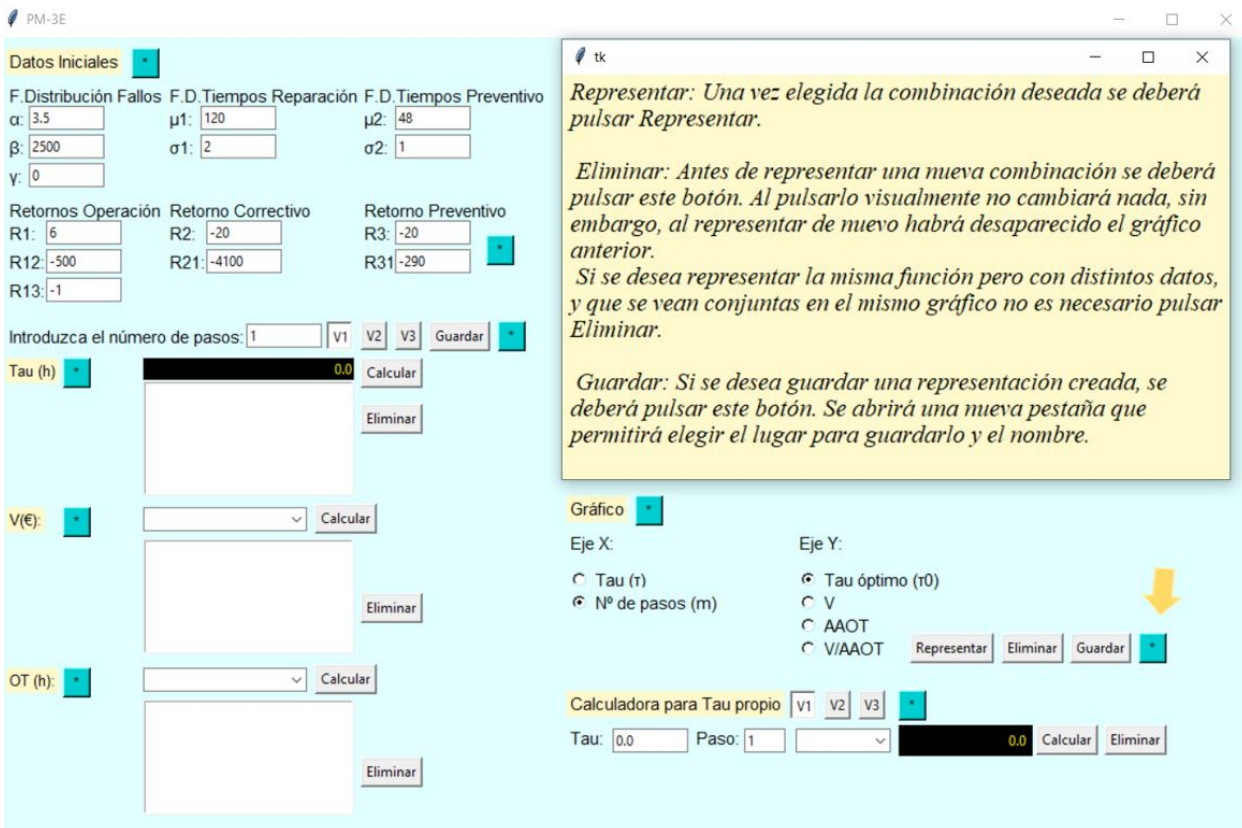


Figura 4-47: Nota explicativa emergente al clicar en un botón Ayuda asociado a la representación gráfica de los datos.

The screenshot shows a software window titled "Metodo stm 3-estados". The interface includes several input fields for parameters: α (3.5), β (2500), γ (0), μ_1 (120), μ_2 (48), and σ_1 (tk). There are also sections for "Retornos Operación" (R1: 6, R12: -500, R13: -1) and "Retornos Preventivo" (R2, R21). A section for "Introduzca el número de p" has a "Tau (h)" dropdown menu. At the bottom, there is a "Calculadora para Tau propio" section with input fields for "Tau" (0.0) and "Paso" (1), and buttons for "Calcular" and "Eliminar". A yellow help note is overlaid on the interface, providing instructions for using the calculator.

Calculadora para Tau propio

1. Introducir el valor de tau.
2. Introducir el paso que le interese.
3. Marcar V1 si el sistema comenzó en estado operativo / V2 si el sistema comenzó en estado correctivo/ V3 Si es sistema comenzó en estado preventivo
4. Seleccionar una de las opciones de cálculo

Opciones:

- V: Cálculo del vector de retorno promedio acumulado para el paso introducido.
- AOT: Cálculo del tiempo de operación promedio.
- AAOT: Cálculo del tiempo de operación promedio acumulado.
- V/AAOT: Cálculo del retorno promedio

4. Pulsar el botón Calcular que se encuentra a continuación.
5. El resultado para la opción elegida aparecerá en el recuadro negro.
6. Si desea calcular un nuevo elemento pulse previamente Eliminar.

Figura 4-48: Nota explicativa emergente al clicar en un botón Ayuda asociado al cálculo de parámetros con el uso de un tau dado.

Las notas explicativas se pueden abrir siempre que el usuario las necesite y cerrar cuando ya las haya leído.

5 RESULTADOS EXPERIMENTALES

Se lucha, pero no se ruega.

- Frida Kahlo -

En este apartado se comprobará el funcionamiento de la herramienta. Para ello se compararán los resultados que se obtienen en la misma con los resultados que se expresan en el caso de estudio, descrito anteriormente.

Como se ha descrito en la presentación del caso de estudio, el cálculo objetivo es el valor del tiempo óptimo hasta el mantenimiento preventivo y la obtención del valor del retorno promedio (AR). Para llegar al valor del retorno promedio es necesario calcular otros elementos previos. Por ello, se va a comparar en cada apartado el resultado obtenido para cada elemento, hasta llegar al valor del retorno promedio. La distribución de las comprobaciones se presenta así:

- Apartado 5.1: Se muestran los parámetros iniciales tomados del estudio e incluidos en el programa.
- Apartado 5.2: Comprobación del resultado para el valor de tau óptimo (tiempo óptimo hasta el mantenimiento preventivo) para tres casos distintos.
- Apartado 5.3: Comprobación del primer vector de retorno promedio acumulado, V1, para tres valores pares de m (pasos de transición distintos).
- Apartado 5.4: Comprobación del valor de V1, para tres valores impares de m.
- Apartado 5.5: Comprobación del tiempo de operación promedio, AOT.
- Apartado 5.6: Comprobación del primer vector de retorno acumulado para 10 pasos de transición.
- Apartado 5.7: Comprobación del valor de AAOT, es decir, el tiempo de funcionamiento promedio acumulado, en horas.
- Apartado 5.8: Comprobación del valor obtenido para el retorno promedio, AR. Haciendo referencia al cociente de V, en este caso V1, entre AAOT.

Para cada uno de los apartados se incluyen imágenes tomadas del artículo de referencia donde se muestran los resultados de cada elemento. Para poder verificar estos resultados con los de la herramienta, se han tomado imágenes del programa mostrando los resultados obtenidos tras calcular el elemento indicado en cada caso.

5.1 Datos Iniciales

Para este primer apartado simplemente se verifica que los datos iniciales introducidos necesarios para desarrollar los cálculos, coinciden con los valores que se utilizaron para el estudio en el artículo [1].

En primer lugar, se exponen los datos del artículo:

Tabla 5–1. Valores de los parámetros utilizados para el caso de estudio [1]

Time to failure probability distribution		Repair time probability distribution		Preventive time probability distribution	
Weibull (α , β)		Normal (μ_1 , σ_1)		Normal (μ_2 , σ_2)	
α	β	μ_1	σ_1	μ_2	σ_2
3.50	2,500	120	2	48	1

Tabla 5-2. Valores de los retornos utilizados para el caso de estudio [1].

When operation (E1): Returns and transition costs			When under repair (E2): Returns and transitions costs		When under PM (E3): Returns and transitions costs	
R ₁ (€/h)	R ₁₂ (€)	R ₁₃ (€)	R ₂ (€/h)	R ₂₁ (€)	R ₃ (€/h)	R ₃₁ (€)
45/12/6	-500	-1.0	-20	-4100	-20	-290

En segundo lugar, se presenta la imagen tomada de la herramienta creada, donde se verifican los valores introducidos para cada uno de los parámetros.



Figura 5-1: Datos iniciales. [Imagen tomada de la herramienta PM-3E].

Si se comparan las imágenes (Tabla 5-1 y Tabla 5-2 vs. Figura 5-1), se puede observar que se han introducido los mismos valores iniciales en la herramienta que los utilizados en el caso de estudio del artículo [1].

En la Figura 5-1 se puede apreciar que para el retorno de operación R1 se ha introducido únicamente 6, mientras que en los datos del artículo aparecen tres valores: 45, 12 y 6. Estos tres valores se refieren a tres casos de estudio diferentes, en los que únicamente se ha variado el valor de R1. En el próximo apartado se puede ver como no sólo se ha resuelto con la herramienta el caso para R1 = 6, sino que también se dan los resultados para los otros dos casos al igual que se ha desarrollado en el artículo.

5.2 Comprobación del Valor Óptimo de τ

El primer elemento calculado que se presenta en el artículo [1] es el valor óptimo de τ en horas, que se indica como τ_0 , inicializando el sistema en el estado *Operativo*. Los resultados que se tomaron del artículo se enseñan en esta tabla:

Tabla 5-3. τ_0 para diferentes pasos y retornos de operación [1].

Transition number	Optimal value τ_0 (hours)		
	Case 1 (R1=45€/h)	Case 2 (R1=12€/h)	Case 3 (R1=6€/h)
m			
1	13,229.20	7,796.88	5,908.93
3	6,351.69	3,743.49	2,837.04
5	5,754.31	3,391.41	2,570.21
7	5,520.24	3,253.46	2,465.66
9	5,394.75	3,179.50	2,409.61
Even Step	4,976.65	2,933.08	2,222.86

Como se puede apreciar en la tabla, se estudia el valor óptimo de tau para 9 pasos de transición, es decir m = 9, partiendo el sistema del estado *Operativo*, y para tres valores de R1 distintos.

Acto seguido se exponen los resultados que se obtuvieron con la herramienta al calcular el valor óptimo de tau

para 9 pasos de transición para cada valor de R1. Para cada caso, el resultado se presenta con dos imágenes de la herramienta, ya que en una única imagen no se llega a ver el resultado para el paso de transición 7 y en adelante. A efectos comparativos se incluyen los datos calculados en el artículo de referencia.

5.2.1 Caso 1 (R1=45)

Comparativa de resultados que se obtuvieron para este caso:

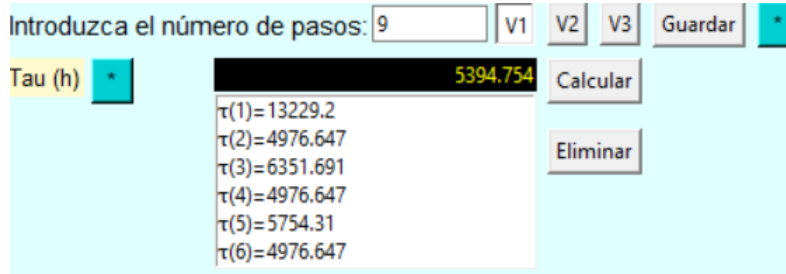


Figura 5-2: τ_0 para 9 pasos de transición y R1=45. Resultado de 1-6 pasos. [Imagen tomada de la herramienta PM-3E].

Tabla 5-4. τ_0 para R1=45 [1]

Transition number	Optimal value τ_0
m	Case 1 (R1=45€/h)
1	13,229.20
3	6,351.69
5	5,754.31
7	5,520.24
9	5,394.75
Even Step	4,976.65

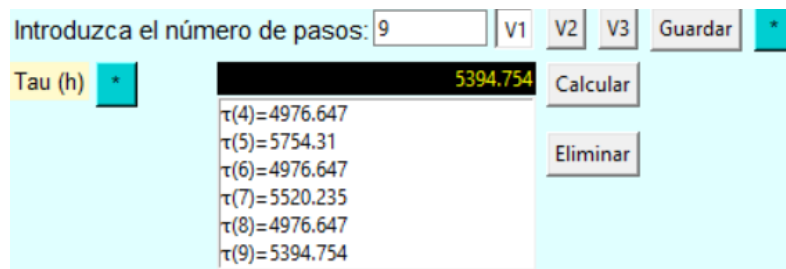


Figura 5-3: τ_0 para 9 pasos de transición y R1=45. Resultado de 4-9 pasos. [Imagen tomada de la herramienta PM-3E].

5.2.2 Caso 2 (R1=12)

Comparativa de resultados que se obtuvieron para este caso:

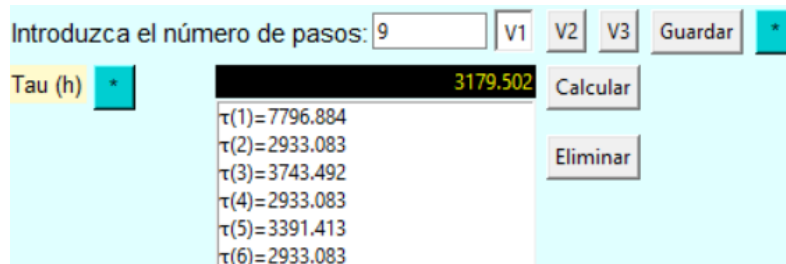


Figura 5-4: τ_0 para 9 pasos de transición y R1=12. Resultado de 1-6 pasos. [Imagen tomada de la herramienta PM-3E].

Tabla 5-5. τ_0 para R1=12 [1]

Transition number	Optimal value τ_0
m	Case 2 (R1=12€/h)
1	7,796.88
3	3,743.49
5	3,391.41
7	3,253.46
9	3,179.50
Even Step	2,933.08

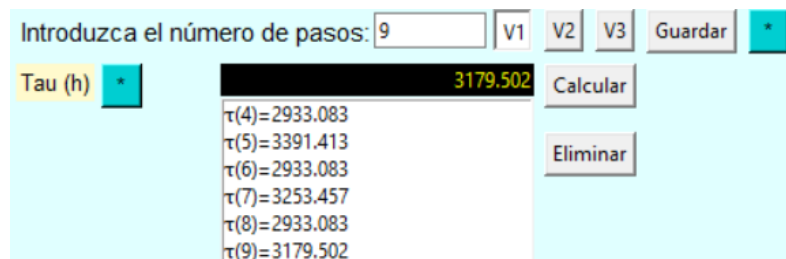


Figura 5-5: τ_0 para 9 pasos de transición y R1=12. Resultado de 4-9 pasos. [Imagen tomada de la herramienta PM-3E].

5.2.3 Caso 3 (R1= 6)

Comparativa de resultados que se obtuvieron para este caso:



Figura 5-6: τ_0 para 9 pasos de transición y R1=6. Resultado de 1-6 pasos. [Imagen tomada de la herramienta PM-3E].

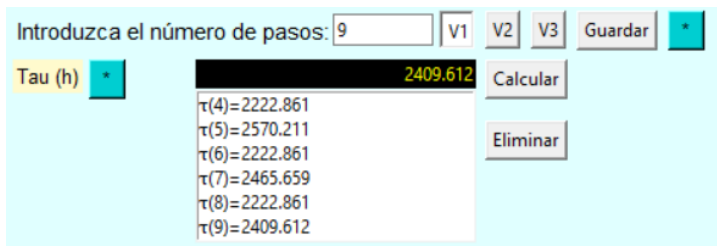


Figura 5-7: τ_0 para 9 pasos de transición y R1=6. Resultado de 4-9 pasos. [Imagen tomada de la herramienta PM-3E].

Tabla 5-6. τ_0 para R1=6 [1]

Transition number	Optimal value τ_0
Case 3 (R1=6€/h)	
m	
1	5,908.93
3	2,837.04
5	2,570.21
7	2,465.66
9	2,409.61
Even Step	2,222.86

Como se puede ver en las imágenes tomadas de la herramienta, ésta calcula el valor de tau óptimo desde 1 hasta el valor de pasos de transición introducido. En los resultados del artículo únicamente se muestra el valor de los pasos de transición impares, ya que el valor óptimo de tau para los valores pares es siempre el mismo. Este valor se señala en las tablas de resultados del artículo como el “Even step” (Paso par). Si se comparan los resultados obtenidos con la herramienta y los de las tablas de resultados del artículo, se puede ver que son los mismos.

Resumen de resultados para tau óptimo para el paso de transición m=9 y el paso par:

- Caso 1 (R1=45): $\tau_0 (9) = 5,394.75h$; $\tau_0 (even\ step) = 4,976.65h$
- Caso 2 (R1=12): $\tau_0 (9) = 3,179.50h$; $\tau_0 (even\ step) = 2,933.08h$
- Caso 3 (R1=6): $\tau_0 (9) = 2,409.61h$; $\tau_0 (even\ step) = 2,222.86h$

Por lo que se puede verificar que la herramienta funciona correctamente para el cálculo del valor óptimo de tau.

