

APRENDIZAJE REFORZADO PARA POLÍTICAS DE REPARACIÓN DE SISTEMAS DE VARIAS COMPONENTES

**Memoria para optar al Título profesional de
Ingeniero Civil Matemático**

CRISTÓBAL ALEXIS REYES PAVEZ

Profesores Guía

Prof. Guido Lagos (Ingeniería - UAI)
Prof. Héctor Olivero Q. (Ingemat - UV)

Miembros de la comisión:

Prof. Rolando Rebolledo (Ingemat - UV)

INSTITUTO DE INGENIERÍA MATEMÁTICA
VALPARAÍSO, 1 de abril de 2025

Resumen

La confiabilidad de un sistema se define como su capacidad para cumplir su propósito, operando de manera continua y sin interrupciones. Para garantizar esta estabilidad, se implementan estrategias de mantenimiento preventivo que buscan maximizar el tiempo operativo y minimizar tanto las fallas imprevistas como los costos asociados a las reparaciones.

Este trabajo se centra en la optimización de políticas de reparación en entornos estocásticos, especialmente en sistemas con múltiples componentes con tiempos de vida aleatorios. Para abordar el problema, se analizarán distintas distribuciones de tiempo de vida y se utilizarán procesos de decisión de Markov como marco teórico para su optimización.

La efectividad de las estrategias propuestas se evaluará mediante simulaciones, demostrando su impacto en la mejora de la gestión del mantenimiento. A través del uso de algoritmos de aprendizaje automático, se diseñan estrategias que optimizan la disponibilidad del sistema y reducen los costos asociados a su mantenimiento.

Agradecimientos

Quiero expresar mi más sincero agradecimiento a mis profesores guías Hector Olivero y Guido Lagos

Este trabajo fue financiado por la Agencia Nacional de Investigación y Desarrollo de Chile (ANID), a través del proyecto Fondecyt de Iniciación en Investigación N° 11230256.

En primer lugar, quiero expresar mi más profundo agradecimiento a mi familia, quienes en todo momento me brindaron su apoyo y cariño. En especial, a mis padres, cuyo esfuerzo y dedicación me han permitido llegar hasta aquí. Me inculcaron los valores que hoy definen quién soy, y con su arduo trabajo aseguraron que nunca me faltara nada, ni en el hogar ni en mis estudios. Su amor y sacrificio es la base de mi crecimiento personal y académico, y por ello les estaré eternamente agradecido.

Asimismo, deseo extender mi gratitud a mis profesores guías, Héctor Olivero y Guido Lagos, por su paciencia y orientación a lo largo de este trabajo. En especial, al profesor Héctor quien no solo fue un pilar fundamental en este proyecto, sino también un apoyo incondicional durante mi paso por la universidad. Con especial paciencia, me guió en las etapas más importantes de mi formación, brindándome siempre un consejo oportuno y valioso. Aprecio profundamente cada conversación que tuvimos.

Finalmente, quiero agradecer al Instituto de Ingeniería Matemática, a sus profesores y a los compañeros que lo conforman. Durante estos años, compartí con ellos muy agradables momentos y valoro enormemente las experiencias y aprendizajes que vivimos juntos.

Índice general

Resumen	2
Algunas Palabras	3
1. Introducción	6
2. Procesos de decisión de Markov	8
2.1. Modelo de decisión secuencial	9
2.2. Elementos de un MDP	10
2.3. Políticas óptimas	13
2.4. Proceso estocástico inducido	14
2.5. Procesos de recompensa y criterio de optimización	16
2.6. Ecuación Optimalidad	20
2.7. Solución de la Ecuación de Optimalidad	24
3. Algoritmos de Optimización e implementaciones	27
3.1. Algoritmo de Iteración de Valores	27
3.1.1. Ejemplo de aplicación	29
4. Modelos para tiempos de vida	33

4.1. Conformación de los Sistemas Coherentes	34
4.2. Distribución Marshall-Olkin	36
4.3. Lévy-Frailty Marshall-Olkin	38
4.3.1. Procesos de Lévy	39
4.3.2. Distribución LFMO	45
5. Estudio de Caso	48
5.1. Modelación sistema para determinar una política de mantenimiento óptima	48
5.2. Formulación del problema de optimización	50
5.3. Simulaciones y Resultados	53
5.4. Evaluación políticas Optimas	58
6. Conclusiones y trabajo futuro	66
A. Códigos empleados	68
A.1. Simulación LFMO	68
A.2. Iteración De Valor	73

Capítulo 1

Introducción

El análisis de fiabilidad de sistemas complejos desempeña un papel crucial en la ingeniería y la gestión industrial, ya que permite diseñar planes de mantenimiento, que busquen asegurar el funcionamiento continuo de procesos, minimizar tiempos de inactividad y reducir costos asociados a reparaciones imprevistas. En este proyecto se define el concepto de “*sistema*” como una colección de componentes o subsistemas organizados conforme a un diseño específico con el propósito de realizar una tarea determinada.

Un sistema se considera “*operativo*” si cumple su función de acuerdo con los requisitos establecidos, en caso contrario, se dice que presenta un “*fallo*”, lo que puede deberse a la interrupción total o parcial de su funcionamiento. La capacidad del sistema para mantenerse operativo a lo largo del tiempo, a pesar del desgaste natural o posibles fallos repentinos, se conoce como **fiabilidad**.

En este trabajo, el término fiabilidad será tratado desde una perspectiva cuantitativa, definiéndolo de manera precisa en el capítulo 4. Bajo esta descripción, la fiabilidad estará determinada tanto por las características intrínsecas de los componentes del sistema como por la forma en que estos interactúan para mantener la operatividad global.

El estudio aborda cómo la disminución en la operatividad y fiabilidad de un sistema, debido a fallos, puede mitigarse mediante una estrategia de reparación óptima. Para ello, se modela la situación como un *problema de decisión de Markov* en el que un agente se encarga de gestionar las reparaciones de los componentes defectuosos, considerando los costos asociados a cada operación. El objetivo es determinar

políticas que indiquen cuándo y qué componentes reparar, logrando así equilibrar los costos de reparación con la mejora en el tiempo de operatividad del sistema.

Para abordar el problema, en el capítulo 2 se explorarán los fundamentos clave de la teoría de Procesos de Decisión de Markov, un marco matemático enfocado en modelar la toma de decisiones en ambientes con incertidumbre. Este capítulo no solo introducirá las bases teóricas del modelo, sino que también definirá los métodos esenciales que permitirán formular y resolver el problema de manera eficiente, estableciendo las bases para una solución robusta.

El capítulo 3 se centrará en el diseño y presentación del algoritmo de "*Iteración De Valor*" para la obtención de políticas óptimas. Se presentaran los aspectos importantes inherentes a este algoritmo y se analizarán los resultados obtenidos con un ejemplo propuesto.

Dado que el funcionamiento de sistemas complejos está sujeto a incertidumbre, se modelará el tiempo de vida de sus componentes mediante variables aleatorias. En la sección 4 se describirán los modelos probabilísticos empleados, considerando tanto eventos que afectan individualmente a cada componente como aquellos que influyen de manera simultánea a múltiples componentes.

Finalmente, se aplicará todo el material desarrollado al análisis de un sistema clásico en el capítulo 5, realizando diversas mediciones para diferentes costos y parámetros. Se llevarán a cabo comparaciones con medidas propuestas, evaluando el desempeño y la eficiencia de las políticas obtenidas.

Capítulo 2

Procesos de decisión de Markov

En este capítulo se da una breve introducción a los **Procesos de Decisión de Markov (MDP, por sus siglas en inglés)**. Los MDP son utilizados como modelo matemático para la toma de decisiones en entornos estocásticos, donde las acciones de un agente influyen en la evolución de un sistema. El objetivo principal de este capítulo es describir la formulación de los MDP, identificar sus componentes esenciales y analizar, los resultados teóricos y propiedades que sustentan el modelo.

Se definirá formalmente el **problema de decisión de Markov**, donde el objetivo es encontrar una *política óptima* que maximice (o minimice) una función objetivo, generalmente relacionada con la recompensa acumulada en el tiempo.

El análisis se enfocará exclusivamente en los **modelos MDP de horizonte temporal infinito** con un *factor de descuento* asociado. En este contexto, se presentará la **ecuación de Bellman**, estudiando sus propiedades fundamentales. Se demostrará que, bajo ciertas condiciones, dicha ecuación posee existencia y unicidad de soluciones, las cuales corresponden a la solución óptimas para el problema de decisión planteado.

Se tomará como principal referencia el libro “Markov decision processes: discrete stochastic dynamic programming” de M. Puterman (2014) [Put14], en cuyos capítulos 2 y 6 se desarrolla la teoría más robusta sobre el tema. Estos conceptos se complementarán con los aportes fundamentales de R. Bellman, expuestos en su artículo “The theory of dynamic programming” (1954) [Bel54] y en su libro “Dynamic Programming” (1957) [Bel57].

2.1. Modelo de decisión secuencial

Un **modelo de decisión secuencial**, representa situaciones donde las decisiones afectan la evolución del sistema a lo largo del tiempo. En estos modelos, se introduce un ente de referencia denominado **decisor** o **agente**, quien debe tomar decisiones en un entorno dinámico determinado por la estructura del sistema, el cual está sujeto a la incertidumbre de procesos subyacentes.

El objetivo principal de estos modelos es optimizar una función objetivo específica. Este objetivo puede ser, por ejemplo, maximizar recompensas acumuladas a lo largo del tiempo o minimizar costos esperados. La optimización de dicha función requiere seleccionar acciones óptimas, considerando no solo sus efectos inmediatos, sino también su impacto a largo plazo en la evolución del sistema.

El esquema general de un modelo de decisión secuencial se ilustra en la Figura 2.1. En dicha figura describe la evolución del sistema en un conjunto de tiempos discretos, $T = \{0 = t_0 \leq t_1 \leq \dots \leq t_n\}$ a través de estados discretos s_t observados en dichos instantes. Con respecto a la dinámica del sistema, el agente elige una acción a_t del conjunto de acciones disponibles que denotamos $\mathcal{A}$, y como resultado de esta acción, el agente recibe una **retribución** r_t , que se considera como una **recompensa inmediata** si esta es positiva o un **costo inmediato** si es negativa. Además, la acción tomada afecta directamente al sistema, provocando su transición hacia un nuevo estado s_{t+1} .

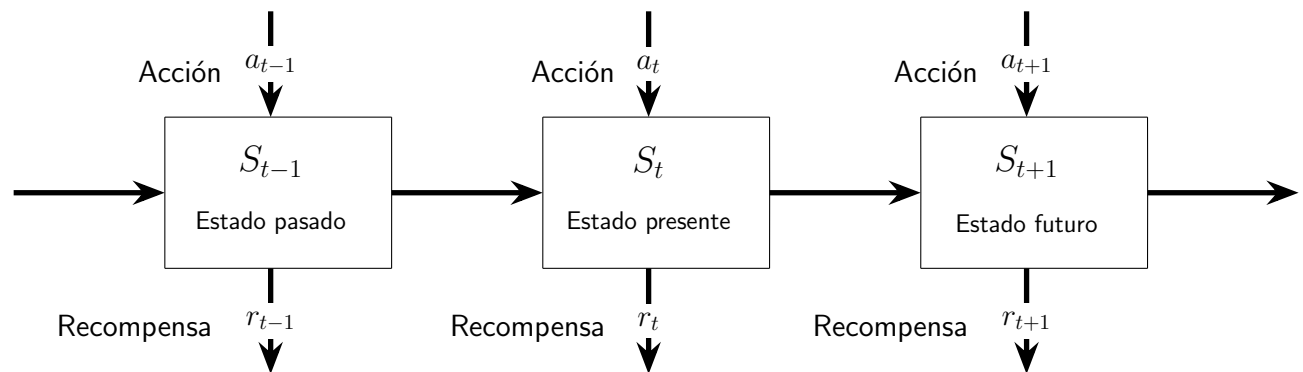


Figura 2.1: Diagrama de transición de estados

La estructura presentada anteriormente en la Figura 2.1 incorpora transiciones aleatorias entre estados. Estas transiciones, transiciones son representadas mediante una **distribución de probabilidad** denotada como $p(s_{t+1} \mid s_t, a_t)$ que depende del estado actual s_t , de la acción seleccionada a_t y de la dinámica inherente del sistema. Este enfoque permite capturar la evolución incierta del sistema,

generando una secuencia de estados aleatorios que se denomina **proceso de decisión de Markov**.

Aunque un MDP se basa en conceptos markovianos, estrictamente no es un proceso puramente markoviano, ya que incorpora decisiones del agente. Sin embargo, mantiene una estructura **markoviana**, que establece que la probabilidad de transición hacia un nuevo estado s_{t+1} depende únicamente del estado actual s_t y de la acción a_t , ignorando la secuencia completa de estados y acciones anteriores.

2.2. Elementos de un MDP

Para definir un proceso de decisión de Markov, es fundamental introducir varios elementos clave que serán descritos a continuación:

Definición 2.1 (Épocas de decisión) *El conjunto de épocas de decisión T representa los momentos en los cuales se toman decisiones. Este conjunto puede ser continuo o discreto, según la naturaleza del problema. En este proyecto, se considerarán los casos discretos:*

	Horizonte	
	Finito	Infinito
Tiempo Discreto	$\{0, 1, \dots, N\}$	$\mathbb{N}_0$

En el caso discreto, la acción del agente se realiza en una época de decisión determinada. Por otro lado, en el caso continuo, la acción del agente debe ocurrir dentro del intervalo entre épocas de decisión y puede ejecutarse de tres maneras: en un instante aleatorio, en un momento que sea conveniente para el agente, o exactamente en la época de decisión (para más detalles sobre el caso continuo, se recomienda consultar la bibliografía sobre MDP).

Es importante remarcar que las épocas de decisión ocurren secuencialmente, pero no necesariamente de forma equidistante. La distribución de estos tiempos puede ajustarse a una determinada ley de probabilidad, dependiendo de la naturaleza del problema.

Definición 2.2 (Espacio de estados) *El espacio de estados, denotado por S , define todas las posibles configuraciones en las que el sistema puede encontrarse. Este espacio puede ser finito, numerable,*

compacto o más general, según el caso particular. Cuando el espacio es finito, la cantidad de estados se denota por $|S|$ o simplemente N .

Por lo tanto, para cada época de decisión, el agente observa el estado del sistema $s_t \in S$. En dicho estado la interacción del agente sobre el sistema se realizan mediante una de las acciones que selecciona.

En el contexto de los MDP, el agente selecciona acciones para hacer transicionar el sistema entre estados, lo que resulta en retribuciones inmediatas que evalúan la calidad de las decisiones. Este conjunto de elementos define el marco sobre el cual se analizan y optimizan los procesos de decisión.

Definición 2.3 (Conjuntos de Acciones) Para cada $s \in S$, se denota por A_s al conjunto de acciones permitidas en dicho estado. Al igual que el espacio de estados, el conjunto de acciones puede ser finito, numerable, compacto o más general.

Al tomar una acción a_t en el estado s_t , el sistema transita a un nuevo estado s_{t+1} . Este proceso de transición está acompañado de una retribución inmediata, que cuantifica el impacto directo de la acción sobre el sistema:

Definición 2.4 (Recompensas/Costes Inmediatos) Para cada acción $a \in A_s$ tomada en la época de decisión t , se define una función de recompensa $r_t(s, a) \rightarrow \mathbb{R}$. En el caso de un horizonte de tiempo finito, la recompensa en la última época de decisión N se expresa mediante la función $r_N(s) \rightarrow \mathbb{R}$, denominada **valor de rescate o valor de desecho**.

En problemas de optimización de ingresos, si $r_t(s, a) > 0$, este valor representa un ingreso, por el contrario, si $r_t(s, a) < 0$, este se interpreta como un costo.

Cuando las transiciones entre estados son inciertas, es fundamental describir la evolución del sistema mediante funciones de probabilidad:

Definición 2.5 (Probabilidades de Transición) Para un estado $s \in S$, al elegir una acción $a \in A_s$ en la época de decisión t , el sistema evoluciona a un nuevo estado en la época $t + 1$ según una ley de transición descrita por la **función de transición de probabilidad**, $p_t(\cdot | s, a) \rightarrow [0, 1]$.

La información sobre las probabilidades de transición entre pares de estados, condicionada a una acción, se representa mediante una función matricial $P : \mathcal{A} \rightarrow S^2$, definida para cada acción $a \in \mathcal{A}$ como:

$$P(a) = (p_{ij}(a)) \quad \text{donde} \quad p_{ij}(a) = p_t(j|i, a), \quad p_{ij}(a) \geq 0, \quad \text{y} \quad \sum_j p_{ij}(a) = 1 \quad \text{para todo } i. \quad (2.1)$$

Notar aquí que las transiciones entre estados dependen únicamente del estado s , la acción a y el tiempo t sin considerar la historia previa del proceso. Esta característica justifica su denominación como procesos markovianos.

Observación 2.6 Se omitirá la notación temporal en las retribuciones y transiciones de probabilidad, dado que se asume que estos valores son estacionarios, es decir, que permanecen constantes a lo largo del tiempo, de acuerdo con el problema que se desea modelar.

Con estos elementos básicos definidos, podemos introducir la definición formal de un proceso de decisión de Markov:

Definición 2.7 (Proceso de Decisión de Markov) El conjunto de elementos

$$\{T, S, \mathcal{A}_s, p(j | i, a), (s, a)\}$$

se denomina **proceso de decisión de Markov (MDP)**, donde:

- T : Representa el horizonte temporal, que puede ser finito o infinito, indicando las etapas de decisión del proceso.
- S : Es el conjunto de estados posibles del sistema.
- $\mathcal{A}_s$: Denota el conjunto de acciones disponibles en el estado s .
- $p(i | j, a)$: Es la función de probabilidad de transición, que para una acción a especifica las probabilidades de moverse de un estado i a un estado j , es decir: $\mathbb{P}(s_{t+1} = j | s_t = i, a_t = a)$.

