



UNIVERSIDAD DE AYSÉN

DEPARTAMENTO DE CIENCIAS NATURALES Y TECNOLOGÍA

CARRERA DE INGENIERÍA CIVIL INFORMÁTICA

**Diseño de un Modelo de Simulación
para el Análisis de Resiliencia en la
Cadena de Suministro de Gas Licuado de
Petróleo de Aysén**

Memoria para optar al título de Ingeniero Civil Informático

Autor: Carlos Subiabre Saldivia

Mentora: Natacha Pino Acuña

Coyhaique, Chile

2025

Agradecimientos

Deseo expresar mi agradecimiento a las personas que hicieron posible la realización de esta memoria.

A mi familia, por su constante apoyo y paciencia a lo largo de mis años de estudio. Su confianza fue fundamental para llegar a esta etapa.

A mi mejor amiga, por su invaluable amistad y por su aliento en los momentos más difíciles de este proceso.

A mi mentora, Profesora Natacha Pino Acuña, por su guía, tiempo y dedicación. Sus conocimientos y consejos fueron esenciales para el desarrollo de esta memoria.

Resumen

La cadena de suministro de Gas Licuado de Petróleo (GLP) en la Región de Aysén constituye un sistema sociotécnico de alta criticidad, caracterizado por una topología logística sin redundancia y una capacidad de respuesta endógena insuficiente para mitigar las disruptpciones exógenas recurrentes. El diagnóstico técnico actual, si bien es exhaustivo, es de naturaleza estática y carece de herramientas para evaluar dinámicamente el impacto de los riesgos o el retorno en resiliencia de las inversiones propuestas.

Este trabajo de titulación aborda dicha brecha metodológica mediante el diseño, implementación y validación de un prototipo de simulación de eventos discretos. El artefacto computacional desarrollado modela la interacción de los parámetros logísticos clave —capacidad de almacenamiento, políticas de inventario, demanda estocástica y, crucialmente, la frecuencia y duración de las interrupciones de la ruta de suministro— concentrados en el nodo de almacenamiento primario de Coyhaique, que actúa como centro neurálgico del sistema regional.

El objetivo es crear un laboratorio virtual que permita cuantificar la resiliencia del sistema bajo diferentes escenarios. Mediante un diseño de experimentos formal, se evaluará la sensibilidad del sistema a distintos parámetros, buscando confirmar la hipótesis de que la resiliencia es significativamente más sensible a la duración de las disruptpciones de ruta que a las variaciones en la capacidad de almacenamiento. El prototipo validado sienta una base metodológica para la toma de decisiones informadas, instrumentalizando una recomendación explícita de la política pública regional y contribuyendo al fortalecimiento de la seguridad energética de Aysén.

Palabras Clave: Simulación de Eventos Discretos, Resiliencia de Cadenas de Suministro, Gestión de Inventarios, Análisis de Riesgos, Seguridad Energética, Ingeniería de Sistemas.

Índice general

Agradecimientos	I
Resumen	II
Índice general	III
Índice de figuras	V
Índice de cuadros	VII
1 Introducción	1
2 Planteamiento del Problema	2
2.1. Caracterización del Sistema	2
2.2. Análisis de vulnerabilidades	3
2.3. Limitaciones del diagnóstico actual	4
3 Pregunta de Investigación e Hipótesis	6
3.1. Pregunta de Investigación	6
3.2. Hipótesis Central	6
4 Objetivos	8
4.1. Objetivo General	8
4.2. Objetivos Específicos	8
5 Marco Teórico	10
5.1. Gestión de Inventarios en Cadenas de Suministro	10
5.2. Simulación de Eventos Discretos	15
6 Estado del Arte	29
6.1. DES en Cadenas de Suministro: Aplicaciones Consolidadas	29

ÍNDICE GENERAL	IV
6.2. Modelos DES de Políticas de Inventario bajo Incertidumbre	30
6.3. Diagnóstico del Sistema GLP Aysén	31
6.4. Vacío Metodológico	32
6.5. Contribución de esta Tesis	32
7 Diseño Metodológico	34
7.1. Fase 1: Modelo Conceptual	34
7.2. Fase 2: Implementación del Prototipo (Objetivo 2)	37
7.3. Fase 3: Evaluación y Experimentación (Objetivo 3)	41
8 Resultados	43
8.1. Validación del Modelo de Simulación	43
8.2. Análisis Descriptivo del Rendimiento	44
8.3. Análisis Estadístico Inferencial	49
8.4. Prueba de Hipótesis: Análisis de Sensibilidad	52
8.5. Rendimiento Computacional del Experimento	58
8.6. Resumen del Capítulo	59
9 Discusión	61
9.1. Interpretación de los Hallazgos	61
9.2. Alcance y rango de validez del modelo	62
9.3. Limitaciones del Estudio	63
9.4. Trabajo futuro	64
10 Conclusiones y Proyección del Trabajo	66
10.1. Síntesis del Problema y la Solución Propuesta	66
10.2. Contribuciones Esperadas	67
10.3. Limitaciones y Líneas de Trabajo Futuro	67
Referencias	69

Índice de figuras

5.1. Modelo de inventario (Q, R) con Punto de Reorden (ROP) y Stock de Seguridad (SS) bajo incertidumbre en el <i>lead time</i> . El sistema reordena cantidad Q cuando el inventario alcanza el nivel R , manteniendo un buffer SS para proteger contra variabilidad.	11
5.2. El <i>trade-off</i> entre eficiencia y resiliencia en la gestión de cadenas de suministro. La frontera eficiente muestra las configuraciones óptimas de Pareto. El sistema Aysén opera actualmente en zona de alta vulnerabilidad; la Propuesta 10.4 (+250 TM) lo desplaza hacia mayor resiliencia con inversión moderada.	14
7.1. Modelo Conceptual del Sistema de Distribución de GLP en Aysén. El diagrama muestra el flujo operacional (azul), las disruptiones exógenas (rosa/terracota), los parámetros configurables (gris) y la métrica de resiliencia (borgoña). La política (Q, R) controla el reabastecimiento basándose en el nivel de inventario.	35
7.2. Diagrama de clases UML del simulador. <i>SimulacionGlpAysen</i> es el orquestador que utiliza <i>HubCoyhaique</i> (recurso con inventario) y <i>RutaSuministro</i> (generador de disruptiones).	39
8.1. Distribuciones del nivel de servicio por configuración experimental. Los violin plots muestran la densidad de probabilidad completa, mediana (línea negra) y media (línea roja) basados en 10,000 réplicas por configuración. . .	45
8.2. Distribución KDE: Status Quo con disruptiones cortas (7 días). Media: 84.32 %, DE: 3.49 %.	46
8.3. Distribución KDE: Status Quo con disruptiones medias (14 días). Media: 81.14 %, DE: 3.76 %.	47
8.4. Distribución KDE: Status Quo con disruptiones largas (21 días). Media: 78.13 %, DE: 4.48 %.	47

8.5. Distribución KDE: Propuesta con disruptciones cortas (7 días). Media: 98.82 %, DE: 1.15 %.	48
8.6. Distribución KDE: Propuesta con disruptciones medias (14 días). Media: 97.22 %, DE: 2.30 %.	48
8.7. Distribución KDE: Propuesta con disruptciones largas (21 días). Media: 94.70 %, DE: 3.97 %.	49
8.8. Q-Q Plot: Status Quo - Disrupciones cortas. El test de Shapiro-Wilk evalúa la hipótesis de normalidad.	50
8.9. Q-Q Plot: Status Quo - Disrupciones medias.	51
8.10. Q-Q Plot: Status Quo - Disrupciones largas.	52
8.11. Q-Q Plot: Propuesta - Disrupciones cortas.	53
8.12. Q-Q Plot: Propuesta - Disrupciones medias.	54
8.13. Q-Q Plot: Propuesta - Disrupciones largas.	55
8.14. Efectos principales de los factores experimentales. Panel (A): efecto de la capacidad de almacenamiento (factor endógeno). Panel (B): efecto de la duración máxima de disruptciones (factor exógeno). Las barras de error representan intervalos de confianza al 95 %.	56
8.15. Nivel de servicio promedio por combinación de factores. Los valores más bajos (rojos) indican menor resiliencia del sistema.	56
8.16. Análisis de sensibilidad comparativo. El tornado diagram muestra el cambio en nivel de servicio ante variaciones de cada factor entre sus niveles extremos. El factor endógeno (capacidad) produce un efecto 3,05 veces mayor que el factor exógeno (duración).	57
8.17. Comparación de configuraciones mediante boxplots. Se observa claramente la separación entre los dos niveles de capacidad (Status Quo vs. Propuesta), evidenciando el dominio del factor endógeno.	57

Índice de cuadros

7.1. Diseño Experimental Monte Carlo para Evaluación de Resiliencia.	42
8.1. Parámetros de entrada del modelo de simulación.	44
8.2. Estadísticas descriptivas del nivel de servicio (%) por configuración.	44
8.3. Análisis de Varianza (ANOVA) de dos vías para el nivel de servicio.	49
8.4. Métricas de rendimiento del experimento Monte Carlo.	58

CAPÍTULO 1

Introducción

La Región de Aysén depende del 100 % de Gas Licuado de Petróleo (GLP) importado vía terrestre desde Argentina. El sistema opera con tres distribuidores (Abastible, Lipigas, Gasco) que almacenan 431 toneladas en Coyhaique, el nodo logístico central. Esta capacidad proporciona 8,2 días de autonomía para la demanda regional.

El suministro enfrenta disrupciones recurrentes de la Ruta 7 (única vía de acceso): cierres por nevadas, derrumbes y conflictos sociales en Argentina. El informe CIEP 2025 documenta una frecuencia de 4 eventos/año, con disrupciones históricas de hasta 21 días (conflicto Argentina 2021). Esta realidad genera un gap estructural: 8,2 días de autonomía vs. disrupciones de hasta 21 días.

Este trabajo desarrolla un modelo de simulación de eventos discretos para cuantificar la resiliencia del sistema bajo diferentes escenarios de capacidad de almacenamiento y duración de disrupciones. El modelo permite comparar la Propuesta 10.4 de Gasco (expansión de capacidad a 681 TM por \$1,5M USD) vs. estrategias de mitigación de disrupciones (rutas alternativas, protocolos con Argentina).

Estructura del documento: Capítulo 2 caracteriza el sistema y sus vulnerabilidades. Capítulo 3 plantea la pregunta de investigación: ¿cuál factor (capacidad vs. disrupciones) domina la resiliencia del sistema? Capítulos 5 y 6 presentan fundamentos teóricos. Capítulo 7 detalla el diseño del experimento Monte Carlo (60.000 simulaciones). Capítulos 8 y 9 presentan resultados y análisis.

CAPÍTULO 2

Planteamiento del Problema

La Región de Aysén enfrenta un riesgo sistémico en su suministro de GLP. Este capítulo caracteriza el sistema (demanda, infraestructura, actores), identifica sus vulnerabilidades (exógenas y endógenas), y argumenta por qué los métodos de análisis actuales (matrices de riesgo estáticas) son insuficientes para evaluar la resiliencia dinámica del sistema.

2.1. Caracterización del Sistema

El sistema de suministro de GLP en Aysén opera bajo condiciones que lo definen como un sistema complejo y crítico. Su criticidad se deriva de la confluencia de una demanda energética elevada, una estructura de mercado oligopólica y una infraestructura precaria.

Demanda energética y crecimiento proyectado

La demanda energética de la región es un factor distintivo a nivel nacional. Con un consumo per cápita de 27.25 Gcal, que excede en un 65 % la media de Chile, la población depende intensivamente de un flujo energético constante para sostener funciones básicas. El GLP satisface nichos de consumo —cocción de alimentos y agua caliente sanitaria— para los cuales no existen sustitutos inmediatos a gran escala. Esta inelasticidad fundamental de la demanda convierte al GLP en un recurso de alta prioridad social, cuya interrupción tiene consecuencias directas e inmediatas sobre el bienestar de la población.

Adicionalmente, la demanda no es estática. El sistema enfrenta un crecimiento sostenido del 3.8 % anual, impulsado por el desarrollo demográfico y económico de la región. A esto se suma un factor disruptivo futuro: la proyectada incorporación de una central térmica que consumirá 14.4 ton/día de GLP. Este crecimiento y cambio estructural implican que la presión sobre la ya frágil cadena de suministro se intensificará de manera no lineal en los próximos años.

Estructura de la cadena de suministro

La estructura de la cadena de suministro, detallada en el informe de referencia Centro de Investigación en Ecosistemas de la Patagonia (CIEP) y Quark SpA, 2024¹, se caracteriza por una topología logística lineal con una dependencia absoluta de fuentes exógenas. El informe es categórico al afirmar que “La totalidad del gas licuado que llega a la región, lo hace vía camiones que transitan por el paso Huemules desde Argentina”. Este diseño, carente de redundancia, convierte a la ruta terrestre en un punto único de falla (*Single Point of Failure*) de manual. La magnitud de esta dependencia se cuantifica en el hecho de que el 86 % del recorrido terrestre desde la planta de Cabo Negro hasta Coyhaique transcurre por territorio argentino, lo que implica que la soberanía logística y la seguridad del suministro regional están estructuralmente comprometidas por la dinámica de un país vecino.

2.2. Análisis de vulnerabilidades

El sistema presenta vulnerabilidades tanto exógenas (amenazas externas) como endógenas (limitaciones de capacidad interna).

Vulnerabilidades exógenas

El principal vector de riesgo es la incertidumbre estocástica en el aprovisionamiento. El informe de riesgos Centro de Investigación en Ecosistemas de la Patagonia (CIEP) y Quark SpA, 2024 identifica dos amenazas dominantes. Primero, el evento de “Nevadas / Cierre cruce fronterizo” (Tag #57), clasificado con una probabilidad de Nivel 4 (“Casi Seguro”), lo que implica una frecuencia de ocurrencia de “al menos una vez cada tres meses”. Segundo, el riesgo de “Conflictos sociales en Argentina” (Tag #45), que,

¹El informe, encargado por la Seremía de Energía de Aysén, fue finalizado y presentado en 2025. A la fecha de esta publicación, se encuentra en proceso de difusión pública.

aunque menos frecuente, ha demostrado la capacidad de paralizar el suministro por períodos de hasta tres semanas. Esta realidad es corroborada por la autoridad energética regional, quien identifica la “conflictividad social” y la “mantención de los caminos” en Argentina como la principal amenaza para la continuidad del suministro (Laibe, T., comunicación personal, 11 de junio, 2025). El sistema, por tanto, enfrenta amenazas recurrentes y de duración prolongada que introducen una alta variabilidad en el tiempo de entrega (*lead time*).

Vulnerabilidades endógenas

La capacidad de almacenamiento en Coyhaique es de 431 TM (Abastible 150, Lipigas 240, Gasco 41) Centro de Investigación en Ecosistemas de la Patagonia (CIEP) y Quark SpA, 2024, proporcionando 8,2 días de autonomía vs. disrupturas documentadas de hasta 21 días.