5.2.4 Representación gráfica del valor óptimo de tau versus el número de transiciones

El artículo incluye una gráfica donde se reflejan los resultados obtenidos para τ_0 respecto a los tres casos. De igual forma se va a comprobar que la gráfica que se obtiene a través de la herramienta es correcta.

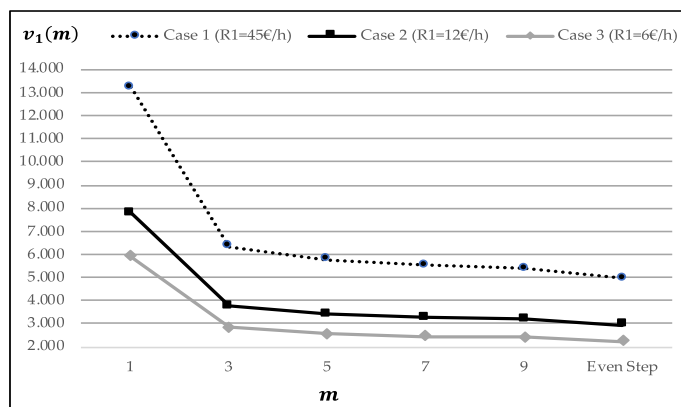


Figura 5-8: Tiempo hasta la acción preventiva versus el número de transiciones [1].

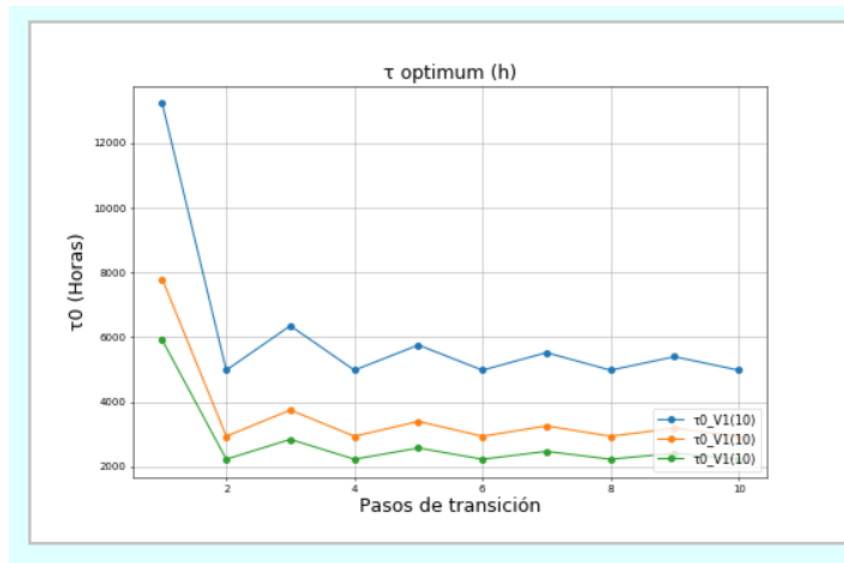


Figura 5-9: Representación gráfica del valor óptimo de tau, para 9 pasos en tres casos distintos. [Imagen tomada de la herramienta PM-3E].

Al igual que en la gráfica del artículo se han representado los tres casos distintos. Para la herramienta, la línea azul coincide con el caso 1 ($R1=45$), la línea naranja con el caso 2 ($R1=12$) y la línea verde con el caso 3 ($R1=6$).

Puesto que, en la gráfica del artículo, Figura 5-8, únicamente se han representado los pasos de transición impares y en la de la herramienta se representan tanto todos, las gráficas no se manifiestan exactamente iguales. Sin embargo, si de la Figura 5-9 se presta atención únicamente a los valores obtenidos para los pasos de transición impares, se ve como ambas siguen el mismo recorrido, siendo el caso 1 donde se obtienen valores más altos.

5.3 Comprobación del Primer Componente del Vector de Retorno Promedio Acumulado para pasos de transición pares, V1(m par)

Para esta comprobación de funcionamiento de la herramienta, se va a verificar si el valor del primer componente del vector de retorno promedio acumulado es correcto. Este trabajo se va a realizar a través de la comparativa de gráficas, ya que el resultado de los valores de V1 que nos aporta el artículo de referencia [1] se muestran a través de la imagen que se exhibe a continuación:

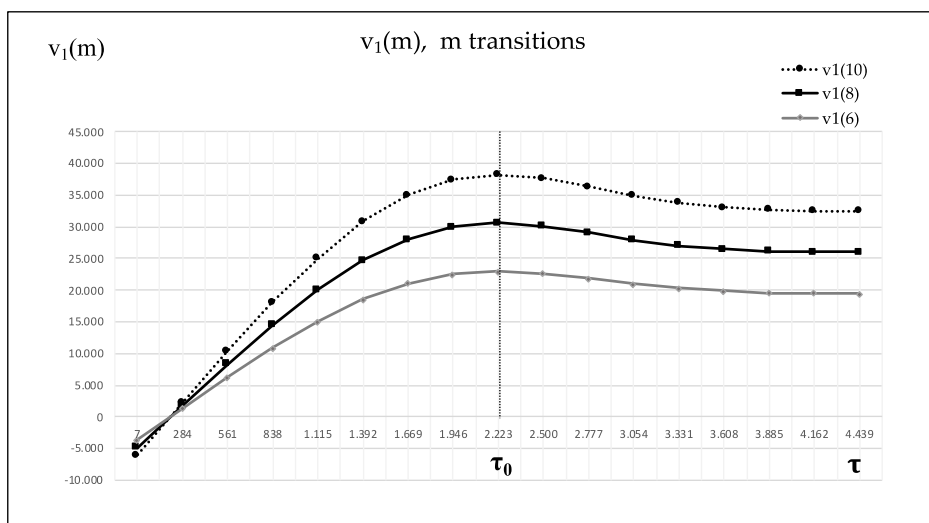


Figura 5-10: Primer componente del vector de retorno acumulado promedio versus tiempo para τ preventivo, para tres valores de transiciones pares diferentes [1].

Como se indica en el pie de la figura Figura 5-10, ésta se corresponde con la representación de V_1 versus τ para tres valores de transiciones (m) diferentes, en este caso 10, 8 y 6. Igualmente, en el gráfico se nos señala el punto más alto de cada uno de los casos, siendo éste el valor de τ óptimo, τ_0 .

Para que la herramienta pueda generar un gráfico como el mostrado en la Figura 5-10, no es necesario calcular ningún valor previo, directamente se introduce el paso de transición deseado y se pasa a la sección “Gráfico” del programa.

5.3.1 Caso 1 ($m=10$)

Como se explica en el Manual de Usuario del Programa PM-3E, primero se debe introducir el valor de m que se desee para obtener los valores de V , y seleccionar “V1” en caso de buscar el primer vector, es decir, inicializando el sistema en el estado *Operativo*.



Figura 5-11: Introducción de valor de $m=10$. [Imagen tomada de la herramienta PM-3E].

Seguidamente en la sección de gráfico se marcan estas opciones y se pulsa “Representar”:

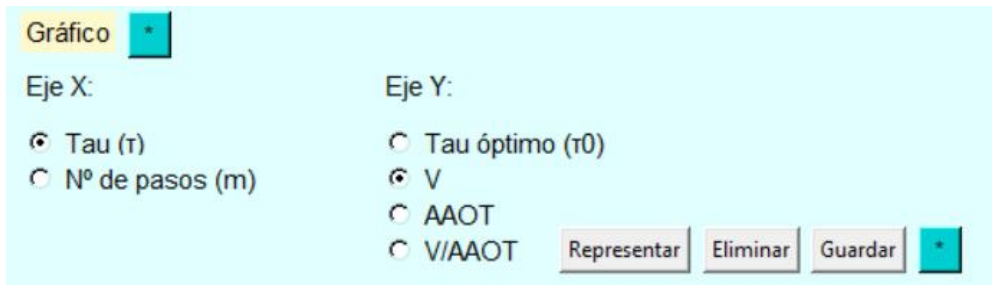


Figura 5-12: Selección para la representación gráfica del tiempo preventivo frente al vector de retorno promedio acumulado. [Imagen tomada de la herramienta PM-3E].

Resultado al seleccionar “Representar”:

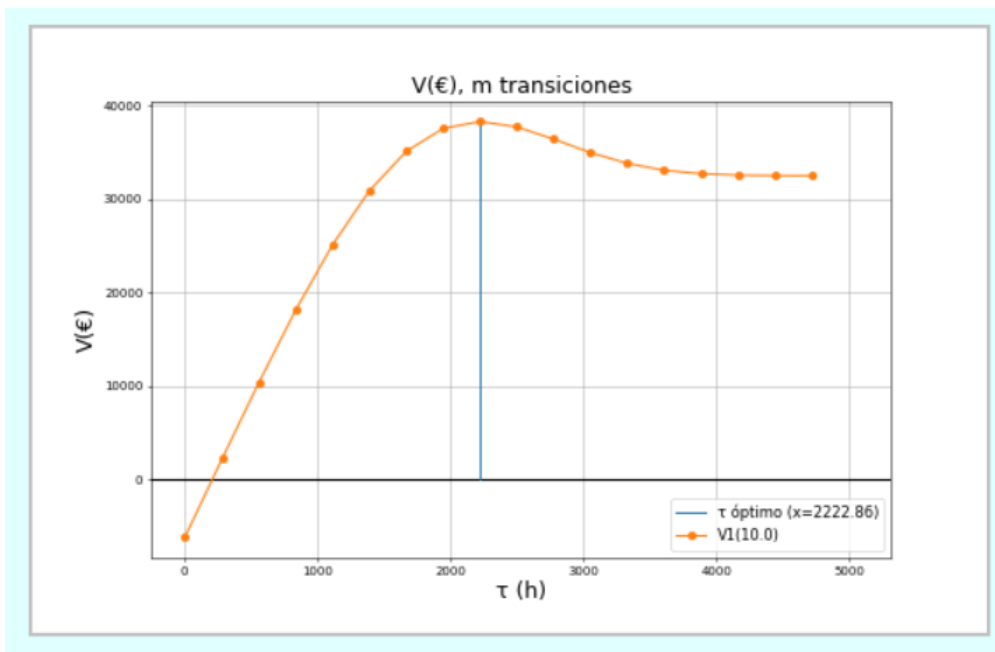


Figura 5-13 Primer componente del vector de retorno acumulado promedio versus tiempo para τ preventivo, para un valor de transición par ($m=10$). [Imagen tomada de la herramienta PM-3E].

5.3.2 Caso 2 (m=8)

Para poder mostrar varias representaciones en una misma gráfica se introducen sucesivamente los valores de m para los distintos casos y, sin pulsar “Eliminar”, se clicca en “Representar” como en el caso anterior:

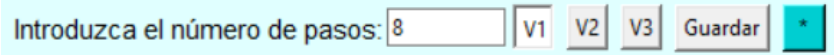


Figura 5-14: Introducción de valor de $m=8$. [Imagen tomada de PM-3E].

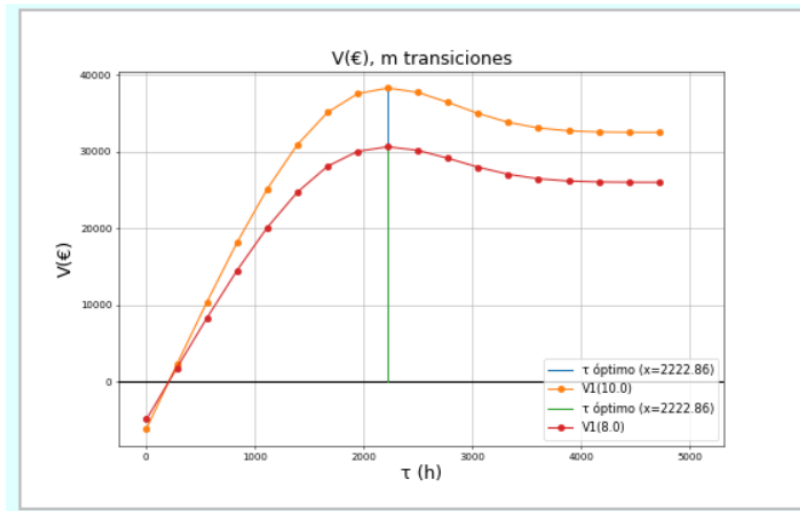


Figura 5-15: Primer componente del vector de retorno acumulado promedio versus tiempo para τ preventivo, para dos valores de transiciones pares diferentes ($m=10$, línea naranja y $m=8$, línea roja). [Imagen tomada de PM-3E].

5.3.3 Caso 3 (m=6)

Se introduce el último caso:

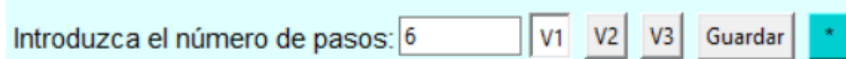


Figura 5-16: Introducción de valor de $m=6$. [Imagen tomada de PM-3E]

El resultado para los tres casos conjuntamente es:

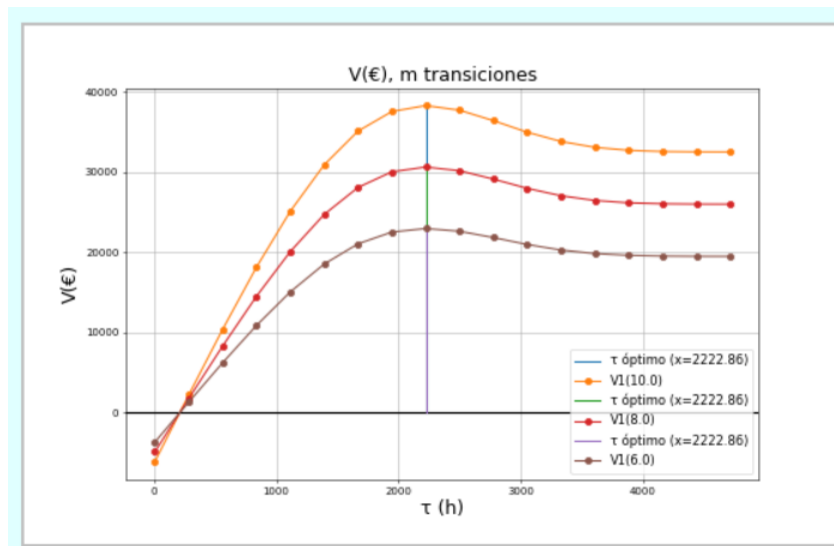


Figura 5-17: Representación gráfica del primer componente del vector retorno promedio acumulado para τ preventivo, para tres valores de transiciones pares diferentes ($m=10$, línea naranja; $m=8$, línea roja y $m=6$, línea marrón). [Imagen tomada de PM-3E].

Si se comparan ambas gráficas (Figura 5-10 vs. Figura 5-15Figura 5-17) se puede ver que el resultado que se obtiene es igual para cada uno de los tres casos, siendo el valor de V_1 más alto el que se corresponde con x igual a tau óptimo. Como se ha explicado previamente, el valor de tau óptimo para las transiciones pares, es el mismo. Éste se representa en ambos gráficos con una línea vertical y para este caso, tau es igual a 2223 horas.

5.4 Comprobación del Primer Componente del Vector de Retorno Promedio Acumulado para pasos de transición impares, $V_1(m \text{ impar})$

Para esta comprobación se va a seguir exactamente el procedimiento del apartado anterior, únicamente que se van a estudiar casos con pasos de transición de valor impar: 9, 7 y 5.

En primer lugar, se muestra el gráfico de resultados que se presenta en el artículo de referencia:

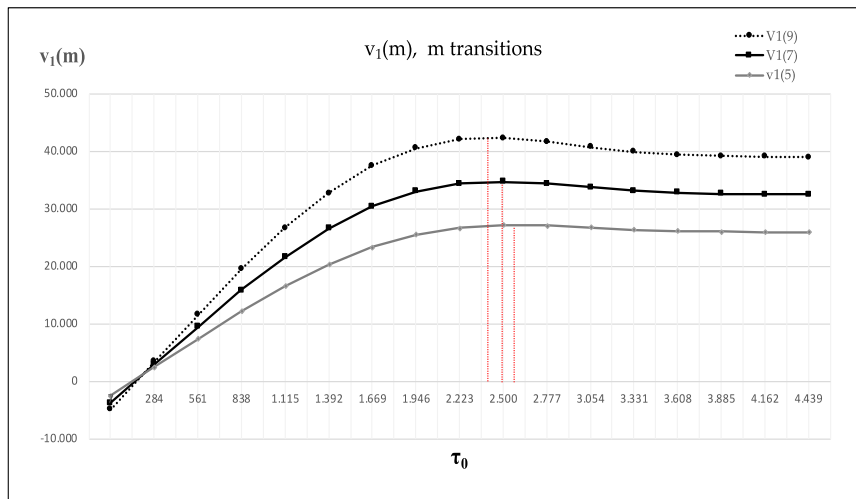


Figura 5-18: Primer componente del vector de retorno acumulado promedio versus tiempo para τ preventivo, para tres valores de transiciones impares diferentes [1].