- $r(s, a)$: Representa la retribución inmediata obtenida al tomar la acción a en el estado s . Es una función que asigna un valor (recompensa o costo) a cada combinación de estado y acción.

2.3. Políticas óptimas

Una **política** es una estrategia que guía al agente en la toma de decisiones a lo largo del tiempo. La forma que toma una política depende del tipo de problema y de las restricciones que se impongan.

Definición 2.8 Una política π es una secuencia de funciones de decisión:

$$\pi = (\pi_0, \pi_1, \pi_2, \dots),$$

donde cada función $\pi_t : S \rightarrow A_s$ asigna una acción $a \in A_s$ a cada estado $s \in S$ en el tiempo t . Según la forma de asignación de acciones, las políticas pueden clasificarse en:

- **Deterministas**: $\pi_t(s) = a$ selecciona una única acción a para cada estado s .
- **Aleatorias**: $\pi_t(a | s)$ define una distribución de probabilidad sobre las acciones para cada estado.

Una política es **estacionaria** si su regla de decisión no varía con el tiempo, es decir, $\pi_0 = \pi_1 = \dots$. Este tipo de políticas, denotadas por π^∞ son particularmente relevantes en la teoría, ya que su estructura invariable en el tiempo permite desarrollar análisis matemáticos más manejables. Más adelante, veremos cómo estas políticas pueden desempeñar un papel clave en la búsqueda de soluciones óptimas en ciertos contextos.

En este trabajo, nos enfocaremos en políticas que dependen únicamente del estado actual, ignorando el historial previo del proceso. Bajo esta suposición, las definiciones presentadas son suficientes. Para una exploración más profunda, se recomienda consultar [Put14] o [Der70].

Finalmente, denotaremos por Π el conjunto de todas las políticas

Al fijar una política π , las probabilidades de transición entre estados del sistema quedan completa-

mente determinadas por las acciones. Estas probabilidades se representan de la siguiente forma:

$$P^\pi := (p_{i,j}^\pi(a)) \quad \text{donde} \quad p_{i,j}^\pi(a) := \mathbb{P}(j \mid i, a = \pi(i))\pi(i).$$

De manera análoga, el vector de recompensas asociado a la política π se define para cada estado del sistema como:

$$r^\pi := (r(i)) \quad \text{donde} \quad r(i) := r(i, \pi(i)).$$

El objetivo principal del agente es encontrar una política π que maximice la utilidad total a lo largo del tiempo. Para ello, es fundamental analizar cómo dicha política influye en la evolución del proceso de toma de decisiones.

2.4. Proceso estocástico inducido

La evolución de un MDP puede describirse a través de un proceso estocástico que depende de la política elegida. Para cada instante de tiempo t , definimos las siguientes variables aleatorias clave:

$$X_t = s_t, \quad Y_t = a_t,$$

donde X_t representa el estado del proceso en el tiempo t , y Y_t denota la acción seleccionada en dicho instante.

Inicialmente, en $t = 0$, el sistema comienza en un estado inicial s_0 , definido por una distribución inicial sobre el conjunto de estados, $\gamma(s)$. Esta distribución es independiente de la política escogida:

$$\gamma(s) := \mathbb{P}(X_0 = s).$$

En ciertos problemas, el estado inicial puede ser conocido con certeza. Esto conduce a una distribución

determinista:

$$\gamma(s) := \begin{cases} 1, & \text{si } s = s^*, \\ 0, & \text{si } s \neq s^*. \end{cases}$$

Teniendo en cuenta el problema que se quiere abordar, se asumirá que el estado inicial es determinista.

Una vez determinado el estado inicial, se selecciona una acción, guiada por la política π . Esto induce una distribución de probabilidad $\mathbb{P}^\pi$, donde la probabilidad de seleccionar una acción a es:

$$\mathbb{P}^\pi(Y_0 = a) = \sum_{s \in S} \gamma(s) \pi(a | s).$$

En otras palabras, esta probabilidad depende tanto de la distribución inicial como de la aplicación de la política sobre cada estado.

Después de elegir la acción, el sistema evoluciona a un nuevo estado, determinado por la función de transición:

$$\mathbb{P}^\pi(X_1 = s_1 | X_0 = s_0, Y_0 = a_0) = p^\pi(s_1 | s_0, a_0).$$

Este proceso continúa iterativamente, seleccionando acciones y realizando transiciones a lo largo del horizonte temporal. Así, una política π y una distribución inicial γ inducen un proceso estocástico sobre las trayectorias:

$$\begin{aligned} (X_0, Y_0, \dots, Y_{N-1}, X_N), & \quad \text{en un horizonte finito,} \\ (X_0, Y_0, \dots), & \quad \text{en un horizonte infinito.} \end{aligned}$$

A cada realización de este proceso se le denomina **trayectoria**, las cuales se representan según la Tabla 2.1.

Si la política π depende únicamente del estado actual, el proceso inducido constituye una **cadena de Markov** con valores en $S \times A$, ya que la probabilidad de una trayectoria se determina de la siguiente forma:

	Horizonte Finito	Horizonte Infinito
Espacio Ω	$\{S \times \mathcal{A}\}^{N-1} \times S$	$\{S \times \mathcal{A}\}^\infty$
Trayectorias ω	$(s_0, a_0, s_1, a_1, \dots, a_{N-1}, s_N)$	$(s_0, a_0, s_1, a_1, \dots)$

Cuadro 2.1: Notación para trayectorias en horizontes finitos e infinitos.

$$\mathbb{P}^\pi(s_0, a_0, \dots, a_{N-1}, s_N) = \gamma(s_0) \prod_{i=0}^{N-1} p(s_{i+1} \mid s_i, a_i) \pi(a_i \mid s_i),$$

$$\mathbb{P}^\pi(s_0, a_0, \dots, s_N, a_N) = \gamma(s_0) \pi(a_0 \mid s_0) \prod_{i=1}^N p(s_i \mid s_{i-1}, a_{i-1}) \pi(a_i \mid s_i).$$

Un razonamiento similar se aplica para horizontes infinitos. Además, si π es determinista, es decir, $\pi(s) = a$ para cada estado s , el proceso inducido deja de depender explícitamente de las acciones y se convierte en una **cadena de Markov** sobre el espacio de estados S .

$$\mathbb{P}^\pi(s_0, s_1, \dots, s_N) = \gamma(s_0) \prod_{i=0}^{N-1} p(s_{i+1} \mid s_i \pi(s_i)).$$

En este caso, la notación y las expresiones se simplifican considerablemente al no requerir la explicación de las acciones en la trayectoria.

2.5. Procesos de recompensa y criterio de optimización

El proceso de trayectorias induce un **proceso de recompensas**. La notación correspondiente es la siguiente:

- **Horizonte Finito:** El espacio de recompensas se representa mediante $\mathbb{R}^N$, y el proceso se denota por

$$(r_0(X_0, Y_0), \dots, r_N(X_N)).$$

- **Horizonte Infinito:** El espacio de recompensas es $\mathbb{R}^\infty$, y el proceso se expresa como

$$(r_0(X_0, Y_0), r_1(X_1, Y_1), \dots).$$

Para simplificar la notación, denotaremos la recompensa en la época R_n como

$$R_n = r(X_n, Y_n),$$

donde r es la recompensa obtenida al estar en el estado X_n y tomar la acción Y_n . Denotaremos R al vector (o sucesión) que representa la realización del proceso.

Como se ha mencionado, el objetivo es determinar la mejor política a lo largo del tiempo. Para ello, es necesario establecer un **criterio de comparación**. Se considerará el concepto de **dominación estocástica** como base para esta comparación:

Definición 2.9 (Dominación Estocástica[OT10]) Sean dos variables aleatorias X_1 y X_2 cuyas funciones de distribución acumulada son F_1 y F_2 , respectivamente. Se dice que F_1 domina estocásticamente a F_2 si, para todo $x \in \mathbb{R}$:

$$F_1(x) \leq F_2(x), \quad \text{lo que equivale a} \quad \mathbb{P}(X_1 > x) \geq \mathbb{P}(X_2 > x).$$

La dominación estocástica implica que, bajo F_1 , los valores más altos tienen mayor probabilidad acumulada que bajo F_2 . Si ambas funciones son estrictamente crecientes, entonces para todo $x \in \mathbb{R}$, las funciones cuantil F_1^{-1} y F_2^{-1} cumplen:

$$x \leq F_1^{-1}(F_2(x)).$$

Si se define $\tilde{X}_1 = F_1^{-1}(U)$ y $\tilde{X}_2 = F_2^{-1}(U)$, donde $U \sim U[0, 1]$, entonces $\tilde{X}_1 \stackrel{d}{=} X_1$ y $\tilde{X}_2 \stackrel{d}{=} X_2$. Al estar definidas en el mismo espacio de probabilidad, se pueden comparar y $\tilde{X}_1 \geq \tilde{X}_2$ casi seguramente. Este resultado refuerza que F_1 "domina" a F_2 al asignar más peso a valores altos.

Teorema 2.10 Para cualquier par de políticas π_1 y π_2 que inducen distribuciones $\mathbb{P}^{\pi_1}$ y $\mathbb{P}^{\pi_2}$, respecti-

vamente, se cumple que $\mathbb{P}^{\pi_1}$ domina estocásticamente a $\mathbb{P}^{\pi_2}$ si y solo si:

$$\mathbb{E}^{\pi_1}(u(X)) \geq \mathbb{E}^{\pi_2}(u(X))$$

para toda función de utilidad u creciente.

La **función de utilidad** captura el grado de satisfacción o preferencia del agente frente a distintas remuneraciones cuantificables. Este concepto permite traducir valoraciones subjetivas de preferencias a una representación matemática objetiva.

Un aspecto asociado a la forma de $u(x)$ es la actitud del agente frente al riesgo. No todas las funciones de utilidad representan de igual manera la actitud del agente frente al riesgo como se ve en los siguientes ejemplos:

- **Aversión al riesgo:** Se dice que un agente es adverso al riesgo cuando su función de utilidad $u(x)$ es cóncava. Esto implica que la utilidad marginal disminuye conforme aumenta x , lo que lleva al agente a preferir resultados ciertos frente a aquellos inciertos, incluso si ambos tienen el mismo valor esperado. Esto se observa en la siguiente desigualdad, que se deduce de la desigualdad de Jensen:

$$u(\mathbb{E}[X]) \geq \mathbb{E}[u(X)]$$

refleja que la utilidad de obtener de forma segura el valor $\mathbb{E}[X]$ es mayor que la utilidad promedio al enfrentar la incertidumbre en X . lo que significa que el agente valora más obtener un pago seguro equivalente a $\mathbb{E}[X]$ que enfrentar la incertidumbre de un pago aleatorio representado por X .

- **Preferencia por el riesgo:** Se dice que un agente muestra preferencia por el riesgo cuando su función de utilidad $u(x)$ es convexa. Esto implica que la utilidad marginal aumenta conforme crece x , lo que evidencia que el agente esta dispuesto a asumir incertidumbre en busca de mayores recompensas. En términos matemáticos, se expresa mediante la desigualdad:

$$u(\mathbb{E}[X]) \leq \mathbb{E}[u(X)],$$

lo que indica que la utilidad promedio obtenida de resultados inciertos supera la utilidad de recibir de manera segura el valor $\mathbb{E}[X]$.

- **Neutralidad al riesgo:** Cuando $u(x)$ es lineal, el agente no tiene inclinación particular hacia o contra el riesgo, basando sus decisiones únicamente en el valor esperado, sin considerar la variabilidad.

Si ninguna distribución inducida por una política domina estocásticamente a las demás, la elección de la política óptima dependerá de la forma de $u(x)$. Por ejemplo, con una función $u(x) = x^2$ que favorece valores altos, se preferirá una política que maximice la probabilidad de grandes recompensas, incluso con mayor incertidumbre. Con una función como $u(x) = \sqrt{x}$, se priorizará una política que garantice recompensas moderadas y menor variabilidad.

El estudio de la teoría de utilidad esperada es fundamental en economía y finanzas, ya que proporciona las bases para analizar decisiones óptimas en contextos de incertidumbre. Textos clásicos como *Microeconomic Analysis* (Hal R. Varian, 1992) y *Risk Aversion in the Small and in the Large* (John W. Pratt, 1978) ofrecen una discusión profunda sobre este tema.

Para este trabajo, se revisará la siguiente función de utilidad:

- Utilidad aditiva con descuentos:

$$u(R) = \sum_{n=0}^{\infty} \alpha^n R_n, \quad (2.2)$$

donde $\alpha \in (0, 1)$ representa el factor de descuento, que mide la valoración decreciente de las recompensas a lo largo del tiempo.

El criterio de optimización se plantea como la determinación de una política óptima $\pi^* \in \Pi$ tal que, para cualquier estado $s \in S$, se cumple:

$$\mathbb{E}_s^{\pi^*}[u(R)] \geq \mathbb{E}_s^{\pi}[u(R)] \quad \forall \pi \in \Pi. \quad (2.3)$$

2.6. Ecuación Optimalidad

En el desarrollo subsecuente se asumirán los siguientes supuestos:

- Las transiciones de probabilidad y las retribuciones son estacionarias, es decir, no varían entre épocas de decisión.
- Las retribuciones están acotadas, es decir, $|r(s, a)| \leq M < \infty$ para toda acción $a \in \mathcal{A}$ y estado $s \in \mathcal{S}$.
- El espacio de estados $\mathcal{S}$ es discreto, pudiendo ser finito o contable.
- Las retribuciones serán descontadas por un factor $\alpha \in (0, 1)$.

Tal y como se presentó en la sección anterior, se prosigue con el análisis de utilidad descontada a lo largo del tiempo.

Definición 2.11 (Función de utilidad total esperada descontada) *Para una política π , se define la función $v_\alpha^\pi(s)$ como:*

$$v_\alpha^\pi(s) = \mathbb{E}_s^\pi \left[\sum_{t=1}^{\infty} \alpha^{t-1} R_t \right], \quad (2.4)$$

Esta función representa, bajo la política π , el valor esperado de las retribuciones futuras, dado que el sistema comienza en el estado s , considerando que las retribuciones están ponderadas por el factor de descuento α .

Dado que $\alpha \in (0, 1)$ y que las retribuciones $R_t = r(X_t, Y_t)$ están acotadas, la serie

$$\sum_{t=1}^{\infty} \alpha^{t-1} R_t \quad (2.5)$$

es geométrica y convergente. Esto justifica que v_α^π esté correctamente definida.

En la sección 2.5, se explicó que la optimización se realiza en términos de dominación estocástica. Bajo este criterio, se define el valor óptimo de la función de utilidad total esperada descontada como

$$v^*(s) = \sup_{\pi \in \Pi} v_\alpha^\pi(s), \quad (2.6)$$

donde el supremo se toma considerando el orden inducido por la dominación estocástica.

La expresión (2.6) es bastante general y trabajar directamente con ella no es aconsejable. Una propiedad clave de esta formulación se presenta en la siguiente proposición, basada en lo abordado en [Put14]. Esta proposición establece una relación recursiva para la función de utilidad.

Proposición 2.12 *Para toda política $\pi \in \Pi$, la función $v_\alpha^\pi(s)$ definida en (2.4) cumple la siguiente relación recursiva para cada estado $s \in S$:*

$$v_\alpha^\pi(s) = r(s, \pi_1(s)) + \alpha \sum_{s' \in S} p(s' | s, \pi_1(s)) v_\alpha^{\pi'}(s'), \quad (2.7)$$

donde $\pi' = (\pi_2, \pi_3, \dots)$ representa la política a partir de la época de decisión 2 en adelante.

Demostración: Sea $\pi = (\pi_1, \dots)$ una política $\pi \in \Pi$. Como las retribuciones son acotadas, por teorema de Fubini-Tonelli:

$$\begin{aligned} v_\alpha^\pi(s) &= \mathbb{E}_s^\pi \left[\sum_{t=1}^{\infty} \alpha^{t-1} r(X_t, Y_t) \right] \\ &= \sum_{t=1}^{\infty} \alpha^{t-1} \mathbb{E}_s^\pi [r(X_t, Y_t)] \\ &= \mathbb{E}_s^\pi [r(X_1, Y_1)] + \sum_{t=2}^{\infty} \alpha^{t-1} \mathbb{E}_s^\pi [r(X_t, Y_t)] \\ &= r(s, \pi_1(s)) + \alpha \sum_{t=2}^{\infty} \alpha^{t-2} \mathbb{E}_s^\pi [r(X_t, Y_t)] \end{aligned}$$

Por otro lado por propiedad de la torre para la esperanza condicional se tiene que para todo $t > 2$:

$$\begin{aligned} \mathbb{E}_s^\pi [r(X_t, Y_t)] &= \mathbb{E}_s^\pi [\mathbb{E}^\pi [r(X_t, Y_t) | X_2 = s']] \\ &= \sum_{s' \in S} p(s' | s, \pi_1(s)) \mathbb{E}_{s'}^\pi [r(X_t, Y_t)]. \end{aligned}$$

Entonces reemplazando el valor esperado en la funcion valor:

$$v_{\alpha}^{\pi}(s) = r(s, \pi_1(s)) + \alpha \sum_{t=2}^{\infty} \alpha^{t-2} \sum_{s' \in S} p(s' | s, \pi(s)) \mathbb{E}_{s'}^{\pi} [r(X_t, Y_t)]$$

Se define $\pi' = (\pi_2, \pi_3, \dots)$. Para esta politica la siguiente expresion queda:

$$\sum_{t=2}^{\infty} \alpha^{t-2} \mathbb{E}_{s'}^{\pi'}(r(X_t, Y_t)) = v_{\alpha}^{\pi'}(s')$$

Luego reemplazando esta expresión en la función valor

$$v_{\alpha}^{\pi}(s) = r(s, \pi(s)) + \alpha \sum_{s' \in S} p(s' | s, \pi(s)) v_{\alpha}^{\pi'}(s')$$

□

La proposición muestra que el valor esperado para $s \in S$ depende de la retribución inmediata recibida más las recompensas futuras ponderadas por el factor de descuento α .