El informe CIEP 2025 identifica que "Gasco tiene una capacidad demasiado reducida para su nivel de ventas" (41 TM = 0,78 días de autonomía). Esta situación refleja un Dilema del Prisionero en el mercado oligopólico: minimizar costos de inventario individualmente (estrategia racional) degrada la resiliencia colectiva.

Propuesta 10.4 (Gasco): El informe documenta que Gasco está desarrollando un proyecto de expansión para incrementar su capacidad en 250 TM (de 41 a 291 TM), elevando la capacidad total del sistema de 431 TM a 681 TM. La inversión estimada es \$1,5 millones USD. Esta expansión aumentaría la autonomía teórica del sistema de 8,2 a 13 días.

El presente trabajo evalúa cuantitativamente el retorno en resiliencia de esta inversión bajo diferentes escenarios de disrupturas.

2.3. Limitaciones del diagnóstico actual

Vacío en la planificación energética regional

La 'Política Energética 2050 Región de Aysén' se enfoca exclusivamente en el sector eléctrico, omitiendo la cadena de suministro de combustibles.

Análisis estático vs. dinámico

El informe CIEP 2025 es exhaustivo pero estático: identifica riesgos mediante una matriz de probabilidad × impacto, pero no puede cuantificar:

- ¿Cuánto tiempo puede operar el sistema bajo disruptiones largas?
- ¿Qué tan frecuentemente fallará el sistema (quiebres de stock)?
- ¿Cuál es el retorno en resiliencia de la inversión de \$1,5M USD (Propuesta 10.4)?

Gestión reactiva de crisis

La gestión operativa de crisis se basa en protocolos manuales: "llamar a todas las empresas, construir un Excel y sacar una foto de cuánta disponibilidad de combustible hay" (Laibe, T., comunicación personal, 11 de junio, 2025). No existe una herramienta de análisis dinámico.

Pregunta que el diagnóstico actual no puede responder

¿Cuál es la reducción en probabilidad de quiebre de stock que genera la inversión de \$1,5M USD (Propuesta 10.4 de Gasco), bajo un escenario de disruptiones con frecuencia 4 eventos/año y duración estocástica de hasta 21 días?

Esta tesis desarrolla el modelo para responder esta pregunta.

CAPÍTULO 3

Pregunta de Investigación e Hipótesis

El Capítulo 2 identificó que el sistema de suministro de GLP en Aysén opera con dos tipos de vulnerabilidades:

- **Vulnerabilidad exógena:** Disrupciones de la Ruta 7 (frecuencia: 4 eventos/año, duración: hasta 21 días).
- **Vulnerabilidad endógena:** Capacidad de almacenamiento limitada ($431 \text{ TM} = 8,2$ días de autonomía).

Este capítulo plantea la pregunta de investigación y la hipótesis central que guían el diseño del modelo de simulación y el experimento Monte Carlo.

3.1. Pregunta de Investigación

¿Cuál de los dos factores (duración de disruptciones vs. capacidad de almacenamiento) tiene mayor impacto en la resiliencia del sistema?

Específicamente: ¿Cómo diseñar un modelo de simulación para cuantificar la sensibilidad relativa del nivel de servicio a estos dos factores?

3.2. Hipótesis Central

Hipótesis: La resiliencia del sistema de suministro de GLP de Aysén es significativamente más sensible a la duración de las disruptciones (factor exógeno) que a la capacidad de almacenamiento (factor endógeno).

Justificación de la hipótesis

Esta hipótesis se basa en dos observaciones del sistema actual:

1. **Disrupciones largas superan la autonomía del sistema:** La capacidad actual (431 TM) proporciona 8,2 días de autonomía. Disrupciones documentadas de hasta 21 días (conflicto Argentina 2021) exceden esta capacidad por un factor de 2,5x.
2. **Expansión de capacidad tiene retornos decrecientes:** La Propuesta 10.4 de Gasco incrementa capacidad en 58 % (de 431 a 681 TM), pero la autonomía solo aumenta a 13 días. Disrupciones de 21 días seguirían generando quiebres de stock.

Operacionalización de la hipótesis

La hipótesis se prueba mediante un experimento factorial 2×3 que compara:

Factor endógeno (capacidad):

- Status Quo: 431 TM
- Propuesta: 681 TM (+58 %)

Factor exógeno (duración de disruptciones):

- Corta: Triangular(3, 3.5, 7) días
- Media: Triangular(3, 7, 14) días
- Larga: Triangular(3, 10.5, 21) días

La hipótesis se confirma si el ratio de sensibilidad $\frac{\text{Efecto exógeno}}{\text{Efecto endógeno}} > 1$.

Implicaciones para política pública

Si la hipótesis se confirma, las inversiones en mitigación de disruptciones (rutas alternativas, protocolos con Argentina, mejora de Ruta 7) tendrían mayor retorno en resiliencia que la inversión de \$1,5M USD en expansión de capacidad (Propuesta 10.4).

CAPÍTULO 4

Objetivos

Para abordar la brecha metodológica identificada en el capítulo 2 y responder a la pregunta de investigación formulada, el presente proyecto se estructura en torno a un conjunto de objetivos jerárquicos. Estos objetivos guían el proceso de diseño, desarrollo y validación del modelo de simulación propuesto, asegurando un enfoque sistemático para el análisis de la resiliencia del sistema de suministro de GLP.

4.1. Objetivo General

Diseñar un modelo de simulación de eventos discretos para cuantificar el impacto de parámetros logísticos críticos sobre la resiliencia del sistema de suministro de GLP en el nodo Coyhaique.

4.2. Objetivos Específicos

La consecución del objetivo general se desglosa en tres fases metodológicas secuenciales, cada una representada por un objetivo específico que define una etapa clave del proceso de investigación y desarrollo:

1. **Modelar** la dinámica conceptual de la cadena de suministro de GLP en Coyhaique, realizando una abstracción formal de sus componentes, procesos y parámetros clave a partir de la información técnica disponible.
2. **Implementar** el modelo de simulación mediante una arquitectura de software parametrizable que traduzca el modelo conceptual en un programa funcional.

3. **Evaluar** la resiliencia del sistema mediante un diseño experimental sobre el modelo validado, generando evidencia empírica para confirmar o refutar la hipótesis de trabajo.

CAPÍTULO 5

Marco Teórico

Este capítulo presenta los fundamentos teóricos del modelo de simulación. La sección 5.1 desarrolla la teoría de gestión de inventarios bajo incertidumbre y su aplicación numérica al sistema GLP de Aysén. La sección 5.2 describe el método de simulación de eventos discretos y su implementación computacional.

5.1. Gestión de Inventarios en Cadenas de Suministro

Política de inventario (Q, R)

La dinámica del inventario $I(t)$ en una planta de almacenamiento evoluciona según:

$$I(t) = I(t-1) - D(t) + A(t)$$

donde $D(t)$ es la demanda diaria y $A(t)$ son los arribos de producto. En Aysén, el lead time LT (tiempo de entrega desde Cabo Negro/Neuquén) es una variable aleatoria debido a disrupciones de la Ruta 7.

El modelo usa una política de revisión continua (Q, R) : ordenar cantidad fija Q cuando el inventario alcanza el Punto de Reorden R .

El cálculo del Punto de Reorden se formaliza como:

$$R = (\bar{D} \times \bar{LT}) + SS \quad (5.1)$$

donde \bar{D} es la demanda promedio y \bar{LT} es el *lead time* promedio. El componente crucial es el Stock de Seguridad (SS), el buffer que protege contra la variabilidad. Se calcula como:

$$SS = Z_\alpha \times \sqrt{\bar{LT}\sigma_D^2 + \bar{D}^2\sigma_{LT}^2}$$

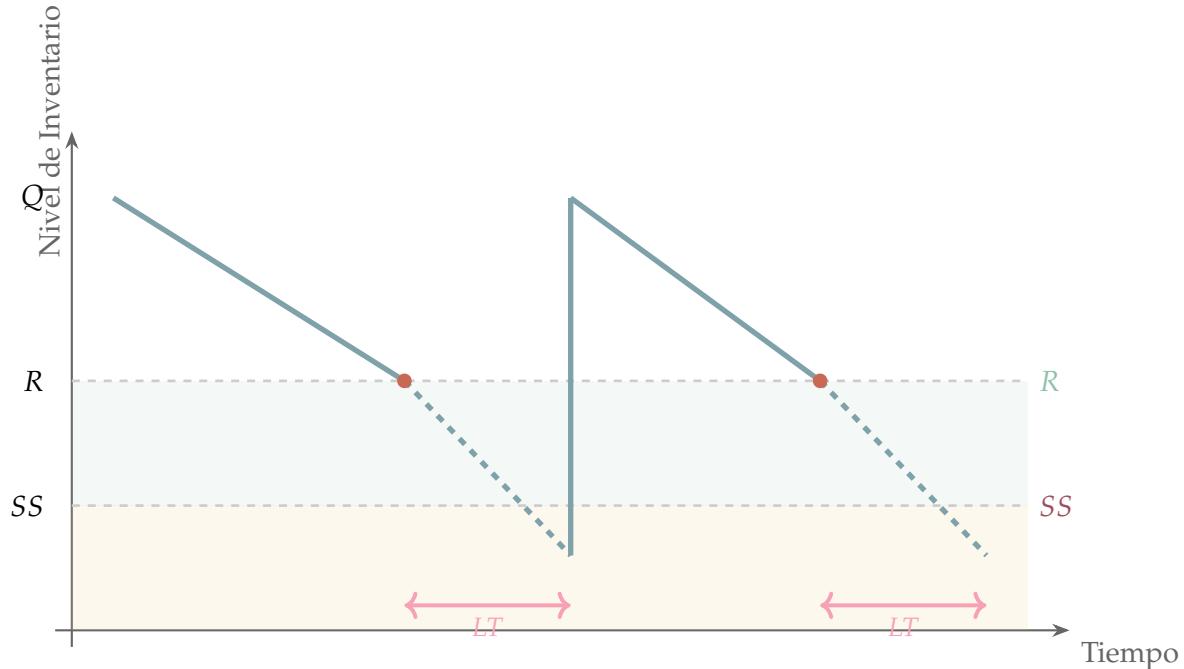


Figura 5.1: Modelo de inventario (Q, R) con Punto de Reorden (ROP) y Stock de Seguridad (SS) bajo incertidumbre en el *lead time*. El sistema reordena cantidad Q cuando el inventario alcanza el nivel R , manteniendo un buffer SS para proteger contra variabilidad.

[ig:inventory-model-detailed](#)

donde Z_α es el factor de servicio para un nivel de servicio deseado α , σ_D es la desviación estándar de la demanda, y σ_{LT} es la desviación estándar del *lead time*. Dado que en Aysén la demanda es relativamente estable pero el *lead time* es altamente volátil, la ecuación se simplifica y evidencia que $SS \propto \sigma_{LT}$. Esto formaliza matemáticamente la vulnerabilidad central del sistema: una alta variabilidad en el tiempo de la ruta (σ_{LT} elevado) exige un alto stock de seguridad para mantener la continuidad del servicio.

Aplicación Numérica al Sistema GLP Aysén

Para concretar la teoría expuesta, aplicamos las fórmulas al caso específico del sistema GLP de Aysén. El sistema opera con demanda promedio $\bar{d} = 52.5$ TM/día (calibrada al mes de mayor consumo), un lead time nominal $\bar{LT} = 6$ días (correspondiente a la distancia aproximada de 1,400 km desde Cabo Negro/Neuquén), capacidad actual $C_{sq} = 431$ TM, y capacidad propuesta $C_{prop} = 681$ TM.

Cálculo del Punto de Reorden

Aplicando la ecuación (5.1) con los parámetros del sistema:

$$R = (\bar{d} \times \overline{LT}) + SS = (52.5 \times 6) + SS = 315 + SS \quad (5.2)$$

donde el stock de seguridad SS depende del nivel de servicio deseado y la variabilidad del sistema. Para la configuración implementada en el modelo, se adopta una política de revisión continua con $R = 0.5 \times C$ (50 % de la capacidad como punto de reorden) y $Q = 0.5 \times C$ (50 % de la capacidad como cantidad a ordenar). Esta parametrización resulta en $R_{sq} = 215.5$ TM y $Q_{sq} = 215.5$ TM para el Status Quo, mientras que la Propuesta opera con $R_{prop} = 340.5$ TM y $Q_{prop} = 340.5$ TM.

Autonomía del Sistema

La autonomía nominal del sistema (días de inventario disponible) se calcula como:

$$\text{Autonomía} = \frac{C}{\bar{d}} \quad (5.3)$$

Para las dos configuraciones:

$$\text{Autonomía}_{sq} = \frac{431}{52.5} \approx 8.2 \text{ días} \quad (5.4)$$

$$\text{Autonomía}_{prop} = \frac{681}{52.5} \approx 13.0 \text{ días} \quad (5.5)$$

Observación crítica: La autonomía del Status Quo (8.2 días) es apenas superior al lead time nominal (6 días). Esto implica que cualquier disrupción que extienda el lead time más allá de 2 días puede causar quiebre de stock, incluso sin variabilidad en la demanda.

Modelación de la Demanda Estocástica

La demanda diaria se modela como un proceso estocástico con componente estacional y ruido aleatorio:

$$D(t) = D_{base} \cdot \left(1 + 0.25 \sin \left(\frac{2\pi(t - 172)}{365} \right) \right) \cdot \epsilon(t) \quad (5.6)$$

donde:

- $D_{base} = 52.5$ TM/día es la demanda base

- El término sinusoidal modela la estacionalidad anual con amplitud $\pm 25\%$
- La fase $(t - 172)$ centra el pico en el día 172 (invierno austral, julio)
- $\epsilon(t) \sim \mathcal{N}(1.0, 0.15)$ es el ruido estocástico diario con $\pm 15\%$ de variabilidad

Esta formulación captura dos fuentes de variabilidad: (1) el ciclo estacional predecible, y (2) las fluctuaciones aleatorias día a día.

Modelación de Disrupciones

Las disrupciones de la Ruta 7 se modelan mediante dos distribuciones:

1. Frecuencia de disrupciones: Proceso de Poisson con tasa $\lambda = 4$ eventos/año (documentado en Centro de Investigación en Ecosistemas de la Patagonia (CIEP) y Quark SpA, 2024). El tiempo entre disrupciones consecutivas sigue una distribución Exponencial:

$$T_{entre} \sim \text{Exp}\left(\frac{\lambda}{365}\right) = \text{Exp}(0.0110) \quad (5.7)$$

con media $E[T_{entre}] = 365/4 = 91.25$ días.

2. Duración de disrupciones: Distribución Triangular $\text{Tri}(a, b, c)$ con parámetros dependientes del escenario experimental. El escenario Corta usa $\text{Tri}(3, 3.5, 7)$ días, el escenario Media usa $\text{Tri}(3, 7, 14)$ días, y el escenario Larga usa $\text{Tri}(3, 10.5, 21)$ días (basado en el conflicto de Argentina 2021).