En el artículo, adicionalmente se nos indica el valor de tau para el resultado más alto que se obtuvo de V_1 :

- Para $m = 5$, el tiempo $\tau_0(5) = 2,570.21$ horas da el valor más alto para $v_1(5)$.
- Para $m = 7$, el tiempo $\tau_0(7) = 2,465.66$ horas da el valor más alto para $v_1(7)$.
- Para $m = 9$, el tiempo $\tau_0(9) = 2,409.61$ da el mayor valor de $v_1(9)$.

Como en el apartado anterior, los resultados se obtienen dentro del programa directamente en la sección “Gráfico”.

5.4.1 Caso 1 (m=9)

Primero se debe introducir el valor de m para el cual queremos obtener los valores de V_1 y se selecciona “ V_1 ”.

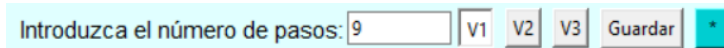


Figura 5-19: Introducción de valor de $m=9$. [Imagen tomada de PM-3E].

Seguidamente en la sección de gráfico se seleccionan estas opciones y se pulsa “Representar”:

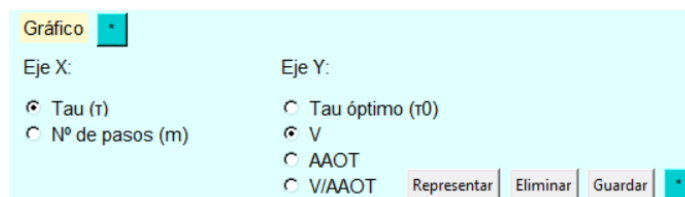


Figura 5-20: Selección para la representación gráfica del tiempo preventivo frente al vector de retorno promedio acumulado. [Imagen tomada de PM-3E].

Resultado al seleccionar "Representar":

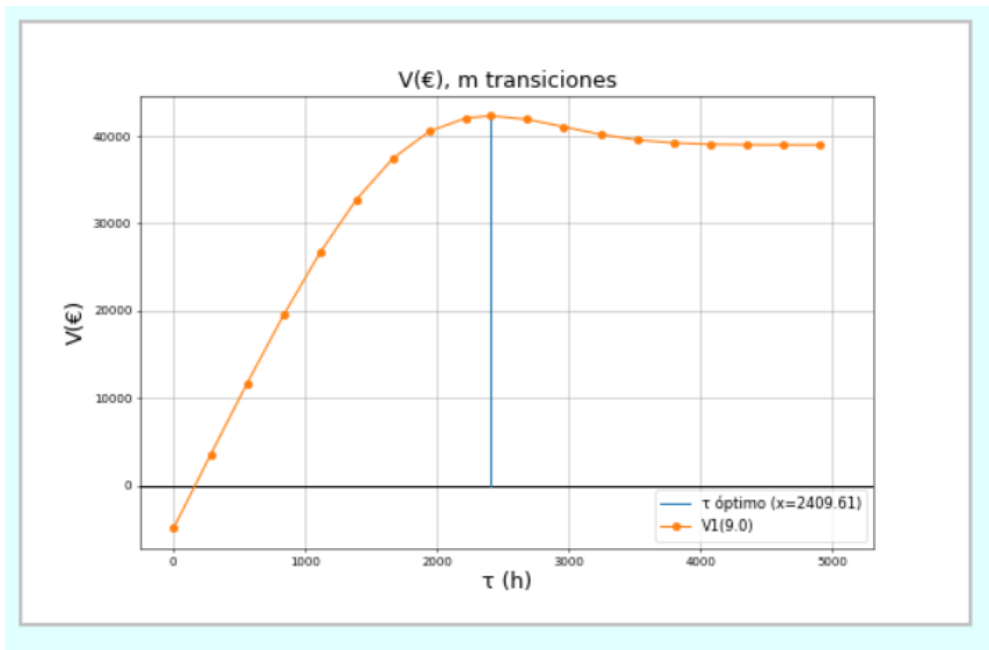


Figura 5-21: Primer componente del vector de retorno acumulado promedio versus tiempo para τ preventivo, para un valor de transición impar ($m=9$). [Imagen tomada de PM-3E].

5.4.2 Caso 2 ($m=7$)

El resultado que se obtiene para este caso es:

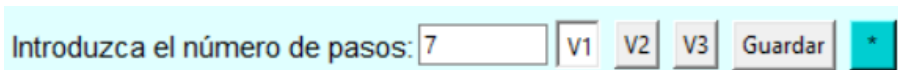


Figura 5-22: Introducción de valor de $m=7$. [Imagen tomada de PM-3E].

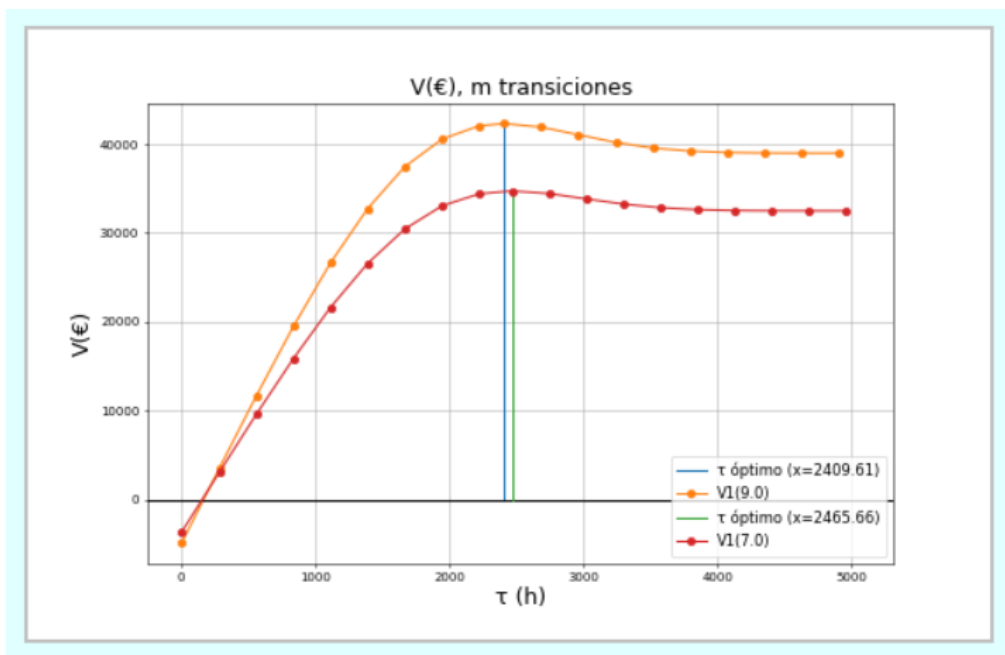


Figura 5-23: Primer componente del vector de retorno acumulado promedio versus tiempo para τ preventivo, para dos valores de transiciones impares diferentes ($m=9$, línea naranja y $m=7$, línea roja). [Imagen tomada de PM-3E].

5.4.3 Caso 3 (m=5)

El resultado que se obtiene para este caso es:

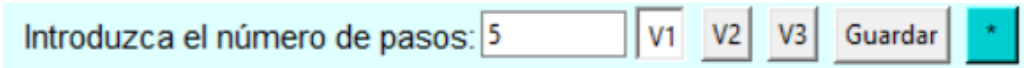


Figura 5-24: Introducción de valor de m=5. [Imagen tomada de PM-3E].

La representación gráfica de los resultados obtenidos se refleja de esta manera:

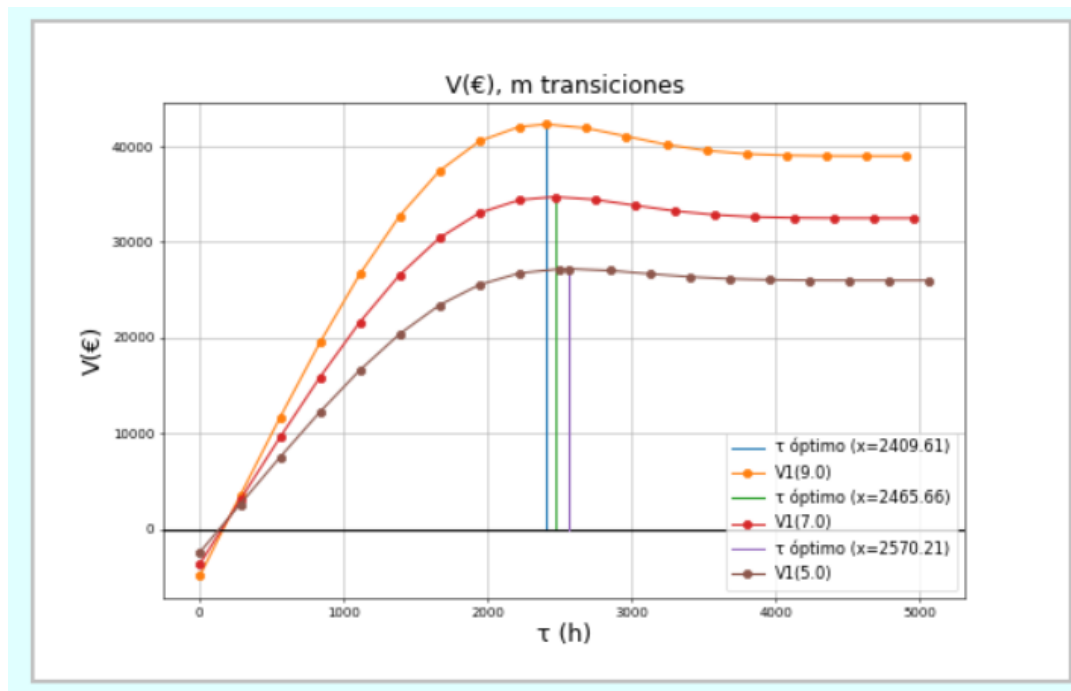


Figura 5-25: Representación gráfica del primer componente del vector retorno promedio acumulado para τ preventivo, para tres valores de transiciones impares diferentes ($m=9$, línea naranja; $m=7$, línea roja y $m=5$, línea marrón). [Imagen tomada de PM-3E].

En la Figura 5-25, como se indica en la esquina inferior derecha, los valores representados en naranja reflejan el caso 1 ($m=9$), siendo éste el que obtiene valores más altos para V_1 . Los valores representados en rojo y marrón corresponden al caso 2 ($m=7$) y al caso 3 ($m=5$), respectivamente.

Se puede apreciar que los resultados de la herramienta coinciden con el gráfico del artículo de referencia, Figura 5-18, siendo los valores de tau para el resultado más alto que se obtuvo de V_1 los siguientes:

- Para $m = 5$, el tiempo $\tau_0(5) = 2,570.21$ horas da el valor más alto para $v_1(5)$.
- Para $m = 7$, el tiempo $\tau_0(7) = 2,465.66$ horas da el valor más alto para $v_1(7)$.
- Para $m = 9$, el tiempo $\tau_0(9) = 2,409.61$ horas da el mayor valor de $v_1(9)$.

5.5 Comprobación del Tiempo de Operación Promedio, AOT

El resultado que se expresa en el artículo para el valor de AOT es:

$$AOT = \tau_0 (1 - p) + A \cdot p = 1,948 \text{ hours} \quad (5-1)$$

Siendo los valores de cada una de las variables:

$$\tau_0 = 2,222.86, p = 0.4846 \text{ y } A = 802.53$$

Para el cálculo de AOT, la herramienta debe contener los valores de τ_0 , p y A . Estos valores se deben calcular a través de la misma previamente, de este modo se guardarán y podrán utilizarse más tarde en el cálculo de AOT. El valor de τ_0 que se utiliza en el artículo de referencia para este apartado corresponde con el Caso 3 (R1= 6) para un paso de transición par. En este caso se ha escogido 10 pasos de transición.

En primer lugar, se obtendrá el resultado de τ_0 :

Figura 5-26: Cálculo de τ_0 para $m=10$. [Imagen tomada de PM-3E].

En segundo lugar, para obtener el valor de p y A es necesario realizar en la herramienta el cálculo de $V1$:

Figura 5-27: Cálculo de $V1$ pasos de transición impares. [Imagen tomada de PM-3E].

En este caso, se ha seleccionado la opción del cálculo de $V1$ únicamente para los pasos de transición impares. Es una demostración de que no tiene importancia que valores de V se haya elegido calcular, ya que para todas las opciones (Par, Impar y Todo) queda registrado el valor de p y de A .

Finalmente, se realiza la operación de AOT:

Figura 5-28: Cálculo de AOT para 10 pasos de transición. [Imagen tomada de PM-3E].

Con la Figura 5-28 tomada de la herramienta, se puede comprobar que el resultado calculado para AOT, 10 pasos de transición, es correcto:

$$AOT (10) = 1,948 \text{ horas}$$

5.6 Comprobación del Primer Vector de Retorno Acumulado para 10 pasos de transición

En el artículo se enseña la siguiente tabla con el resultado del primer vector de retorno acumulado, V_1 , para 10 pasos de transición:

Tabla 5-7: Resultado del retorno promedio acumulado para 10 pasos de transición.

m	Average Accumulated Return (€) $v_1(m)$
1	11446,223
2	7652,0197
3	19098,243
4	15304,039
5	26750,262
6	22956,059
7	34402,282
8	30608,079
9	42054,302
10	38260,098

El resultado que se obtuvo con el uso de la herramienta para este caso es:

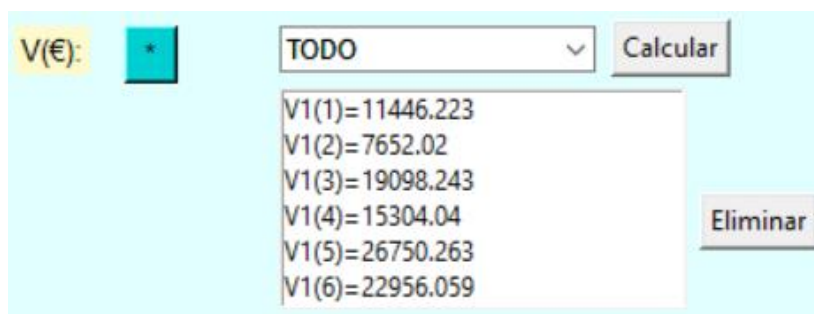


Figura 5-29: Resultado de V_1 de $m=1$ hasta $m=6$ para 10 pasos de transición. [Imagen tomada de PM-3E].

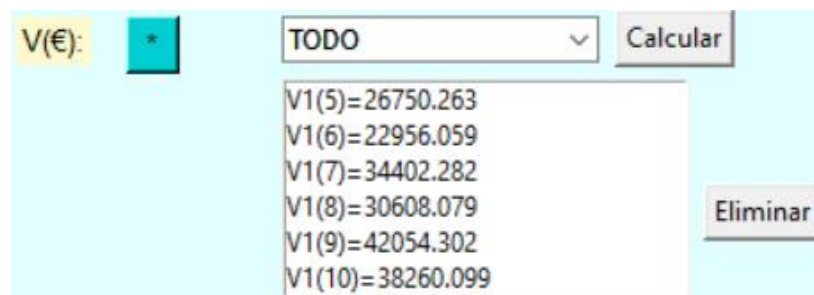


Figura 5-30: Resultado de V_1 de $m=5$ hasta $m=10$ para 10 pasos de transición. [Imagen tomada de PM-3E].

Si se comparan ambos resultados se puede apreciar que los valores son iguales para cada uno de los pasos, siendo:

$$V_1(m = 10) = 38260,099\text{€}$$

Quedando verificado el correcto funcionamiento de la herramienta para el cálculo de V_1 .

5.7 Comprobación del Valor del Tiempo de Funcionamiento Promedio Acumulado, AAOT

Los resultados que se obtienen en el artículo de referencia para el valor de AAOT se expresan en esta tabla:

Tabla 5-8: Resultado para AAOT correspondiente a 10 pasos de transición [1]

m	Average Accumulated Operating Time (hours) AAOT
1	1948,17
2	1948,17
3	3896,35
4	3896,35
5	5844,52
6	5844,52
7	7792,70
8	7792,70
9	9740,87
10	9740,87

Con los resultados guardados en la herramienta para el cálculo de V1, como se indica en el apartado anterior, se puede realizar directamente el cálculo de AAOT.