Sin embargo partiendo de la expresión (2.7) que evalúa el valor de una política fija se deduce que al tomar el supremo sobre todas las acciones disponibles en cada estado, se obtiene una formulación equivalente que caracteriza el valor óptimo.

Definición 2.13 (Ecuación de Bellman) *La ecuación de Bellman para el problema de decisión se define como:*

$$v(s) = \sup_{a \in \mathcal{A}} \left\{ r(s, a) + \alpha \sum_{s' \in S} p(s' | s, a) v(s') \right\}. \quad (2.8)$$

Esta ecuación constituye el núcleo teórico de la programación dinámica. Al descomponer el problema en decisiones locales seleccionando la mejor acción $a \in \mathcal{A}$, se obtiene una ecuación para cada estado $s \in S$, presentando así un sistema de ecuaciones lineales que puede ser abordado con herramientas de análisis funcional, como veremos a continuación.

Consideremos V como el espacio de funciones de utilidad acotadas con valores en los reales. Sobre

este espacio se define la norma

$$\|v\| = \sup_{s \in S} |v(s)|,$$

y la relación de orden se establece de acuerdo con lo expuesto en la sección 2.5. En particular, se demuestra que V es un espacio de Banach con esta norma (véase este resultado en el libro de Erwin y Kreyszig [Kre91]).

Sobre V se define el operador $L_\pi : V \rightarrow V$ como:

$$L_\pi(v)(s) = r(s, \pi(s)) + \alpha \sum_{s' \in S} p(s' | s, \pi(s))v(s')$$

Así, la definición del operador da lugar a la siguiente formulación para la ecuación 2.7:

$$v_\alpha^\pi(s) = L_\pi(v_\alpha^\pi)(s)$$

De forma similar, para reformular la ecuación 2.8, se define el operador $L : V \rightarrow V$ tal que:

$$L(v)(s) = \sup_{a \in A} \left\{ r(s, a) + \alpha \sum_{s' \in S} p(s' | s, a)v(s') \right\} \quad (2.9)$$

La definición del operador da lugar a la siguiente formulación para la ecuación (2.8):

$$v(s) = L(v)(s) \quad \forall s \in S \quad (2.10)$$

Un concepto clave aquí es el de **punto fijo**. Para ambos operadores, un punto fijo es un elemento $v \in V$ tal que $L(v)(s) = v(s)$ y $L_\pi(v)(s) = v(s)$. En particular, para la ecuación 2.7, si se utiliza una política estacionaria π^∞ , esta cumple que $\pi = \pi'$, por lo que:

$$L_\pi(v_\alpha^{\pi^\infty}) = v_\alpha^{\pi^\infty},$$

lo que establece que $v_\alpha^{\pi^\infty}$ es precisamente un punto fijo para el operador L_π . Este concepto será clave para la construcción de la solución óptima.

Si bien moralmente el problema planteado en (2.8) y el planteado en (2.6) no son el mismo, el

siguiente teorema presenta que las soluciones para ambos problemas son equivalentes.

Teorema 2.14 *Suponga que existe $v \in V$ tal que:*

a.- $v \geq L(v)$, entonces $v \geq v_\alpha$.

b.- $v \leq L(v)$, entonces $v \leq v_\alpha$.

c.- $v = L(v)$, entonces v es el único elemento de V con esta propiedad; además, $v = v_\alpha^*$.

Demostración: Consultar [Put14] pp. 148-149 para la demostración. □

2.7. Solución de la Ecuación de Optimalidad

Para abordar de manera rigurosa la solución de la ecuación de Bellman, es esencial introducir herramientas del análisis funcional, como el concepto de contracción y el **teorema del punto fijo de Banach**.

Definición 2.15 *Sea V un espacio de Banach. Se dice que el operador $T : V \rightarrow V$ es una contracción si existe $\alpha \in [0, 1)$ tal que*

$$\|T(v) - T(u)\| \leq \alpha \|v - u\|$$

para todo $u, v \in V$.

Teorema 2.16 (Teorema del Punto Fijo de Banach) *Sea V un espacio de Banach y $T : V \rightarrow V$ una contracción. Entonces:*

(a) Existe un único v en V tal que $T(v) = v$.

(b) Para cualquier $v^0 \in V$, la secuencia $v^n_{n \in \mathbb{N}}$ definida por:

$$v^{n+1} = T(v^n), \tag{2.11}$$

converge a v^* .

Demostración: Consultar [Put14] 150 o [Kre91] 300-302 para la demostración □

Entonces para hacer uso de este teorema lo primero que hay que demostrar es que el operado L definido en la sección anterior es un operador contractivo.

Proposición 2.17 *Sea L el operador definido en 2.10 es un operado contractivo sobre V .*

Demostración: Sean v y u elementos de V y consideramos de manera arbitraria $s \in S$ fijo. En primera instancia asumamos que $Lv(s) \geq Lu(s)$.

$$a_s^* \in \arg \max_{a \in A} \{r(s, a) + \alpha \sum_{s' \in S} p(s' | s, a)v(s')\}$$

Es decir:

$$Lv(s) = r(s, a_s^*) + \alpha \sum_{s' \in S} p(s' | s, a_s^*)v(s')$$

Por otro lado

$$Lu(s) \geq r(s, a_s^*) + \alpha \sum_{s' \in S} p(s' | s, a_s^*)u(s')$$

Entonces:

$$\begin{aligned} |Lv(s) - Lu(s)| &\leq r(s, a_s^*) + \alpha \sum_{s' \in S} p(s' | s, a_s^*)v(s') - r(s, a_s^*) - \alpha \sum_{s' \in S} p(s' | s, a_s^*)u(s') \\ &\leq \alpha \sum_{s' \in S} p(s' | s, a_s^*) [v(s') - u(s')] \\ &\leq \alpha \sum_{s' \in S} p(s' | s, a_s^*) \sup_{s \in S} |v(s) - u(s)| \\ &= \alpha \|v - u\| \end{aligned}$$

De manera análoga se puede hacer para el caso en que $Lu(s) \geq Lv(s)$. Con ambos casos se obtiene que para todo $s \in S$:

$$|Lv(s) - Lu(s)| \leq \alpha \|v - u\|$$

Tomando el supremo sobre S en el lado izquierdo se concluye que L es una contracción. □

Con estas este resultado se enuncia el siguiente teorema fundamental.

Teorema 2.18 *Suponga $\alpha \in [0, 1)$, S finito contable y $r(s, a)$ acotada, entonces existe v^* que satisface $Lv^* = v^*$. Es más v^* es el unico elemento en V que cumple con esta propiedad y que además $v^* = v_\alpha^*$*

Demostración: Ya que V es un espacio Banach la proposición 2.17 establece que L es una contracción por lo que es aplicable el teorema 2.16. Esto implica la existencia de una unica solución que nombraremos $v^* \in V$ para $Lv = v$. Por ultimo por el teorema 2.14(c) establece que $v^* = v_\alpha^*$ □

Observación 2.19 *El Teorema 2.18 es uno de los resultados más importantes de este trabajo, ya que garantiza la existencia y unicidad de la función de valor óptima v^* como solución de la ecuación de Bellman $Lv^* = v^*$. Este resultado es clave para justificar la convergencia del método iterativo de Value Iteration, pues al establecer que L al ser una contracción en el espacio de funciones V , asegura que la sucesión generada por iteraciones sucesivas de L converge a la única solución v^* , validando así el uso del método para determinar políticas óptimas.*

Capítulo 3

Algoritmos de Optimización e implementaciones

3.1. Algoritmo de Iteración de Valores

El *algoritmo de iteración de valores* es un método para resolver problemas de **Procesos de Decisión de Markov** (MDP). Este algoritmo tiene como objetivo calcular la función de valor óptimo $v^*(s)$ a cual describe el valor máximo esperado que se puede alcanzar desde cualquier estado s al seguir una política óptima.

Recordar que un MDP se define por los siguientes elementos: el conjunto de estados S , el conjunto de acciones A , la función de transición $p(s' | s, a)$, la función de retribuciones $r(s, a)$ y el factor de descuento $\alpha \in [0, 1)$. El problema consiste en encontrar la función de valor $v(s)$ que satisface la **ecuación de Bellman**, expresada como:

$$v(s) = \max_{a \in A} \left(r(s, a) + \alpha \sum_{s'} p(s' | s, a) v(s') \right) \quad (3.1)$$

El algoritmo de iteración de valores utiliza esta ecuación para actualizar de manera iterativa las estimaciones de $v(s)$ hasta que estas converjan a la función de valor óptima $v^*(s)$. Una vez obtenida $v^*(s)$, se puede derivar directamente la política óptima $\pi^*(s)$, definida como:

$$\pi(s) = \arg \max_{a \in A} \left(r(s, a) + \alpha \sum_{s'} p(s' | s, a) v(s') \right). \quad (3.2)$$

Las instrucciones básicas para el algoritmo de iteración de valores es el siguiente:

Algorithm 1: Iteración de Valores

Input: MDP $(\mathcal{S}, \mathcal{A}, P, R, \alpha)$, tolerancia ϵ
Output: Secuencia de vectores de utilidad v_n

- 1 Inicializar $v_0(s) \leftarrow 0$ para cada $s \in \mathcal{S}$;
- 2 $\delta \leftarrow +\infty$;
- 3 $n \leftarrow 0$;
- 4 **while** $\delta \geq \epsilon \cdot \frac{1-\alpha}{\alpha}$ **do**
- 5 $v_{n+1}(s) \leftarrow v_n(s)$ para todo $s \in \mathcal{S}$;
- 6 **foreach** $s \in \mathcal{S}$ **do**
- 7 $q \leftarrow \max_{a \in \mathcal{A}(s)} \left[r(s, a) + \alpha \sum_{s' \in \mathcal{S}} p(s', a, s') \cdot v_n(s') \right]$;
- 8 $\delta \leftarrow \min(\delta, |q - v_n(s)|)$;
- 9 $v_{n+1}(s) \leftarrow q$;
- 10 $n \leftarrow n + 1$;
- 11 **return** v_{n+1}

Una vez obtenido el valor optimo el siguiente algoritmo rescata la politica optima obtenida al finalizar la iteración de Valores:

Algorithm 2: Política Óptima a partir de Valores de Utilidad

Input: MDP $(\mathcal{S}, \mathcal{A}, P, R, \alpha)$, vector de valores optimos v^*
Output: Política π

- 1 Inicializar la política π como una asignación vacía;
- 2 **foreach** estado $s \in \mathcal{S}$ **do**
- 3 $a^* = \arg \max_{a \in \mathcal{A}(s)} \{ r(s, a) + \alpha \sum_{s' \in \mathcal{S}} p(s' | s, a) \cdot v^*(s') \}$;
- 4 $\pi(s) \leftarrow a^*$;
- 5 **return** π

El algoritmo mencionado cuenta con una implementación en código Julia, disponible en el Apéndice A.

Bajo las hipótesis $\alpha \in [0, 1)$, espacio de estados $\mathcal{S}$ contable y función de recompensas $|r(s, a)|$ acotada, el siguiente teorema garantiza la convergencia del método de iteración de valores:

Teorema 3.1 Sea $v_0 \in V$, $\epsilon > 0$ y sea v_n tal que $v_{n+1} = Lv_n$, entonces v_n converge en norma a v_α^* .

Demostración: La demostración es una consecuencia directa del Teorema 2.18. □

Este resultado justifica la validez del algoritmo para la obtención de la función de valor óptima. Sin embargo, a medida que el tamaño del espacio de estados o el conjunto de acciones crece, el costo computacional del algoritmo aumenta considerablemente. Esto se debe a la necesidad de actualizar cada estado en cada iteración, lo que vuelve el método poco práctico para problemas de alta dimensión o sistemas con un gran número de estados.

Para mitigar estas limitaciones, es común recurrir a implementaciones optimizadas o explorar alternativas más eficientes, pero que exceden los objetivos del presente proyecto.

3.1.1. Ejemplo de aplicación

Consideremos un robot en un entorno cerrado de dimensiones 4×3 casillas, como se muestra en la 3.1(a) donde la celda $(2, 2)$ es una pared por donde este no puede pasar. Comenzando en el estado inicial, el robot debe elegir una acción en cada paso de tiempo. La interacción con el entorno finaliza cuando el agente alcanza uno de los estados objetivo, marcados como $+1$ o -1 .

Para modelar este problema como un proceso de decisión de Markov (MDP), definimos los siguientes elementos:

- **Estados** S : El entorno 4×3 consta de 12 celdas, pero la celda $(2, 2)$ está ocupada por un pilar y no es accesible, por lo que el conjunto de estados se reduce a 11 posiciones alcanzables. Además, existen dos estados terminales con recompensas de $+1$ y -1 , los cuales finalizan el recorrido del robot al ser alcanzados.
- **Acciones** $A(s)$: En cada estado s , el robot tiene cuatro posibles movimientos: *Norte*, *Sur*, *Este* y *Oeste*. Estas acciones conforman el conjunto $A(s)$ para cada estado.
- **Probabilidades de Transición** $(P(s'|s, a))$: Las transiciones del robot pueden presentar desvíos, por lo que se definen de manera estocástica de la siguiente forma:

Avanza Con probabilidad 0.8, se mueve una casilla hacia la dirección seleccionada

Desvío El robot en lugar realizar la acción seleccionada, este avanza hacia la izquierda de la dirección seleccionada con probabilidad 0.1 o a la derecha con probabilidad 0.1

Pared o Pilar Si el robot intenta moverse hacia una pared del entorno o hacia la celda (2,2), que está bloqueada por un pilar, permanecerá en el mismo estado con probabilidad 0.8, mientras que con probabilidad 0.1 se desviara hacia la izquierda y con probabilidad 0.1 hacia la derecha.

■ **Recompensas** ($R(s, a)$):

- El estado (4, 3) entrega una recompensa de valor +1 cuando el robot llega a este estado.
- El estado (4, 2) entrega una penalización de valor -1 cuando el robot llega a este estado.
- En las demás transiciones se asigna una recompensa nula o una leve penalización negativa para incentivar al agente a alcanzar los objetivos de manera eficiente. En este ejemplo, se utiliza una penalización por transición entre celdas de valor -0.04 unidades.

■ **Factor de Descuento** (α): Este parámetro regula la importancia de las recompensas. Con un α cercano a 0, el agente favorece las recompensas inmediatas, ya que las futuras tienen menor peso. En cambio, un α cercano a 1 asigna casi la misma relevancia a recompensas las futuras que a las inmediatas. En este ejemplo se utilizan dos valores de α , 0.9 y 0.2 para comparar su efecto en la política que entrega el algoritmo.

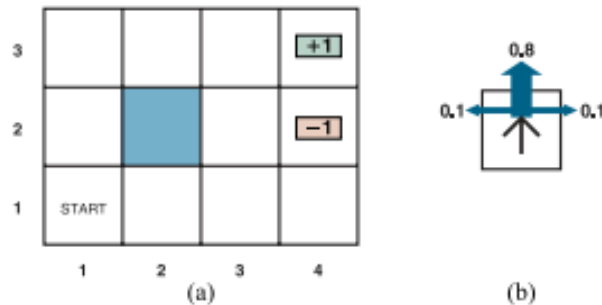


Figura 3.1: Entorno 4×3 y modelo de movimiento estocástico.

El factor de descuento en la ecuación de Bellman puede interpretarse como un parámetro que regula la importancia de las recompensas futuras en las decisiones del agente. Al ajustar este factor, se puede controlar el comportamiento del agente frente a los riesgos asociados con sus acciones. Un factor de descuento bajo hará que el agente dé mayor prioridad a las recompensas inmediatas, evitando tomar decisiones que impliquen altos riesgos para el futuro. En cambio, un valor alto del factor de descuento fomenta una mayor disposición a asumir riesgos, ya que el agente valorará más las recompensas a largo

plazo, lo que podría llevarlo a tomar rutas menos seguras pero con mayor potencial de beneficio en el futuro. El siguiente ejemplo muestra como opera esta propiedad en un ejemplo específico.

Solución camino optimo para recorrido del robot.

Se busca determinar una ruta optima que maximice la recompensa esperada del robot al hacer su recorrido. El algoritmo de **Iteración de Valores** presentado en la sección 3.1 retorna una politica que entrega a cada estado/posicion del robot la dirección en la cual se maximiza la recompensa total esperada.

El resultado obtenido para una política calculada con un factor de descuento $\alpha = 0.9$ se presenta en la Figura 3.2. Con este valor alto de α , el agente valora de forma similar las recompensas inmediatas y futuras, lo que lo impulsa a dirigirse directamente al estado objetivo $[4, 3]$, que otorga una recompensa de $+1$. Es decir, la política es directa y se enfoca a alcanzar el objetivo lo más rápido posible.

Por otro lado, la Figura 3.3 muestra la política óptima cuando se utiliza un factor de descuento $\alpha = 0.1$. Con un α tan bajo, el agente da mayor relevancia a las recompensas inmediatas. En consecuencia, aunque sigue intentando llegar al estado objetivo, la política prefiere evitar transitar por el estado $[4, 2]$, donde se aplica una penalización de -1 . Esto se debe a que al estar cerca de las penalización inmediata esta pesa significativamente en comparación de una recompensa futura.

Esta representación refleja cómo el algoritmo de iteración de valores calcula una política óptima que maximiza las recompensas acumuladas para el agente, considerando el factor de descuento y la estructura de recompensas del entorno.