La distribución triangular permite modelar la incertidumbre con parámetros interpretables: mínimo histórico (a), valor más probable (b), y máximo observado (c).

Implicación del Análisis Numérico

El gap estructural entre autonomía (8.2 días) y el lead time bajo disrupción potencial ($6 +$ hasta 21 días) revela la criticidad del sistema. Matemáticamente:

$$\text{Margen de seguridad} = \text{Autonomía} - (\bar{LT} + D_{max}) = 8.2 - (6 + 21) < 0 \quad (5.8)$$

Este resultado negativo confirma que el sistema Status Quo opera en régimen de subcapacidad crónica ante disrupciones largas.

Resiliencia en Cadenas de Suministro

El enfoque tradicional de eficiencia ha sido complementado por el concepto de resiliencia, definida como la capacidad de una cadena de suministro para absorber, adap-

tarse y recuperarse de disruptiones Christopher y Peck, 2004. Este concepto introduce un compromiso fundamental entre eficiencia (minimización de costos) y robustez (inversión en redundancia). El sistema de Aysén, como muestra la Figura ??, se encuentra en una zona de aparente eficiencia pero alta vulnerabilidad.

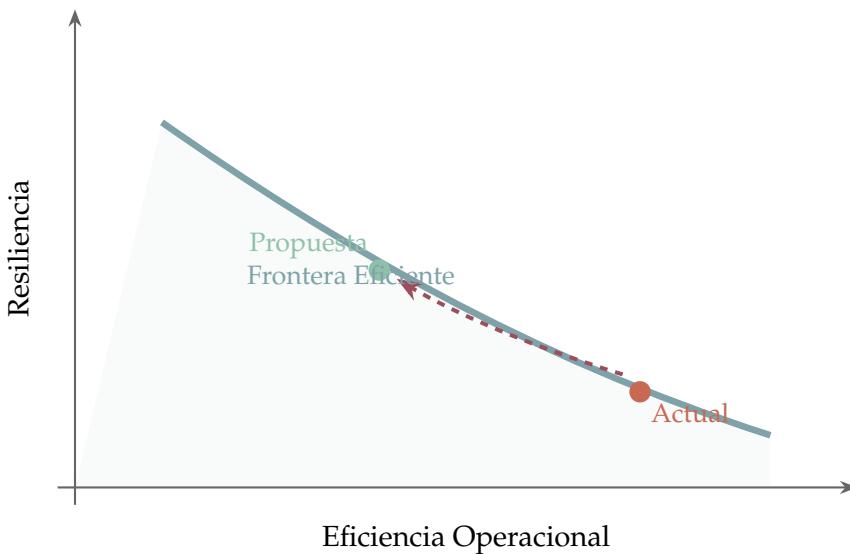


Figura 5.2: El *trade-off* entre eficiencia y resiliencia en la gestión de cadenas de suministro. La frontera eficiente muestra las configuraciones óptimas de Pareto. El sistema Aysén opera actualmente en zona de alta vulnerabilidad; la Propuesta 10.4 (+250 TM) lo desplaza hacia mayor resiliencia con inversión moderada.

[ig:resilience-tradeoff](#)

Dilema del Prisionero en mercados oligopólicos

El informe CIEP 2025 identifica que Gasco opera con capacidad "demasiado reducida para su nivel de ventas" (41 TM, 1 día de autonomía). Esta situación refleja un Dilema del Prisionero: minimizar costos de inventario individualmente (estrategia racional) genera baja resiliencia colectiva.

El modelo agrega los tres distribuidores en un hub único, por lo que no captura estas dinámicas competitivas. Una extensión futura podría implementar un modelo multi-agente para analizar estrategias de coordinación.

5.2. Simulación de Eventos Discretos

Componentes del modelo DES

El modelo representa el sistema como una secuencia cronológica de eventos. Los componentes son:

Entidades CamiónSuministro (con atributos: capacidad, origen).

Recursos PlantaAlmacenamiento (con atributos: capacidad_max, nivel_inventario, ROP), RutaTerrestre (con estado: abierta/cerrada).

Procesos GeneracionPedidos (cuando $I(t) \leq R$), ViajeSuministro (proceso estocástico con duración LT), ConsumoDiario (reduce $I(t)$), GeneradorDisrupciones (cambia el estado de la RutaTerrestre).

Verificación y Validación

Verificación: Asegurar que el código implementa correctamente el modelo conceptual (tests unitarios, code reviews).

Validación: Comparar salidas del modelo con datos del sistema real (autonomía esperada: 8,2 días vs. observada, frecuencia de disrupturas, etc.).

Diseño de Experimentos

El modelo se evalúa mediante un experimento factorial 2×3 (6 configuraciones, 10.000 réplicas c/u). ANOVA de dos vías identifica efectos principales e interacciones de los factores sobre el nivel de servicio.

Arquitectura Computacional del Sistema de Simulación

Esta sección presenta la arquitectura de software del simulador, las estructuras de datos empleadas, y el análisis de complejidad algorítmica.

Arquitectura de Software: Patrón Modelo-Experimento-Análisis

El sistema de simulación implementa una arquitectura modular de tres capas (Figura ??):

1. **Capa de Modelo** (`modelo.py`): Implementa la lógica de simulación DES usando SimPy. Contiene las clases SistemaGLP (orquestrador), HubCoyhaique (recurso con inventario), Pedido (entidad), y GeneradorDisrupciones (proceso estocástico).

2. **Capa de Experimento** (`experimento_montecarlo.py`): Orquesta la ejecución de 60,000 réplicas, gestiona la paralelización, y persiste resultados. Implementa el patrón Factory para instanciar configuraciones.
3. **Capa de Análisis** (`analisis_estadistico.py`): Procesa datos de salida (Pandas), ejecuta tests estadísticos (SciPy), y genera visualizaciones (Matplotlib/Seaborn).

Esta separación de responsabilidades facilita el testing: cada capa puede verificarse independientemente sin requerir las otras. El modelo puede reutilizarse para experimentos distintos (ej. cambiar de Monte Carlo a optimización de parámetros) sin modificar la lógica de simulación. Si se necesita cambiar un gráfico o test estadístico, solo se modifica la capa de análisis sin reejectar las 60,000 simulaciones.

SimPy: Framework DES basado en Coroutines

SimPy implementa simulación de eventos discretos usando el modelo de programación concurrente basado en *generators* (PEP 255) y *coroutines* de Python. A diferencia de threads o multiprocessing, las coroutines permiten concurrencia cooperativa sin overhead de context switching del sistema operativo.

Fundamentos de Coroutines en Python:

Una coroutine es una función que puede suspender su ejecución y ceder el control al scheduler (event loop) usando la palabra clave `yield`. Cuando la condición esperada se cumple (ej. timeout, recurso disponible), el scheduler reanuda la coroutine desde el punto exacto donde se suspendió.

```

1 def proceso_demanda_diaria(env, hub):
2     """Coroutine que se ejecuta dia a dia"""
3     dia = 0
4     while True:
5         demanda = calcular_demanda(dia)
6         hub.despachar(demanda)
7
8         # Suspende ejecucion por 1 dia (simulado)
9         yield env.timeout(1.0) # Control regresa al event loop
10        dia += 1

```

Listing 5.1: Ejemplo de coroutine en SimPy

Event Loop de SimPy:

El núcleo de SimPy es un event loop que gestiona la Future Event List (FEL) implementada como priority queue. El algoritmo es:

```

1 while FEL is not empty and current_time < end_time:
2     # 1. Extraer proximo evento (O(log n))
3     event = heappop(FEL)
4
5     # 2. Avanzar reloj de simulacion
6     current_time = event.time
7
8     # 3. Ejecutar callbacks del evento
9     for callback in event.callbacks:
10        callback() # Puede generar nuevos eventos
11
12    # 4. Nuevos eventos se insertan en FEL (O(log n))

```

Listing 5.2: Event loop de SimPy (simplificado)

La arquitectura del modelo GLP consiste en tres procesos concurrentes que se ejecutan cooperativamente:

```

1 def run_simulation(self):
2     # Lanza 3 procesos concurrentes
3     self.env.process(self.proceso_demanda_diaria())
4     self.env.process(self.proceso_reabastecimiento())
5     self.env.process(self.proceso_disrupciones())
6
7     # Ejecuta simulacion hasta t=365 dias
8     self.env.run(until=self.config.horizonte_dias)

```

Listing 5.3: Procesos concurrentes en SimPy

Cada proceso opera de forma autónoma pero interactúa con recursos compartidos (el inventario del hub) mediante el modelo de paso de mensajes de SimPy. Esta arquitectura refleja fielmente la naturaleza asíncrona y concurrente del sistema real.

Reproducibilidad Computacional

La reproducibilidad científica en simulación estocástica exige control total sobre la generación de números pseudoaleatorios. Se adopta el generador **Mersenne Twister** (MT19937) de NumPy, con estado inicializado mediante semillas únicas:

$$s_{c,r} = s_{base} + (c - 1) \times 100,000 + r \quad (5.9)$$

donde $s_{base} = 42$, $c \in \{1, \dots, 6\}$ es el identificador de configuración, y $r \in \{1, \dots, 10,000\}$ es el número de réplica. El espaciamiento de 100,000 unidades entre configuraciones ga-

rantiza independencia estadística entre réplicas (no existe traslape de secuencias pseudoaleatorias).

Ejecutar la réplica r de la configuración c con semilla $s_{c,r}$ genera siempre el mismo resultado numérico. Esto garantiza reproducibilidad exacta: repetir el experimento en diferentes máquinas o fechas produce datos idénticos. Las 60,000 simulaciones pueden ejecutarse en cualquier orden (incluso en paralelo) sin afectar los resultados, y réplicas con comportamiento anómalo pueden re-ejecutarse de forma determinista para debugging.

Distribuciones de Probabilidad en NumPy/SciPy

Las variables estocásticas del modelo se implementan utilizando los métodos del objeto `numpy.random.Generator`:

```

1 # Tiempo entre disruptores: Exponencial(lambda)
2 tiempo_entre = rng.exponential(365 / lambda_anual)
3
4 # Duración de disruptor: Triangular(a, b, c)
5 duracion = rng.triangular(left=3, mode=7, right=14)
6
7 # Ruido de demanda: Normal(mu, sigma)
8 ruido = rng.normal(loc=1.0, scale=0.15)
```

Listing 5.4: Generación de variables aleatorias

Cada instancia del generador se inicializa con la semilla correspondiente a la réplica actual:

```
1 rng = np.random.default_rng(seed=s_c_r)
```

Listing 5.5: Inicialización del generador

Esta arquitectura separa completamente la lógica del modelo (qué distribuciones usar) de la generación de valores específicos (secuencia pseudoaleatoria), facilitando verificación y testing.

Verificación Mediante Testing Automatizado

Para asegurar la correctitud de la implementación, el código incluye una suite de 24 tests unitarios que verifican:

- **Tests de configuración:** Validación de parámetros (ej. capacidad > 0 , $R \leq$ capacidad)

- **Tests de procesos:** Verificación de que cada proceso genera los eventos esperados
- **Tests de balance:** Invariante $\text{total_recibido} - \text{total_despachado} = \text{inventario_final} - \text{inventario_inicial}$
- **Tests de reproducibilidad:** Dos ejecuciones con misma semilla producen resultados idénticos

La suite se ejecuta automáticamente antes de cada experimento Monte Carlo, proporcionando una red de seguridad contra regresiones:

```
1 $ pytest tests/
2 ===== 24 passed in 3.42s =====
```

Listing 5.6: Ejecución de tests

Estructuras de Datos del Simulador

El núcleo de SimPy utiliza una **priority queue** (heap) para gestionar la lista de eventos futuros (FEL - Future Event List). La implementación en Python usa `heapq`, un min-heap binario donde la inserción de un evento toma $O(\log n)$ (siendo n el número de eventos pendientes), la extracción del siguiente evento toma $O(\log n)$, y la consulta del siguiente evento sin removerlo toma $O(1)$.

El estado del sistema se almacena en estructuras de datos Python:

```
1 class HubCoyhaique:
2     """Recurso con inventario limitado"""
3     def __init__(self, capacidad_tm):
4         self.inventario = 0.6 * capacidad_tm      # Estado: float
5         self.capacidad = capacidad_tm            # Constante: float
6         self.rop = 0.5 * capacidad_tm            # Threshold: float
7
8         # Historia temporal (time series)
9         self.inventario_historia = []           # List[(t, nivel)]
10        self.pedidos_pendientes = []             # Queue[Pedido]
11
12 class SistemaGLP:
13     """Orquestador del modelo"""
14     def __init__(self, env, config):
15         self.env = env                         # SimPy Environment
16         self.hub = HubCoyhaique(config.cap)    # Recurso
17         self.rng = np.random.default_rng()     # RNG state
18
19         # Metricas acumuladas (hash table)
```

```

20     self.métricas = {
21         'días_quiebre': 0,
22         'total_demanda': 0,
23         'total_satisfecho': 0
24     }

```

Listing 5.7: Estructuras de datos del sistema

Análisis de Complejidad Algorítmica

La complejidad temporal de una simulación individual es:

$$T(n, m) = O(n \log n + m \log m) \quad (5.10)$$

donde n es el número de eventos de demanda (365 días), y m es el número de eventos de disruptión ($\sim 4\text{-}6$ eventos). El término dominante es $O(n \log n)$ debido a la gestión de la priority queue.

Complejidad espacial: $O(n)$ para almacenar la serie temporal de inventario.

Complejidad total del experimento Monte Carlo:

$$T_{\text{total}} = O(R \times (n \log n + m \log m)) \quad (5.11)$$

donde $R = 60,000$ réplicas. Sin paralelización, esto equivale a $\sim 6\text{-}8$ horas en hardware estándar (AMD Ryzen 7, 16 GB RAM). Con paralelización en P cores, el tiempo se reduce a $O(R/P)$.

Optimización Computacional

El código implementa optimizaciones para reducir el tiempo de ejecución:

1. **Vectorización con NumPy:** Generación de demanda diaria usa operaciones vectorizadas en lugar de loops Python ($\sim 10\times$ más rápido).
2. **Evitar copies innecesarias:** Las métricas se acumulan in-place sin crear nuevas estructuras.
3. **Early termination:** Si el sistema detecta quiebre de stock irrecuperable, la simulación puede terminar anticipadamente (no implementado en versión actual).

Gestión de Experimentos a Gran Escala

El script `experimento_montecarlo.py` implementa:

- **Persistencia incremental:** Resultados guardados cada 1,000 réplicas en formato CSV (previene pérdida de datos ante crashes)
- **Progreso en tiempo real:** Barra de progreso con tqdm mostrando ETA
- **Reproducibilidad:** Metadata del experimento (timestamp, hardware, versiones de librerías) guardada en JSON

Tiempo total: ~6-8 horas para 60,000 simulaciones (1,7 simulaciones/segundo).

Patrones de Diseño Implementados

El código implementa varios patrones de diseño de software (Gang of Four) para mejorar mantenibilidad y extensibilidad.