Los resultados que se obtienen son los mostrados:

Figura 5-31: Resultado de AAOT de m=1 hasta m=6 para 10 pasos de transición. [Imagen tomada de PM-3E].

Figura 5-32: Resultado de AAOT de m=5 hasta m=10 para 10 pasos de transición. [Imagen tomada de PM-3E].

Los valores obtenidos coinciden con los que se exponen en el artículo de referencia, teniendo como resultado para $m = 10$:

$$AAOT(m = 10) = 9740,87 \text{ horas}$$

Quedando verificado el correcto funcionamiento de la herramienta para el cálculo del tiempo de operación promedio acumulado.

5.8 Comprobación del Valor del Retorno Promedio, V1/AAOT

Por último, una vez que se obtienen V1 y AAOT se puede calcular el retorno promedio. Los valores que se obtienen en el artículo de referencia para este parámetro considerando 10 pasos de transición son:

Tabla 5-9: Resultados para V1/AAOT correspondientes a 10 pasos de transición [1]

m	Average Return (€/h) $v_1(m)/AAOT$
1	5,875
2	3,928
3	4,902
4	3,928
5	4,577
6	3,928
7	4,415
8	3,928
9	4,317
10	3,928

Los resultados que se obtuvieron para el retorno promedio a partir de la herramienta son:

Figura 5-33: Resultado de V1/AAOT de m=1 hasta m=5 para 10 pasos de transición. [Imagen de PM-3E].

Figura 5-34: Resultado de V1/AAOT de m=5 hasta m=10, para 10 pasos de transición. [Imagen de PM-3E].

Los valores obtenidos coinciden con los que se exponen en el artículo de referencia, teniendo como resultado para $m = 10$:

$$\text{Retorno promedio } (m = 10) = 3,928\text{€/h}$$

Con esta última comprobación, se verifica el correcto funcionamiento del programa creado al completo.

6 CONCLUSIONES

Si eres valiente para decir adiós, la vida te recompensará con un nuevo hola.

- Paulo Coelho-

En el presente trabajo se avanza en el conocimiento de la resolución de un problema de optimización del mantenimiento, así como en su implantación y gestión en problemas reales. Se trata de un problema de mantenimiento para un horizonte de tiempo finito de m pasos, donde se buscan los intervalos de mantenimiento preventivo óptimos. Se parte de una resolución del problema desde un enfoque de procesos Semi-Markoviano (SMP) formado por sistema de tres estados, que demuestra que el retorno esperado puede expresarse como la solución de un sistema de ecuaciones en diferencias resuelto a través de la técnica de la transformada z . Para maximizar este retorno promedio acumulado esperado en m pasos, $V(m)$, se ha utilizado el procedimiento directo de derivación con respecto al tiempo hasta el mantenimiento preventivo τ , igualando esta derivada a cero y estudiando las condiciones de existencia de la solución. Este método tomado de referencia [1], inicializaba el sistema desde un estado *Operativo*. Los resultados indican que:

1. La aportación al método creada en este trabajo, siguiendo la misma metodología antes descrita, permite calcular el tiempo hasta el mantenimiento preventivo óptimo, inicializando el sistema desde los estados *Correctivo* y *Preventivo*. Tras el desarrollo de las ecuaciones matemáticas, se obtiene el resultado del tiempo óptimo en el que el sistema debe pasar a un estado de mantenimiento preventivo, iniciando el sistema desde el estado *Correctivo* o desde el estado *Preventivo*, con idéntico valor para ambos estados y superior al que se obtiene desde el estado *Operativo*.
2. Respecto al desarrollo de la herramienta para la implantación y gestión del mantenimiento preventivo, se concluye que gracias al lenguaje de programación Python y la implantación de librerías como Tkinter, se ha podido crear una interfaz gráfica eficiente y amigable para el usuario.
3. El programa desarrollado permite el cálculo de los siguientes parámetros: tiempo óptimo hasta la intervención de mantenimiento preventivo (Tau óptimo, τ), vector de retorno promedio acumulado (V), tiempo de operación promedio (Average Operating Time, AOT), tiempo de operación promedio acumulado (Average Accumulated Operating Time, AAOT). Alternativamente, permite el cálculo de valores concretos de dichos parámetros para un valor de tau dado.
4. La herramienta proporciona una selección de representaciones gráficas de dichos parámetros: Número de pasos de transición frente el tau óptimo, número de pasos de transición frente al vector de retorno promedio acumulado, tau frente al vector de retorno promedio acumulado, número de pasos de transición frente al tiempo de operación promedio acumulado y número de pasos de transición frente al retorno promedio. De forma individualizada para un único caso o de forma combinada para estudios comparativos.
5. Esta interfaz permite de manera sencilla el cálculo y la comparativa de los diferentes escenarios que puede presentar el sistema, simplemente cambiando cualquier parámetro en la distribución de Weibull o los valores de los retornos en cada estado.
6. Se ha demostrado el correcto funcionamiento del programa mediante la comparativa de los resultados obtenidos con el mismo los desarrollados en el método de referencia.
7. El ahorro de tiempo que supone la rapidez en la obtención de resultados es una ventaja adicional de la herramienta desarrollada. De esta forma, se puede favorecer la implantación de políticas de mantenimiento, en particular la planificación y gestión del mantenimiento preventivo, partiendo de cualquiera de los tres estados posibles del sistema, *Operativo*, *Correctivo* o *Preventivo*.
8. Gracias al programa se facilita la implantación del mantenimiento preventivo, y a través de un método que maximiza el retorno, se reduce el impacto económico que implicaría la implantación de dicho mantenimiento. Estas acciones de mantenimiento aumentan la vida útil de un sistema, además de reducir los fallos graves del mismo, lo que implica la reducción de posibles impactos tanto ambientales como a las personas.

7 TRABAJO FUTURO

El pasado es experiencia que el presente aprovecha y el futuro perfecciona.

- A. B. Angélica Araujo-

Puesto que el programa creado PM-3E, está preparado para un sistema que puede encontrarse en tres estados diferentes, sería interesante realizar una ampliación del programa donde se pueda elegir el número de estados en los que se puede encontrar el sistema.

Por último, el programa se ha creado a partir de un caso práctico donde se estudió que la función que representa la distribución del tiempo de fallo del sistema sigue una función Weibull de tres parámetros. Sin embargo, no todos los sistemas siguen esta distribución. Por ello, complementar el programa con una sección en la cual se pudieran cargar los datos históricos sobre los instantes de fallo del sistema y que ésta te indicara a qué distribución se asemeja, aumentaría la aplicabilidad del programa a otros sistemas.

ANEXO A: CODIFICACIÓN EN PYTHON DEL PROGRAMA PM-3E

Tienes que entender su oscuridad para apreciar su luz.

- David Sant-

La herramienta objeto del presente proyecto, se ha desarrollado a través del lenguaje de programación Python. A continuación, se revela el código realizado para el desarrollo de cada una de las secciones del programa. Adicionalmente se explican, a través de breves descripciones, los operadores, estructuras, librerías y demás componentes que el programa contiene.

A.1 Importación de Librerías

Una librería se define como un conjunto de implementaciones funcionales. Python viene con una biblioteca de módulos estándar. La biblioteca estándar ofrece una gran variedad de módulos que realizan distintos tipos de funciones.

Fuera de la biblioteca estándar, existen librerías para visualización, cálculo numérico, análisis de datos, aprendizaje automático y Deep Learning. En resumen, estas librerías son fuentes de recursos con varias funcionalidades. Para poder desarrollar distintos componentes dentro del programa ha sido necesaria la implantación de librerías externas como: *Matplotlib* para generar gráficos, *Numpy* que proporciona una estructura de datos universal que posibilita el análisis y el intercambio de datos entre distintos algoritmos, *Sympy* para matemáticas simbólicas o *Tkinter* para la creación de una interfaz gráfica de usuario o GUI [39].

```
import tkinter as tk
from tkinter import *
from tkinter import END
from tkinter import StringVar
from tkinter import DoubleVar
from tkinter import IntVar
from tkinter import ttk
from tkinter import filedialog as fd
import matplotlib.pyplot as plt
import numpy as np
from matplotlib.backends.backend_tkagg import FigureCanvasTkAgg
```

A.2 Creación de la Ventana Principal

El elemento fundamental de una GUI de Tkinter es la ventana. Las ventanas son los contenedores en los que se encuentran todos los demás elementos de la interfaz gráfica.

```
class Aplicacion():
    def __init__(self):
```

```

self.raiz = tk.Tk()
self.raiz.title("PM-3E")
self.raiz.geometry("1100x700")
self.raiz.configure(background='light cyan')

```

A.3 Declaración de las Variables Principales

Una variable es un espacio reservado donde se guarda una determinada información. Hace referencia a un objeto que reside en la memoria del programa. En función del tipo de información que se guarde (texto, números, booleanas, etc.), la variable será de un tipo u otro. Cada variable debe tener un nombre único llamado identificador.

```

self.α= DoubleVar(value=3.5)
self.β= DoubleVar(value=2500)
self.γ= DoubleVar(value=0)
self.μ1= DoubleVar(value=120)
self.σ1= DoubleVar(value=2)
self.μ2= DoubleVar(value=48)
self.σ2= DoubleVar(value=1)
self.R1= DoubleVar(value=6)
self.R12= DoubleVar(value=-500)
self.R13= DoubleVar(value=-1)
self.R2= DoubleVar(value=-20)
self.R21= DoubleVar(value=-4100)
self.R3= DoubleVar(value=-20)
self.R31= DoubleVar(value=-290)
self.paso=IntVar(value=1)
self.valor_optimo= DoubleVar()

```

A.4 Declaración de Widgets

Los widgets son pequeñas aplicaciones cuyo objetivo es dotar de información visual y facilitar el acceso a las funciones que se desarrollan en el programa. De este modo, cualquiera de los componentes básicos que forman parte de una aplicación en una interfaz gráfica de usuario, se conoce como widget [40].

A.4.1 Datos iniciales: títulos

Widget Label: Una etiqueta es un widget que muestra texto o imágenes, generalmente que el usuario solo verá, no podrá interactuar con ella. Las etiquetas se utilizan para identificar controles u otras partes de la interfaz de usuario, proporcionar comentarios o resultados textuales, etc. [41]

```

self.marco = ttk.Frame(self.raiz, borderwidth=2,
                        relief="raised", padding=(10,10))
fuente = font=('Arial', 11)
color = bg=('lightcyan')

```

```

self.titulo=tk.Label(self.raiz, text="Datos Iniciales",
                    bg='lemon chiffon', font=fuente)
self.fallos=tk.Label(self.raiz, text="F.Distribución
                    Fallos",bg=color, font=fuente)
self.reparacion=tk.Label(self.raiz, text="F.D.Tiempos
                    Reparación", font=fuente, bg=color)
self.preventivo=tk.Label(self.raiz, text="F.D.Tiempos
                    Preventivo", font=fuente,bg=color)
self.ret_ope=tk.Label(self.raiz, text="Retornos Operación",
                    font=fuente,bg=color)
self.ret_cor=tk.Label(self.raiz, text="Retorno Correctivo",
                    font=fuente,bg=color)
self.ret_pre=tk.Label(self.raiz, text="Retorno Preventivo",
                    font=fuente,bg=color)

```

A.4.2 Datos iniciales: etiquetas y casillas de escritura

Un widget de *Entry* crea una casilla en la cual el usuario podrá escribir y el programa tomar la información que éste contenga [42].

```

#-----Botones para introducir datos con sus etiquetas-----#
self.etq1= tk.Label(self.raiz, text="α:", font=fuente,
                    bg=color)
self.landa= ttk.Entry(self.raiz, textvariable=self.α,
                    width=10)

self.etq2= tk.Label(self.raiz, text="β:", font=fuente,
                    bg=color)
self.beta= ttk.Entry(self.raiz, textvariable=self.β,
                    width=10)

self.etq3= tk.Label(self.raiz, text="γ:", font=fuente,
                    bg=color)
self.gamma= ttk.Entry(self.raiz, textvariable=self.γ,
                    width=10)

self.etq4= tk.Label(self.raiz, text="μ1:", font=fuente,
                    bg=color)
self.mil= ttk.Entry(self.raiz, textvariable=self.μ1,
                    width=10)

self.etq5= tk.Label(self.raiz, text="σ1:", font=fuente,
                    bg=color)
self.sigma1= ttk.Entry(self.raiz, textvariable=self.σ1,
                    width=10)

self.etq6= tk.Label(self.raiz, text="μ2:", font=fuente,
                    bg=color)
self.mi2= ttk.Entry(self.raiz, textvariable=self.μ2,
                    width=10)

self.etq7= tk.Label(self.raiz, text="σ2:", font=fuente,
                    bg=color)
self.sigma2= ttk.Entry(self.raiz, textvariable=self.σ2,
                    width=10)

```



```

        indicatoron=False)

self.radiobutton_V3 = tk.Radiobutton(self.raiz,
        text="V3",
        variable=self.sel_v, value='V3',
        indicatoron=False)

```

A.4.4 Cálculo de tau óptimo

Un *Button* es un widget que puede estar encendido o apagado. Cuando un usuario hace clic en él, el botón emite un evento/función.

El *Listbox* es un cuadro de texto, que presenta el contenido en forma de lista, y permite al usuario elegir entre un conjunto de opciones [42].

```

self.etq16= tk.Label(self.raiz, text='Tau (h) ' ,font="Arial
        10", bg='lemon chiffon')

self.resultado= ttk.Label(self.raiz,
        textvariable=self.valor_optimo, foreground="yellow",
        background="black",borderwidth=5, anchor="e",
        width=30)

self.calcular = tk.Button(self.raiz, text="Calcular",
        command=self.calcular)

self.limpiar =tk.Button(self.raiz, text="Eliminar",
        command=self.limpiar)

self.listbox = tk.Listbox(self.raiz, width=30, height=6)

self.listbox2= tk.Listbox()

self.salir = tk.Button(self.raiz, text="Salir",command=quit)

self.guardar = tk.Button(self.raiz,
        text="Guardar",command=self.guardar)

```

A.4.5 Cálculo de V

El widget *Combobox* crea una lista desplegable. Cuando se hace clic en la flecha de la izquierda, aparece un menú desplegable que expone todas las opciones disponibles. El usuario puede seleccionar cualquiera de ellas.

```

self.v= tk.Label(self.raiz, text="V(€):",font="Arial 10",
        bg='lemon chiffon')

self.desplegable= ttk.Combobox(self.raiz, state="readonly")

self.desplegable["values"] = ["PAR", "IMPAR", "TODO"]

self.calcular_v=tk.Button(self.raiz, text="Calcular",
        command=self.calcular_v)

self.resultado_v= tk.Listbox(self.raiz, width=30, height=6)

self.eliminar_v=tk.Button(self.raiz, text="Eliminar",
        command=self.limpiar_v)

self.t_par=tk.Listbox()

self.t_impar=tk.Listbox()

self.datos_v=tk.Listbox()

```

A.4.6 Cálculo de OT

```

self.OT=tk.Label(self.raiz, text="OT (h):",font="Arial 10",
                 bg='lemon chiffon')

self.desplegable_ot= ttk.Combobox(self.raiz,
                                  state="readonly")

self.desplegable_ot["values"] = ["AOT", "AAOT", "V/AAOT"]

self.resultado_ot=tk.Listbox(self.raiz, width=30, height=6)

self.calcular_ot=tk.Button(self.raiz, text="Calcular",
                            command=self.calcular_ot)

self.eliminar_ot=tk.Button(self.raiz, text="Eliminar",
                            command=self.limpiar_ot)

self.datos_A=tk.Listbox()

self.datos_AAOT=tk.Listbox()

self.datos_AR=tk.Listbox()

```

A.4.7 Sección plantilla gráfico

```

right_frame = tk.Frame(self.raiz, bg='#C0C0C0', bd=1.5)
right_frame.place(x=500, y=20, width=550, height=350 )
self.figure = plt.Figure(figsize=(5,6), dpi=50)
self.plot = self.figure.add_subplot(111)

self.plot.grid(True),self.plot.set_xlabel('$x$'),self.plot.set_ylabel('$y(x)$')
self.line = FigureCanvasTkAgg(self.figure, right_frame)

self.line.get_tk_widget().pack(side=tk.LEFT,fill=tk.BOTH,expand=1)