Observación 3.2 *Notar que si el entorno fuera determinista (sin transiciones aleatorias entre celdas), una secuencia específica de acciones llevaría al agente al objetivo $+1$ de manera óptima. En este caso se puede tomar la ruta dada por la secuencia de acciones: Norte, Norte, Este, Este, Este.*

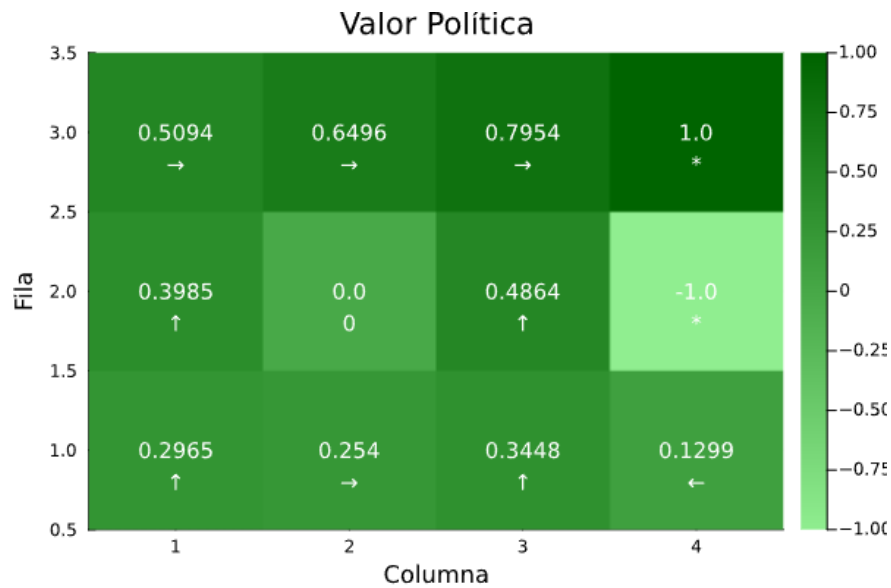


Figura 3.2: Política óptima calculada mediante el algoritmo de Iteración de Valores con factor de descuento $\alpha = 0.9$. Los valores en cada celda indican la recompensa acumulada esperada desde ese estado, y las flechas señalan la acción óptima para el robot.

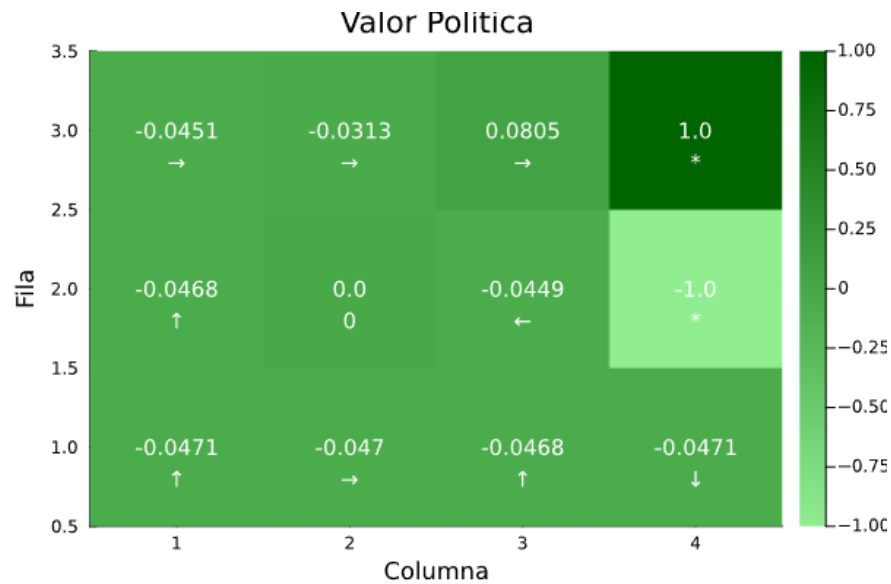


Figura 3.3: Política óptima calculada mediante el algoritmo de Iteración de Valores con factor de descuento $\alpha = 0.1$. Los valores en cada celda indican la recompensa acumulada esperada desde ese estado, y las flechas señalan la acción óptima para el robot.

Capítulo 4

Modelos para tiempos de vida

En este capítulo, se presentarán el concepto de sistema coherente, su modelación mediante funciones estructurales, y los conceptos de relevancia e irrelevancia de componentes dentro de un sistema. Asimismo, se abordará el concepto de fiabilidad y se introducirá la función de confiabilidad, el cual pretende cuantificar este concepto. Posteriormente, se presentará la distribución Marshall-Olkin y se presentarán sus propiedades más importantes. Por último, se describen los modelos Lévy Frailty Marshall Olkins (LFMO), un caso particular de la distribución Marshall-Olkin, para modelar el tiempo de vida de las componentes, cuya principal ventaja radica en su menor complejidad paramétrica en comparación con el modelo original.

Para modelar los tiempos de vida de las componentes, un punto de partida intuitivo es suponer que estas se distribuyen de manera independiente e idéntica (IID) siguiendo una distribución exponencial. Este modelo es ampliamente utilizado debido a su simplicidad matemática y a la propiedad de “pérdida de memoria”, que facilita su tratamiento analítico. Sin embargo, este enfoque presenta una limitación significativa: bajo este supuesto la probabilidad de fallos simultáneos entre componentes es nula, lo cual puede no reflejar adecuadamente la realidad y la dinámica del sistema donde existe dependencia implícita entre componentes.

Para superar esta restricción y representar posibles interdependencias entre componentes, se recurre al modelo multivariado de Marshall-Olkin [MO67], que permite representar de forma natural eventos de fallos conjuntos siendo más realista.

4.1. Conformación de los Sistemas Coherentes

Los sistemas son estructuras construidas utilizando varias componentes. En el análisis de fiabilidad, un sistema está compuesto por componentes cuya interacción determina su operatividad global. Este comportamiento se modela mediante una **función estructural**, que vincula el estado de las componentes con el estado del sistema.

Definición 4.1 (Función Estructural) *Un sistema (binario) con componentes (binarias) de orden n es una función estructural booleana*

$$\phi : \{0, 1\}^n \rightarrow \{0, 1\} \quad (4.1)$$

donde $\phi(x_1, \dots, x_n) \in \{0, 1\}$ representa el estado del sistema, determinado por los estados de los componentes $x_1, \dots, x_n \in \{0, 1\}$.

Ejemplo 4.2 *En la Figura 4.1 se presenta un sistema de 3 componentes que se representaran bajo el simbolismo de x_1, x_2 y x_3 respectivamente. La estructura tiene dos secciones en paralelo la primera, con una componente (x_1) y la segunda con dos componentes conectadas en serie (x_2, x_3). Suponiendo la representación binaria de las componentes x_i , se puede representar la interacción de estar en paralelo utilizando la función $\max(\cdot, \cdot)$ y estar conectadas en serie mediante la función $\min(\cdot, \cdot)$. Bajo esta representación se deduce la siguiente función estructural para las condiciones del sistema:*

$$\phi(x) = \max(x_1, \min(x_2, x_3)) \quad (4.2)$$

Este sistema se ilustra en la Figura 4.1.

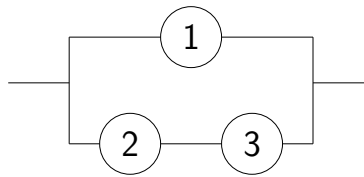


Figura 4.1: Sistema de 3 Componentes. Las componentes 1, 2 y 3 están conectadas de manera que el fallo del sistema depende del fallo de la componente 1 y de la componente 2 o 3.

Con el fin de buscar representar sistemas “realistas”, es natural asumir algunas propiedades adicionales para la función estructural que se introducen a continuación:

Definición 4.3 (Componente Irrelevante) Se dice que un componente i -ésimo es irrelevante para el sistema ϕ si

$$\phi(x_1, \dots, x_{i-1}, 0, x_{i+1}, \dots, x_n) = \phi(x_1, \dots, x_{i-1}, 1, x_{i+1}, \dots, x_n) \quad (4.3)$$

para todo $x_1, \dots, x_{i-1}, x_{i+1}, \dots, x_n \in \{0, 1\}$. En caso contrario, se dice que esta componente es relevante. En particular una componente irrelevante no afecta al estado global del sistema.

Definición 4.4 (Sistemas Semi-coherentes y Coherentes) Un sistema se llama semi-coherente de orden n si satisface las siguientes propiedades:

(i) ϕ es creciente:

$$\phi(x_1, \dots, x_n) \leq \phi(y_1, \dots, y_n)$$

para todo $x_1 \leq y_1, \dots, x_n \leq y_n$.

(ii) $\phi(0, \dots, 0) = 0$ y $\phi(1, \dots, 1) = 1$

Se dice que el sistema es coherente de orden n si la condición (ii) es reemplazada por:

(ii') ϕ es estrictamente creciente en cada variable en al menos un punto.

Por lo tanto, la idea detrás de un sistema coherente es que todos los componentes contribuyen de manera “consistente” o “coherente” a la fiabilidad general del sistema siendo esto la propiedad mínima requerida para trabajar.

En un sistema coherente la confiabilidad se entiende como la probabilidad de que el sistema mantenga su funcionamiento durante un periodo determinado. La confiabilidad se determina a partir de sus componentes y su función estructural. Cada componente puede encontrarse en estado operativo o inoperativo, y el estado del sistema se define a partir de la función estructural ϕ descrita en (4.1). El tiempo de vida del sistema depende tanto de la vida útil de sus componentes como de la forma en que estos afectan la operatividad del sistema, es decir, su función estructural.

Definición 4.5 (Tiempo de vida del sistema) Se denomina tiempo de vida del sistema al momento

aleatorio en el cual es sistema deja de funcionar. Esto se representa por $\mathbf{T}_{sis}$.

$$\mathbf{T}_{sis} = \inf\{t \geq 0 : \phi(X(t)) = 0\}.$$

Para cada componente i , se define se tiempo de vida como T_i como.

$$T_i = \inf\{t \geq 0 : X_i(t) = 0\}$$

Definición 4.6 Sea ϕ una función estructural, como la definida en (4.1). La función de confiabilidad del sistema, denotada por $R(t)$ se define como la probabilidad de que el sistema esté operativo en el tiempo t es decir:

$$R(t) = \mathbb{P}(\mathbf{T}_{sis} > t). \quad (4.4)$$

Además, $R(t)$ puede expresarse en términos de la función estructural ϕ y los estados de las componentes como:

$$R(t) = \mathbb{P}(\phi(X(t)) = 1) = \mathbb{E}[\phi(X(t))]$$

4.2. Distribución Marshall-Olkin

Dado que el problema en estudio incorpora elementos de incertidumbre, resulta natural adoptar modelos estocásticos para su estudio. En este contexto, se trabaja en un espacio de probabilidad filtrado $(\Omega, \mathcal{F}, (\mathcal{F}_t)_{t \geq 0}, \mathbb{P})$, donde se definen todas las variables y procesos aleatorios relevantes para el análisis del sistema analizado.

Sea C un conjunto de componentes. Definimos para cada componente i el conjunto

$$S(i) := \{S \subset 2^C : i \in S\}.$$

Para cada subconjunto $S \in S(i)$, V_S representa el tiempo en el que ocurre un "shock" que deja

inoperativas todas las componentes en S . Así, el tiempo de vida de la componente i , se define como:

$$T_i = \min_{S \in S(i)} V_S$$

En este caso se dice que el vector aleatorio $T = (T_1, \dots, T_n)$ sigue una distribución Marshall-Olkin, parametrizada por los valores λ_S para cada $S \subseteq 2^C$. Bajo esta construcción, se obtiene la siguiente proposición.

Proposición 4.7 *Marginalmente, T_i se distribuye como $\text{Exp}\left(\sum_{S \subseteq S(i)} \lambda_S\right)$.*

Demostración: La función de distribución acumulada de T_i es:

$$\mathbb{P}(T_i < t) = \mathbb{P}\left(\min_{S \in S(i)} V_S < t\right) = 1 - \mathbb{P}\left(\min_{S \in S(i)} V_S \geq t\right)$$

Calculando la probabilidad del minimo se obtiene:

$$\mathbb{P}\left(\min_{S \in S(i)} V_S \geq t\right) = \mathbb{P}(V_S \geq t \text{ Para todo } S \in S(i))$$

Como los V_S son exponenciales independientes de parámetro λ_S se tiene que:

$$\mathbb{P}\left(\min_{S \in S(i)} V_S \geq t\right) = \prod_{S \in S(i)} \mathbb{P}(V_S \geq t) = \prod_{S \in S(i)} e^{-\lambda_S t} = e^{-t \sum_{S \in S(i)} \lambda_S}$$

Por lo tanto

$$\mathbb{P}(T_i \leq t) = 1 - e^{-t \sum_{S \in S(i)} \lambda_S}$$

□

El modelo Marshall-Olkin es particularmente útil por su capacidad de modelar fallos simultáneos de componentes, como se enuncia en el siguiente lema:

Lema 4.8 *Sean i y j elementos distintos de C . La probabilidad de que $T_i = T_j$ es estrictamente mayor que cero.*

Demostración: Por definición

$$\mathbb{P}(T_i = T_j) = \mathbb{P}\left(\min_{S \subset S(i)} V_S = \min_{S \subset S(j)} V_S\right)$$

Si existe un subconjunto S tal que $S \in S(i) \cap S(j)$ entonces este evento ocurre. Se considera en tal caso el conjunto:

$$S(i, j) := \{S \text{ tales que } S \in S(i) \cap S(j)\}$$

Por lo tanto:

$$\mathbb{P}\left(\min_{S \subset S(i)} V_S = \min_{S \subset S(j)} V_S\right) = \mathbb{P}\left(\min_{S \subset S(i, j)} V_S\right)$$

y dado que $S(i, j)$ es no vacío (ya que en particular se tiene el subconjunto $S = \{i, j\}$), es inmediato que:

$$\mathbb{P}\left(\min_{S \subset S(i, j)} V_S\right) > 0$$

□

A pesar de sus ventajas, el modelo Marshall-Olkin, presenta una alta complejidad paramétrica en sistemas de gran dimensión, ya que requiere un parámetro λ_S para cada subconjunto $S \subseteq C$, es decir este modelo tiene una complejidad paramétrica exponencial en relación con el número de componentes. Esto puede resultar prohibitivo en términos almacenamiento y obtención de datos.

Para abordar esta limitación, se considera la sub-familia Lévy-Fraily Marshall-Olkin (LFMO), que como se vera a continuación, reduce la complejidad paramétrica del modelo.

4.3. Lévy-Fraily Marshall-Olkin

En esta sección se introduce el modelo propuesto en Mai y Scherer [MS09] para la degradación estocástica acumulada de un sistema, caracterizada por incrementos discontinuos en instantes y magnitudes aleatorias, modelados mediante procesos estocásticos. Este modelo se basa en las propiedades de los subordinadores de Lévy, los cuales pertenecen a la clase de procesos de Lévy no decrecientes.

Los procesos de Lévy generalizan modelos estocásticos clásicos, como el movimiento Browniano y el proceso de Poisson. Estos permiten modelar sistemas con dinámicas complejas que incluyen tanto

saltos esporádicos como degradación continua del sistema.

4.3.1. Procesos de Lévy

Se presentarán los procesos de Lévy siguiendo lo presentado por Bertoin en su libro Lévy Processes [Ber96]. Antes de introducir estos procesos, definiremos el concepto de variable aleatoria infinitamente divisible.

Definición 4.9 *Una variable aleatoria X es infinitamente divisible si para todo $n \in \mathbb{N}$ se puede descomponer como:*

$$X \stackrel{d}{=} X_1 + \dots + X_n \quad (4.5)$$

donde $\stackrel{d}{=}$ indica equivalencia en distribución.

Para un proceso de Lévy X_t , esta propiedad implica que puede representarse como incrementos independientes e identificativamente distribuidos en intervalos de tiempo $\frac{t}{n}$:

$$X_t = \sum_{k=1}^n (X_{\frac{kt}{n}} - X_{\frac{(k-1)t}{n}}) \quad (4.6)$$

donde $X_0 = 0$ y los incrementos $(X_{\frac{kt}{n}} - X_{\frac{(k-1)t}{n}})$ son variables aleatorias i.i.d.

En tal caso para un proceso de Lévy, debido a 4.6 la expresión de su función característica está directamente relacionado con el tiempo t y el exponente característico de ψ_1 en X_1 :

$$\psi_t(\theta) = -\log(\mathbb{E}[e^{i\theta X_t}]) = t\psi_1(\theta). \quad (4.7)$$

Esta propiedad facilita los cálculos y será utilizada en algunas partes de esta sección. A continuación, se presentan dos procesos estocásticos conocidos que sirven como base para construir un ejemplo de proceso de Lévy.

Definición 4.10 *Un proceso con valores reales $B = \{B_t : t \geq 0\}$ se denomina movimiento Browniano*

si cumple con las siguientes propiedades:

1. $\mathbb{P}(B_0 = 0) = 1$.
2. Las trayectorias de B son continuas $\mathbb{P}$ -casi seguramente.
3. B tiene incrementos independientes; es decir, $B_t - B_s$ es independiente de B_u para todo $0 \leq u \leq s \leq t$.
4. Para todo $t > s$, $B_t - B_s \sim \mathcal{N}(0, t - s)$, lo que implica que $B_t - B_s$ tiene la misma distribución que B_{t-s} .

Definición 4.11 Un proceso con valores enteros no negativos $N = \{N_t : t \geq 0\}$ se denomina proceso de Poisson de intensidad $\lambda > 0$ si cumple con las siguientes propiedades:

1. $\mathbb{P}(N_0 = 0) = 1$.
2. Las trayectorias de N son $\mathbb{P}$ -casi seguramente continuas por la derecha y con límites por la izquierda.
3. N tiene incrementos independientes; es decir, $N_t - N_s$ es independiente de N_u para todo $0 \leq u \leq s \leq t$.
4. Para todo $t > s$, $N_t - N_s \sim \mathcal{P}(\lambda(t - s))$, lo que implica que $N_t - N_s$ tiene la misma distribución que N_{t-s} .

Es claro que los procesos de Poisson y el movimiento browniano presentan tanto similitudes como diferencias. Ambos procesos comienzan en cero casi seguramente y tienen incrementos independientes y estacionarios. Sin embargo, mientras que las trayectorias del movimiento browniano son continuas, los procesos de Poisson exhiben discontinuidades en forma de saltos. Además, los incrementos estacionarios de ambos procesos siguen distribuciones diferentes: en el caso del Poisson, las trayectorias son crecientes con incrementos controlados, mientras que en el movimiento browniano los incrementos estacionarios, al seguir una distribución normal, sus trayectorias no pueden ser monótonas y su variación en un intervalo de tiempo finito es no acotada.