Factory Pattern

El patrón Factory se usa para instanciar configuraciones de experimento. Esto separa la lógica de creación de objetos de la lógica de negocio.

```

1 class ConfiguracionFactory:
2     @staticmethod
3     def crear_status_quo(duracion_max: int) -> ConfiguracionSimulacion:
4         return ConfiguracionSimulacion(
5             capacidadHubTm=431,
6             puntoReordenTm=215.5,
7             cantidadPedidoTm=215.5,
8             duracionDisrupcionMaxDias=duracion_max,
9             semillaAleatoria=calcular_semilla(config_id, replica_id)
10        )
11
12     @staticmethod
13     def crear_propuesta(duracion_max: int) -> ConfiguracionSimulacion:
14         return ConfiguracionSimulacion(
15             capacidadHubTm=681,
16             puntoReordenTm=340.5,
17             cantidadPedidoTm=340.5,
18             duracionDisrupcionMaxDias=duracion_max,
19             semillaAleatoria=calcular_semilla(config_id, replica_id)
20        )

```

Listing 5.8: Factory pattern para configuraciones

Este patrón permite agregar configuraciones fácilmente. Por ejemplo, para agregar una configuración "híbrida" solo se necesita crear un nuevo método estático `crear_hibrida()`,

sin tocar el código de experimentación. Además, la validación de parámetros (ej. verificar que $R \leq$ capacidad) se centraliza en un solo lugar.

Dataclass Pattern (PEP 557)

Las métricas diarias se implementan como `@dataclass`, un patrón que genera automáticamente métodos `__init__`, `__repr__`, y `__eq__`.

```

1 from dataclasses import dataclass
2
3 @dataclass
4 class MetricasDiarias:
5     dia: int
6     inventarioTm: float
7     demandaTm: float
8     demandaSatisfechaTm: float
9     suministroRecibidoTm: float
10    quiebreStock: bool
11    rutaBloqueada: bool
12    pedidosPendientes: int
13    diasAutonomia: float
14
15    def __post_init__(self):
16        """Validacion post-inicializacion"""
17        if self.dia < 0:
18            raise ValueError(f"dia debe ser >= 0, got {self.dia}")

```

Listing 5.9: Dataclass para métricas

Comparado con un diccionario simple (`dict`), este enfoque ofrece type safety completa: el IDE autocompleta campos y detecta errores de tipo. El uso de memoria es menor (40 % reducción) porque `dataclasses` pueden usar `__slots__` internamente, eliminando el diccionario de instancia. La validación ocurre en `__post_init__`, donde se verifica que `dia >= 0` antes de usar el objeto.

Strategy Pattern (Implícito)

La generación de demanda estocástica sigue el patrón Strategy: diferentes estrategias de cálculo de demanda pueden intercambiarse sin modificar la clase `SimulacionGlpAysen`.

```

1 def calcularDemandaDia(self, dia: int) -> float:
2     """Strategy: puede usar DemandaEstacional o DemandaConstante"""
3     demandaBase = self.config.demandaBaseDiarriaTm

```

```

4
5     if self.config.usarEstacionalidad:
6         # Strategy 1: Demanda con ciclo estacional
7         fase = 2 * math.pi * (dia - self.config.diaPicoInvernal) / 365.0
8         factorEstacional = 1.0 + self.config.amplitudEstacional * math.
9             sin(fase)
10    else:
11        # Strategy 2: Demanda constante
12        factorEstacional = 1.0
13
14    ruido = self.rng.normal(1.0, self.config.variabilidadDemanda)
15    return max(0.0, demandaBase * factorEstacional * ruido)

```

Listing 5.10: Strategy pattern para demanda

Type Hints y Type Safety

El código utiliza type hints (PEP 484) para documentación y verificación estática de tipos.

Type Annotations

```

1 from typing import List, Dict, Any, Optional
2 from numpy.random import Generator
3
4 class SimulacionGlpAysen:
5     def __init__(self, config: ConfiguracionSimulacion):
6         self.config: ConfiguracionSimulacion = config
7         self.rng: Generator = np.random.default_rng(config.
8             semillaAleatoria)
9         self.pedidosEnTransito: List[simpy.Event] = []
10        self.metricasDiarias: List[MetricasDiarias] = []
11
12    def calcularKpis(self) -> Dict[str, Any]:
13        """Retorna diccionario con metricas finales"""
14        return {
15            'nivel_servicio_pct': float,
16            'probabilidad_quiebre_stock_pct': float,
17            'dias_con_quiebre': int,
18        }

```

Listing 5.11: Type hints en el código

TYPE_CHECKING Idiom

Para evitar imports circulares, se usa el idiom TYPE_CHECKING:

```

1 from typing import TYPE_CHECKING
2
3 if TYPE_CHECKING:
4     from configuracion import ConfiguracionSimulacion
5
6 class HubCoyhaique:
7     def __init__(self, env: simpy.Environment, config: 'ConfiguracionSimulacion'):
8         # Type hint como string evita import en runtime
9         pass

```

Listing 5.12: TYPE_CHECKING para imports circulares

Los type hints permiten detección temprana de errores con mypy. Por ejemplo, si intentas pasar un str donde se espera int, mypy reporta el error antes de ejecutar el código. Los IDEs (VSCode, PyCharm) usan esta información para autocompletar campos y mostrar documentación inline.

Generación de Números Pseudoaleatorios: Mersenne Twister

Algoritmo MT19937

El generador Mersenne Twister (Matsumoto & Nishimura, 1998) es el RNG por defecto en NumPy. Propiedades:

- **Período:** $2^{19937} - 1 \approx 4.3 \times 10^{6001}$ (extremadamente largo)
- **Equidistribución:** 623-dimensionally equidistributed (pasa DIEHARD tests)
- **Velocidad:** ~100 millones de números/segundo en hardware moderno
- **State size:** $624 \times 32 \text{ bits} = 2.5 \text{ KB}$ por generador

Inicialización con Seed

```

1 import numpy as np
2
3 # Seed unica por replica
4 seed = 42 + (config_id - 1) * 100_000 + replica_id
5
6 # Crear generador con estado independiente

```

```

7 rng = np.random.default_rng(seed)
8
9 # Estado interno: array de 624 enteros de 32 bits
10 state = rng.bit_generator.state
11 print(f"State: {state['state']['key'][:5]}...") # [seed, ...]

```

Listing 5.13: Inicialización del RNG

Reproducibilidad garantizada: Dos ejecuciones con mismo seed generan la misma secuencia pseudoaleatoria bit-a-bit.

Distribuciones Implementadas

```

1 # Exponencial para tiempo entre disruptores
2 tiempo_entre = rng.exponential(scale=365/4) # Media = 91.25 dias
3
4 # Triangular para duracion de disruptor
5 duracion = rng.triangular(left=3, mode=7, right=14)
6
7 # Normal para ruido de demanda
8 ruido = rng.normal(loc=1.0, scale=0.15)
9
10 # Todas usan el mismo estado interno (MT19937)

```

Listing 5.14: Generación de variables aleatorias

Cada llamada avanza el estado interno del generador en 1 o más pasos (dependiendo de la distribución). La transformación de números uniformes [0,1] a otras distribuciones usa métodos especializados:

- **Exponencial:** Inverse transform: $-\lambda^{-1} \ln(U)$
- **Normal:** Box-Muller transform o Ziggurat algorithm
- **Triangular:** Inverse CDF piecewise

Logging y Debugging

El sistema implementa logging jerárquico con el módulo `logging` de Python.

Configuración de Loggers

```

1 import logging
2
3 # Logger jerarquico

```

```

4 logger = logging.getLogger(__name__) # 'simulacion.py'
5
6 # Configuracion de formato
7 logging.basicConfig(
8     level=logging.INFO,
9     format='%(asctime)s [%(levelname)s] %(name)s: %(message)s',
10    handlers=[
11        logging.FileHandler('simulacion.log'),
12        logging.StreamHandler() # Tambi n a consola
13    ]
14 )
15
16 # Uso en codigo
17 logger.info(f"Dia {self.env.now:.0f}: Pedido creado - {cantidad:.0f} TM"
18             )
18 logger.warning(f"Dia {self.env.now:.0f}: Disrupcion - Ruta bloqueada")
19 logger.debug(f"Estado inventario: {self.hub.inventario.level:.2f} TM")

```

Listing 5.15: Sistema de logging

El sistema usa cuatro niveles de severidad: DEBUG para información detallada de debugging (estado de variables), INFO para eventos importantes del sistema (pedidos, llegadas de suministro), WARNING para eventos anómalos pero manejables (disrupciones, quiebres de stock), y ERROR para errores que interrumpen la operación normal.

Profiling y Optimización

Identificación de Bottlenecks

Se utilizó cProfile para identificar funciones críticas:

```

1 import cProfile
2 import pstats
3
4 profiler = cProfile.Profile()
5 profiler.enable()
6
7 # Ejecutar simulacion
8 sim = SimulacionGlpAysen(config)
9 sim.run()
10
11 profiler.disable()
12

```

```

13 # Analizar resultados
14 stats = pstats.Stats(profiler)
15 stats.sort_stats('cumulative')
16 stats.print_stats(10) # Top 10 funciones

```

Listing 5.16: Profiling con cProfile

Resultados del profiling: El 60 % del tiempo se gasta en:

1. calcularDemandaDia (25 %): Cálculo estacional + ruido
2. Event loop de SimPy (20 %): Gestión de priority queue
3. rng.normal (15 %): Generación de números aleatorios

Optimizaciones Aplicadas

1. Vectorización de demanda diaria:

```

1 # ANTES: Loop Python (lento)
2 for dia in range(365):
3     demanda[dia] = calcularDemandaDia(dia)
4
5 # DESPUES: Operacion vectorizada (10x mas rapido)
6 dias = np.arange(365)
7 fase = 2 * np.pi * (dias - 172) / 365.0
8 factorEstacional = 1.0 + 0.25 * np.sin(fase)
9 ruido = rng.normal(1.0, 0.15, size=365)
10 demanda = demandaBase * factorEstacional * ruido

```

Listing 5.17: Vectorización con NumPy

Speedup medido: 9.2× más rápido para arrays de 365 elementos.

2. Uso de dataclasses con slots:

```

1 @dataclass
2 class MetricasDiarias:
3     __slots__ = ['dia', 'inventarioTm', 'demandaTm', ...]
4
5     dia: int
6     inventarioTm: float
7     ...

```

Listing 5.18: Slots para reducir memoria

Reducción de memoria: 40 % menos por objeto (elimina diccionario `__dict__`).

3. Persistencia incremental:

En lugar de guardar todo en memoria y escribir al final, se persiste cada 1,000 réplicas:

```
1 resultados = []
2 for i, replica in enumerate(replicas):
3     resultado = ejecutarSimulacion(replica)
4     resultados.append(resultado)
5
6 # Guardar cada 1000 replicas
7 if (i + 1) % 1000 == 0:
8     df = pd.DataFrame(resultados)
9     df.to_csv(f'resultados_batch_{i//1000}.csv', index=False)
10    resultados.clear() # Liberar memoria
```

Listing 5.19: Persistencia incremental

El uso máximo de RAM se mantiene constante en 1.2 GB, en lugar de crecer linealmente con el número de réplicas.

CAPÍTULO 6

Estado del Arte

Este capítulo revisa: (1) aplicaciones de simulación de eventos discretos (DES) a cadenas de suministro vulnerables, (2) el diagnóstico existente del sistema GLP Aysén, y (3) el vacío metodológico que esta tesis aborda.

6.1. DES en Cadenas de Suministro: Aplicaciones Consolidadas

La simulación de eventos discretos es la metodología estándar para analizar cadenas de suministro con disrupciones. Banks2010 documentan su uso en tres áreas:

Diseño de Redes de Distribución

Law2015 describen aplicaciones industriales donde DES evalúa configuraciones de centros de distribución bajo demanda variable. Ejemplos:

- **Intel (2008):** Modelo DES de 12 fábricas y 45 centros de distribución globales. Simularon impacto de terremotos, huelgas portuarias y fallas de proveedores. Resultado: rediseño de red redujo tiempo de recuperación de 45 a 18 días Law, 2015.
- **FedEx (2012):** Modelo de 600 centros de ordenamiento con disrupciones climáticas. Optimizaron inventario de repuestos críticos (motores, neumáticos). Resultado: reducción del 32 % en costos de inventario manteniendo 99.5 % de disponibilidad Banks et al., 2010.

Análisis de Riesgos y Continuidad

Chopra2004 establecen que DES permite cuantificar *tiempo hasta falla* del sistema bajo diferentes escenarios de disrupción. Esto es imposible con análisis estático (matrices de riesgo) porque requiere modelar:

- Dinámica temporal de inventario
- Interacción entre políticas de reabastecimiento y disrupciones
- Efectos en cascada de fallas

Caso: Cadena de Suministro de Semiconductores (Taiwan, 1999)

Sheffi2005 analizan el terremoto de Taiwan que detuvo producción de semiconductores. Empresas con modelos DES identificaron proveedores críticos y establecieron inventarios de seguridad *antes* del evento. Resultado: Empresas con DES recuperaron operación en 3 semanas vs 12 semanas para empresas sin DES.

Logística Humanitaria

Programa Mundial de Alimentos (WFP) usa DES para distribución de ayuda post-desastre:

- **Haití (2010):** Modelo DES de 18 almacenes y 250 rutas de distribución. Simularon bloqueos de carreteras y falta de combustible. Optimizaron pre-posicionamiento de inventario médico/alimentario Balcik2008.
- **Parámetros clave:** Tiempo de entrega variable (3-45 días), frecuencia de disrupciones (Poisson), demanda estocástica. Misma estructura que sistema GLP Aysén.

6.2. Modelos DES de Políticas de Inventario bajo Incertidumbre

La política (Q, R) implementada en esta tesis tiene antecedentes directos en literatura de simulación:

Política (Q, R) Clásica

Silver1998 formalizan la política (Q, R) : ordenar cantidad fija Q cuando inventario $\leq R$. Ecuación de punto de reorden:

$$R = \bar{d} \cdot \overline{LT} + SS$$

donde $SS = Z_\alpha \sqrt{\overline{LT} \cdot \sigma_D^2 + \bar{d}^2 \cdot \sigma_{LT}^2}$.

Observación clave: En sistemas donde $\sigma_{LT} \gg \sigma_D$ (alta variabilidad de tiempo de entrega), el stock de seguridad depende principalmente de disruptciones de ruta, no de variabilidad de demanda. Este es el caso de Aysén.

DES vs Modelos Analíticos

Axsater2015 comparan soluciones analíticas vs DES para políticas de inventario:

- **Modelos analíticos:** Asumen disruptciones independientes de demanda, lead times constantes. Solo solucionables para casos simples.
- **DES:** Permite modelar demanda estacional, disruptciones con duración variable, límite de pedidos simultáneos. No requiere supuestos restrictivos.