```

A.4.8 Sección gráfico

```

self.crear_grafico = tk.Button(self.raiz,
                               text="Representar",command=self.crear_grafico)

self.titulografico=tk.Label(self.raiz,
                             text="Gráfico",bg='lemon chiffon', font=fuente)

self.varx = StringVar(value='m')

self.var_x=tk.Label(self.raiz, text="Eje X:",bg=color,
                   font=fuente)

self.tau_x= tk.Radiobutton(self.raiz, text='Tau ( $\tau$ )',
                           variable=self.varx, value='tx',bg=color,font=fuente)

self.m= tk.Radiobutton(self.raiz, text='N° de pasos
(m)',variable=self.varx, value='m',bg=color,font=fuente)

self.vary=StringVar(value='ty')

self.var_y=tk.Label(self.raiz, text="Eje Y:",bg=color,
                   font=fuente)

self.tau_y= tk.Radiobutton(self.raiz, text='Tau óptimo ( $\tau_0$ )',
                           variable=self.vary, value='ty',bg=color,font=fuente)

self.v_y= tk.Radiobutton(self.raiz, text='V',
                          variable=self.vary, value='v',bg=color,font=fuente)

self.aaot_y=tk.Radiobutton(self.raiz, text='AAOT',

```

```

        variable=self.vary, value='AAOT',bg=color,font=fuente)

self.v1_aaot_y=tk.Radiobutton(self.raiz, text='V/AAOT',
    variable=self.vary, value='V/AAOT',bg=color,font=fuente)

self.eliminar_grafico=tk.Button(self.raiz,
    text="Eliminar",command=self.eliminar_graf)

self.guardar_grafico=tk.Button(self.raiz,
    text="Guardar",command=self.guardar_graf)

self.datos_v_comp=tk.Listbox()

```

A.4.9 Cálculo para tau propio

```

self.titu_tau_pro=tk.Label(self.raiz, text="Calculadora para Tau
    propio",bg='lemon chiffon', font=fuente)
self.etq_tau= tk.Label(self.raiz, text="Tau:",font=fuente, bg=color)
self.pro_tau=DoubleVar()
self.propio_tau=ttk.Entry(self.raiz, textvariable=self.pro_tau,
    width=10)
self.etq_paso= tk.Label(self.raiz, text="Paso:",font=fuente,
    bg=color)
self.pro_paso=IntVar(value=1)
self.propio_paso=ttk.Entry(self.raiz, textvariable=self.pro_paso,
    width=5)
self.desplegable_tau_pro= ttk.Combobox(self.raiz,
    state="readonly",width=10)
self.desplegable_tau_pro["values"] = ["V","AOT", "AAOT", "V/AAOT"]
self.resultado_tau_prop=DoubleVar(0.0)
self.resultado_tau_pro=tk.Label(self.raiz,
    textvariable=self.resultado_tau_prop,width=15,foreground="yel
    low", background="black", borderwidth=5, anchor="e")
self.calcular_tau_pro=tk.Button(self.raiz, text="Calcular",
    command=self.calcular_prop_tau)
self.sel_v_tau = StringVar()
self.sel_v_tau.set('V1')
self.radiobutton_V1_tau = tk.Radiobutton(self.raiz,
    text="V1",
    variable=self.sel_v_tau, value='V1',
    indicatoron=False)
self.radiobutton_V2_tau = tk.Radiobutton(self.raiz,
    text="V2",
    variable=self.sel_v_tau, value='V2',
    indicatoron=False)
self.radiobutton_V3_tau = tk.Radiobutton(self.raiz,
    text="V3",

```

```

        variable=self.sel_v_tau, value='V3',
        indicatoron=False)
self.eliminar_tau_pro=tk.Button(self.raiz, text="Eliminar",
        command=self.eliminar_prop_tau)

```

A.4.10 Creación botones ayuda

```

color_2='dark turquoise'
self.ayuda_datos=tk.Button(self.raiz,
        text="*",height=1,background=color_2,width=2,
        command=self.ayuda_datos)
self.ayuda_OP=tk.Button(self.raiz,
        text="*",height=1,background=color_2,width=2,
        command=self.ayuda_OP)
self.ayuda_V1=tk.Button(self.raiz,
        text="*",height=1,background=color_2,width=2,
        command=self.ayuda_V1)
self.ayuda_OT=tk.Button(self.raiz,
        text="*",height=1,background=color_2,width=2,
        command=self.ayuda_OT)
self.ayuda_guard=tk.Button(self.raiz,
        text="*",height=1,background=color_2,width=2,
        command=self.ayuda_guard)
self.ayuda_graf_ejes=tk.Button(self.raiz,
        text="*",height=1,background=color_2,width=2,
        command=self.ayuda_graf_ejes)
self.ayuda_graf=tk.Button(self.raiz,
        text="*",height=1,background=color_2,width=2,
        command=self.ayuda_graf)
self.ayuda_retornos=tk.Button(self.raiz,
        text="*",height=1,background=color_2,width=2,
        command=self.ayuda_retornos)
self.ayuda_tau_propio=tk.Button(self.raiz,
        text="*",height=1,background=color_2,width=2,
        command=self.ayuda_tau_propio)

```

A.5 Posicionamiento de Widgets

La librería Tkinter provee tres métodos para establecer la posición de los *widgets* dentro de una ventana. Éstas se corresponden con las funciones *pack()*, *place()* y *grid()*. Para este caso se ha utilizado la función *place()*. [43]

La función *place()* permite ubicar elementos indicando su posición (X e Y) respecto de un elemento padre, siendo la coordenada (0,0) la esquina superior izquierda de la ventana principal. Dentro de esta función también se puede indicar el tamaño que ocupa el widget en píxeles con los parámetros *width* y *height*.

A.5.1 Datos iniciales

```

self.titulo.place(x=10, y=10)

self.fallos.place(x=10, y=40)
self.etq1.place(x=10, y=60)
self.landa.place(x=30, y=60)

```

```
self.etq2.place(x=10, y=85)
self.beta.place(x=30, y=85)

self.etq3.place(x=10, y=110)
self.gamma.place(x=30, y=110)

self.etq4.place(x=150, y=60)
self.mi1.place(x=180, y=60)

self.reparacion.place(x=150, y=40)
self.etq5.place(x=150, y=85)
self.sigma1.place(x=180, y=85)

self.preventivo.place(x=320, y=40)
self.etq6.place(x=320, y=60)
self.mi2.place(x=350, y=60)

self.etq7.place(x=320, y=85)
self.sigma2.place(x=350, y=85)

self.ret_ope.place(x=10, y=140)
self.etq8.place(x=10, y=160)
self.retorno1.place(x=45, y=160)

self.etq9.place(x=10, y=185)
self.retorno12.place(x=45, y=185)

self.etq10.place(x=10, y=210)
self.retorno13.place(x=45, y=210)

self.ret_cor.place(x=150, y=140)
self.etq11.place(x=150, y=160)
self.retorno2.place(x=185, y=160)

self.etq12.place(x=150, y=185)
self.retorno21.place(x=185, y=185)

self.ret_pre.place(x=320, y=140)
self.etq13.place(x=320, y=160)
self.retorno3.place(x=350, y=160)

self.etq14.place(x=320, y=185)
self.retorno31.place(x=350, y=185)
```

A.5.2 Botones V1, V2 y V3

```
self.radiobutton_V1.place(x=290, y=248)
self.radiobutton_V2.place(x=320, y=248)
self.radiobutton_V3.place(x=350, y=248)
```

A.5.3 Cálculo de tau óptimo

```
self.guardar.place(x=380, y=248)
self.etq15.place(x=10, y=250)
self.numeropasos.place(x=220, y=250)
```

```
self.calcular.place(x=320, y=280)
self.etq16.place(x=10, y=280)
self.resultado.place(x=130, y=280)
self.limpiar.place(x=320, y=320)
self.listbox.place(x=130, y=300)
```

A.5.4 Cálculo de V

```
self.v.place(x=10, y=410)
self.desplegable.place(x=130, y=410)
self.calcular_v.place(x=280, y=407)
self.resultado_v.place(x=130, y=438)
self.eliminar_v.place(x=320, y=485)
self.propio_tau.place(x=290, y=410)
```

A.5.5 Cálculo de OT

```
self.OT.place(x=10, y=550)
self.desplegable_ot.place(x=130, y=550)
self.resultado_ot.place(x=130, y=578)
self.calcular_ot.place(x=280, y=547)
self.eliminar_ot.place(x=320, y=628)
```

A.5.6 Representación gráfica

```
self.titulografico.place(x=500, y=400)
self.var_x.place(x=500, y=430)
self.tau_x.place(x=500, y=460)
self.m.place(x=500, y=480)
self.var_y.place(x=700, y=430)
self.tau_y.place(x=700, y=460)
self.v_y.place(x=700, y=480)
self.aoot_y.place(x=700, y=500)
self.v1_aoot_y.place(x=700, y=520)
self.crear_grafico.place(x=800, y=520)
self.eliminar_grafico.place(x=880, y=520)
self.guardar_grafico.place(x=940, y=520)
```

A.5.7 Cálculo para tau propio

```

self.titu_tau_pro.place(x=500, y=570)
self.radiobutton_V1_tau.place(x=695, y=570)
self.radiobutton_V2_tau.place(x=725, y=570)
self.radiobutton_V3_tau.place(x=755, y=570)
self.etq_tau.place(x=500, y=600)
self.propio_tau.place(x=540, y=603)
self.etq_paso.place(x=610, y=600)
self.propio_paso.place(x=655, y=603)
self.desplegable_tau_pro.place(x=700, y=603)
self.resultado_tau_pro.place(x=790, y=600)
self.calcular_tau_pro.place(x=910, y=600)
self.eliminar_tau_pro.place(x=970, y=600)

```

A.5.8 Botones ayuda

```

self.ayuda_datos.place(x=120, y=10)
self.ayuda_retornos.place(x=430, y=173)
self.ayuda_OP.place(x=60, y=280)
self.ayuda_V1.place(x=60, y=410)
self.ayuda_OT.place(x=60, y=550)
self.ayuda_guard.place(x=440, y=248)
self.ayuda_graf_ejes.place(x=560, y=400)
self.ayuda_graf.place(x=1000, y=520)
self.ayuda_tau_propio.place(x=790, y=570)
#final widgets
self.raiz.mainloop()

```

A.6 Definición Botones Ayuda

A.6.1 Retornos

```

def ayuda_retornos(self):
    color='lemon chiffon'
    letra=('times', 14, 'italic')
    master = tk.Tk()
    master.geometry("550x400")
    master.configure(background=color)
    whatever_you_do = "RETORNOS"\
        "\t\n-R1, ingreso por unidad de tiempo que el sistema\
        permanece en el estado 1 (E1: Operativo)"\

```

```

"\t\n -R2, costo por unidad de tiempo que el sistema
permanece en el estado 2 (E2: Correctivo)"\

"\t\n -R3, costo por unidad de tiempo que el sistema
permanece en el estado 3 (E3: Preventivo)."\

"\t\n -R12, costo de transición del estado operativo al
correctivo."

"\t\n -R21, costo de transición del estado correctivo al
operativo."

"\t\n -R13, costo de transición del estado operativo al
preventivo."

"\t\n -R31, costo de transición del estado preventivo al
operativo.\t\n"

"\t\n Los costos se escriben con signo negativo"

msg = tk.Message(master, text = whatever_you_do)
msg.config(bg=color, font=letra, width="450", padx="1")
msg.pack()
tk.mainloop()

```

A.6.2 Datos iniciales

```

def ayuda_datos(self):
    color='lemon chiffon'
    letra=('times', 14, 'italic')
    master = tk.Tk()
    master.configure(background=color)
    whatever_you_do ="Introduzca los datos de su problema"\

    "\t\n Programa diseñado para un sistema que pasa por tres
    estados diferentes."

    "\n •E1: Estado Operativo. Hace referencia al momento en
    el que el sistema se encuentra funcionando correctamente."

    "\n •E2: Estado Correctivo. Hace referencia a cuando el
    sistema está siendo reparado debido a un fallo en el
    mismo."

    "\n •E3: Estado Preventivo. Se refiere al momento, el cual
    el sistema se encuentra en un estado en el que se están
    llevando a cabo acciones de mantenimiento
    preventivas.\t\n"

    "\t\n El tiempo que el sistema permanece en cada estado se
    expresa con funciones de distribución."

    "\n •E1: Función de distribución de Fallos: Función Weibull
    de tres parámetros ( $\alpha$ ,  $\beta$ ,  $\gamma$ )"

    "\n •E2: Función de distribución Tiempos de Reparación:
    Normal ( $\mu_1$ ,  $\sigma_1$ )"

    "\n •E3: Función de distribución Tiempos Preventivo: Normal
    ( $\mu_2$ ,  $\sigma_2$ )\t\n"

    "\t\n La permanencia del sistema en cada estado, lleva
    asociado unos ingresos y costos, llamados retornos."

```



```

"\t\n El objetivo de este programa es el cálculo del tiempo
óptimo (Tau óptimo) en el que el sistema debe pasar al
mantenimiento preventivo de manera que se maximicen los
retornos acumulados."\

"\t\n Para introducir los datos solo debe clicar en la
casilla que desee, borrar su contenido y escribir el valor
deseado."\

"\t\n Utilizar punto no coma para marca la parte decimal."
msg = tk.Message(master, text = whatever_you_do)
msg.config(bg=color, font=letra,width="500")
msg.pack()
tk.mainloop()

```

A.6.3 Cálculo de tau óptimo

```

def ayuda_OP(self):
    color='lemon chiffon'
    letra=('times', 14, 'italic')
    master = tk.Tk()
    whatever_you_do = "Cálculo de Tau Óptimo"\

    "\t\n 1.Introducir el número de pasos para el cual desea
    calcular el tau óptimo."\

    "\t\n 2.Pulsar V1 si el sistema parte del estado
    operativo. V2 si el sistema parte del estado correctivo.
    V3 Si es sistema comenzó en estado preventivo."\

    "\t\n 3. Pulsar el botón de Calcular que se encuentra a
    continuación."\

    "\t\n 4. El resultado aparecerá en el recuadro
    correspondiente a esta sección."\

    "\t\n  $\tau$  (paso de transición)= valor de tau óptimo para ese
    paso en horas."\

    "\t\n 5. Si se desea eliminar el resultado para calcular
    uno nuevo, pulse Eliminar."

    msg = tk.Message(master, text = whatever_you_do)
    msg.config(bg=color, font=letra)
    msg.pack()
    tk.mainloop()

```

A.6.4 Cálculo de V

```

def ayuda_V1(self):
    color='lemon chiffon'
    letra=('times', 16, 'italic')

```

```

master = tk.Tk()

whatever_you_do = "Cálculo vector de retorno promedio
acumulado. "\

"\t\n 1. Seleccionar una de las opciones del
desplegable.\t\n"\

"\t\n Opciones:"\

"\t\n -PAR: Resultado únicamente para los pasos de
transición pares de 1 al introducido."\

"\t\n -IMPAR: Resultado únicamente para los pasos de
transición impares de 1 al introducido."\

"\t\n -TODO: Resultado de V para cada uno de los valores
de m, desde 1 hasta el número de pasos."\

"\t\n 2. Pulsar el botón de Calcular que se encuentra a
continuación.\t\n"\

"\t\n 3. El resultado aparecerá en el recuadro
correspondiente a esta sección."\

" V(paso de transición) = valor de V para ese paso en
euros.\t\n"\

"\t\n 4. Si se desea eliminar el resultado para calcular
uno nuevo, pulse Eliminar."

msg = tk.Message(master, text = whatever_you_do)
msg.config(bg=color, font=letra)
msg.pack()
tk.mainloop()

```

A.6.5 Cálculo de OT

```

def ayuda_OT(self):
    color='lemon chiffon'
    letra=('times', 16, 'italic')
    master = tk.Tk()
    whatever_you_do = "Cálculo del tiempo de operación"\

    "\t\n 1. Seleccionar una de las opciones del
desplegable.\t\n"\

    "\t\n Opciones:"\

    "\t\n -AOT: Cálculo del tiempo de operación promedio. "\
    "\t\n AOT (paso de transición)= valor de AOT para ese
paso en horas.\t\n"\

    "\t\n -AAOT: Cálculo del tiempo de operación promedio
acumulado. "\

    "\t\n AAOT (paso de transición)= valor de AAOT para ese
paso en horas.\t\n"\

    "\t\n -V/AAOT: Cálculo del retorno promedio. Para poder
seleccionar esta opción es necesario haber calculado V
para TODO"\

    "\t\n AR (paso de transición)= valor de AR para ese paso
en euros/horas.\t\n"\

    "\t\n 2. Pulsar el botón de Calcular que se encuentra a

```

```

        continuación.\t\n"
        "\t\n 3. El resultado aparecerá en el recuadro
        correspondiente a esta sección."
        "\t\n 4. Si se desea eliminar el resultado para calcular
        uno nuevo, pulse Eliminar."
    msg = tk.Message(master, text = whatever_you_do)
    msg.config(bg=color, font=letra)
    msg.pack()
    tk.mainloop()