Sin embargo las características compartidas de ambos procesos los ubican dentro de una familia más amplia de procesos estocásticos: los procesos de Lévy. Estos procesos generalizan el movimiento browniano al combinar variación continua con saltos discretos.

Definición 4.12 *Un proceso $X = \{X_t : t \geq 0\}$ se denomina proceso de Lévy si posee las siguientes propiedades:*

1. $\mathbb{P}(X_0 = 0) = 1$.
2. *Las trayectorias de X son $\mathbb{P}$ -casi seguramente continuas por la derecha y con límites por la izquierda.*
3. *X tiene incrementos independientes; es decir, $X_t - X_s$ es independiente de X_u para todo $0 \leq u \leq s \leq t$*
4. *Para todo $t \geq s$, $X_t - X_s$ tiene la misma distribución que X_{t-s} .*

En general, los procesos de Lévy no poseen una expresión explícita para sus funciones de densidad. Una herramienta fundamental para estudiar estos procesos es la función característica, lo cual resulta muy útil para analizar distribuciones sin forma analítica explícita.

Dada una variable aleatoria X y su medida de probabilidad μ en $\mathbb{R}$, la función característica se define como una transformación μ mediante la expresión:

$$\phi_X(\theta) = \int_{\mathbb{R}} e^{i\theta x} \mu(dx), \quad \theta \in \mathbb{R}. \quad (4.8)$$

Esta función permite describir diversas propiedades de la distribución de X , incluso cuando no se dispone de su densidad de probabilidad explícita.

Para ciertos procesos estocásticos, incluyendo los procesos de Lévy, la función característica puede representarse de forma más conveniente mediante el exponente característico, definido como:

$$\psi(\theta) = -\log(\phi_X(\theta)) \quad (4.9)$$

Bajo la fórmula de Lévy-Khintchine [Ber96, Cap. 1, p.17], si X es un proceso de Lévy su función característica se puede expresar mediante su exponente característico $\psi(\theta)$ de la siguiente forma:

$$\psi(\theta) = ia\theta + \frac{1}{2}\sigma^2\theta^2 + \int_{\mathbb{R}} (1 - e^{i\theta x} + i\theta x \mathbb{1}_{|x|<1}) \nu(dx), \quad a, \sigma \in \mathbb{R}, \quad (4.10)$$

donde el triplete de elementos (a, σ, ν) son únicos. Estos elementos cumplen que: $a, \sigma \in \mathbb{R}$ y ν es una medida en $\mathbb{R}$ que satisface que la condición de inteligibilidad siguiente:

$$\int_{\mathbb{R}} (1 \wedge x^2) \nu(dx) < \infty, \quad \nu(0) = 0. \quad (4.11)$$

Exploraremos algunos procesos de Lévy utilizando esta propiedad.

Ejemplo 4.13 (Proceso de Poisson) Consideremos un proceso de Poisson N_t con parámetro $\lambda > 0$. El exponente característico de N_t es:

$$\psi_t(\theta) = t\lambda(1 - e^{i\theta}) \quad (4.12)$$

Si lo comparamos con la fórmula de Lévy-Khintchine:

$$\psi(\theta) = \underbrace{i\theta a}_{a=0} + \underbrace{\frac{1}{2}\sigma^2\theta^2}_{\sigma^2=0} + \int_{\mathbb{R}} (1 - e^{i\theta x} + i\theta x \mathbb{1}_{|x|<1}) \underbrace{\nu(dx)}_{\lambda\delta_1} \quad (4.13)$$

donde δ_1 es la distribución delta de Dirac con peso en 1.

Ejemplo 4.14 Consideremos un movimiento browniano B_t . Su exponente característico es:

$$\psi_t(\theta) = \frac{t\theta^2}{2}. \quad (4.14)$$

Luego en la comparativa con la formula de Lévy-Khintchine:

$$\psi(\theta) = \underbrace{i\theta a}_{a=0} + \underbrace{\frac{1}{2}\sigma^2\theta^2}_{\sigma^2=1} + \int_{\mathbb{R}} (1 - e^{i\theta x} + i\theta x \mathbb{1}_{|x|<1}) \underbrace{\nu(dx)}_0 \quad (4.15)$$

En este contexto, surge una clase especial de procesos de Lévy que es fundamental en la definición de una distribución LFMO: los subordinadores de Lévy, caracterizados por tener trayectorias no decrecientes.

Definición 4.15 Sea un proceso estocástico $\Lambda = \{\Lambda_t : t \geq 0\}$. Se dice que Λ_t es un subordinador de Lévy si cumple las siguientes propiedades:

1. Es un proceso de Lévy
2. Sus trayectorias son no decrecientes, es decir $\Lambda_t \geq \Lambda_s$ para todo $t \geq s \geq 0$.

Un ejemplo clásico de subordinador es el proceso de Poisson compuesto, donde los saltos están determinados por una distribución específica:

Ejemplo 4.16 Sea $\{N_t : t \geq 0\}$ un proceso de Poisson con tasa $\lambda > 0$, y sea $\{J_i\}_{i=1}^{\infty}$ una secuencia de variables aleatorias independientes entre sí y de N_t , idénticamente distribuidas con valores en $\mathbb{R}^+$. Se define el proceso Λ_t como:

$$\Lambda_t = \sum_{i=1}^{N_t} J_i, \quad t \geq 0$$

En términos simples, Λ_t es una suma aleatoria de variables aleatorias, que como veremos a continuación corresponde a un subordinador de Lévy. Primero, dado que las variables aleatorias J_i son no negativas por construcción, es inmediato que el proceso Λ_t es creciente.

Para Λ_t , su función característica se calcula como sigue:

$$\begin{aligned}
\phi_{\Lambda_t}(\theta) &= \mathbb{E}(e^{i\theta\Lambda_t}) \\
&= \mathbb{E}\left(e^{i\theta\sum_{i=1}^{N_t} J_i}\right) \\
&= \mathbb{E}\left(\mathbb{E}\left(e^{i\theta\sum_{i=1}^n J_i} \mid N_t = n\right)\right) \\
&= \mathbb{E}\left(\mathbb{E}\left(e^{i\theta J_1}\right)^n \mid N_t = n\right) \\
&= \mathbb{E}\left((\phi_{J_1}(\theta))^{N_t}\right) \\
&= \sum_{n=0}^{\infty} (\phi_{J_1}(\theta))^n \frac{e^{-\lambda t} (\lambda t)^n}{n!} \\
&= e^{-\lambda t} \left(\sum_{n=0}^{\infty} \frac{(\phi_{J_1}(\theta) \lambda t)^n}{n!} \right) = e^{-\lambda t + \phi_{J_1}(\theta) \lambda t} = e^{\lambda t (\phi_{J_1}(\theta) - 1)},
\end{aligned}$$

donde se ha utilizado: la definición de Λ_t en las dos primeras igualdades, la propiedad de la torre para esperanzas condicionales en la tercera, la independencia de las variables J_i en la cuarta, y la distribución de N_t como proceso de Poisson en las siguientes.

Ahora sean $t_4 \geq t_3 \geq t_2 \geq t_1 \geq 0$. Se busca calcular la función característica conjunta de los incrementos:

$$\begin{aligned}
\phi_{(\Lambda_{t_4}-\Lambda_{t_3}, \Lambda_{t_2}-\Lambda_{t_1})}(\theta_1, \theta_2) &= \mathbb{E}\left(e^{i(\theta_1(\Lambda_{t_4}-\Lambda_{t_3})+\theta_2(\Lambda_{t_2}-\Lambda_{t_1}))}\right) \\
&= \mathbb{E}\left(e^{i\theta_1(\Lambda_{t_4}-\Lambda_{t_3})} e^{i\theta_2(\Lambda_{t_2}-\Lambda_{t_1})}\right) \\
&= \mathbb{E}\left(\mathbb{E}\left(e^{i\theta_1(\Lambda_{t_4}-\Lambda_{t_3})} e^{i\theta_2(\Lambda_{t_2}-\Lambda_{t_1})}\right) \mid N_{t_i} = n_i, i = \{1, 2, 3, 4\}\right) \\
&= \mathbb{E}\left((\phi_{J_1}(\theta_1))^{n_4-n_3} (\phi_{J_1}(\theta_2))^{n_2-n_1} \mid N_{t_i} = n_i, i = \{1, 2, 3, 4\}\right) \\
&= \mathbb{E}\left((\phi_{J_1}(\theta_1))^{N_{t_4}-N_{t_3}} (\phi_{J_1}(\theta_2))^{N_{t_2}-N_{t_1}}\right) \\
&= \mathbb{E}\left((\phi_{J_1}(\theta_1))^{N_{t_4}-N_{t_3}}\right) \mathbb{E}\left((\phi_{J_1}(\theta_2))^{N_{t_2}-N_{t_1}}\right) \\
&= e^{\lambda(t_4-t_3)(\phi_{J_1}(\theta_1)-1)} e^{\lambda(t_2-t_1)(\phi_{J_1}(\theta_2)-1)},
\end{aligned}$$

donde se ha usado la independencia de los incrementos de N_t para factorizar la esperanza conjunta y la propiedad de estacionalidad de los incrementos para calcular las expectativas de cada término.

El cálculo realizado para los tiempos específicos $t_4 \geq t_3 \geq t_2 \geq t_1 \geq 0$ se puede generalizar para

cualquier conjunto de tiempos ordenados, gracias a la independencia de los incrementos de N_t y la propiedad de estacionalidad de los incrementos. Esto indica que para cualquier conjunto de tiempos, los incrementos de Λ_t serán independientes y estacionarios, y las trayectorias seguirán siendo crecientes. Por lo tanto, se puede concluir que Λ_t es un subordinador de Lévy.

4.3.2. Distribución LFMO

Sea $(\Lambda_t : t \geq 0)$ un subordinador de Lévy. Se define la secuencia de variables aleatorias $\{\epsilon_i\}_{i \in C}$, las cuales son independiente e idénticamente distribuida (IID) y además independiente del proceso Λ_t . Cada variable ϵ_i sigue una distribución exponencial de parámetro 1.

Se dice que un vector aleatorio $T = (T_1, \dots, T_n)$ en $\mathbb{R}^n$ sigue una distribución Lévy-Fraily Marshall-Olkin (LFMO) cuando, para cada $i = 1, \dots, n$, se cumple que

$$T_i = \min\{t \geq 0 : \Lambda_t \geq \epsilon_i\}.$$

Cada ϵ_i actúa como un “trigger” correspondiente a la componente i . La distribución LFMO es una versión particular de la distribución Marshall-Olkin, con parámetros definidos por:

$$\lambda_S = \sum_{j=0}^{|S|-1} \binom{|S|-1}{j} (-1)^j (\psi(n - |S| + j + 1) - \psi(n - |S| + j)), \quad (4.16)$$

En esta formulación, la tasa λ_S depende únicamente del tamaño de S , lo cual, a diferencia de la distribución Marshall-Olkin, implica que cada subconjunto de igual cardinalidad puede estar asociado a diferentes tasas de muerte. Esta formulación se puede encontrar la con un desarrollo teórico completo en [Mai10], mientras que una descripción más directa de la LFMO se presenta en [HM12]. Aquí, $\psi(x) = -\log \mathbb{E}[\exp(-x\Lambda_1)]$ es el exponente de Laplace del subordinador, con $x \geq 0$, y S representa un subconjunto de componentes. Dado que Λ_t es un proceso no decreciente, se tiene la equivalencia de eventos:

$$\{T_i \geq t\} = \{\Lambda_t \leq \epsilon_i\}. \quad (4.17)$$

Esta propiedad permite establecer la siguiente relación con la función de fiabilidad:

$$R_i(t) = \mathbb{P}(T_i \geq t) = \mathbb{P}(\Lambda_t \leq \epsilon_i) = F_{\Lambda_t}(\epsilon_i), \quad (4.18)$$

donde F_{Λ_t} es la función de distribución acumulada del subordinador en el tiempo t .

De esta forma, $R_i(t)$ expresa la probabilidad de que cada componente T_i continúe funcionando al menos hasta el tiempo t , lo cual está determinado por el proceso subordinador y los umbrales ϵ .

Ejemplo 4.17 *Consideremos el sistema ilustrado en la Figura 4.1. En este sistema, cada componente i falla en el instante en que el proceso Λ_t supera el umbral ϵ_i , obteniéndose así el tiempo de vida T_i . La Figura 4.2 muestra un ejemplo concreto de esta realización, en el que se aprecia la trayectoria del proceso Λ_t junto a los distintos umbrales asignados a cada componente. El cruce de alguno de estos umbrales determina el fallo de la correspondiente componente, mostrando la interacción entre el proceso subordinador y los tiempos de vida de cada elemento. Como se puede notar, el modelo LFMO resulta eficiente, pues solo necesita 3 parámetros para su simulación, mientras que el modelo Marshall-Olkin requeriría 2^3 parámetros.*

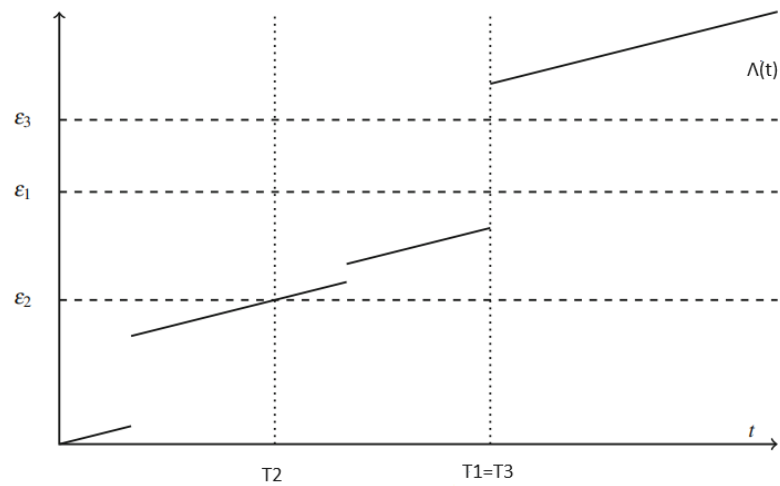


Figura 4.2: Ejemplo de realización de la distribución LFMO. Se aprecia que la degradación constante del proceso cruza el umbral ϵ_2 , obteniéndose así el tiempo de vida T_2 pero el sistema sigue en funcionamiento. De igual forma, el proceso experimenta un salto que cruza simultáneamente los umbrales ϵ_1 y ϵ_3 , haciendo que los tiempos de vida T_1 y T_3 sean iguales, terminando con la ejecución del sistema.

Capítulo 5

Estudio de Caso

En este capítulo se utilizarán los conceptos estudiados en los capítulos anteriores para abordar el problema de determinar una política de mantenimiento óptima en sistemas cuyas componentes pueden fallar de forma aleatoria. Se asumirá que se harán reparaciones de dichas componentes cuando sea necesario o adecuado con el fin de abaratar los costes totales. Se trabajará en un ejemplo que ponga a prueba la factibilidad del modelo propuesto utilizando distintos criterios de optimización y se analizarán los resultados obtenidos.

5.1. Modelación sistema para determinar una política de mantenimiento óptima

Se considera el sistema $\text{Sys}(3)$, de 3 componentes presentado en el Ejemplo 4.2, el que queda representado por la Figura 5.1.

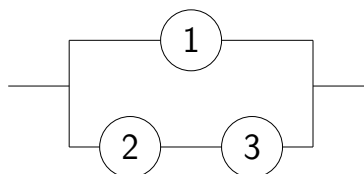


Figura 5.1: Ejemplo de sistema de tres componentes.

En este ejemplo, la operatividad del sistema puede modelarse mediante la siguiente función estruc-

tural:

$$\phi(x) = \max\{x_1, \min\{x_2, x_3\}\}, \quad (5.1)$$

donde $x \in \{0, 1\}^3$. En lo que sigue, para $s \in \{0, 1\}^3$ denotaremos $|s| := s_1 + s_2 + s_3$.

Para **Sys(3)**, se modelan los tiempos de vida de los componentes utilizando la distribución Lévy-Frailty-Marshall-Olkins (LFMO).

El proceso de degradación asociado a la distribución LFMO, denotado como $\{\Lambda_t\}_{t \geq 0}$ será un proceso de Poisson compuesto con tasa $\lambda > 0$ al que se añade un drift constante con factor $\mu > 0$:

$$\Lambda_t = \mu t + \sum_{i=1}^{N_t} J_i \quad t > 0, \quad (5.2)$$

donde:

- μ representa la tasa de degradación constante que sufre el sistema a lo largo del tiempo.
- N_t es el número de eventos de degradación instantánea que ocurren hasta el tiempo t .
- J_i son variables aleatorias i.i.d., que representan la magnitud de cada evento catastrófico y la degradación que produce.

El exponente de Laplace asociado a Λ_t se define como::

$$\psi(b) = -\log \mathbb{E} \left[e^{-b\Lambda_1} \right] = \mu b + \lambda(1 - \mathbb{E}(e^{-bJ_1})). \quad (5.3)$$

La elección de la distribución de J_1 depende del fenómeno que se modela. En este caso, se opta por una distribución exponencial con tasa $\gamma > 0$. Por lo tanto:

$$\mathbb{E}(e^{-bJ_1}) = \frac{\gamma}{\gamma + b} \quad b > 0,$$

y el exponente de Laplace queda expresado como:

$$\psi(b) = \mu b - \lambda \left(1 - \frac{\gamma}{\gamma + b}\right) = \mu b - \lambda \frac{\gamma}{\gamma + b} \quad (5.4)$$

En esta caracterización, la dinámica del sistema está determinada por los parámetros μ , λ y γ .

Alternativamente, se podrían considerar otras distribuciones para J_1 , como las distribuciones Gamma o Weibull, dependiendo de las características del sistema a modelar. Para mayor información en este tema consultar en [Mai10].