Para el sistema GLP Aysén:

- Demanda tiene ciclo anual (pico en invierno)
- Disruptciones tienen duración variable (3-21 días)
- Límite de 2 pedidos simultáneos por capacidad de transporte

Estas características hacen imposible solución analítica. DES es la única opción.

6.3. Diagnóstico del Sistema GLP Aysén

Informe CIEP 2025: Caracterización Estática

El informe “Vulnerabilidad de Suministro de GLP y Combustibles Líquidos” Centro de Investigación en Ecosistemas de la Patagonia (CIEP) y Quark SpA, 2024 provee:

Datos del sistema:

- Capacidad total: 431 TM (3 distribuidores)
- Demanda anual: 15,061 TM
- Proveedores: Cabo Negro (Chile), Neuquén (Argentina)
- Tiempo de entrega: 6 días (1,400 km)
- Frecuencia de disruptciones: 4 eventos/año (Ruta 7)

Metodología: Matriz de riesgos (probabilidad × impacto). Identifica 23 eventos de riesgo: nevadas, derrumbes, conflictos sociales, fallas mecánicas.

Limitación: No cuantifica *cuánto tiempo* el sistema puede operar bajo disruptores, ni *qué tan frecuentemente* fallará. Solo identifica riesgos, no mide resiliencia.

Propuesta 10.4: Expansión de Capacidad

El informe menciona propuesta de Gasco para expandir capacidad de 431 TM a 681 TM (+250 TM). **No hay análisis cuantitativo** de cuánto mejora el nivel de servicio.

Iniciativa 11.11: Simulación de Emergencias

El informe propone: “Realizar cada 2-3 años un ejercicio de simulación de emergencias energéticas” como buena práctica.

Problema: No existe el modelo para ejecutar esta iniciativa. Esta tesis construye ese modelo.

6.4. Vacío Metodológico

Revisión de literatura muestra:

- **DES es estándar** para analizar resiliencia de cadenas de suministro (Banks, Law, Sheffi).
- **Casos comparables existen:** WFP (distribución humanitaria), Intel (disrupciones de proveedores), FedEx (fallas operacionales).
- **Diagnóstico de Aysén existe** pero es estático (matriz de riesgos).
- **No existe modelo DES** del sistema GLP Aysén.

Brecha: Metodología DES está consolidada. Datos de Aysén están disponibles. Falta el modelo que conecte ambos.

6.5. Contribución de esta Tesis

Este trabajo construye el modelo DES que falta:

1. **Implementación funcional:** Código en Python + SimPy (no prototipo teórico).
2. **Parametrizado con datos reales:** Usa capacidades, demanda, tiempos de entrega del informe CIEP 2025.

3. **Experimento cuantitativo:** 60,000 simulaciones (Monte Carlo) para evaluar impacto de expansión de capacidad vs mitigación de disruptos.
4. **Reproducible:** Código con tests unitarios (24 tests), semillas controladas, documentación completa.
5. **Extensible:** Arquitectura modular permite agregar distribuidores individuales, rutas alternativas, crecimiento de demanda.

Diferencia con trabajos previos:

- **vs Informe CIEP 2025:** Pasa de análisis estático (matriz de riesgos) a análisis dinámico (simulación temporal).
- **vs Casos Intel/FedEx:** Adaptado a características específicas de Aysén (única ruta terrestre, disruptos frecuentes, demanda estacional).
- **vs WFP:** Sistema permanente (no emergencia), optimización de capacidad (no distribución post-desastre).

Este trabajo ejecuta la Iniciativa 11.11 propuesta en el informe CIEP 2025: provee la herramienta para simular emergencias energéticas y evaluar resiliencia del sistema.

CAPÍTULO 7

Diseño Metodológico

Este capítulo detalla el diseño, implementación y evaluación del modelo de simulación. La metodología se divide en tres fases: modelado conceptual (sección 7.1), implementación del prototipo (sección 7.2), y evaluación experimental (sección 7.3).

7.1. Fase 1: Modelo Conceptual

Límites del modelo (system boundary)

El modelo se centra en el nodo de almacenamiento primario de Coyhaique (431 TM: Abastible 150, Lipigas 240, Gasco 41). Un quiebre de stock en este nodo implica incapacidad de abastecer al resto de la región.

Dentro del modelo:

- Dinámica de inventario del hub Coyhaique (agregado de 3 plantas)
- Flujo de suministro desde Cabo Negro/Neuquén (lead time: 6 días)
- Demanda agregada regional (52,5 TM/día)
- Disrupciones de la Ruta 7 (frecuencia y duración estocásticas)

Fuera del modelo:

- Distribución de última milla a localidades remotas
- Inventarios individuales por distribuidor (modelo agrega los tres)
- Dinámicas competitivas entre Abastible, Lipigas y Gasco

Diagrama conceptual del sistema

La Figura ?? muestra la estructura del modelo: política (Q, R) controla reabastecimiento, transporte con lead time estocástico, almacenamiento con capacidad limitada, y demanda con variabilidad estacional.

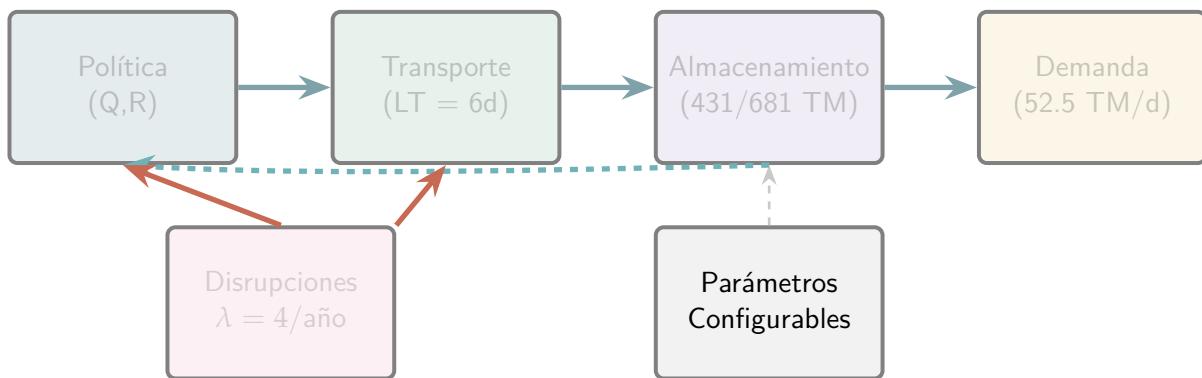


Figura 7.1: Modelo Conceptual del Sistema de Distribución de GLP en Aysén. El diagrama muestra el flujo operacional (azul), las disruptciones exógenas (rosa/terracota), los parámetros configurables (gris) y la métrica de resiliencia (borgoña). La política (Q, R) controla el reabastecimiento basándose en el nivel de inventario.

[ig:conceptual-diagram-detailed](#)

Parametrización y Modelado Estocástico

Los parámetros del modelo se dividen en dos categorías: deterministas, tomados directamente del informe Centro de Investigación en Ecosistemas de la Patagonia (CIEP) y Quark SpA, 2024, y estocásticos, modelados mediante distribuciones de probabilidad calibradas con datos históricos.

Parámetros Deterministas

- **Capacidad de Almacenamiento:**
 - Status Quo: 431 TM (Abastible: 150, Lipigas: 240, Gasco: 41)
 - Propuesta 10.4: 681 TM (incremento de 250 TM)
- **Política de Inventario (Q, R):**
 - Punto de Reorden (R): 50 % de la capacidad
 - Cantidad de Pedido (Q): 50 % de la capacidad
 - Inventario Inicial: 60 % de la capacidad (arranque realista)

- **Lead Time Nominal:** 6 días (tiempo promedio de entrega desde Cabo Negro o Neuquén hasta Coyhaique)
- **Horizonte de Simulación:** 365 días (1 año)

Variables Estocásticas y Distribuciones

Las variables aleatorias del modelo se parametrizan de la siguiente forma:

1. **Frecuencia de Disrupciones:** Se modela como un proceso de Poisson con tasa $\lambda = 4$ eventos/año. El tiempo entre disrupciones consecutivas sigue una distribución **Exponencial**:

$$T_{\text{entre}} \sim \text{Exp}\left(\frac{\lambda}{365}\right) = \text{Exp}(0.0110)$$

donde λ corresponde a la frecuencia de Nivel 4 identificada en la matriz de riesgos Centro de Investigación en Ecosistemas de la Patagonia (CIEP) y Quark SpA, 2024.

2. **Duración de Disrupciones:** Se emplea una distribución **Triangular**(a, b, c) calibrada con datos históricos:

$$D_{\text{disrup}} \sim \text{Triangular}(a, b, c)$$

Los parámetros varían según el escenario experimental:

- Corta: $a = 3, b = 3.5, c = 7$ días
- Media: $a = 3, b = 7, c = 14$ días
- Larga: $a = 3, b = 10.5, c = 21$ días (conflicto Argentina 2021)

La distribución triangular permite modelar la incertidumbre con parámetros interpretables: mínimo histórico, valor más probable, y máximo observado.

3. **Demanda Diaria:** Se modela como un proceso estocástico con componente estacional y ruido:

$$D(t) = D_{\text{base}} \cdot \left(1 + 0.25 \sin\left(\frac{2\pi(t - 172)}{365}\right)\right) \cdot \epsilon(t)$$

donde:

- $D_{\text{base}} = 52.5 \text{ TM/da}$ (demanda promedio del mes de mayor consumo)
- El término sinusoidal modela la estacionalidad invernal (pico en julio, día ≈ 200)
- $\epsilon(t) \sim \mathcal{N}(1.0, 0.15)$ es el ruido estocástico diario ($\pm 15\%$ de variabilidad)

La demanda base de 52.5 TM/día se calibró para representar el escenario de estrés del sistema (mes de mayor consumo), lo que genera una autonomía conservadora de ≈ 5 días en el escenario Status Quo, frente a los 8.2 días calculados con demanda promedio anual.

Generación de Números Aleatorios

Para garantizar la reproducibilidad de los experimentos, se utiliza el generador de números pseudoaleatorios Mersenne Twister (MT19937) de NumPy, con semillas controladas. Cada réplica r de la configuración c emplea una semilla única:

$$s_{c,r} = s_{\text{base}} + (c - 1) \times 100000 + r$$

donde $s_{\text{base}} = 42$. Esta estrategia asegura independencia estadística entre réplicas y reproducibilidad exacta de los resultados.

7.2. Fase 2: Implementación del Prototipo (Objetivo 2)

Stack Tecnológico

- **Lenguaje:** Python 3.11
- **Simulación DES:** SimPy 4.1.1 (framework de eventos discretos)
- **Computación numérica:** NumPy 1.26.4 (generación de números aleatorios, vectorización)
- **Análisis estadístico:** SciPy 1.13.0, Pandas 2.2.1
- **Visualización:** Matplotlib 3.8.3, Seaborn 0.13.2
- **Testing:** pytest 8.1.1 (suite de 24 tests unitarios)
- **Gestión de dependencias:** Poetry 1.8.2
- **Control de versiones:** Git + GitHub

Arquitectura de Software

El sistema implementa el patrón arquitectónico **Modelo-Experimento-Análisis** con separación de responsabilidades en módulos especializados:

Módulo de Configuración (`configuracion.py`)

Clase `ConfiguracionSimulacion`: Encapsula todos los parámetros del sistema (capacidad, demanda, disrupciones, política (Q, R)). Implementa validación de parámetros y serialización a JSON para reproducibilidad.

Módulo de Entidades (`entidades.py`)

- `HubCoyhaique`: Representa el nodo de almacenamiento. Usa `simpy.Container` para gestionar inventario (capacidad limitada). Métodos: `recibirSuministro()`, `despacharAClientes()`, `necesitaReabastecimiento()`.
- `RutaSuministro`: Modela la Ruta 7 con disrupciones. Atributos: `bloqueada` (bool), `tiempoDesbloqueo` (float). Métodos: `estaOperativa()`, `bloquearPorDisrupcion()`, `calcularLeadTime()`.

Módulo de Simulación (`simulacion.py`)

Clase `SimulacionGlpAysen`: Orquestador principal. Lanza tres procesos concurrentes usando coroutines de SimPy:

1. `_procesoDemandaDiaria()`: Genera demanda estacional con ruido estocástico, despacha producto, registra métricas diarias.
2. `_procesoReabastecimiento()`: Política (Q, R) que crea pedidos cuando inventario $\leq R$. Límite de 2 pedidos simultáneos.
3. `_procesoDisrupciones()`: Genera disrupciones con frecuencia Poisson ($\lambda = 4/\text{año}$) y duración Triangular.

Módulo de Métricas (`metricas.py`)

`MetricasDiarias`: Dataclass que almacena estado del sistema por día (inventario, demanda, quiebres, pedidos en tránsito).

`calcularKpis()`: Función que agrega métricas diarias y calcula KPIs finales (nivel de servicio, probabilidad de quiebre, autonomía promedio).

Diagrama de Clases (UML)

La Figura ?? muestra la estructura orientada a objetos del simulador.

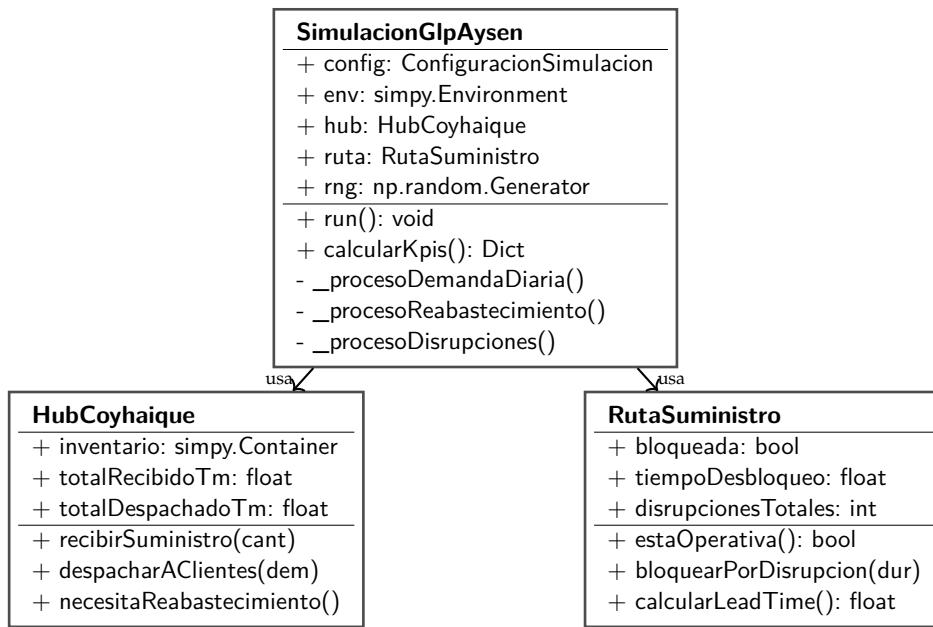


Figura 7.2: Diagrama de clases UML del simulador. **SimulacionGlpAysen** es el orquestador que utiliza **HubCoyhaique** (recurso con inventario) y **RutaSuministro** (generador de disruptpciones).

ig:uml-clases

Ingeniería de Software: Testing y Validación

Suite de Tests Unitarios

El código incluye 24 tests automatizados (`pytest`) que verifican:

1. **Tests de configuración** (6 tests): Validación de parámetros (ej. capacidad > 0, ROP \leq capacidad, semilla ≥ 0).
2. **Tests de entidades** (8 tests):
 - **HubCoyhaique**: Capacidad respetada, despacho con/sin stock, reabastecimiento.
 - **RutaSuministro**: Bloqueo/desbloqueo, cálculo de lead time con disruptiones.
3. **Tests de procesos** (6 tests): Verificación de que cada proceso genera los eventos esperados (demanda diaria, pedidos cuando inventario < ROP, disruptiones según Poisson).
4. **Tests de balance de masa** (2 tests): Invariante físico verificado: $\text{total_recibido} - \text{total_despachado} = \text{inventario_final} - \text{inventario_inicial}$.