```

A.6.6 Guardar

```

def ayuda_guard(self):
    color='lemon chiffon'
    letra=('times', 16, 'italic')
    master = tk.Tk()
    whatever_you_do = "El botón guardar permite generar un archivo
    de texto con los resultados obtenidos en las diferentes
    secciones.\t\n"
    "\t\n 1. Pulsar el botón guardar que se encuentra a
    continuación.\t\n"
    "\t\n 2. Aparecerá una nueva pantalla donde se podrá
    seleccionar el lugar donde guardar el archivo nuevo y el
    nombre del mismo.\t\n"
    "\t\n 3. Al pulsar guardar en la segunda pantalla, se
    creará el archivo de texto con los resultados obtenidos."
    msg = tk.Message(master, text = whatever_you_do)
    msg.config(bg=color, font=letra)
    msg.pack()
    tk.mainloop()

```

A.6.7 Opciones representación gráfica

```

def ayuda_graf_ejes(self):
    color='lemon chiffon'
    letra=('times', 16, 'italic')
    master = tk.Tk()
    whatever_you_do = "Se podrán representar únicamente los datos
    calculados y las siguientes combinaciones:"
    "\t\n Eje X: Tau; Eje Y: V \t\n"
    "\t\n Eje X: N.º de pasos (m); Eje Y: Tau óptimo\n"
    "\n Eje X: N.º de pasos (m); Eje Y: V\t\n"

```

```

        "\t\n Eje X: N.º de pasos (m); Eje Y: AAOT\t\n"
        "\t\n Eje X: N.º de pasos (m); Eje Y: V/AAOT\t\n"
msg = tk.Message(master, text = whatever_you_do)
msg.config(bg=color, font=letra)
msg.pack()
tk.mainloop()

```

A.6.8 Representación gráfica

```

def ayuda_graf(self):
    color='lemon chiffon'
    letra=('times', 16, 'italic')
    master = tk.Tk()
    whatever_you_do = "Representar: Una vez elegida la combinación
        deseada se deberá pulsar Representar.\t\n"
        "\t\n Eliminar: Antes de representar una nueva
        combinación se deberá pulsar este botón. Al pulsarlo
        visualmente no cambiará nada, sin embargo, al
        representar de nuevo habrá desaparecido el gráfico
        anterior."
        "\t\n Si se desea representar la misma función pero con
        distintos datos, y que se vean conjuntas en el mismo
        gráfico no es necesario pulsar eliminar.\t\n"
        "\t\n Guardar: Si se desea guardar una representación
        creada, se deberá pulsar este botón. Se abrirá una nueva
        pestaña que permitirá elegir el lugar donde guardarlo y
        el nombre.\t\n"
    msg = tk.Message(master, text = whatever_you_do)
    msg.config(bg=color, font=letra)
    msg.pack()
    tk.mainloop()

```

A.6.9 Cálculo para tau propio

```

def ayuda_tau_propio(self):
    color='lemon chiffon'
    letra=('times', 16, 'italic')
    master = tk.Tk()
    whatever_you_do = "Calculadora para Tau propio"
        "\t\n 1. Introducir el valor de tau.\t"
        "\t\n 2. Introducir el paso que le interese.\t"
        "\t\n 3. Marcar V1 si el sistema comenzó en estado
        operativo / V2 si el sistema comenzó en estado
        correctivo/ V3 Si es sistema comenzó en estado
        preventivo"

```

```

"\t\n 4. Seleccionar una de las opciones de cálculo\t"\
"\t\n Opciones:"\
"\t\n -V: Cálculo del vector de retorno promedio
acumulado para el paso introducido."\<
"\t\n -AOT: Cálculo del tiempo de operación promedio."\<
"\t\n -AAOT: Cálculo del tiempo de operación promedio
acumulado."\<
"\n -V/AAOT: Cálculo del retorno promedio"\
"\t\n 4. Pulsar el botón Calcular que se encuentra a
continuación."\<
"\t\n 5. El resultado para la opción elegida aparecerá
en el recuadro negro."\<
"\t\n 6. Si desea calcular un nuevo elemento pulse
previamente Eliminar."

msg = tk.Message(master, text = whatever_you_do)
msg.config(bg=color, font=letra)
msg.pack()
tk.mainloop()

```

A.7 Definición Botón Eliminar

Función *delete ()*: Esta función permite borrar el contenido del widget. Es decir, si se ha creado un *listbox* con resultados, esta función permite borrar el contenido de la lista para poder incorporar los nuevos.

A.7.1 Resultado de tau óptimo

```

def limpiar(self):
    self.listbox.delete(0, tk.END)
    self.listbox2.delete(0, tk.END)
    self.t_0=0
    self.A=0
    self.F=0

```

A.7.2 Resultado de V

```

def limpiar_v(self):
    self.resultado_v.delete(0, tk.END)
    self.datos_v.delete(0, tk.END)
    self.t_impar.delete(0, tk.END)
    self.t_par.delete(0, tk.END)
    self.y2=0
    self.datos_A.delete(0, tk.END)

```

A.7.3 Resultado de OT

```
def limpiar_ot(self):
    self.resultado_ot.delete(0, tk.END)
    self.datos_AAOT.delete(0, tk.END)
    self.datos_AR.delete(0, tk.END)
```

A.7.4 Gráfico

```
def eliminar_graf(self):
    self.plot.clear()
```

A.7.5 Resultado para tau propio

```
def eliminar_prop_tau(self):
    self.resultado_tau_prop.set(0.0)
```

A.8 Función: Cálculo de Tau Óptimo

Una estructura de control, es un bloque de código que permite agrupar instrucciones de forma controlada. Se debe prestar atención a las líneas de código que están dentro de una misma estructura, ya que estas suelen causar errores, debido a que es necesario agruparlas teniendo el mismo número de espacios a la izquierda de cada línea.

Las estructuras de control condicionales, son aquellas que permiten evaluar si una o más condiciones se cumplen, para decir qué acción se va a ejecutar. La evaluación de condiciones, solo puede arrojar 1 de 2 resultados: verdadero o falso (True o False). La pieza de código siguiente se ejecutará siempre y cuando el resultado sea verdadero. Las estructuras de control de flujo condicionales, se definen mediante el uso de tres palabras claves reservadas, del lenguaje: *if* (si), *elif* (sino, si) y *else* (sino). [44]

```
def calcular(self):
    import sympy as sp
    error_dato = False
    valor_optimo = 0
    m=1
    try:
         $\alpha$  = float(self. $\alpha$ .get())
         $\beta$  = float(self. $\beta$ .get())
         $\gamma$  = float(self. $\gamma$ .get())
         $\mu_1$  = float(self. $\mu_1$ .get())
         $\sigma_1$  = float(self. $\sigma_1$ .get())
         $\mu_2$  = float(self. $\mu_2$ .get())
         $\sigma_2$  = float(self. $\sigma_2$ .get())
        R1 = float(self.R1.get())
        R12 = float(self.R12.get())
```

```

R13 = float(self.R13.get())
R2 = float(self.R2.get())
R21 = float(self.R21.get())
R3 = float(self.R3.get())
R31 = float(self.R31.get())
pasos = float(self.paso.get())

except:
    error_dato = True
if not error_dato:
    if self.sel_v.get()=="V1":
        tau = ((beta**alpha)/alpha*(-R1)*(1+2*pasos+(-1)**(pasos-1)) / ((1+2*pasos+(-1)**(pasos-1))*(R12-R13)+(-1+2*pasos-(-1)**(pasos-1))*(R2*mu1+R21-R3*mu2-R31))** (1/(alpha-1)))-gamma
        F=1-sp.exp(-(((tau-gamma)/beta)**alpha))
        x = sp.Symbol('x')
        y= alpha/(beta*F)*x*(((x-gamma)/beta)**(alpha-1))*sp.exp(-(((x-gamma)/beta)**alpha))
        A=sp.integrate(y,(x,gamma+0.001,tau))
        self.tau_0=tau
        self.A=A
        self.F=F
        while m <= pasos:
            valor_optimo=((beta**alpha)/alpha*(-R1)*(1+2*m+(-1)**(m-1)) / ((1+2*m+(-1)**(m-1))*(R12-R13)+(-1+2*m-(-1)**(m-1))*(R2*mu1+R21-R3*mu2-R31))** (1/(alpha-1)))-gamma
            self.valor_optimo.set(round(valor_optimo,3))
            self.listbox.insert(tk.END,
                                f"tau({m})={round(valor_optimo,3)}")
            self.listbox2.insert(tk.END,round(valor_optimo,3))
            m += 1
        if self.sel_v.get()=="V2":
            tau = ((beta**alpha)/alpha*(-R1)*(-1+2*pasos-(-1)**(pasos-1)) / ((-1+2*pasos-(-1)**(pasos-1))*(R12-R13)+(-3+2*pasos+(-1)**(pasos-1))*(R2*mu1+R21-R3*mu2-R31))** (1/(alpha-1)))-gamma
            F=1-sp.exp(-(((tau-gamma)/beta)**alpha))
            x = sp.Symbol('x')
            y= alpha/(beta*F)*x*(((x-gamma)/beta)**(alpha-1))*sp.exp(-(((x-gamma)/beta)**alpha))
            A=sp.integrate(y,(x,gamma+0.001,tau))
            self.tau_0=tau
            self.A=A
            self.F=F
            m += 1

```

```

while m <= pasos:
    valor_optimo=((β**α)/α*(-R1)*(-1+2*m-(-1)**(m-1))/((-
        1+2*m-(-1)**(m-1))*(R12-R13)+(-3+2*m+(-1)**(m-
        1))*(R2*μ1+R21-R3*μ2-R31))**(1/(α-1))-γ
    self.valor_optimo.set(round(valor_optimo,3))
    self.listbox.insert(tk.END,
        f"τ({m})={round(valor_optimo,3)}")
    self.listbox2.insert(tk.END,round(valor_optimo,3))
    m += 1
if self.sel_v.get()=="V3":
    τ=((β**α)/α*(-R1)*(-1+2*pasos-(-1)**(pasos-1))/((-
        1+2*pasos-(-1)**(pasos-1))*(R12-R13)+(-
        3+2*pasos+(-1)**(pasos-1))*(R2*μ1+R21-R3*μ2-
        R31))**(1/(α-1))-γ
    F=1-sp.exp(-((τ-γ)/β)**α)
    x = sp.Symbol('x')
    y= α/(β*F)*x*((x-γ)/β)**(α-1)*sp.exp(-((x-γ)/β)**α)
    A=sp.integrate(y,(x,γ+0.001,τ))
    self.τ_0=τ
    self.A=A
    self.F=F
    m += 2
while m <= pasos:
    valor_optimo=((β**α)/α*(-R1)*(-1+2*m-(-1)**(m-1))/((-
        1+2*m-(-1)**(m-1))*(R12-R13)+(-3+2*m+(-1)**(m-
        1))*(R2*μ1+R21-R3*μ2-R31))**(1/(α-1))-γ
    self.valor_optimo.set(round(valor_optimo,3))
    self.listbox.insert(tk.END,
        f"τ({m})={round(valor_optimo,3)}")
    self.listbox2.insert(tk.END,round(valor_optimo,3))
    m += 1

```

A.9 Función: Cálculo de V

A diferencia de las estructuras de control condicionales, las iterativas, también llamadas cíclicas o bucles, permiten ejecutar un mismo código, repetidamente, mientras se cumpla una condición [44].

El bucle *while*, se encarga de ejecutar una misma acción "mientras que" una determinada condición se cumpla.

```

def calcular_v(self):
    import sympy as sp
    error_dato = False
    valor_optimo = 0

```



```

m=1
try:
    α = float(self.α.get())
    β = float(self.β.get())
    γ = float(self.γ.get())
    μ1 = float(self.μ1.get())
    σ1 = float(self.σ1.get())
    μ2 = float(self.μ2.get())
    σ2 = float(self.σ2.get())
    R1 = float(self.R1.get())
    R12 = float(self.R12.get())
    R13 = float(self.R13.get())
    R2 = float(self.R2.get())
    R21 = float(self.R21.get())
    R3 = float(self.R3.get())
    R31 = float(self.R31.get())
    pasos = float(self.paso.get())
except:
    error_dato = True
if not error_dato:
    while m <= pasos:
        if self.desplegable.get()=="IMPAR":
            if self.sel_v.get()=="V1":
                τ=((β**α)/α*(-R1)*(1+2*pasos+(-1)**(pasos-1)))/((1+2*pasos+(-1)**(pasos-1))*(R12-R13)+(-1+2*pasos-(-1)**(pasos-1))*(R2*μ1+R21-R3*μ2-R31))**1/(α-1)-γ
                F=1-sp.exp(-((τ-γ)/β)**α)
                x = sp.Symbol('x')
                y=α/(β)*x*((x-γ)/β)**(α-1)*sp.exp(-((x-γ)/β)**α)
                A=sp.integrate(y, (x, γ+0.001, τ))
                v=1/4*((1+2*m+(-1)**(m-1))*(R1*A+R12*F+(R1*τ+R13)*(1-F))+(-1+2*m-(-1)**(m-1))*(R2*μ1+R21)*F+(R3*μ2+R31)*(1-F))
            if self.sel_v.get()=="V2":
                τ=((β**α)/α*(-R1)*(-1+2*pasos-(-1)**(pasos-1)))/((-1+2*pasos-(-1)**(pasos-1))*(R12-R13)+(-3+2*pasos+(-1)**(pasos-1))*(R2*μ1+R21-R3*μ2-R31))**1/(α-1)-γ
                F=1-sp.exp(-((τ-γ)/β)**α)
                x = sp.Symbol('x')
                y=α/(β)*x*((x-γ)/β)**(α-1)*sp.exp(-((x-γ)/β)**α)

```

```

A=sp.integrate(y, (x, γ+0.001, τ))

v=1/4*((-1+2*m-(-1)**(m-1))*
(R1*A+R12*F+(R1*τ+R13)*(1-F))
+(1+2*m+(-1)**(m-1))*(R2*μ1+R21)+(-3+2*m+(-1)**(m-1))*
((R3*μ2+R31)-(R2*μ1+R21)*(1-F)))

if self.sel_v.get()=="V3":
    τ=((β**α)/α*(-R1)*(-1+2*pasos-(-1)**(pasos-1))
/((-1+2*pasos-(-1)**(pasos-1))
*(R12-R13)+(-3+2*pasos+(-1)**(pasos-1))
*(R2*μ1+R21-R3*μ2-R31)))*(1/(α-1))-γ
    F=1-sp.exp(-((τ-γ)/β)**α)
    x = sp.Symbol('x')
    y=α/β*x*((x-γ)/β)**(α-1)*sp.exp(-((x-γ)/β)**α)
    A=sp.integrate(y, (x, γ+0.001, τ))
    v=1/4*((-1+2*m-(-1)**(m-1))*
(R1*A+R12*F+(R1*τ+R13)*(1-F))
+(1+2*m+(-1)**(m-1))*(R3*μ2+R31)+(-3+2*m+(-1)**(m-1))*
((R2*μ1+R21)-(R3*μ2+R31)*(F)))

self.τ_impar.insert(tk.END, τ)
self.A=(1/F)*A
self.resultado_v.insert(tk.END,
f"{self.sel_v.get()}({m})={round(v, 3)}")
self.datos_v.insert(tk.END, round(v, 3))
m += 2
if self.desplegable.get()=="PAR":
    m +=1
    if self.sel_v.get()=="V1":
        τ=((β**α)/α*(-R1)*(1+2*pasos+(-1)**(pasos-1))
/((1+2*pasos+(-1)**(pasos-1))
*(R12-R13)+(-1+2*pasos-(-1)**(pasos-1))
*(R2*μ1+R21-R3*μ2-R31)))*(1/(α-1))-γ
        F=1-sp.exp(-((τ-γ)/β)**α)
        x = sp.Symbol('x')
        y=α/β*x*((x-γ)/β)**(α-1)*sp.exp(-((x-γ)/β)**α)
        A=sp.integrate(y, (x, γ+0.001, τ))
        v=1/4*((1+2*m+(-1)**(m-1))*
(R1*A+R12*F+(R1*τ+R13)*(1-F))
+(-1+2*m-(-1)**(m-1))*
((R2*μ1+R21)*F+(R3*μ2+R31)*(1-F)))
    if self.sel_v.get()=="V2":
        τ=((β**α)/α*(-R1)*(-1+2*pasos-(-1)**(pasos-1))
/((-1+2*pasos-(-1)**(pasos-1))
*(R12-R13)+(-3+2*pasos+(-1)**(pasos-1))
*(R2*μ1+R21-R3*μ2-R31)))*(1/(α-1))-γ
        F=1-sp.exp(-((τ-γ)/β)**α)