Observación 5.1 *Notemos que la distribución exponencial presenta colas ligeras. Se denomina “cola ligera” a aquella distribución para la cual la probabilidad de obtener valores extremos decrece rápidamente, lo que implica que existe menos masa en la cola. Esto se traduce en que los eventos de magnitud muy alta son extremadamente poco probables.*

En este caso, dado que J_1 se distribuye de forma exponencial, su naturaleza limita la ocurrencia de eventos catastróficos de gran magnitud.

Por otro lado, si se requiere modelar situaciones en las que eventos de alta magnitud ocurren con mayor frecuencia, se puede optar por distribuciones con colas pesadas (por ejemplo, algunas variantes de las distribuciones Gamma o Weibull), en las que la probabilidad de valores extremos decae de manera más lenta.

5.2. Formulación del problema de optimización

El objetivo del problema de optimización que se abordará es minimizar los costos asociados a las reparaciones del sistema. Cada componente tiene un costo fijo por reparación, y si el sistema llega a fallar, se añade un costo adicional por la reparación completa del sistema.

El agente debe determinar cuándo es adecuado realizar reparaciones, considerando las siguientes restricciones:

- Si el sistema falla, debe repararse de manera obligatoria e inmediata.

- Si el sistema está operativo y se decide reparar, la reparación es completa, es decir, todos los componentes vuelven a su estado inicial.

Este sistema es un ejemplo claro de un Proceso de Decisión de Markov (MDP). En la Tabla 5.1 se resumen los elementos que componen el problema.

Elemento	Notación	Consideraciones
Épocas de decisión	Espacio $T \subseteq \mathbb{N}$	Aunque las componentes fallan en tiempos continuos, el MDP se modela en tiempo discreto. Por ello, se definen las épocas de decisión como los instantes aleatorios en los que ocurre algún evento (por ejemplo, el fallo de una componente), momento en el cual se actualiza el estado del sistema y se toma una decisión.
Espacio de estados	Espacio $S = \{0, 1\}^3$	Los estados corresponden a todas las configuraciones posibles para las 3 componentes del sistema.
Acciones	Espacio $\mathcal{A} = \{R, NR\}$ Donde ■ "R": Reparar ■ "NR": No reparar	Las acciones determinan si el sistema se repara completamente o se deja que el sistema siga su dinámica.
Transiciones de probabilidad	$p : S^2 \times \mathcal{A} \rightarrow [0, 1]$ $(s', a, s) \mapsto p(s' s, a)$	Las transiciones están determinadas por la distribución LFMO y las reglas de reparación impuestas.
Retribuciones	$r : S \times \mathcal{A} \rightarrow \mathbb{R}$ ■ C^c: Coste por componentes en mal estado. ■ C^s: Coste por reparación total del sistema.	Solo se consideran retribuciones negativas (costos) asociados a las reparaciones y estados defectuosos.
Criterio de optimización	Se busca $\pi \in \Pi$ tal que: $v^* = \sup_{\pi \in \Pi} \left(\mathbb{E}_s^\pi \left(\sum_{n=1}^{\infty} \alpha^{n-1} r(X_n, Y_n) \right) \right)$	El objetivo es encontrar una política π que maximice el valor esperado de las retribuciones con descuento. El factor $\alpha \in (0, 1)$ controla la importancia relativa de costos presentes y futuros.

Cuadro 5.1: Elementos del MDP. Como el problema supone que las probabilidades de transición y las retribuciones son estacionarias a lo largo del tiempo, se omite la notación temporal.

Probabilidades de Transición: Condicional a que el sistema se encuentre en el estado $s_0 \in \{0, 1\}^3$

- Si el sistema no está operativo, la única acción permitida es reparar y entonces:

$$p(s_1 | s_0, R) = \begin{cases} 1 & \text{si } s_1 = [1]^3 \\ 0 & \text{en caso contrario} \end{cases}$$

donde $[1]^3 = (1, 1, 1)$.

- Si el sistema está operativo:

- Si la acción es no reparar, entonces, sólo hay probabilidad de transitar a estados s_1 con menos componentes activas, es decir $|s_1| \leq |s_0|$ y

$$p(s_1 | s_0, R) = H \left[n+1-|s_0|, n+1-|s_1| \right] \binom{|s_0|}{|s_1|}^{-1},$$

para todo $s_0, s_1 \in S$ tal que s_1 sea un estado de transición posible a partir de s_0 , consideramos la matriz H , de dimensiones $n \times n$, cuyos términos se definen de la siguiente forma:

$$H[i, j] = \binom{n-i}{j-i} \frac{\lambda_{[n-i, j-i]}}{\psi(n-i)} \quad \text{si } i < j,$$

para $i, j \in \{0, \dots, n\}$.

En esta expresión, $\lambda_{[d, I]}$ representa la tasa a la cual, dada una cantidad I de componentes en funcionamiento, se observa que solo d de ellas continúan operativas. Esta tasa se define por:

$$\lambda_{[d, I]} = \sum_{i=0}^{I-1} (-1)^i \binom{I-1}{i} (\psi(d-I+i+1) - \psi(d-I+i)).$$

El resultado que origina la definición de $\lambda_{[d, I]}$ se deduce de la tasa presentada en (4.16) (véase [BL25]).

Todas estas transiciones se resumen en la Figura 5.2.

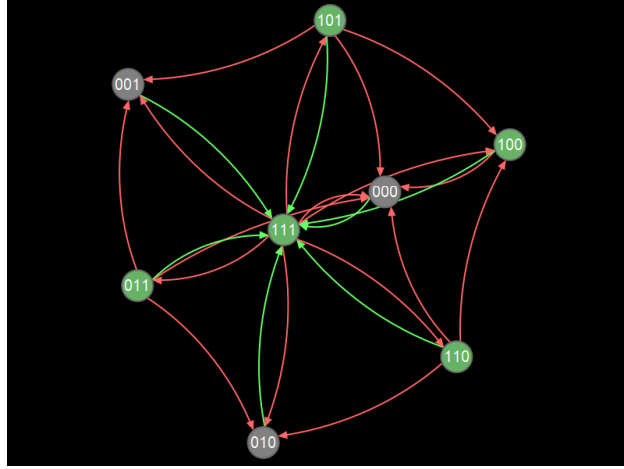


Figura 5.2: Grafo que representa las transiciones del sistema. Cada nodo representa un estado del sistema: los nodos en verde indican que en ese estado, el sistema está operativo, mientras que los nodos en gris señalan que el sistema está inoperativo. Las aristas representan las transiciones resultantes de la acción tomada: las aristas rojas indican que la transición se produce por la decisión de no reparar el sistema y las aristas verdes indican que la transición es consecuencia de la acción de reparar.

Observación 5.2 *Notemos que el proceso analizado se desarrolla en tiempo continuo, y los eventos ocurren de manera aleatoria. Al definir los tiempos de decisión como los instantes en que se produce un evento, el factor de descuento en la formulación de Bellman no se aplica de forma uniforme (es decir, α^t con t variable). Formalmente, tanto la variación temporal como el descuento al largo del tiempo deberían considerarse aleatorios. Sin embargo, el tiempo de vida de las componentes se asume que distribuyen LFMO lo que implica que los tiempos entre épocas de decisión distribuyen de manera exponencial. Entonces debido a que la distribución exponencial tiene cola ligera, la probabilidad de observar grandes variaciones entre los tiempos de salto es pequeña. Además, dado que las políticas obtenidas se muestran suficientemente robustas ante pequeñas variaciones del factor de descuento, dichas fluctuaciones no deberían alterar de manera significativa la política óptima. Esto justifica en cierta forma la aplicación de un factor de descuento con incrementos de tiempo constantes, como se utiliza en el presente análisis.*

5.3. Simulaciones y Resultados

La formulación del problema como un Proceso de Decisión de Markov (MDP) permite aplicar las herramientas presentadas en el capítulo 2. En particular, se empleará el algoritmo de Iteración de

Valor descrito en el capítulo 3 para determinar una política óptima.

En este caso, se aborda el problema bajo el criterio de valor esperado descontado, resolviendo la ecuación de Bellman:

$$v(s) = \sup_{a \in \mathcal{A}} \left[r(s, a) + \alpha \sum_{s' \in \mathcal{S}} p(s' | s, a) v(s') \right] \quad (5.5)$$

Se efectuaron simulaciones considerando todas las combinaciones posibles de los parámetros especificados en la Tabla 5.2.

Parámetro	Valores posibles
(μ, λ, γ)	$(1, 1, 1), (100, 1, 1), (1, 100, 1)$
(C^c, C^s)	$(-1, -1), (-1, -0.1), (-1, -10)$
α	$\{0.01, 0.02, \dots, 0.99\}$

Cuadro 5.2: Parámetros y valores analizados.

Se deja invariante el valor para las componentes y se varia el coste del sistema ya que el interes es revisar como influye la razón entre el costo del sistema y el costo de la componente (C^s/C^c) , en los resultados.

Los experimentos numéricos revelaron únicamente cuatro tipos de políticas óptimas, detalladas a continuación y representadas gráficamente en la Figura 5.3.

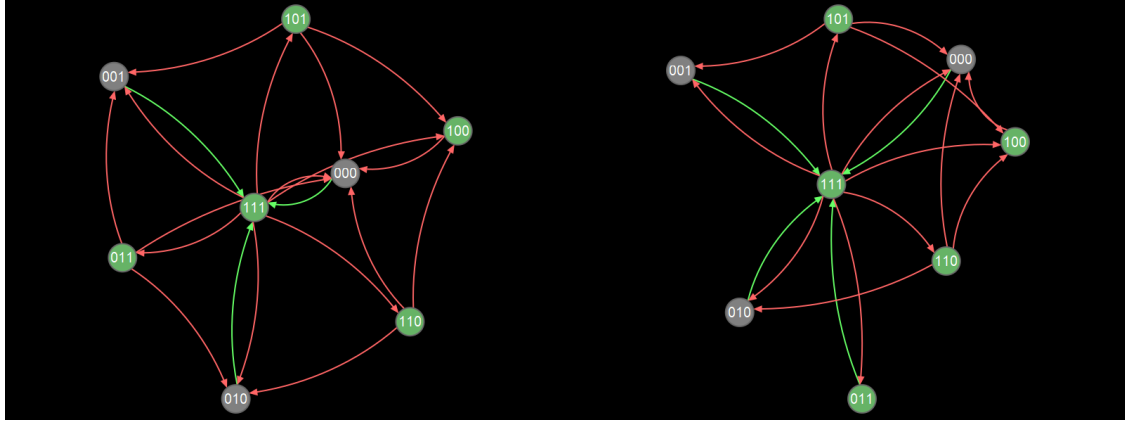
Esta políticas, para valores fijos de los parámetros (μ, λ, γ) y (C^c, C^s) , dividen el rango de α en cuatro subintervalos, dentro de cada uno de los cuales la política óptima se mantiene constante.

1. **Tipo (a): Indiferente a fallas de componentes.** En este caso, el costo de reparar el sistema es relativamente pequeño en la función valor, lo que significa que, comparado con otros términos, el impacto de una reparación es mínimo. Esto se traduce en que el agente opta por no reparar salvo cuando la falla es inminente. La reparación del sistema (es decir, reparar la totalidad de las componentes) resulta, relativamente menos “atractiva” que mantener la operatividad de cada componente de forma individual.
2. **Tipo (b): Comportamiento intermedio.** Aquí la decisión de reparar depende de la gravedad de la falla. Cuando el sistema tiene una probabilidad alta de fallar, el agente decide reparar; en caso contrario, se abstiene. Este comportamiento refleja un balance en el que aunque el

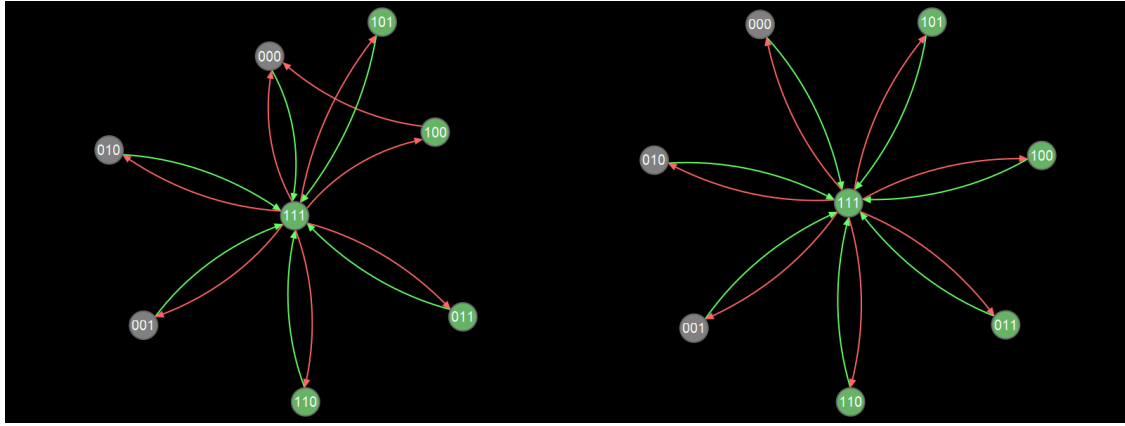
costo de reparar sistema en la función valor es pequeño en comparación con otros costos, su impacto se vuelve relevante si la próxima falla compromete la operatividad del sistema.

3. **Tipo (c): Comportamiento intermedio con tendencia a reparar.** En este caso, el agente repara en casi todas las situaciones, excepto cuando la falla es tan probable que no hay un cambio significativo entre reparar ahora o cuando el sistema deje de operar. Esto indica que, a pesar de que el costo de reparar es pequeño dentro de la función valor, existe un umbral donde si la falla se acerca demasiado a la falla total del sistema, la reparación no aporta lo suficiente al beneficio esperado, por lo que se evita la acción de reparar.
4. **Tipo (d): Reparación inmediata ante cualquier indicio de fallo.** Aquí, el costo de reparar es tan bajo en comparación con el impacto negativo de una falla del sistema que el agente opta por reparar inmediatamente al detectar cualquier tipo de fallo. La estrategia es prevenir los elevados costes o pérdidas asociados a la inoperatividad.

El análisis también reveló que, aunque las políticas óptimas permanecen invariantes al modificar las probabilidades de transición y los costos, los intervalos de α en los que estas políticas se mantienen constantes varían significativamente. Se observó que las transiciones entre políticas dependen principalmente de la relación entre el costo de reparación de una componente y el costo de reparación del sistema y en menor medida de los parámetros del subordinador. Las Tablas 5.3, 5.4 y 5.5 presentan estos intervalos para distintas configuraciones de los parámetros (μ, λ, γ) , mostrando la comparación para cada costo.



(a) Política indiferente a la falla del sistema. (b) Política con baja prevención frente a fallas



(c) Política con alta prevención frente a fallas. (d) Política comprometida con el funcionamiento.

Figura 5.3: Representación gráfica de políticas de reparación según el nivel de prevención del agente ante las fallas del sistema. Las políticas de reparación están indicadas por las aristas verdes. Las flechas rojas muestran las posibles transiciones de estados que seguirá el sistema si no se toman acciones de reparación. Las representaciones (a), (b), (c) y (d) están ordenadas en orden creciente según el factor de descuento.

- Intervalos de α para $(\mu, \lambda, \gamma) = (1, 1, 1)$

Costos (C^c, C^s)	Intervalos de α			
	(a)	(b)	(c)	(d)
(1,1)	(0.01, 0.36)	(0.36, 0.47)	(0.47, 0.57)	(0.57, 1)
(1,10)	(0.01, 0.09)	(0.09, 0.14)	(0.14, 0.16)	(0.16, 1)
(1,0.1)	(0.01, 0.55)	(0.55, 0.64)	(0.64, 0.78)	(0.78, 1)

Cuadro 5.3: Intervalos de α para diferentes configuraciones de costos (C^c, C^s) .

- Intervalos de α para $(\mu, \lambda, \gamma) = (10, 1, 1)$
- Intervalos de α para $(\mu, \lambda, \gamma) = (1, 10, 1)$

Costos (C^c, C^s)	Intervalos de α			
	(a)	(b)	(c)	(d)
(1,1)	(0.01, 0.36)	(0.36, 0.49)	(0.49, 0.56)	(0.56, 1)
(1,10)	(0.01, 0.09)	(0.09, 0.15)	(0.15, 0.16)	(0.16, 1)
(1,0.1)	(0.01, 0.57)	(0.57, 0.65)	(0.65, 0.76)	(0.76, 1)

Cuadro 5.4: Intervalos de α para diferentes configuraciones de costos (C^c, C^s).

Costos (C^c, C^s)	Intervalos de α			
	(a)	(b)	(c)	(d)
(1,1)	(0.01, 0.35)	(0.35, 0.44)	(0.44, 0.63)	(0.63, 1)
(1,10)	(0.01, 0.09)	(0.09, 0.12)	(0.12, 0.17)	(0.17, 1)
(1,0.1)	(0.01, 0.53)	(0.53, 0.61)	(0.61, 0.88)	(0.88, 1)

Cuadro 5.5: Intervalos de α para diferentes configuraciones de costos (C^c, C^s).

El factor de descuento determina la relevancia de los costos futuros en relación con los costos actuales. Un factor de descuento alto asigna un peso similar a los costos futuros y presentes, lo que favorece estrategias de mantenimiento preventivo, como en los casos (c) y (d). En este escenario, reparar el sistema de manera anticipada es una decisión óptima, ya que las fallas futuras se consideran tan relevantes como los costos inmediatos de reparación. Esto es especialmente importante en sistemas críticos, como plantas de energía o centros de datos, donde la interrupción del funcionamiento implica pérdidas significativas.