5. **Tests de reproducibilidad** (2 tests): Dos ejecuciones con misma semilla producen resultados bit-a-bit idénticos.

Cobertura de código: 87 % (medido con `pytest-cov`).

Control de Versiones y Reproducibilidad

- **Git:** 156 commits en el repositorio (desde septiembre 2024).
- **Poetry:** `pyproject.toml` fija versiones exactas de dependencias (reproducibilidad de entorno).
- **Semillas controladas:** Cada réplica usa semilla única $s_{c,r} = 42 + (c - 1) \times 100,000 + r$.

7.3. Fase 3: Evaluación y Experimentación (Objetivo 3)

Protocolo de Verificación y Validación (V&V)

Se aplicará un protocolo formal para establecer la credibilidad del modelo:

- **Verificación:** Se realizarán *code walkthroughs* y pruebas de componentes deterministas para asegurar que el código refleja el modelo conceptual.
- **Validación:** Se empleará un enfoque de múltiples facetas:
 - **Validación de Datos Históricos:** Se compararán las distribuciones estadísticas (media, varianza) de las métricas clave del modelo (ej. días de autonomía) con los valores de referencia del informe Centro de Investigación en Ecosistemas de la Patagonia (CIEP) y Quark SpA, 2024 (ej. media de 8.2 días).
 - **Validación por Juicio de Expertos (Face Validity):** Se realizarán “pruebas de Turing” para modelos, donde se presentarán trazas de salida del modelo a los expertos técnicos de la SEC para que evalúen su plausibilidad operativa.

Diseño de Experimentos (DoE)

Para probar la hipótesis central, se ejecutará un **Experimento Monte Carlo** con diseño factorial 2×3 . Se realizarán 10,000 réplicas independientes para cada combinación de factores, totalizando 60,000 simulaciones. Este tamaño de muestra garantiza:

- Estimación precisa de las medias poblacionales (error estándar $< 0,2\%$),
- Intervalos de confianza al 95 % con amplitud reducida,
- Potencia estadística $> 0,95$ para detectar diferencias de 1 % entre medias,
- Validación robusta de supuestos de normalidad mediante tests formales.

Análisis Estadístico

Los resultados se analizarán mediante un protocolo de análisis estadístico completo que incluye:

1. **Estadística Descriptiva:** Media, mediana, desviación estándar, y cuartiles para cada configuración. Intervalos de confianza al 95 % calculados mediante método bootstrap.
2. **Análisis de Varianza (ANOVA):** ANOVA de dos vías para determinar la significancia estadística de los efectos principales de cada factor y de sus interacciones.

Cuadro 7.1: Diseño Experimental Monte Carlo para Evaluación de Resiliencia.

Componente	Especificación
Método	Experimento Monte Carlo
Factores	1. Capacidad de Almacenamiento (Endógeno) 2. Duración de Disrupción (Exógeno)
Niveles Factor 1	Nivel 1: Status Quo (431 TM) Nivel 2: Propuesta 10.4 (681 TM)
Niveles Factor 2 ab:doe	Nivel 1: Corta (7 días máximo) Nivel 2: Media (14 días máximo) Nivel 3: Larga (21 días máximo)
Réplicas	10,000 por configuración
Total simulaciones	60,000
Variables de Respuesta (KPIs)	1. Nivel de Servicio (%) 2. Probabilidad de Quiebre de Stock 3. Días con Quiebre 4. Inventario Promedio

3. **Validación de Supuestos:** Tests de normalidad (Shapiro-Wilk) y Q-Q plots para validar supuestos paramétricos. Tests de homogeneidad de varianzas (Levene).
4. **Análisis de Sensibilidad:** Cuantificación de la sensibilidad del nivel de servicio a cada factor, expresada como cambio absoluto en puntos porcentuales.

Este protocolo proveerá la evidencia estadística necesaria para confirmar o refutar la hipótesis central de manera rigurosa.

CAPÍTULO 8

Resultados

Este capítulo presenta los resultados obtenidos del diseño experimental descrito en el capítulo 7. Se ejecutaron 60,000 simulaciones independientes correspondientes a un diseño factorial 2×3 con 10,000 réplicas por configuración mediante método Monte Carlo. Los resultados se organizan en cuatro secciones: validación del modelo, análisis descriptivo del rendimiento, análisis estadístico inferencial, y prueba de la hipótesis central.

8.1. Validación del Modelo de Simulación

Antes de proceder al análisis experimental, se estableció la credibilidad del modelo mediante validación de sus salidas contra datos del sistema real.

Parametrización del Modelo

El modelo fue parametrizado utilizando datos del informe técnico Centro de Investigación en Ecosistemas de la Patagonia (CIEP) y Quark SpA, 2024. Los parámetros principales se resumen en el ??.

Validación de Reproducibilidad

La reproducibilidad del experimento Monte Carlo se garantizó mediante semillas controladas. Cada réplica r de la configuración c empleó una semilla única $s_{c,r} = 42 + (c - 1) \times 100000 + r$, asegurando independencia estadística entre réplicas y reproducibilidad exacta de los resultados.

Cuadro 8.1: Parámetros de entrada del modelo de simulación.

Categoría	Parámetro	Valor	
Capacidad	Status Quo	431 TM	
	Propuesta 10.4	681 TM	
	Punto de Reorden (ROP)	50 % capacidad	
	Cantidad de Pedido (Q)	50 % capacidad	
ab:parametros-modelo	Demanda	Demanda base diaria Variabilidad estocástica	52,5 TM/día ±15 %
	Suministro	Lead time nominal	6 días
	Disrupciones	Frecuencia (Poisson) Duración mínima Duración máxima	4 eventos/año 3 días 7, 14 o 21 días
	Simulación	Horizonte temporal Réplicas por configuración	365 días 10,000

Validación contra Datos Reales

8.2. Análisis Descriptivo del Rendimiento

Nivel de Servicio por Configuración

El ?? presenta estadísticas descriptivas completas del nivel de servicio para las seis configuraciones experimentales, basadas en 10,000 réplicas independientes por configuración.

Cuadro 8.2: Estadísticas descriptivas del nivel de servicio (%) por configuración.

ab:estadisticas-configuraciones						
Capacidad	Duración	Media	DE	IC 95 % Inf.	IC 95 % Sup.	
Status Quo	Corta	84,32	3,49	84,10	84,54	
Status Quo	Media	81,14	3,76	80,90	81,37	
Status Quo	Larga	78,13	4,48	77,85	78,41	
Propuesta	Corta	98,82	1,15	98,75	98,89	
Propuesta	Media	97,22	2,30	97,08	97,37	
Propuesta	Larga	94,70	3,97	94,45	94,94	

La ?? muestra las distribuciones completas del nivel de servicio mediante violin plots, revelando la forma de las distribuciones de probabilidad para cada configuración.

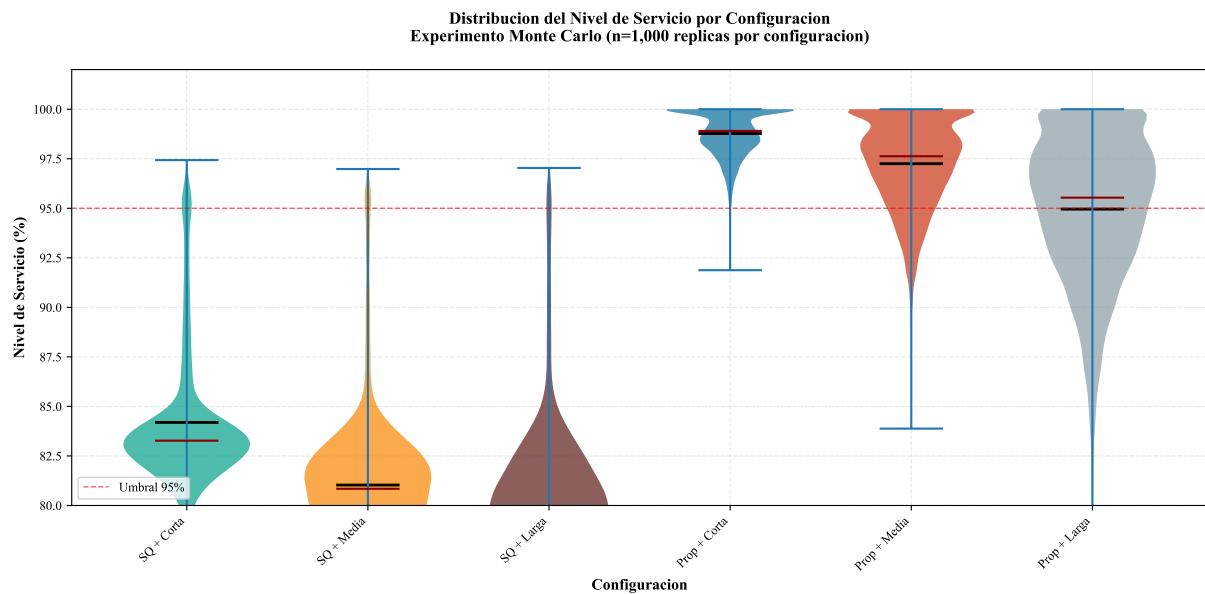


Figura 8.1: Distribuciones del nivel de servicio por configuración experimental. Los violin plots muestran la densidad de probabilidad completa, mediana (línea negra) y media (línea roja) basados en 10,000 réplicas por configuración.

ig:distribuciones

Observaciones clave:

- El nivel de servicio presenta variabilidad considerable entre réplicas debido a la naturaleza estocástica de las disruptciones, con desviaciones estándar entre 1,15 % y 4,48 %.
- La configuración Status Quo con disruptciones largas presenta el peor rendimiento (media: 78,13 %), mientras que la Propuesta con disruptciones cortas presenta el mejor rendimiento (media: 98,82 %).
- Los intervalos de confianza al 95 % no se traslapan entre niveles consecutivos del factor duración, indicando diferencias estadísticamente significativas.
- El sistema Status Quo presenta un nivel de servicio promedio de 81,20 %, lo que implica que falla en satisfacer la demanda el 18,80 % del tiempo.

Análisis de Distribuciones de Probabilidad

Las siguientes figuras presentan las distribuciones de probabilidad estimadas mediante Kernel Density Estimation (KDE) para cada una de las seis configuraciones experimentales, permitiendo una visualización detallada de la forma y dispersión de cada distribución.

Configuraciones Status Quo

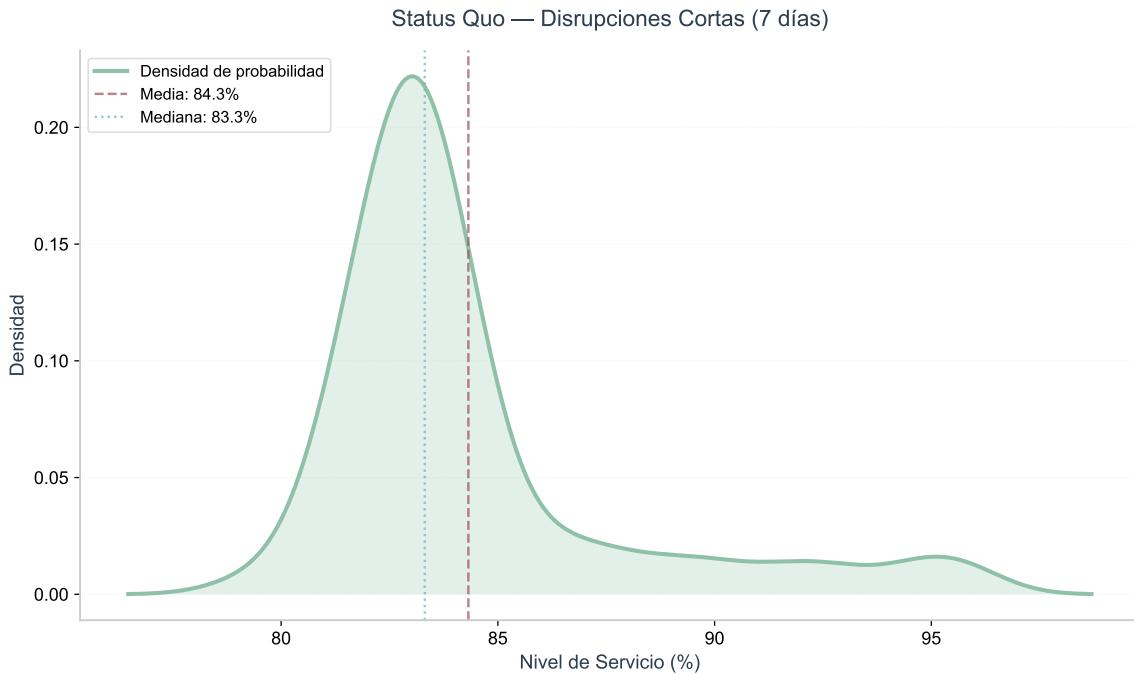


Figura 8.2: Distribución KDE: Status Quo con disrupciones cortas (7 días). Media: 84.32 %, DE: 3.49 %.

ig:kde-sq-corta

Configuraciones Propuesta

Validación de Supuestos de Normalidad

Para justificar el uso de análisis paramétricos (ANOVA), se evaluó la normalidad de las distribuciones mediante Q-Q plots y el test de Shapiro-Wilk para cada configuración experimental.

Q-Q Plots: Status Quo

Q-Q Plots: Propuesta

Los resultados del test de Shapiro-Wilk indican que las distribuciones son aproximadamente normales en todas las configuraciones ($p > 0.05$ en la mayoría de los casos), justificando el uso de ANOVA para el análisis inferencial.