```

```

x = sp.Symbol('x')
y=α/(β)*x*((x-γ)/β)**(α-1)*sp.exp(-((x-γ)/β)**α)
A=sp.integrate(y,(x,γ+0.001,τ))
v=1/4*((-1+2*m-(-1)**(m-1))*(R1*A+R12*F+(R1*τ+R13)*(1-F))+(1+2*m+(-1)**(m-1))*(R2*μ1+R21)+(-3+2*m+(-1)**(m-1))*((R3*μ2+R31)-(R2*μ1+R21))*(1-F))

if self.sel_v.get()=="V3":
    τ=((β**α)/α*(-R1)*(-1+2*pasos-(-1)**(pasos-1))/((-1+2*pasos-(-1)**(pasos-1))*(R12-R13)+(-3+2*pasos+(-1)**(pasos-1))*(R2*μ1+R21-R3*μ2-R31)))*(1/(α-1))-γ
    F=1-sp.exp(-((τ-γ)/β)**α)
    x = sp.Symbol('x')
    y=α/(β)*x*((x-γ)/β)**(α-1)*sp.exp(-((x-γ)/β)**α)
    A=sp.integrate(y,(x,γ+0.001,τ))
    v=1/4*((-1+2*m-(-1)**(m-1))*(R1*A+R12*F+(R1*τ+R13)*(1-F))+(1+2*m+(-1)**(m-1))*(R3*μ2+R31)+(-3+2*m+(-1)**(m-1))*((R2*μ1+R21)-(R3*μ2+R31))*(F))

self.τ_par.insert(tk.END,τ)
self.A=(1/F)*A
self.resultado_v.insert(tk.END,
    f"{self.sel_v.get()}({m})={round(v,3)}")
self.datos_v.insert(tk.END,round(v,3))
m +=1

if self.desplegable.get()=="TODO":
    if self.sel_v.get()=="V1":
        τ=((β**α)/α*(-R1)*(1+2*pasos+(-1)**(pasos-1))/((1+2*pasos+(-1)**(pasos-1))*(R12-R13)+(-1+2*pasos-(-1)**(pasos-1))*(R2*μ1+R21-R3*μ2-R31)))*(1/(α-1))-γ
        F=1-sp.exp(-((τ-γ)/β)**α)
        x = sp.Symbol('x')
        y=α/(β)*x*((x-γ)/β)**(α-1)*sp.exp(-((x-γ)/β)**α)
        A=sp.integrate(y,(x,γ+0.001,τ))
        v=1/4*((1+2*m+(-1)**(m-1))*(R1*A+R12*F+(R1*τ+R13)*(1-F)))+(-1+2*m-(-1)**(m-1))*((R2*μ1+R21)*F+(R3*μ2+R31)*(1-F))

    if self.sel_v.get()=="V2":
        τ=((β**α)/α*(-R1)*(-1+2*pasos-(-

```

```

1)**(pasos-1))/((-1+2*pasos-(-1)**(pasos-
1))*(R12-R13)+(-3+2*pasos+(-1)**(pasos-
1))*(R2*μ1+R21-R3*μ2-R31))**(1/(α-1))-γ

F=1-sp.exp(-((τ-γ)/β)**α)

x = sp.Symbol('x')

y=α/(β)*x*((x-γ)/β)**(α-1)*sp.exp(-((x-
γ)/β)**α)

A=sp.integrate(y,(x,γ+0.001,τ))

v=1/4*((-1+2*m-(-1)**(m-
1))*(R1*A+R12*F+(R1*τ+R13)*(1-
F))+(1+2*m+(-1)**(m-1))*(R2*μ1+R21)+(-
3+2*m+(-1)**(m-1))*((R3*μ2+R31)-(R2*
μ1+R21)*(1-F)))

if self.sel_v.get()=="V3":

    τ=(β**α)/α*(-R1)*(-1+2*pasos-(-
1)**(pasos-1))/((-1+2*pasos-(-1)**(pasos-
1))*(R12-R13)+(-3+2*pasos+(-1)**(pasos-
1))*(R2*μ1+R21-R3*μ2-R31))**(1/(α-1))-γ

    F=1-sp.exp(-((τ-γ)/β)**α)

    x = sp.Symbol('x')

    y=α/(β)*x*((x-γ)/β)**(α-1)*sp.exp(-((x-
γ)/β)**α)

    A=sp.integrate(y,(x,γ+0.001,τ))

    v=1/4*((-1+2*m-(-1)**(m-
1))*(R1*A+R12*F+(R1*τ+R13)*(1-
F))+(1+2*m+(-1)**(m-1))*(R3*μ2+R31)+(-
3+2*m+(-1)**(m-1))*((R2*μ1+R21)-(R3*
μ2+R31)*(F)))

self.A=(1/F)*A

self.resultado_v.insert(tk.END,
    f"{self.sel_v.get()}({m})={round(v,3)}")

self.datos_v.insert(tk.END,round(v,3))

m +=1

```

A.10 Función: Cálculo de OT

```

def calcular_ot(self):

    m=(self.paso.get())

    paso=1

    V1=list(self.datos_v.get(0, END))

    i=0

    τ=self.τ_0

    A=self.A

    F=self.F

```

```

if self.desplegable_ot.get()=="AOT":
    AOT=tau*(1-F)+A*F
    self.resultado_ot.insert(tk.END, f"AOT({round(m)})={round(AOT,3)}")

if self.desplegable_ot.get()=="AAOT":
    AOT=tau*(1-F)+A*F

    while paso<=m:
        if paso % 2 == 0:
            AAOT=(paso/2)*AOT

        else:
            AAOT=(paso+1)/2)*AOT
            self.resultado_ot.insert(tk.END, f"AAOT({paso})={round(AAOT,3)}")
            self.datos_AAOT.insert(tk.END, round(AAOT,2))
            paso+=1

if self.desplegable_ot.get()=="V/AAOT":
    if self.desplegable.get()=="TODO"or
        self.desplegable.get()=="PROPIO TAU":
        AOT=tau*(1-F)+A*F
        while paso<=m:
            if paso % 2 == 0:
                AAOT=(paso/2)*AOT
            else:
                AAOT=(paso+1)/2)*AOT
            paso+=1
            self.datos_AAOT.insert(tk.END, AAOT)
            AAOT_L=list(self.datos_AAOT.get(0, END))
            while i<=m-1:
                AR=V1[i]/float(AAOT_L[i])
                self.resultado_ot.insert(tk.END, f"AR({i+1})={round(AR,3)}")
                self.datos_AR.insert(tk.END, round(AR,2))
                i+=1

```

A.11 Función: Guardar Resultados

El objeto *file()* permite la creación de un archivo de texto. Para ello es necesario la función *open()*, ya que no es posible escribir ni leer en un archivo cerrado.

```
def guardar(self):
```

```

#obtener valores
α = float(self.α.get())
β = float(self.β.get())
γ = float(self.γ.get())
μ1 = float(self.μ1.get())
σ1 = float(self.σ1.get())
μ2 = float(self.μ2.get())
σ2 = float(self.σ2.get())
R1 = float(self.R1.get())
R12 = float(self.R12.get())
R13 = float(self.R13.get())

R2 = float(self.R2.get())
R21 = float(self.R21.get())
R3 = float(self.R3.get())
R31 = float(self.R31.get())
pasos = float(self.paso.get())
valor_optimo_info = self.valor_optimo.get()
desplegable=self.desplegable.get()
desplegable_ot=self.desplegable_ot.get()

#crear archivo txt
nombreach=fd.asksaveasfilename(initialdir = "/",title =
"Guardar como",filetypes = (("txt files","*.txt"),("todos los
archivos","*.*")),defaultextension='*.txt')
if nombreach!='':
    file=open(nombreach, "w", encoding="utf-8")
    file.write("F.Distribución Fallos")
    file.write("\t")
    file.write("α:{{}; β:{{}; γ:{{}}".format(α, β, γ))
    file.write("\t\n")

    file.write("F.D.Tiempos Reparación")
    file.write("\t")
    file.write(" μ1:{{}; σ1:{{}}".format(μ1, σ1))
    file.write("\t\n")

    file.write("F.D.Tiempos Preventivo")
    file.write("\t")
    file.write(" μ2:{{}; σ2:{{}}".format(μ2, σ2))
    file.write("\t\n")

    file.write("Retornos Operación")
    file.write("\t")

```

```
file.write("R1:{}; R12:{}; R13:{}".format(R1, R12, R13))
file.write("\t\n")

file.write("Retorno Correctivo")
file.write("\t")
file.write(" R2:{}; R21:{}".format(R2, R21))
file.write("\t\n")

file.write("Retorno Preventivo")
file.write("\t")
file.write(" R3:{}; R31:{}".format(R3, R31))
file.write("\t\n")

file.write("\t\n")
file.write("NºPasos:{}".format(pasos))
file.write("\t\n")

file.write("\t\n")
file.write("VALOR ÓPTIMO (h):")
file.write(str(valor_optimo_info))
file.write("\t\n")
file.write('\n'.join(self.listbox.get(0, END)))
file.write("\t\n")

file.write("\t\n")
file.write("V(€):{}".format(desplegable))
file.write("\t\n")
file.write('\n'.join(self.resultado_v.get(0, END)))
file.write("\t\n")

file.write("\t\n")
file.write("Tiempo de Operación:{}".format(desplegable_ot))
file.write("\t\n")
file.write('\n'.join(self.resultado_ot.get(0, END)))

file.close()
```

A.12 Función: Guardar Gráfico

La función *savefig()* permite guardar figuras creadas como los gráficos.

```
def guardar_graf(self):
    nombreach=fd.asksaveasfilename(initialdir = "/",title =
    "Guardar como",filetypes = (("PNG Image", "*.png"),("todos los
    archivos","*.")),defaultextension='.png')
    self.figure.savefig(nombreach)
```

A.13 Función: Representación de Resultados

```
def crear_grafico(self):
    m=int(self.paso.get())
    if self.varx.get()=="m":
        if self.vary.get()=="ty":
            if self.sel_v.get()=="V1":
                x=np.arange(1,m+1)
            if self.sel_v.get()=="V2":
                x=np.arange(2,m+1)
            if self.sel_v.get()=="V3":
                x=np.arange(3,m+1)
            y=self.listbox2.get(0, END)
            self.plot.set_title('τ optimum (h)',fontsize=18)
            self.plot.set_xlabel('Pasos de transición',fontsize=18)
            self.plot.set_ylabel('τ0 (Horas)',fontsize=18)
            self.plot.plot(x,y,'o-', label=f'τ0_{self.sel_v.get()}({m})'),
            self.plot.grid(True)
            self.plot.legend(loc=4,fontsize=12)

        if self.varx.get()=="m":
            if self.vary.get()=="v":
                if self.desplegable.get()=="PAR":
                    x =np.arange(start=2, stop=m+2, step=2)
                    y=np.array(self.datos_v.get(0, END))
                    #self.plot.set_title('τ0 (optimum)')
                    self.plot.set_title('{self.sel_v.get()}(€), para
                    transiciones pares',fontsize=18)
                    self.plot.set_xlabel('Pasos de transición
                    par',fontsize=18)
                    self.plot.set_ylabel('{self.sel_v.get()}(€)',fontsize=18)
                    self.plot.plot(x,y,'o-',label=f'{self.sel_v.get()}({m})
                    par'), self.plot.grid(True)
```



```

        self.plot.legend(loc=4, fontsize=12)

    if self.varx.get()=="m":
        if self.vary.get()=="v":
            if self.desplegable.get()=="IMPAR" :
                x =np.arange(start=1, stop=m+1, step=2)
                y=np.array(self.datos_v.get(0, END))
                self.plot.set_title('V(€), para transiciones
impares', fontsize=18)
                self.plot.set_xlabel('Pasos de transición
impar', fontsize=18)
                self.plot.set_ylabel('V(€)', fontsize=18)
                self.plot.plot(x, y, 'o-
', label=f'{self.sel_v.get()}({m}) impar'),
                self.plot.grid(True)
                self.plot.legend(loc=4, fontsize=12)

            if self.desplegable.get()=="TODO" :
                x =np.arange(start=1, stop=m+1)
                y=np.array(self.datos_v.get(0, END))
                #self.plot.set_title('τ0 (optimum)')
                self.plot.set_title('V(€)', fontsize=18)
                self.plot.set_xlabel('Pasos de transición', fontsize=18)
                self.plot.set_ylabel('V(€)', fontsize=18)
                self.plot.plot(x, y, 'o-', label=f'{self.sel_v.get()}({m})'),
                self.plot.grid(True)
                self.plot.legend(loc=4, fontsize=12)

        if self.varx.get()=="m":
            if self.vary.get()=="AAOT":
                if self.desplegable_ot.get()=="AAOT":
                    x =np.arange(start=1, stop=m+1)
                    y=list(self.datos_AAOT.get(0, END))
                    self.plot.set_title('Tiempo de funcionamiento promedio
acumulado (h)', fontsize=18)
                    self.plot.set_xlabel('Pasos de transición', fontsize=18)
                    self.plot.set_ylabel('AAOT(h)', fontsize=18)
                    self.plot.plot(x, y, 'o-', label=f'AAOT({m})'),
                    self.plot.grid(True)
                    self.plot.legend(loc=4, fontsize=12)

```

```

if self.varx.get()=="m":
    if self.vary.get()=="V/AAOT":
        if self.desplegable_ot.get()=="V/AAOT":
            x = np.arange(start=1, stop=m+1)
            y=self.datos_AR.get(0, END)
            self.plot.set_title('Retorno promedio (€/h)', fontsize=18)
            self.plot.set_xlabel('Pasos de transición', fontsize=18)
            self.plot.set_ylabel('V/AAOT(€/h)', fontsize=18)
            self.plot.plot(x,y, 'o-
            ', label=f'{self.sel_v.get()}/AAOT({m})'),
            self.plot.grid(True)
            self.plot.legend(loc=4, fontsize=12)
if self.varx.get()=="tx":
    if self.vary.get()=="v":
        self.datos_v_comp.delete(0, tk.END)
        import sympy as sp
        error_dato = False
        valor_optimo = 0
        m=1
        try:
            α = float(self.α.get())
            β = float(self.β.get())
            γ = float(self.γ.get())
            μ1 = float(self.μ1.get())
            σ1 = float(self.σ1.get())
            μ2 = float(self.μ2.get())
            σ2 = float(self.σ2.get())
            R1 = float(self.R1.get())
            R12 = float(self.R12.get())
            R13 = float(self.R13.get())
            R2 = float(self.R2.get())
            R21 = float(self.R21.get())
            R3 = float(self.R3.get())
            R31 = float(self.R31.get())
            pasos = float(self.paso.get())
        except:
            error_dato = True
        if not error_dato:
            m=pasos
            if self.sel_v.get()=="V1":
                self.op_dec=round(( (β**α) /α* (-R1) * (1+2*m+ (-1) ** (m-
                1)) / ((1+2*m+ (-1) ** (m-1)) * (R12-R13) + (-1+2*m- (-1) ** (m-
                1)) * (R2*μ1+R21-R3*μ2-R31)) ** (1/ (α-1)) -γ, 2)

```

```

if self.sel_v.get()=="V2":
    self.op_dec=round(( $(\beta^{**\alpha})/\alpha^{(-R1)}*(-1+2^{*pasos}-(-1)^{**}(pasos-1))/((-1+2^{*pasos}-(-1)^{**}(pasos-1))*(R12-R13)+(-3+2^{*pasos}+(-1)^{**}(pasos-1))*(R2*\mu1+R21-R3*\mu2-R31))^{**}(1/(\alpha-1))-\gamma$ )

if self.sel_v.get()=="V3":
    self.op_dec=round(( $(\beta^{**\alpha})/\alpha^{(-R1)}*(-1+2^{*pasos}-(-1)^{**}(pasos-1))/((-1+2^{*pasos}-(-1)^{**}(pasos-1))*(R12-R13)+(-3+2^{*pasos}+(-1)^{**}(pasos-1))*(R2*\mu1+R21-R3*\mu2-R31))^{**}(1/(\alpha-1))-\gamma$ )

self.op=round(self.op_dec)
tau=list(range(7, self.op, 277))+list(range(self.op,self.op+3000, 277))
i=0

while tau[i]<= self.op+3000-277:
    F=1-sp.exp(-((tau[i]- $\gamma$ )/ $\beta$ )^{** $\alpha$ ))
    x = sp.Symbol('x')
    y=  $\alpha/(\beta)*x^{((x-\gamma)/\beta)^{**(\alpha-1)}}*sp.exp(-((x-\gamma)/\beta)^{**\alpha})$ 
    A=sp.integrate(y, (x, $\gamma$ +0.001,tau[i]))
    self.datos_A.insert(tk.END, round (A,3))

    if self.sel_v.get()=="V1":
        v=1/4*( $(1+2^{*m}+(-1)^{**}(m-1))*(R1*A +R12*F+(R1*\tau[i] +R13)*(1-F))+(-1+2^{*m}-(-1)^{**}(m-1))*(R2*\mu1 +R21)*F+(R3*\mu2+R31)*(1-F)$ ))

    if self.sel_v.get()=="V2":
        v=1/4*( $(-1+2^{*m}-(-1)^{**}(m-1))*(R1*A+R12*F+(R1*\tau[i] +R13)*(1-F))+(1+2^{*m}+(-1)^{**}(m-1))*(R2*\mu1 +R21)+(-3+2^{*m}+(-1)^{**}(m-1))*((R3*\mu2+R31)-(R2*\mu1 +R21))*(1-F)$ ))

    if self.sel_v.get()=="V3":
        v=1/4*( $(-1+2^{*m}-(-1)^{**}(m-1))*(R1*A+R12*F+(R1*\tau[i] +R13)*(1-F))+(-1+2^{*m}+(-1)^{**}(m-1))*(R3*\mu2 +R31)+(-3+2^{*m}+(-1)^{**}(m-1))*((R2*\mu1+R21)-(R3*\mu2 +R31))*(F)$ ))

    self.datos_v_comp.insert(tk.END, round (v,3))

    if tau[i]==self.op:
        self.y2=v
        i+=1

x=list(range(7, self.op, 277))+list(range(self.op,self.op+3000-277, 277))

y=np.array(self.datos_v_comp.get(0, END))
x2=[self.op,self.op]
y2=[0,self.y2]

self.plot.set_title("V(€), m transiciones",fontSize=18)
self.plot.set_xlabel('τ (h)',fontSize=18)