Por otro lado, un factor de descuento bajo implica que los costos inmediatos son mucho más significativos que los costos futuros. En estos casos, las políticas como (a) y (b), que postergan la reparación hasta que sea absolutamente necesaria, resultan más favorables. Este enfoque podría ser adecuado en escenarios donde las fallas del sistema no tienen un impacto severo o donde el costo de oportunidad de invertir en mantenimiento es alto. Por ejemplo, en sistemas con equipos de bajo costo que pueden ser fácilmente reemplazados, postergar reparaciones hasta que sea estrictamente necesario puede ser una estrategia económicamente viable.

Es importante destacar que los intervalos de estabilidad de las políticas óptimas experimentan cambios significativos al modificar los costos. Cuando el costo de reparación del sistema es considerablemente mayor que el de los componentes, el intervalo asociado a una política comprometida con el funcionamiento del sistema se amplía, indicando que esta estrategia es más adecuada para una visión de funcionalidad a largo plazo.

En contraste, cuando el costo de reparación del sistema disminuye, las políticas con menor prevención frente a fallas ganan relevancia, especialmente en escenarios donde se prioriza el análisis inmediato.

En términos generales, un factor de descuento alto tiende a favorecer políticas conservadoras, como las mostradas en los casos (c) y (d). Estas políticas prodian ser adecuadas en casos cuando una falla del sistema resulta muy costosa en comparación a la repación de componentes. Por otro lado, un factor de descuento bajo promueve políticas más arriesgadas como son los casos (a) y (b), llegando a ser adecuadas para situaciones donde los costos por fallas del sistema son bajos y el costo de reparaciones no afecta significativamente en comparación con los costos de los componentes.

5.4. Evaluación políticas Optimas

Para validar las políticas óptimas obtenidas, se realizó una simulación del sistema siguiendo las indicaciones de cada política.

En el análisis, cada caso fue simulado múltiples veces en un horizonte temporal fijo, utilizando un enfoque tipo Monte Carlo. Si el sistema fallaba, se procedía a repararlo, sumando el costo correspondiente al total acumulado por simulación.

Para cada intervalo presentado en las Tablas 5.3, 5.4 y 5.5, se tomó el punto medio como un valor representativo de α para dicho intervalo. Con cada α , se determinó su política óptima correspondiente.

Las medidas utilizadas para evaluar las simulaciones son presentadas en la Tabla 5.6:

Cuadro 5.6: Medidas de comparación para el análisis del sistema

Medida de comparación	Descripción
ECcycle	Coste promedio hasta la primera falla del sistema
ETcycle	Tiempo promedio hasta la primera falla
LTMC	Coste medio a largo plazo
LTMN	Cantidad esperada de fallas a largo plazo

Los resultados obtenidos se resumen en las Tablas 5.7, 5.8 y 5.9.

ECycle representa el coste promedio hasta la primera falla del sistema, brindando información sobre la eficiencia económica antes de que ocurra una interrupción. **ETcycle** mide el tiempo promedio hasta la primera falla, lo que permite evaluar la fiabilidad del sistema. **LTMC**, por su parte, corresponde al coste medio en el largo plazo, definido como $\lim_{t \rightarrow \infty} \frac{C[0,t]}{t}$, proporcionando una visión del rendimiento económico general. Finalmente, **LTMN** representa la cantidad esperada de fallas en el largo plazo, útil para estimar la frecuencia de eventos críticos.

Coste $C^c = 1$, $C^s = 1$	Factor de Descuento	ECcycle	ETcycle	LTMC	LTMN
Parámetro : $\mu = 1$ $\lambda = 1$ $\gamma = 1$	$\alpha_d = 0.185$	(mean)=3.375 (std)=0.014	(mean)=0.86 (std)=0.021	(mean)=3.92 (std)=0.097	(mean)=1.161 (std)=0.029
	$\alpha_c = 0.415$	(mean)=-5.147 (std)=0.105	(mean)=1.731 (std)=0.08	(mean)=2.973 (std)=0.093	(mean)=0.578 (std)=0.026
	$\alpha_b = 0.52$	(mean)=8.224 (std)=0.248	(mean)=2.763 (std)=0.149	(mean)=2.976 (std)=0.093	(mean)=0.362 (std)= 0.019
	$\alpha_a = 0.785$	(mean)=8.687 (std)=0.273	(mean)=2.898 (std)=0.163	(mean)=2.997 (std)= 0.095	(mean)=0.345 (std)= 0.019
Parámetro : $\mu = 10$ $\lambda = 1$ $\gamma = 1$	$\alpha_d = 0.185$	(mean)=3.209 (std)=0.004	(mean)=0.134 (std)=0.001	(mean)=-23.953 (std)=0.214	(mean)=7.464 (std)=0.068
	$\alpha_c = 0.425$	(mean)=18.559 (std)=0.78	(mean)=1.53 (std)= 0.076	(mean)=12.13 (std)=0.141	(mean)=0.654 (std)=0.032
	$\alpha_b = 0.525$	(mean)=-35.772 (std)=1.747	(mean)=2.964 (std)=0.165	(mean)=12.06 (std)=0.138	(mean)=0.337 (std)=0.019
	$\alpha_a = 0.78$	(mean)=36.129 (std)=1.77	(mean)=2.994 (std)=0.168	(mean)=12.057 (std)=0.14	(mean)=0.334 (std)=0.019
Parámetro : $\mu = 1$ $\lambda = 10$ $\gamma = 1$	$\alpha_d = 0.18$	(mean)=3.614 (std)=0.007	(mean)=0.19 (std)=0.002	(mean)=19.061 (std)=0.242	(mean)=5.275 (std)=0.067
	$\alpha_c = 0.395$	(mean)=3.937 (std)=0.011	(mean)=0.213 (std)=0.003	(mean)=18.435 (std)=0.245	(mean)=4.681 (std)=0.066
	$\alpha_b = 0.535$	(mean)=4.719 (std)=0.024	(mean)=0.243 (std)=0.004	(mean)=19.382 (std)=0.251	(mean)=4.108 (std)=0.065
	$\alpha_a = 0.815$	(mean)=5.184 (std)=0.034	(mean)=0.255 (std)=0.004	(mean)=20.304 (std)=0.261	(mean)=3.916 (std)=0.065

Cuadro 5.7: Resultados de simulación para las diferentes políticas de mantenimiento α presentadas en 5.3 bajo costos de componente $C^c = 1$ y costos de sistema $C^s = 1$. Se presentan las medidas de valor promedio (mean) y desviación estándar (std) para cada métrica ECcycle, ETcycle, LTMC y LTMN, evaluadas para distintos valores de los parámetros μ , λ y γ .

Coste $C^c = 1, C^s = 10$	Factor de Descuento	ECcycle	ETcycle	LTMC	LTMN
Parámetro : $\mu = 1$ $\lambda = 1$ $\gamma = 1$	$\alpha_d = 0.050$	(mean)=12.375 (std)=0.014	(mean)=0.86 (std)=0.021	(mean)=14.381 (std)=0.35	(mean)=1.162 (std)=0.028
	$\alpha_c = 0.115$	(mean)=14.147 (std)=0.105	(mean)=1.732 (std)=0.079	(mean)=8.171 (std)=0.324	(mean)=0.578 (std)=0.026
	$\alpha_b = 0.15$	(mean)=17.222 (std)=0.249	(mean)=2.762 (std)=0.151	(mean)=6.235 (std)=0.263	(mean)=0.362 (std)=0.02
	$\alpha_a = 0.58$	(mean)=17.692 (std)=0.274	(mean)=2.899 (std)=0.161	(mean)=6.102 (std)=0.257	(mean)=0.345 (std)=0.019
Parámetro : $\mu = 10$ $\lambda = 1$ $\gamma = 1$	$\alpha_d = 0.050$	(mean)=12.209 (std)=0.004	(mean)=0.134 (std)=0.001	(mean)=91.143 (std)=0.818	(mean)=7.465 (std)=0.067
	$\alpha_c = 0.12$	(mean)=27.567 (std)=0.779	(mean)=1.531 (std)=0.075	(mean)=18.01 (std)=0.397	(mean)=0.654 (std)=0.032
	$\alpha_b = 0.155$	(mean)=44.788 (std)=1.757	(mean)=2.966 (std)=0.165	(mean)=15.093 (std)=0.276	(mean)=0.337 (std)=0.018
	$\alpha_a = 0.58$	(mean)=45.119 (std)=1.746	(mean)=2.993 (std)=0.166	(mean)=15.066 (std)=0.278	(mean)=0.334 (std)=0.018
Parámetro : $\mu = 1$ $\lambda = 10$ $\gamma = 1$	$\alpha_d = 0.050$	(mean)=12.613 (std)=0.007	(mean)=0.189 (std)=0.002	(mean)=66.531 (std)=0.841	(mean)=5.274 (std)=0.067
	$\alpha_c = 0.105$	(mean)=12.938 (std)=0.012	(mean)=0.214 (std)=0.003	(mean)=60.576 (std)=0.822	(mean)=4.682 (std)=0.065
	$\alpha_b = 0.145$	(mean)=13.718 (std)=0.024	(mean)=0.243 (std)=0.004	(mean)=56.348 (std)=0.824	(mean)=4.107 (std)=0.065
	$\alpha_a = 0.585$	(mean)=14.184 (std)=0.034	(mean)=0.255 (std)=0.004	(mean)=55.554 (std)=0.832	(mean)=3.916 (std)=0.065

Cuadro 5.8: Resultados de simulación para las diferentes políticas de mantenimiento α presentadas en 5.3 bajo costos de componente $C^c = 1$ y costos de sistema $C^s = 10$. Se presentan las medidas de valor promedio (mean) y desviación estándar (std) para cada métrica ECcycle, ETcycle, LTMC y LTMN, evaluadas para distintos valores de los parámetros μ , λ y γ .

Coste $C^c = 1, C^s = 0.1$	Factor de Descuento	ECcycle	ETcycle	LTC	LTMN
Parámetro : $\mu = 1$ $\lambda = 1$ $\gamma = 1$	$\alpha_d = 0.28$	(mean)=2.475 (std)=0.014	(mean)=0.86 (std)=0.021	(mean)=2.875 (std)= 0.072	(mean)=1.162 (std)=0.029
	$\alpha_c = 0.595$	(mean)=4.247 (std)=0.106	(mean)=1.731 (std)=0.079	(mean)=2.578 (std)=0.026	(mean)=1.067 (std)=0.037
	$\alpha_b = 0.71$	(mean)=7.322 (std)=0.249	(mean)=2.761 (std)=0.15	(mean)=2.652 (std)=0.077	(mean)=0.362 (std)=0.02
	$\alpha_a = 0.89$	(mean)=7.791 (std)=0.273	(mean)= 2.9 (std)=0.162	(mean)=2.686 (std)=0.081	(mean)=0.345 (std)=0.019
Parámetro : $\mu = 10$ $\lambda = 1$ $\gamma = 1$	$\alpha_d = 0.29$	(mean)=2.309 (std)=0.004	(mean)=0.134 (std)=0.001	(mean)=17.236 (std)=0.152	(mean)=7.465 (std)=0.067
	$\alpha_c = 0.61$	(mean)=17.653 (std)=0.786	(mean)=1.53 (std)=0.076	(mean)=11.54 (std)=0.123	(mean)=0.654 (std)=0.032
	$\alpha_b = 0.705$	(mean)=34.875 (std)=1.724	(mean)= 2.964 (std)=0.163	(mean)=11.754 (std)=0.129	(mean)=0.337 (std)=0.018
	$\alpha_a = 0.88$	(mean)=35.224 (std)=1.751	(mean)=2.993 (std)=0.166	(mean)=11.759 (std)=0.129	(mean)=0.334 (std)=0.018
Parámetro : $\mu = 1$ $\lambda = 10$ $\gamma = 1$	$\alpha_d = 0.27$	(mean)=2.714 (std)=0.007	(mean)=0.19 (std)=0.002	(mean)=14.317 (std)=0.183	(mean)=5.276 (std)=0.067
	$\alpha_c = 0.57$	(mean)=3.037 (std)=0.012	(mean)=0.213 (std)=0.003	(mean)=14.228 (std)=0.184	(mean)=4.684 (std)=0.065
	$\alpha_b = 0.745$	(mean)=3.819 (std)=0.024	(mean)=0.243 (std)=0.004	(mean)=15.684 (std)=0.196	(mean)=4.107 (std)=0.065
	$\alpha_a = 0.94$	(mean)=4.285 (std)=0.034	(mean)=0.255 (std)=0.004	(mean)=16.781 (std)=0.202	(mean)=3.916 (std)=0.065

Cuadro 5.9: Resultados de simulación para las diferentes políticas de mantenimiento α presentadas en 5.3 bajo costos de componente $C^c = 1$ y costos de sistema $C^s = 0.1$. Se presentan las medidas de valor promedio (mean) y desviación estándar (std) para cada métrica ECcycle, ETcycle, LTC y LTMN, evaluadas para distintos valores de los parámetros μ , λ y γ .

Los resultados presentados en las Tablas 5.7 , 5.8 y 5.9 fueron obtenidos al realizar un total de 10,000 iteraciones donde cada una de ellas fueron realizadas en un horizonte temporal de 1000 unidades de tiempo.

Los resultados obtenidos de las simulaciones muestran que las políticas tipo (c) y (d), las más preventivas ante fallos del sistema, tienden a mejorar el rendimiento económico a largo plazo (LTMC) cuando el coste del sistema es mayor que el de los componentes ($C^s \gg C^c$). Por el contrario, cuando el coste del sistema es menor que el de los componentes ($C^s \ll C^c$), las políticas tipo (a) y (b), las menos preventivas, resultan más económicas a largo plazo.

En general, las políticas de tipo (c) y (d) presentan diferencias ligeras pero resultados muy similares en la mayoría de los casos, al igual que las políticas de tipo (a) y (b).

Este comportamiento con respecto a los costes se logra visualizar en la Figura 5.4. Para la Figura se mantuvieron constantes los valores de λ , μ , γ y el costo de reparación por componente ($C^c = 1$). Sin embargo, se varió el costo de reparación del sistema completo (C^s), tomando valores entre 0 y 25. A partir de estos, se grafico la tendencia del coste a largo plazo (LTMC).

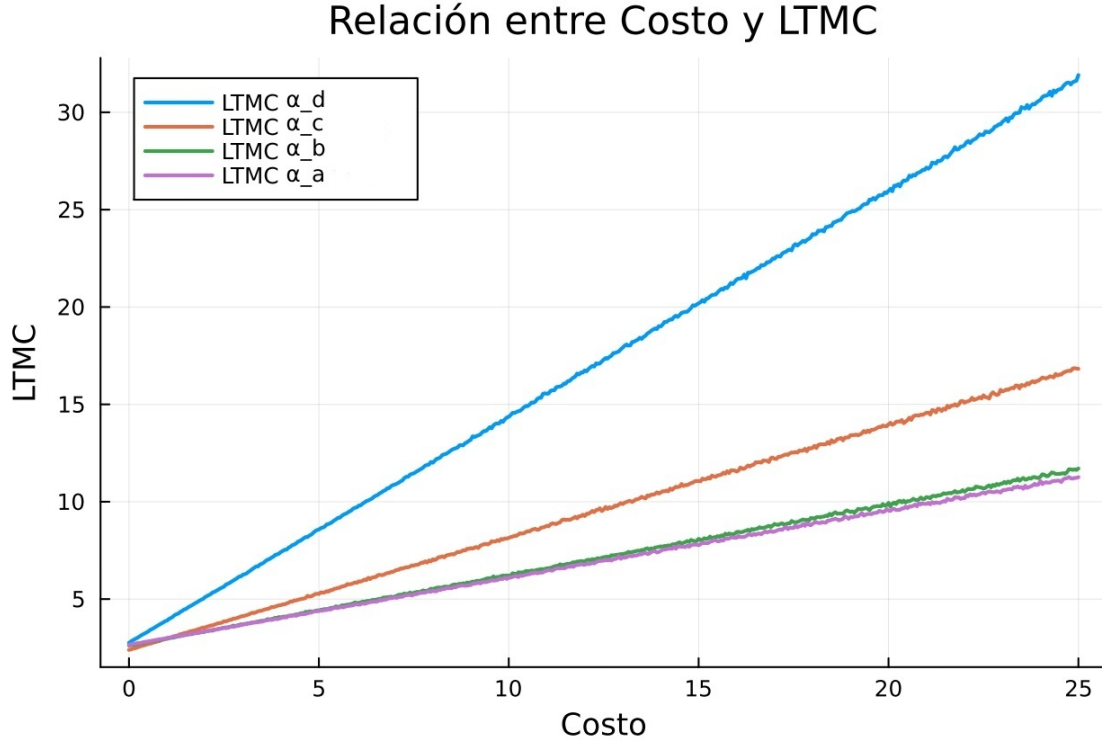


Figura 5.4: Sensibilidad costes para LTMC

A simple vista, las políticas tipo (d) y (c), las más preventivas, son las mejores opciones a medida que aumenta el costo del sistema, siendo prácticamente equivalentes, aunque ligeramente mejor la política que repara inmediatamente ante cualquier falla. Les siguen en rendimiento las políticas tipo (b) y (a), en ese orden, mostrando un desempeño significativamente inferior.

Para un análisis más detallado, se realizó un acercamiento entre los costes comprendidos entre 0 y 2.5. En la Figura 5.5 correspondiente, se observa claramente que cuando $C^s \ll C^c$, la política tipo (b) supera en eficiencia económica a las políticas tipo (c) y (d), siendo el punto de cambio alrededor de 1.