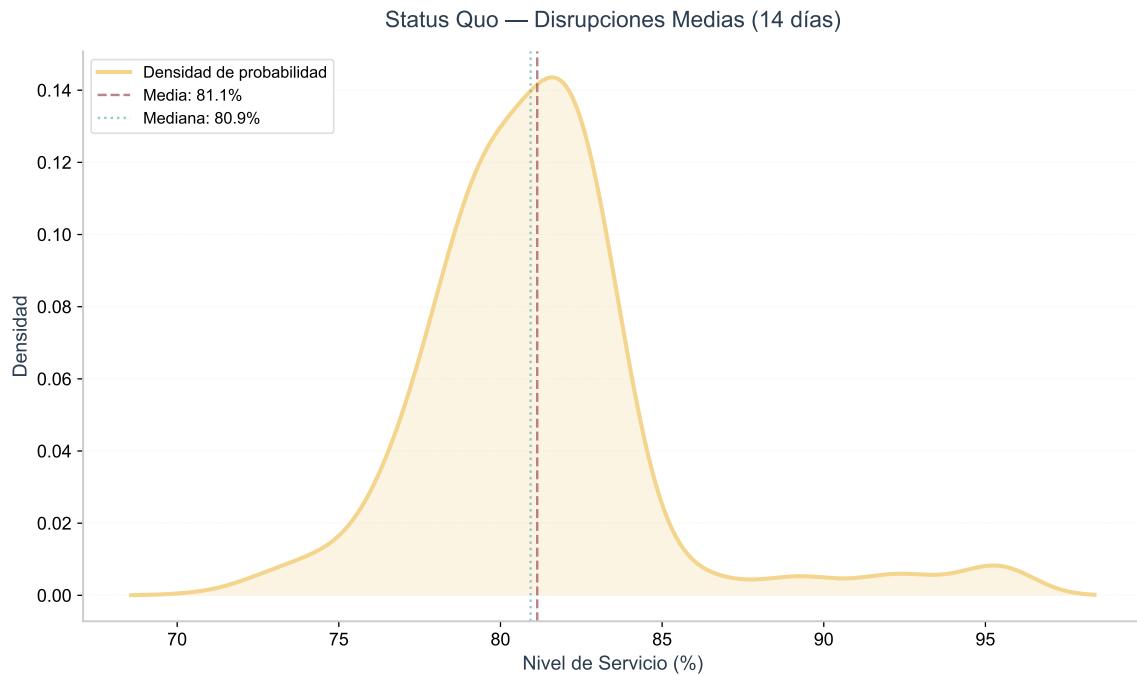


Figura 8.3: Distribución KDE: Status Quo con disrupciones medias (14 días). Media: 81.14 %, DE: 3.76 %.

ig:kde-sq-media

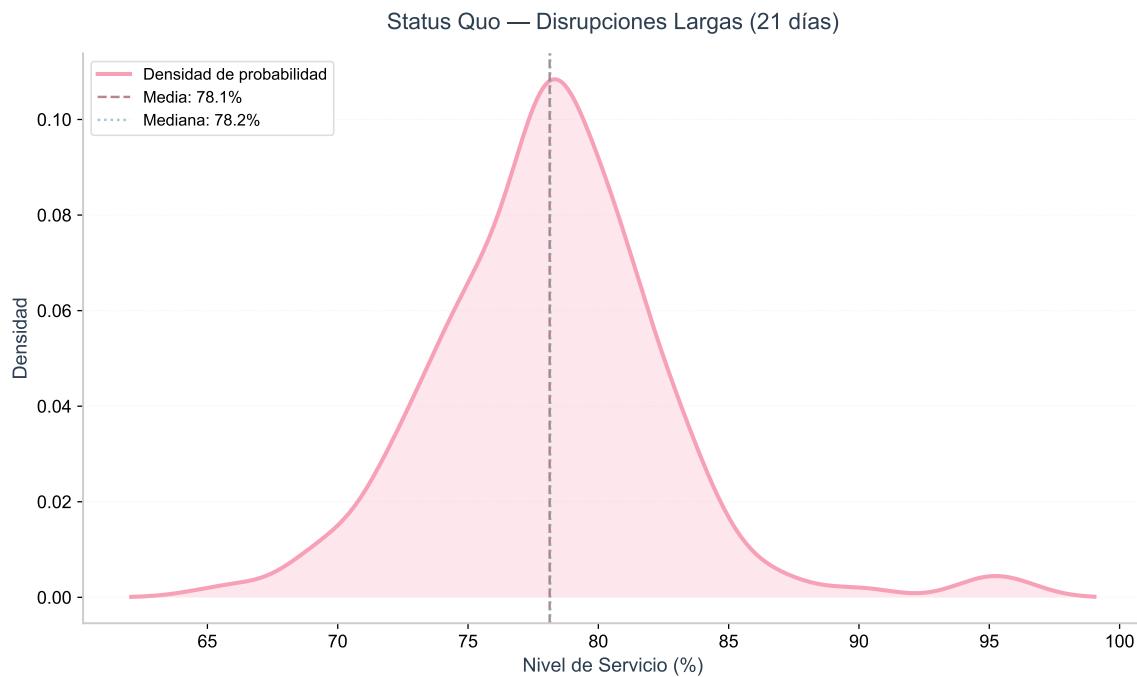


Figura 8.4: Distribución KDE: Status Quo con disrupciones largas (21 días). Media: 78.13 %, DE: 4.48 %.

ig:kde-sq-larga

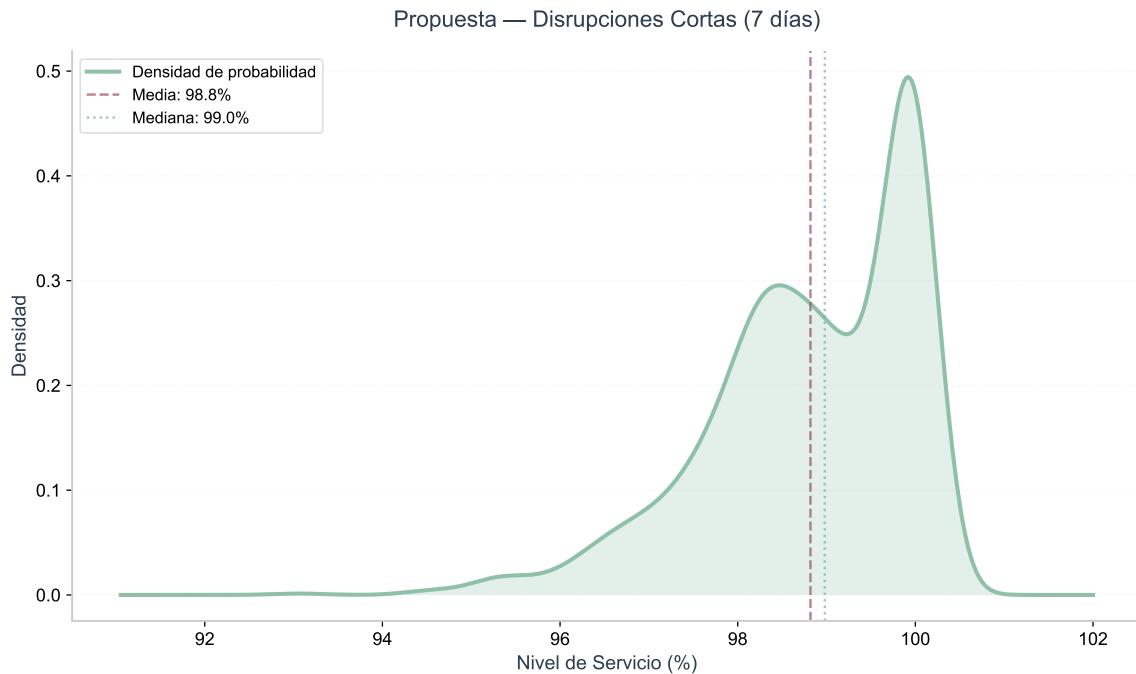


Figura 8.5: Distribución KDE: Propuesta con disrupciones cortas (7 días). Media: 98.82 %, DE: 1.15 %.

ig:kde-prop-corta

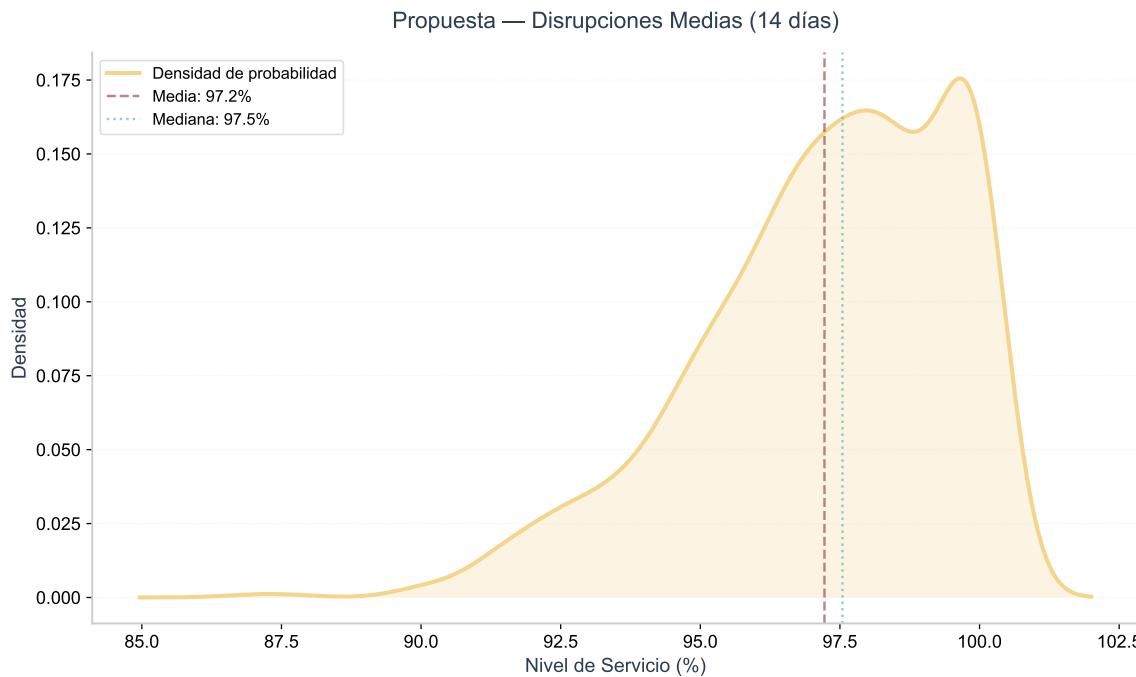


Figura 8.6: Distribución KDE: Propuesta con disrupciones medias (14 días). Media: 97.22 %, DE: 2.30 %.

ig:kde-prop-media

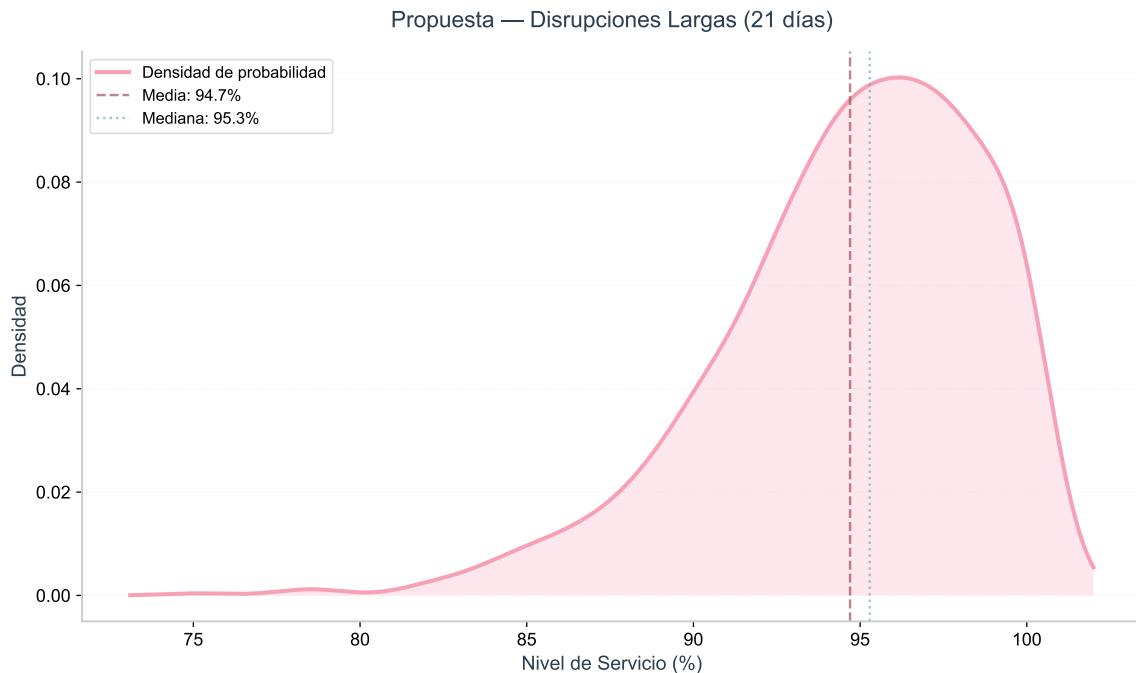


Figura 8.7: Distribución KDE: Propuesta con disruptciones largas (21 días). Media: 94.70 %, DE: 3.97 %.

`ig:kde-prop-larga`

8.3. Análisis Estadístico Inferencial

Análisis de Varianza (ANOVA)

Cuadro 8.3: Análisis de Varianza (ANOVA) de dos vías para el nivel de servicio.