```

```

self.plot.set_ylabel('V(€)', fontsize=18)
self.plot.axhline(0, color='black')
self.plot.plot(x2, y2, label=f'τ óptimo (x={self.op_dec})')
self.plot.plot(x, y, 'o-', label=f"{self.sel_v.get()} ({m})",
self.plot.grid(True)
#self.plot.plot(x2, y2)
self.plot.legend(loc=4, fontsize=12)
self.line.draw()

```

A.14 Función: Cálculo para Tau Propio

```

def calcular_prop_tau(self):

    import sympy as sp
    error_dato = False
    valor_optimo = 0
    m=1
    try:
        α = float(self.α.get())
        β = float(self.β.get())
        γ = float(self.γ.get())
        μ1 = float(self.μ1.get())
        σ1 = float(self.σ1.get())
        μ2 = float(self.μ2.get())
        σ2 = float(self.σ2.get())
        R1 = float(self.R1.get())
        R12 = float(self.R12.get())
        R13 = float(self.R13.get())
        R2 = float(self.R2.get())
        R21 = float(self.R21.get())
        R3 = float(self.R3.get())
        R31 = float(self.R31.get())
        m = float(self.propio_paso.get())
        τ=float(self.pro_tau.get())
    except:
        error_dato = True
    if not error_dato:
        F=1-sp.exp(-((τ-γ)/β)**α)
        x = sp.Symbol('x')
        y=α/(β)*x*((x-γ)/β)**(α-1)*sp.exp(-((x-γ)/β)**α)

```

```

A=sp.integrate(y, (x,γ+0.001,τ))
if self.desplegable_tau_pro.get()=="V":
    if self.sel_v_tau.get()=="V1":
        v=1/4*((1+2*m+(-1)**(m-1))*(R1*A +R12*F+(R1* τ
+R13)*(1- F))+(-1+2*m-(-1)**(m-1))*(R2* μ1
+R21)*F+(R3* μ2+R31)*(1-F))
        self.resultado_tau_prop.set(round(v,3))
    if self.sel_v_tau.get()=="V2":
        v=1/4*((-1+2*m-(-1)**(m-1))*(R1*A+R12*F+(R1* τ
+R13)*(1- F))+(-1+2*m+(-1)**(m-1))*(R2* μ1 +R21)+(-
3+2*m+(-1)**(m-1))*((R3* μ2+R31)-(R2* μ1
+R21)*(1-F)))
        self.resultado_tau_prop.set(round(v,3))
    if self.sel_v_tau.get()=="V3":
        v=1/4*((-1+2*m-(-1)**(m-1))*(R1*A+R12*F+(R1* τ
+R13)*(1- F))+(-1+2*m+(-1)**(m-1))*(R3* μ2 +R31)+(-
3+2*m+(-1)**(m-1))*((R2* μ1+R21)-(R3* μ2
+R31)*(F)))
        self.resultado_tau_prop.set(round(v,3))
if self.desplegable_tau_pro.get()=="AOT":
    A=(1/F)*A
    AOT=τ*(1-F)+A*F
    self.resultado_tau_prop.set(round(AOT,3))
if self.desplegable_tau_pro.get()=="AAOT":
    A=(1/F)*A
    AOT=τ*(1-F)+A*F
    if m % 2 == 0:
        AAOT=(m/2)*AOT
    else:
        AAOT=(m+1)/2)*AOT
    self.resultado_tau_prop.set(round(AAOT,3))
if self.desplegable_tau_pro.get()=="V/AAOT":
    if self.sel_v_tau.get()=="V1":
        v=1/4*((1+2*m+(-1)**(m-1))*(R1*A +R12*F+(R1* τ
+R13)*(1- F))+(-1+2*m-(-1)**(m-1))*(R2* μ1
+R21)*F+(R3* μ2+R31)*(1-F))
    if self.sel_v_tau.get()=="V2":
        v=1/4*((-1+2*m-(-1)**(m-1))*(R1*A+R12*F+(R1* τ
+R13)*(1- F))+(-1+2*m+(-1)**(m-1))*(R2* μ1
+R21)+(-3+2*m+(-1)**(m-1))*((R3* μ2+R31)-(R2*
μ1 +R21)*(1-F)))
    if self.sel_v_tau.get()=="V3":
        v=1/4*((-1+2*m-(-1)**(m-1))*(R1*A+R12*F+(R1* τ
+R13)*(1- F))+(-1+2*m+(-1)**(m-1))*(R3* μ2
+R31)+(-3+2*m+(-1)**(m-1))*((R2* μ1+R21)-(R3*

```

```
         $\mu 2 + R31) * (F) )$   
A=(1/F)*A  
AOT= $\tau * (1-F) + A * F$   
if m % 2 == 0:  
    AAOT=(m/2)*AOT  
else:  
    AAOT=(m+1)/2)*AOT  
print(AAOT)  
print(v)  
AR=v/AAOT  
self.resultado_tau_prop.set(round(AR,3))
```

A.15 Cierre de la Aplicación

```
def main():  
    mi_app = Aplicacion()  
    return 0  
if __name__ == '__main__':  
    main()
```

8 REFERENCIAS

- [1] A. Sánchez Herguedas, «Cálculo del tiempo hasta el mantenimiento preventivo utilizando procesos de Markov. Un caso de estudio,» *III Congreso Internacional de Ingeniería-Facultad de Mecánica de la Escuela Superior Politécnica de Chimborazo*, Riobamba, Ecuador, 2019. Comunicación oral.
- [2] A. E. Narro Ramirez, «Aplicación de algunos modelos matemáticos a la toma de decisiones,» *Política y Cultura*, vol. primavera, nº 6, pp. 183-198, 1996.
- [3] «Economipedia. 2020. Modelo Matemático - Definición, Qué Es Y Concepto | Economipedia.,» [En línea]. Available: <<https://economipedia.com/definiciones/modelo-matematico.html>>. [Último acceso: 10 Mayo 2020].
- [4] «Home.ubalt.edu. 2020. Modelos Deterministas: Optimización Lineal.,» 2001. [En línea]. Available: <http://home.ubalt.edu/ntsbarsh/Business-stat/opre/SpanishD.htm>. [Último acceso: mayo 2020].
- [5] J. Reason, «Cognitive Engineering in Aviation Domain,» *Lawrence Erlbaum Associates, Mahwah, NJ*, 2000.
- [6] L. Swanson, «Linking maintenance strategies to performance,» *International Journal of Economics*, vol. 70, nº 3, pp. 237-244, 2001.
- [7] S. Duffuaa, A. Raouf y J. Campbell, «Planning and Control of Maintenance Systems: Modelling and Analysis.,» *John Wiley & Sons*, pp. New York, NY, 1999.
- [8] A. Ismail, R. Ismail, R. Zulkifli, N. Makhtar y B. Deros, «t Malaysia palm oil mill,» *European Journal Scientific Research*, vol. 29, nº 1, pp. 126-135, 2009.
- [9] J. Wikstan y M. Johannson, «Maintenance and Reliability with Focus on Aircraft Maintenance and Spares Provisioning,» de *Bachelor's thesis, Lulea University of Technology*, Lulea, 2006.
- [10] N. Paz y W. Leigh, «Maintenance scheduling: issues, results and research needs,» *International Journal of Operations and Production Management*, vol. 14, nº 8, pp. 47-69, 1994.
- [11] A. Crespo Márquez, «The Maintenance Management Framework Models and Methods for Complex Systems Maintenance,» Springer London, 2007. [En línea]. Available: <https://doi.org/10.1007/978-1-84628-821-0>.
- [12] Y. Kimura, «Maintenance tribology: its significance and activity in Japan,» *Wear*, vol. 207, nº 1-2, pp. 63-66, 1997.
- [13] C. Mechefske y Z. Wang, «Using fuzzy linguistics to select optimum maintenance and condition monitoring strategies,» *Mechanical System and Signal Processing*, vol. 15, nº 6, pp. 1129-1140., 2001.
- [14] P. Okoh y S. Haugen, «A study of maintenance-related major accident cases in the 21st century,» *Process Saf. Environ. Protect.*, vol. 92, nº 4, pp. 346-356, 2014.

- [15] C. Franciosi, B. Iung, S. Miranda y S. Riemma, «Maintenance for sustainability in the industry 4.0 context: a scoping literature review,» *IFAC-PapersOnLine*, vol. 51, n° 11, pp. 903-908., 2018.
- [16] A. Sutrisno, I. Gunawan, H. Khorshidi y S. Tangkuman, «A new modified FMEA model for ranking the risk of maintenance waste considering hierarchy of root causes and effects,» *Int. J. Qual. Eng. Technol.*, vol. 5, n° 3-4, pp. 217-237, 2015.
- [17] J. Liyanage, F. Badurdeen y R. Ratnayake, «Industrial asset maintenance and sustainability performance: economical, environmental, and societal implications.,» *Handb. Mainten Manag.Eng.*, pp. 665-693, 2009.
- [18] J. Liyanage y F. Badurdeen, «Strategies for integrating maintenance for sustainable manufacturing. In: Engineering Asset Lifecycle Management.,» *Springer London*, pp. 308-315., 2010.
- [19] V. Ajukumar y O. Gandhi, « Evaluation of green maintenance initiatives in design and development of mechanical systems using an integrated approach,» *J. Clean. Prod.*, vol. 51, pp. 34-46, 2013.
- [20] Google Books , «books.google.es,» Principios De Tribología Aplicados En La Ingeniería Mecánica, 2020. [En línea]. Available: https://books.google.es/books?id=XMTEDwAAQBAJ&pg=PA25&lpg=PA25&dq=tribo-materiales&source=bl&ots=DNXUOMUUHN&sig=ACfU3U2F2MR0Tg4Xc0tcXBx1dggx1_4yaQ&hl=es&sa=X&ved=2ahUKEwjhkZuRm5_qAhXaD2MBHQf6BmYQ6AEwB3oECAkQAQ#v=onepage&q=tribo-materiales&f=false.
- [21] J. Souza, J. Sacomano, S. Kyrillos, F. Milreu y F. Paparlado, « The maintenance function in the context of corporate sustainability: a theoreticalanalytical reflexion.,» In: *IFIP International Conference on Advances in Production Managemen*, 2014.
- [22] M. Jasiulewicz-Kaczmarek y Dro_zyner, «Social dimension of sustainable developmentesafety and ergonomics in maintenance activities,» de P. In: *International Conference on Universal Access in Human-Computer Interaction*, Berlin, Heidelberg,, Springer, 2013b, pp. 175-184.
- [23] V. Ajukumar y O. Gandhi, « Evaluation of green maintenance initiatives in design and development of mechanical systems using an integrated approach,» *J. Clean. Prod.*, vol. 51, pp. 34-46, 2013.
- [24] G. Waeyenbergh y L. Pintelon, «A framework for maintenance concept development,» *International Journal of Production Economics*, vol. 77, n° 3, pp. 299-313, 2002.
- [25] C. Franciosi, A. Voisin, S. Miranda y S. Riemma, «Measuring maintenance impacts on sustainability of manufacturing industries: from a systematic literature review to a framework proposal,» *Journal of Cleaner Production*, vol. 260, n° 121065, 2020.
- [26] C. Cassady y E. Kutanoglu, «Integrating preventive maintenance planning and production scheduling for a single machine,» *IEEE Transactions on Reliability*, vol. 54, n° 2, pp. 304-309, 2005.
- [27] Crespo Marquez A y S. H. A., «Models for maintenance optimization: a study for repairable systems and finite time periods.,» *Reliability Engineering and System Safety*, vol. 75, n° 3, p. 367-377, 2002.
- [28] E. I. Basri, I. H. A. Razak, H. Ab-Samat y S. Kamaruddin, «Preventive maintenance (PM) planning: a review,» *Journal of Quality in Maintenance Engineering*, vol. 23, n° 2, pp. 114-143, 2017.
- [29] K. Kobbacy y D. Murthy, *Complex system maintenance handbook*, London: Springer, 2008.
- [30] A. Alabdulkarim, P. Ball y A. Tiwari, «Applications of simulation in maintenance research,» *World Journal of Modelling and Simulation*, vol. 9, n° 1, pp. 14-37, 2013.

- [31] H. Rommelfanger, «Fuzzy linear programming and applications,» *European Journal of Operational Research*, vol. 92, nº 3, pp. 512-527, 1996.
- [32] C. Romesburg, *Cluster Analysis for Researchers*, North Carolina: Lulu.com, 2004.
- [33] Y. Zhao, «On preventive maintenance policy of a critical reliability level for system subject to degradation,» *Reliability Engineering and System Safety*, vol. 79, nº 3, pp. 301-308, 2003.
- [34] A. Labib, R. O'Connor y G. Williams, «An effective maintenance system using the analytic hierarchy process,» *Integrated Manufacturing Systems*, vol. 9, nº 2, pp. 87-98, 1998.
- [35] Pythonizame.s3.amazonaws, «Pythonizame.s3.amazonaws.com,» 2020. [En línea]. Available: <https://pythonizame.s3.amazonaws.com/media/Book/iniciacion-python-3/file/8ba1f99c-7c60-11e5-964d-04015fb6ba01.pdf>.
- [36] Iaa.csic, «Iaa.csic.es,» 2020. [En línea]. Available: <https://www.iaa.csic.es/python/curso-python-para-principiantes.pdf>. [Último acceso: 6 Junio 2020].
- [37] Programoergosum., «www.programoergosum.com,» Pi., ¿Qué Es Python? - Curso De Iniciación A La Programación Con Python En Raspberry, 2020. [En línea]. Available: <https://www.programoergosum.com/cursos-online/raspberry-pi/244-iniciacion-a-python-en-raspberry-pi/que-es-py>. [Último acceso: 15 junio 2020].
- [38] The Best Manuals Online , «the-best-manuals-online.com,» Esquema del sistema hidráulico de camiones fuera de carretera OEM de Caterpillar 777F y 777F, [En línea]. Available: <https://the-best-manuals-online.com/es/products/caterpillar-777f-and-777f-oem-off-highway-truck-hydraulic-system-schematic-diagrams>.
- [39] Think Big., «empresas.blogthinkbig.com,» Python Para Todos (3): Scypy, Numpy, Pandas ...¿Qué Librerías Necesitamos? - Think Big Empresas., 2020. [En línea]. Available: <https://empresas.blogthinkbig.com/python-todos-3-librerias/>.
- [40] Es.qwe.wiki, «Es.qwe.wiki.com,» Tkinter - Qwe.Wiki., 2020. [En línea]. Available: <https://es.qwe.wiki/wiki/Tkinter#Widget>.
- [41] Tkdocs, «tkdocs.com,» TkDocs - Tk Tutorial - Basic Widgets, [En línea]. Available: <https://tkdocs.com/tutorial/widgets.html>.
- [42] Dummies, «Dummies.com,» Tkinter Widgets In Python - Dummies, 2020. [En línea]. Available: <https://www.dummies.com/programming/python/using-tkinter-widgets-in-python/>.
- [43] Recursospython, «Recursospython.com,» Posicionar elementos en Tkinter, 2020. [En línea]. Available: <https://recursospython.com/guias-y-manuales/posicionar-elementos-en-tkinter/>.
- [44] «uniwebsidad.com,» 2.2. Estructuras de Control de Flujo (Python para principiantes), 2020. [En línea]. Available: <https://uniwebsidad.com/libros/python/capitulo-2/estructuras-de-control-de-flujo>.