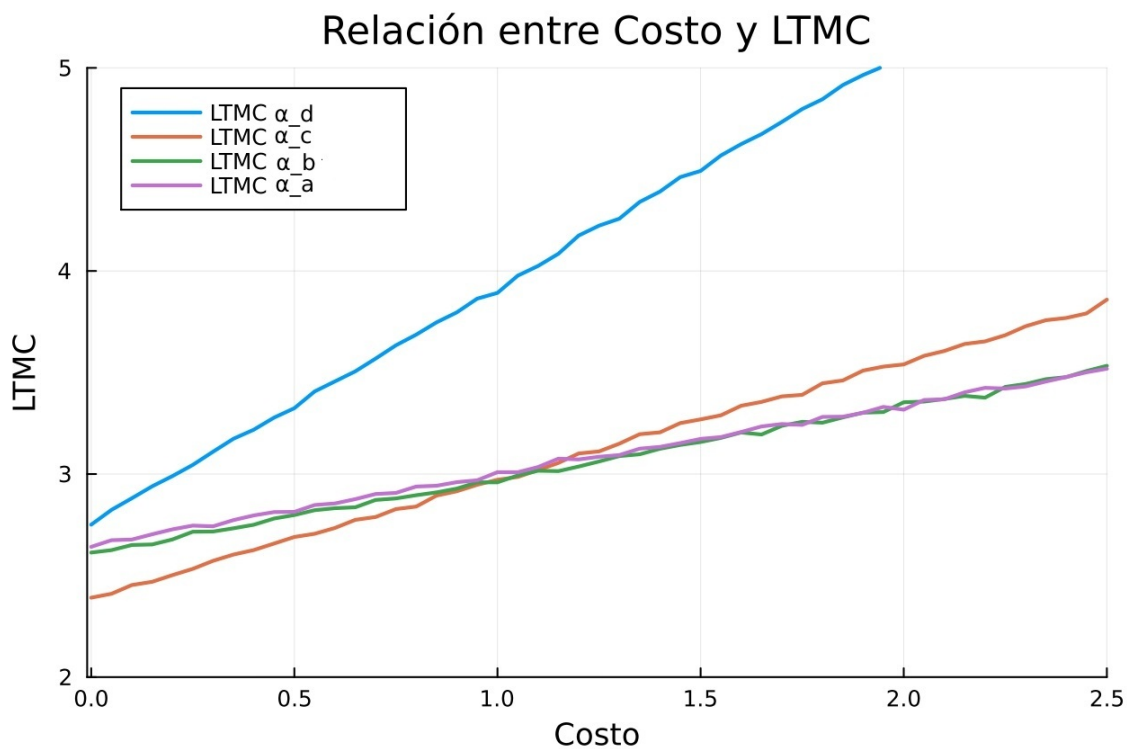


Figura 5.5: Agregar Caption

Si bien las políticas preventivas (c) y (d) presentan mayores costes durante el primer ciclo en comparación con las políticas tipo (a) y (b), estas primeras extienden la vida útil del sistema, aunque a costa de un incremento sostenido en los costes. La elección entre una opción u otra dependerá de las prioridades del agente para el sistema: maximizar la confiabilidad o minimizar los costes iniciales.

Por otro lado, como era de esperarse, las políticas que optan por reparar con mayor frecuencia

logran reducir la cantidad de fallas a largo plazo. Esta relación se mantiene invariante ante cualquier cambio de parámetros.

Al aumentar el parámetro μ , es más probable que falle un componente individual que un conjunto de ellos por un evento de salto. Esto hace que el sistema tenga mayor probabilidad de alcanzar un estado en el que sea necesario reparar componentes individuales antes de la falla completa. Como consecuencia, se produce un aumento considerable en los costes debido a la reparación aislada de componentes.

Cuando se incrementa el valor de λ , es más probable que múltiples componentes fallen de manera simultánea, haciendo más probable que el sistema falle por completo sin pasar por estados intermedios donde puedan realizarse reparaciones. Esto afecta negativamente el tiempo de vida del sistema, generando un aumento significativo en los costes asociados a la reparación total del sistema.

El análisis muestra que para este sistema de tres componentes las políticas más preventivas logran extender la vida útil del sistema y reducir la frecuencia de fallas, aunque presentan mayores costos iniciales. Sin embargo, esta inversión inicial puede ser compensada por una operación más estable y económica a largo plazo. Finalmente, los cambios en los parámetros de degradación (μ) y frecuencia de saltos (λ) reflejan que la dinámica del sistema puede alterar la conveniencia de cada política, siendo importante la consideración del proceso de Lévy utilizado y sus parámetros para seleccionar la estrategia de reparación más adecuada.

Capítulo 6

Conclusiones y trabajo futuro

El presente proyecto abordó el desarrollo de políticas óptimas de reparación para sistemas modelados mediante la distribución Lévy-Fraily Marshall-Olkin (L.F.M.O.), utilizando el algoritmo de Value Iteration.

En este tiempo se logró implementar y ajustar el algoritmo de Value Iteration para un sistema básico de tres componentes, obteniendo políticas de reparación óptimas bajo distintas configuraciones de parámetros del sistema, como costos de falla y probabilidades de transición.

Un hallazgo relevante fue la identificación de intervalos para el factor de descuento, dentro de los cuales las políticas permanecen constantes. Estos intervalos dependen de los parámetros del sistema: por ejemplo, políticas conservadoras que reparan inmediatamente predominaron con intervalos mas largos cuando los costos de fallas del sistema eran elevados, mientras que por otra parte las políticas más permisivas surgieron cuando bajaron los costes del sistema.

Esto sugiere que la amplitud de estos intervalos podría servir como un indicador para evaluar la efectividad de las políticas de mantenimiento. Los resultados preliminares muestran que intervalos más largos suelen estar relacionados con estrategias de mantenimiento más adecuadas y potencialmente más eficientes. No obstante, esta relación requiere un análisis más detallado para ser confirmada.

Un desafío importante surgió al intentar extender el estudio a sistemas de 21 componentes, donde la complejidad computacional se tornó prohibitiva debido al tamaño del espacio de estados (2^{21}

estados). Esto impidió obtener políticas óptimas y realizar simulaciones en tiempos razonables.

En cuanto a las posibles extensiones, en primer lugar se plantea ampliar el análisis a sistemas de mayor tamaño, lo que requeriría desarrollar técnicas de reducción del espacio de estados o métodos de aproximación que permitan manejar la alta dimensionalidad. En segundo lugar, es necesario validar empíricamente los resultados asociados al factor de descuento, lo que podría implicar la comparación con datos experimentales para calibrar este parámetro de forma precisa según las características del sistema. Finalmente, se propone ampliar el modelo para incorporar variaciones en la dinámica de fallas y reparaciones, por ejemplo, incluyendo reparaciones con efecto tardío, utilizando distribuciones alternativas para modelar los tiempos de fallo o integrando escenarios estocásticos de mayor complejidad.

Un ejemplo de extensión sería incorporar un retraso en las reparaciones, de modo que estas no se realicen de inmediato, sino después de un tiempo aleatorio. Esto podría ser útil para modelar por ejemplo sistemas de redes eléctricas donde las reparaciones deben hacerse en el terreno y puede tardar un tiempo aleatorio en efectuarse. Además, se podría considerar la variación en la tasa de fallas durante diferentes períodos, por ejemplo, cuando en sistemas que tienen periodos de alta y baja intensidad de uso, esto afectaría la frecuencia en la ocurrencia de las fallas. Otra posible extensión sería utilizar modelos con colas pesadas, que pueden ser apropiados para sistemas donde las fallas catastróficas ocurren más frecuentemente. Finalmente, sería necesario ampliar el análisis del factor de descuento α^t , que en este trabajo se consideró constante, pero tiene un comportamiento aleatorio que afectara a las políticas de reparación, lo que requiere un estudio más profundo.

Apéndice A

Códigos empleados

A.1. Simulación LFMO

```
# -----  
# FUNCIONES DEL PROCESO  
# -----  
  
# Actualización de costos según el estado  
function update_costs(s, phi, Cc, Cs) # = (state,function, cost value ,cost sistem)  
    reparation_value = count_zeros(~UInt8(2^n-1) | UInt8(s)) * Cc  
    return phi(s) == 0 ? Cs + reparation_value : reparation_value # inoperative system  
end  
  
function initialize_repair(n)  
    epsilon = -log.(rand(n))  
    epsilon_min = minimum(epsilon)  
    s = UInt8(2^n-1)  
    epsilon_t = Set()  
    t, X_t, J, C = [0.0], [0.0], [0.0], [0.0]  
    T, Tm = zeros(n), [0.0]  
    CT= 0.0
```

```

    return (t, X_t, T, Tm, J, C, CT, s, epsilon, epsilon_t, epsilon_min)
end

function update_state_time(t, X_t, J, mu, lambda, gamma ,THorizon)
    dt = -log(rand()) / lambda
    if t[end]+dt > THorizon
        push!(t, t[end] + THorizon)
    else
        push!(t, t[end] + dt)
    end
    J_i = -log(rand()) / gamma
    push!(J, J_i)

    push!(X_t, X_t[end] + mu * dt + J_i)
    return dt,J_i
end

function simulate_system(pi*, phi, mu, lambda, gamma, Cc, Cs, n ,THorizon)
    t, X_t, T, Tm, J,
        C, CT, s, epsilons, epsilons_t, epsilons_min = initialize_repair(n)

    dt,J_i = update_state_time(t, X_t, J, mu, lambda, gamma,THorizon)
    while t[end] < THorizon
        while X_t[end-1] < epsilon_min <= X_t[end]
            #Todo lo que pasa en el drif
            if X_t[end-1] < epsilon_min < X_t[end-1] + mu * dt

                k = findfirst(I -> (I == epsilon_min), epsilon)
                T[k] = t[end-1] + (epsilon_min - X_t[end-1]) / mu
                s = s & ~(UInt8(1) << (n-k))
                push!(epsilon_t, epsilon_min)
            end
        end
    end
end

```

```

if phi(s) == 0
    push!(Tm,T[k])
    CT += update_costs(s, phi, Cc, Cs)
    push!(C,CT)
    CT = 0
    for j in 1:n
        if epsilon[j] <= epsilon[k]
            epsilon[j] = epsilon[k] - log(rand())
            T[j] = 0
        end
    end
    s = UInt8(2^n-1)
elseif phi(s) == 1 && pi*[bitstring(s)[end-n+1:end]] == "R"
    CT += update_costs(s, phi, Cc, Cs)
    for j in 1:n
        if epsilon[j] <= epsilon[k]
            epsilon[j] = epsilon[k] - log(rand())
            T[j] = 0
        end
    end
    s = UInt8(2^n-1)
end

epsilon_min = minimum([Inf; filter(I -> !(I in epsilon_t), epsilon)])

#Todo lo que pasa en el salto

elseif X_t[end-1]+mu*dt <= epsilon_min <= X_t[end-1]+mu*dt+J_i

I=findall(m-> (X_t[end-1]+mu*dt <= m <= X_t[end-1]
+ mu*dt+J_i),epsilon)

```

```

for i in I
    T[i] = t[end]
    s = s & ~(UInt8(1) << (n-i))
    push!(epsilon_t, epsilon[i])
end

if phi(s) == 0
    push!(Tm, T[I[1]])
    CT += update_costs(s, phi, Cc, Cs)
    push!(C, CT)
    CT = 0
    for j in 1:n
        if epsilon[j] <= X_t[end]
            epsilon[j] = X_t[end] - log(rand())
            T[j] = 0
        end
    end
    s = UInt8(2^n-1)
elseif phi(s) == 1 && pi*[bitstring(s)[end-n+1:end]] == "R"
    CT += update_costs(s, phi, Cc, Cs)
    for j in 1:n
        if epsilon[j] <= X_t[end]
            epsilon[j] = X_t[end] - log(rand())
            T[j] = 0
        end
    end
    s = UInt8(2^n-1)
end
end

```

```

        epsilon_min = minimum([Inf; filter(I -> !(I in epsilon_t), epsilon)])
    end

    end

    dt,J_i = update_state_time(t, X_t, J, mu, lambda, gamma,THorizon)

end

push!(C,CT)

Ccycle = mean(C[2:end-1]) # Costo hasta la primera falla del sistema
Tcycle = mean(diff(Tm[2:end-1])) # Tiempo hasta la primera falla del sistema
LTMC = sum(C) / THorizon # Costo medio a largo plazo
LTMN = (length(Tm)-1) / THorizon # Cantidad esperada de fallas en el largo plazo
return THorizon, CT, Ccycle, Tcycle , LTMC, LTMN
end

# -----
# ANÁLISIS DE RESULTADOS
# -----

function analizar_resultados(resultados,text=false)

    grouped_data = combine(groupby(resultados, :THorizonte),

        :Ccycle => mean => :media_ECcycle,
        :Ccycle => std => :desviacion_std_ECcycle,
        :Tcycle=> mean => :media_ETcycle,
        :Tcycle=> std => :desviacion_std_ETcycle,
        :LTMC => mean => :media_LTMC,
        :LTMC => std => :desviacion_std_LTMC,
        :LTMN => mean => :media_LTMN,
        :LTMN => std => :desviacion_std_LTMN)

    # Mostrar resultados

    for row in eachrow(grouped_data)

        if text

```

```

println("THorizonte: $(row.THorizonte)")
println("Media de ECcycle: $(row.media_ECcycle)")
println("Desviación estándar de ECcycle: $(row.desviacion_std_ECcycle)")
println("Media de ETcycle: $(row.media_ETcycle)")
println("Desviación estándar de ETcycle: $(row.desviacion_std_ETcycle)")
println("Media de LTMC: $(row.media_LTMC)")
println("Desviación estándar de LTMC: $(row.desviacion_std_LTMC)")
println("Media de LTMN: $(row.media_LTMN)")
println("Desviación estándar de LTMN: $(row.desviacion_std_LTMN)")
println("-"^30)

else
    continue
end

end

end

```

A.2. Iteración De Valor

```

struct MDPs
    STATES::Vector{UInt8}
    ACTIONS::Vector{String}
    phi::Function
    Cc::Float64
    Cs::Float64
    alpha::Float64
    PP::Matrix{Float64}
end

function componente_operativa(estados::UInt8, componente::Int)
    celda_componente = UInt8(1) << (componente - 1)
    return (estados & celda_componente) != 0
end

```

```

end

function CONNECTIONS(A, B, a, phi)
    if a == "R" && B != (2^n - 1)
        return false
    elseif a == "NR" && (A == B || (phi(A) == 0 && phi(B) == 0) || A & B != B)
        return false
    else
        return true
    end
end

function acciones_permitidas(s,a,phi)
    if a == "R" && s == (2^n-1)
        return false
    elseif phi(s) == 0 && a == "NR"
        return false
    else
        return true
    end
end

function p(PP, s0, s1, a)
    n=length(PP[1:end,1])-1
    if a == "NR"
        prob = PP[(n + 1) - count_ones(s0), (n + 1) - count_ones(s1)]
                                                    / binomial(count_ones(s0), count_ones(s1))
        return prob
    elseif a == "R"
        return 1
    end
end

```

```

end

function r(s, Cc, Cs, a, phi)

    if a == "R"

        return Cc * (n-count_ones(s)) + (1-phi(s))*Cs

    elseif a=="NR"

        return 0

    end

end

function Q_Value(mdp, s, a, U)

    return r(s ,mdp.Cc ,mdp.Cs ,a , mdp.phi)

        + mdp.alpha * sum(p(mdp.PP, s, s1, a) * U[s1+1]

            for s1 in mdp.STATES if CONNECTIONS(s, s1,a, mdp.phi))

end

function VALUE_ITERATIO(mdp, tol)

    U0 = zeros(Float64 , 2^n)
    U = copy(U0)
    Un = [U0]
    delta = Inf

    while delta >= tol * (1 - mdp.alpha) / mdp.alpha

        U = copy(Un[end])

        for s0 in mdp.STATES

            U[s0+1] = maximum(Q_Value(mdp, s0, a, U)

                for a in mdp.ACTIONS if acciones_permitidas(s0, a, mdp.phi))

            end

            delta = maximum(abs(Un[end][s0+1] - U[s0+1]) for s0 in mdp.STATES)

        push!(Un, U)

    end

    return Un

end

function pi_V(mdp, V)

```

```

pi* = Dict( bitstring(estados[i])[end-n+1:end] = > "" for i in 1:2^n)
for s0 in mdp.STATES
    aux = Dict(a = > Q_Value(mdp, s0, a, V)
        for a in mdp.ACTIONS
            if acciones_permitidas(s0, a, mdp.phi))
    pi*[bitstring(s0)[end-n+1:end]] = argmax(aux)
end
return pi*
end

```

Bibliografía

- [Bel54] Richard Bellman. The theory of dynamic programming. *Bulletin of the American Mathematical Society*, 60(6):503–515, 1954. [8](#)
- [Bel57] Richard Bellman. *Dynamic Programming*. Princeton University Press, Princeton, NJ, 1957. Republished 2003: Dover, ISBN 0-486-42809-5. [8](#)
- [Ber96] Jean Bertoin. *Lévy processes*, volume 121. Cambridge university press Cambridge, 1996. [39](#), [42](#)
- [BL25] Javiera Barrera and Guido Lagos. Trabajo en preparación. En preparación, 2025. [52](#)
- [Der70] Cyrus Derman. *Finite state Markovian decision processes*. Academic Press, Inc., 1970. [13](#)
- [HM12] Christian Hering and Jan-Frederik Mai. Moment-based estimation of extendible marshall-olkin copulas. *Metrika*, 75(5):601–620, 2012. [45](#)
- [Kre91] Erwin Kreyszig. *Introductory functional analysis with applications*, volume 17. John Wiley & Sons, 1991. [23](#), [25](#)
- [Mai10] Jan-Frederik Mai. *Extendibility of Marshall-Olkin distributions via Lévy subordinators and an application to portfolio credit risk*. PhD thesis, Technische Universität München, 2010. [45](#), [50](#)
- [MO67] Albert W Marshall and Ingram Olkin. A multivariate exponential distribution. *Journal of the American Statistical Association*, 62(317):30–44, 1967. [33](#)
- [MS09] Jan-Frederik Mai and Matthias Scherer. Lévy-frailty copulas. *Journal of Multivariate Analysis*, 100(7):1567–1585, 2009. [38](#)

- [oT10] Massachusetts Institute of Technology. 14.123 microeconomic theory iii. MIT OpenCourseWare, 2010. Instructor: Prof. Muhamet Yildiz. Retrieved from <http://ocw.mit.edu>.
17
- [Put14] Martin L. Puterman. *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons, 2014. 8, 13, 21, 24, 25