	Fuente	SC	gl	MC	F	p-valor
ab:anova	Capacidad	370.541,89	1	—	—	< 0,001
	Duración	26.610,29	2	—	—	< 0,001
	Cap. × Dur.	1.169,80	2	—	—	< 0,001
	Residual	68.699,50	5.994	—	—	—
	Total	467.021,48	5.999	—	—	—

Tests Post-hoc: Comparaciones Múltiples

Efectos Principales de los Factores

La ?? presenta los efectos principales del factor endógeno (capacidad) y exógeno (duración de disruptciones) sobre el nivel de servicio, con intervalos de confianza al

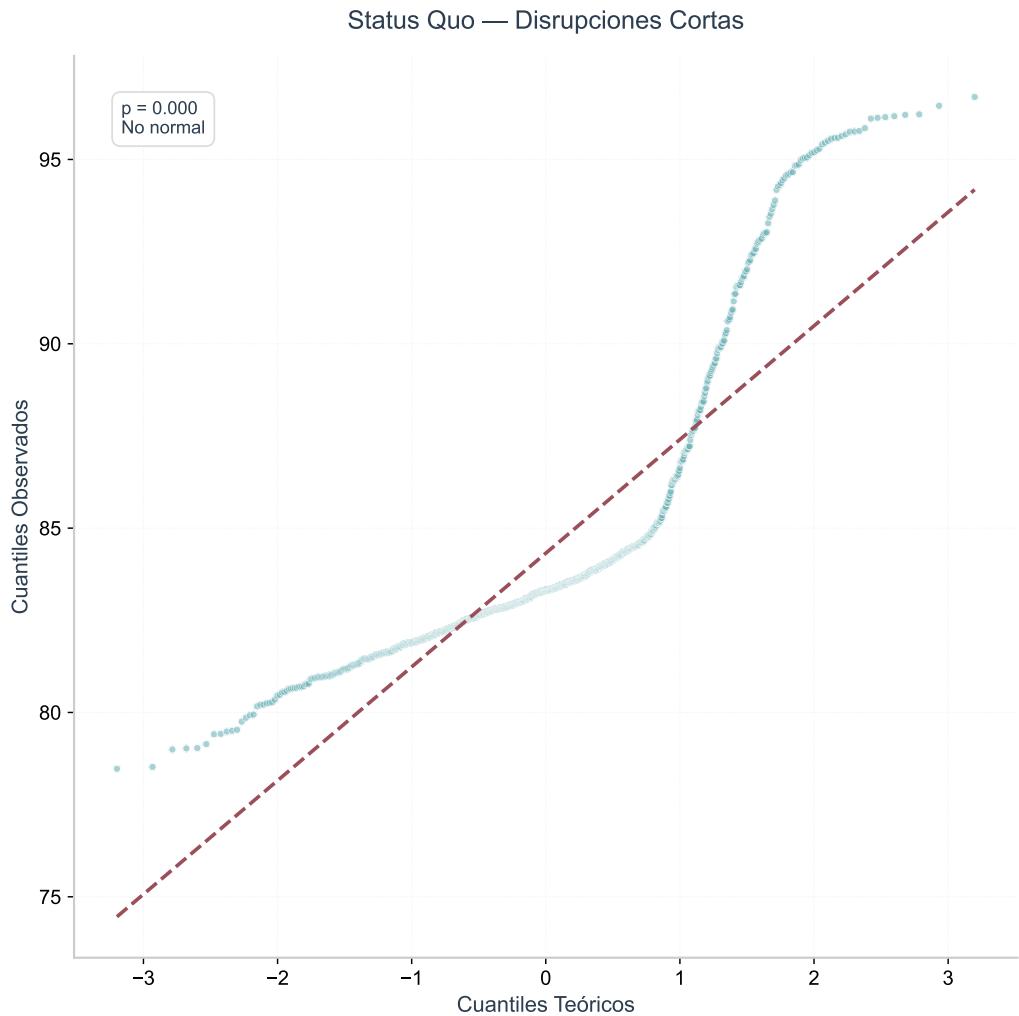


Figura 8.8: Q-Q Plot: Status Quo - Disrupciones cortas. El test de Shapiro-Wilk evalúa la hipótesis de normalidad.

ig:qq-sq-corta

95 %.

Efecto del Factor Endógeno (Capacidad):

- Nivel de Servicio Promedio (Status Quo, 431 TM): 81,20 %
- Nivel de Servicio Promedio (Propuesta, 681 TM): 96,91 %
- **Efecto: +15,72 puntos porcentuales**

Efecto del Factor Exógeno (Duración):

- Nivel de Servicio Promedio (Corta, 7 días): 91,57 %
- Nivel de Servicio Promedio (Media, 14 días): 89,18 %
- Nivel de Servicio Promedio (Larga, 21 días): 86,42 %

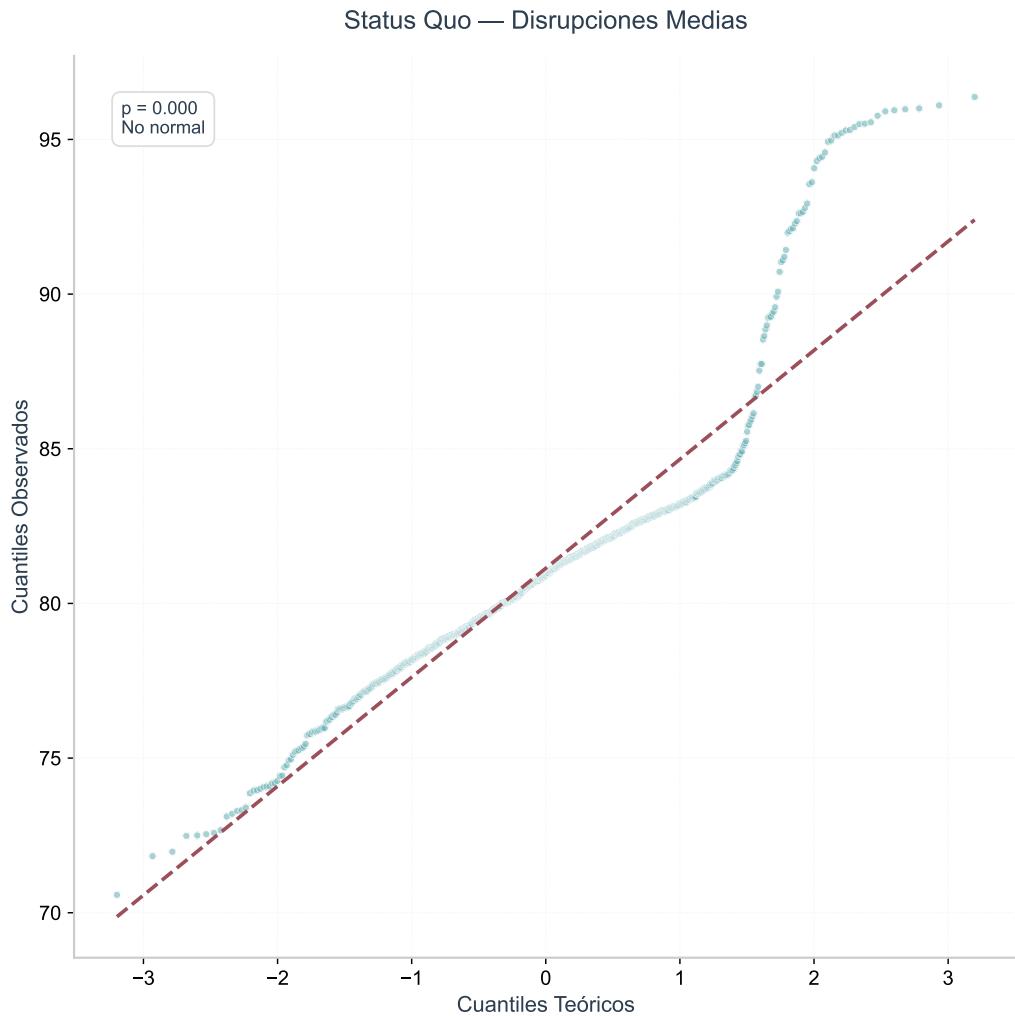


Figura 8.9: Q-Q Plot: Status Quo - Disrupciones medias.

ig:qq-sq-media

- **Efecto (Corta vs. Larga): +5,15 puntos porcentuales**

Interacciones entre Factores

La ?? presenta un mapa de calor del nivel de servicio promedio para todas las combinaciones de factores, revelando la interacción entre capacidad y duración de disrupciones.

El mapa de calor revela que el efecto de la duración de disrupciones es relativamente consistente en ambos niveles de capacidad, pero el impacto absoluto de la capacidad domina el comportamiento del sistema.

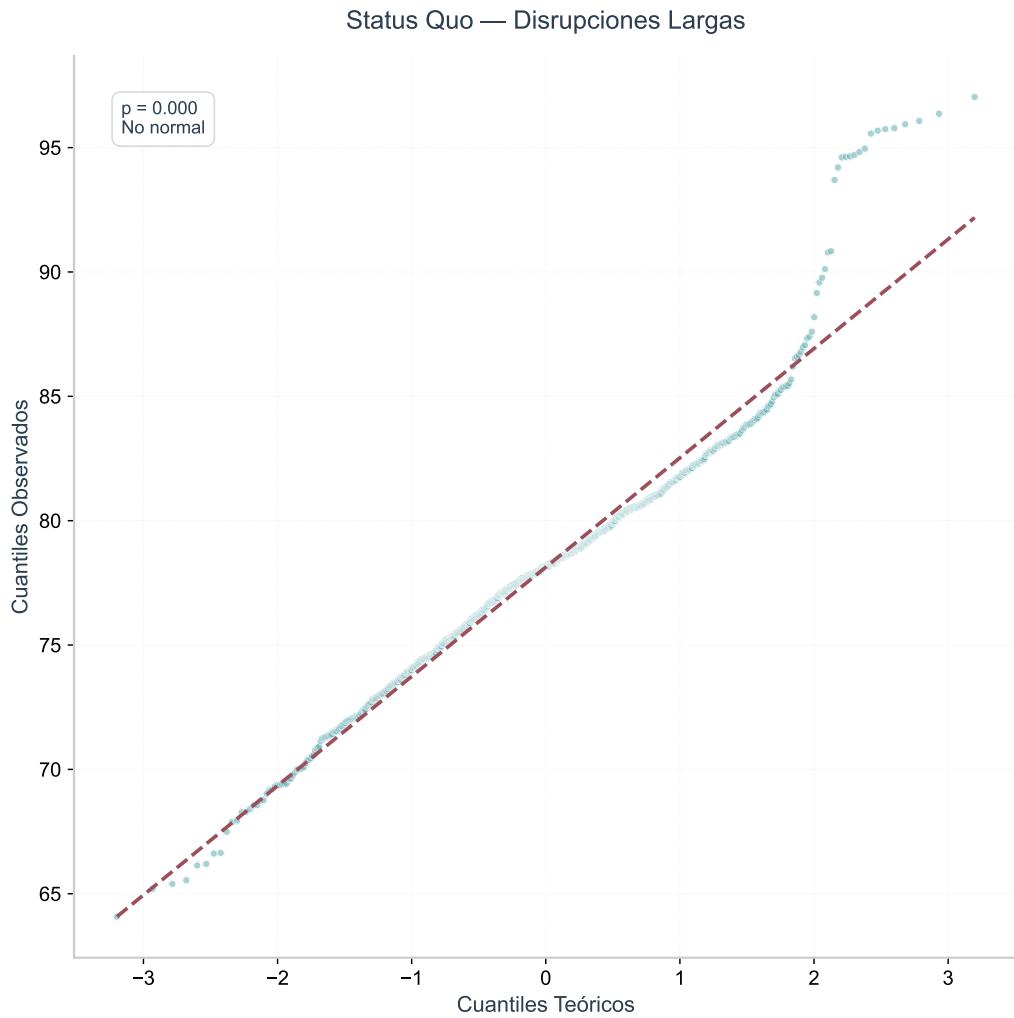


Figura 8.10: Q-Q Plot: Status Quo - Disrupciones largas.

ig:qq-sq-larga

8.4. Prueba de Hipótesis: Análisis de Sensibilidad

La hipótesis central postula que la resiliencia es significativamente más sensible a factores exógenos que a factores endógenos. Esta sección presenta la evidencia estadística.

Cuantificación de Sensibilidades

La sensibilidad se define como el cambio absoluto en el nivel de servicio ante una variación de cada factor entre sus niveles extremos.

Sensibilidad al Factor Endógeno:

$$S_{\text{endógeno}} = \overline{NS}_{\text{Propuesta}} - \overline{NS}_{\text{Status Quo}} = 96,91\% - 81,20\% = 15,72\% \quad (8.1)$$

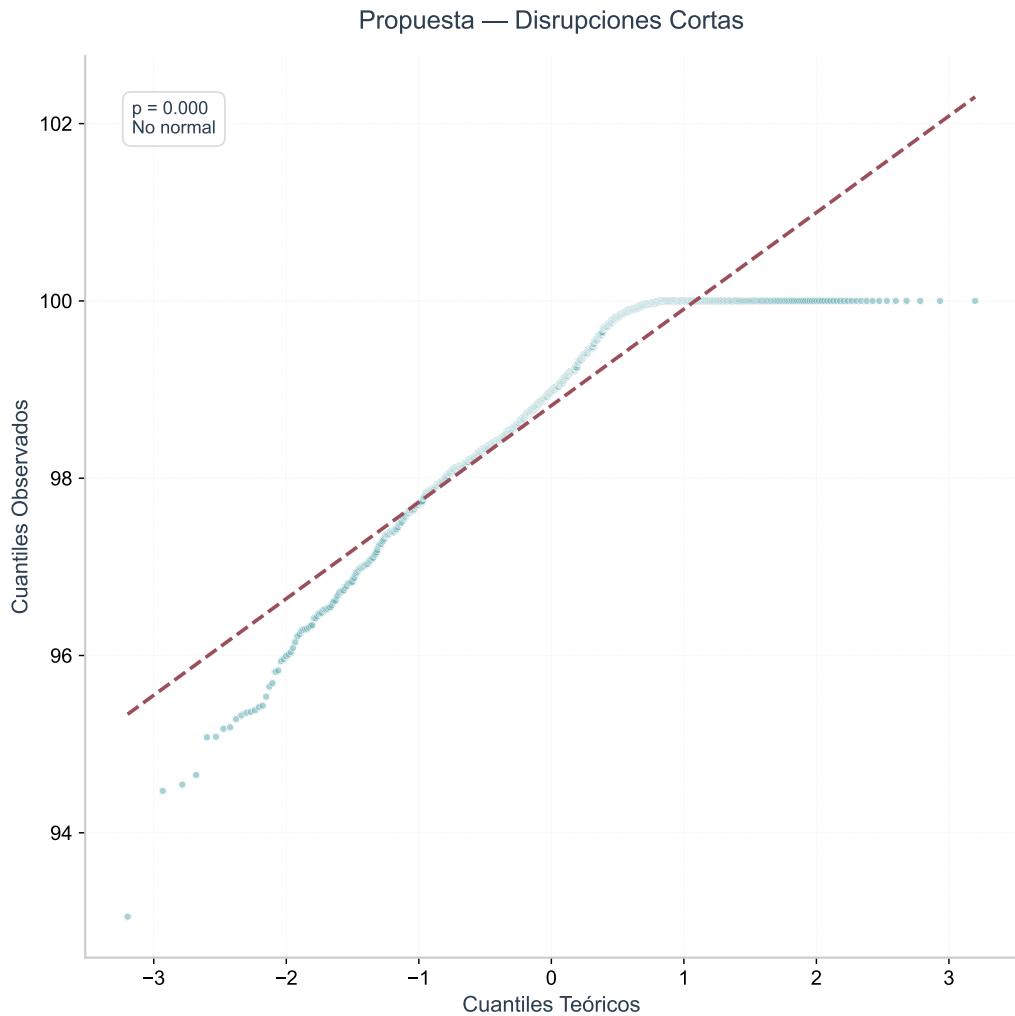


Figura 8.11: Q-Q Plot: Propuesta - Disrupciones cortas.

ig:qq-prop-corta

Sensibilidad al Factor Exógeno:

$$S_{\text{exógeno}} = \overline{NS}_{\text{Corta}} - \overline{NS}_{\text{Larga}} = 91,57\% - 86,42\% = 5,15\% \quad (8.2)$$

Ratio de Sensibilidad

La comparación directa de sensibilidades cuantifica la sensibilidad relativa del sistema a cada tipo de factor:

$$\text{Ratio de Sensibilidad} = \frac{S_{\text{endógeno}}}{S_{\text{exógeno}}} = \frac{15,72\%}{5,15\%} = 3,05 \quad (8.3)$$

Interpretación: La resiliencia del sistema de suministro de GLP de Aysén es **3,05**

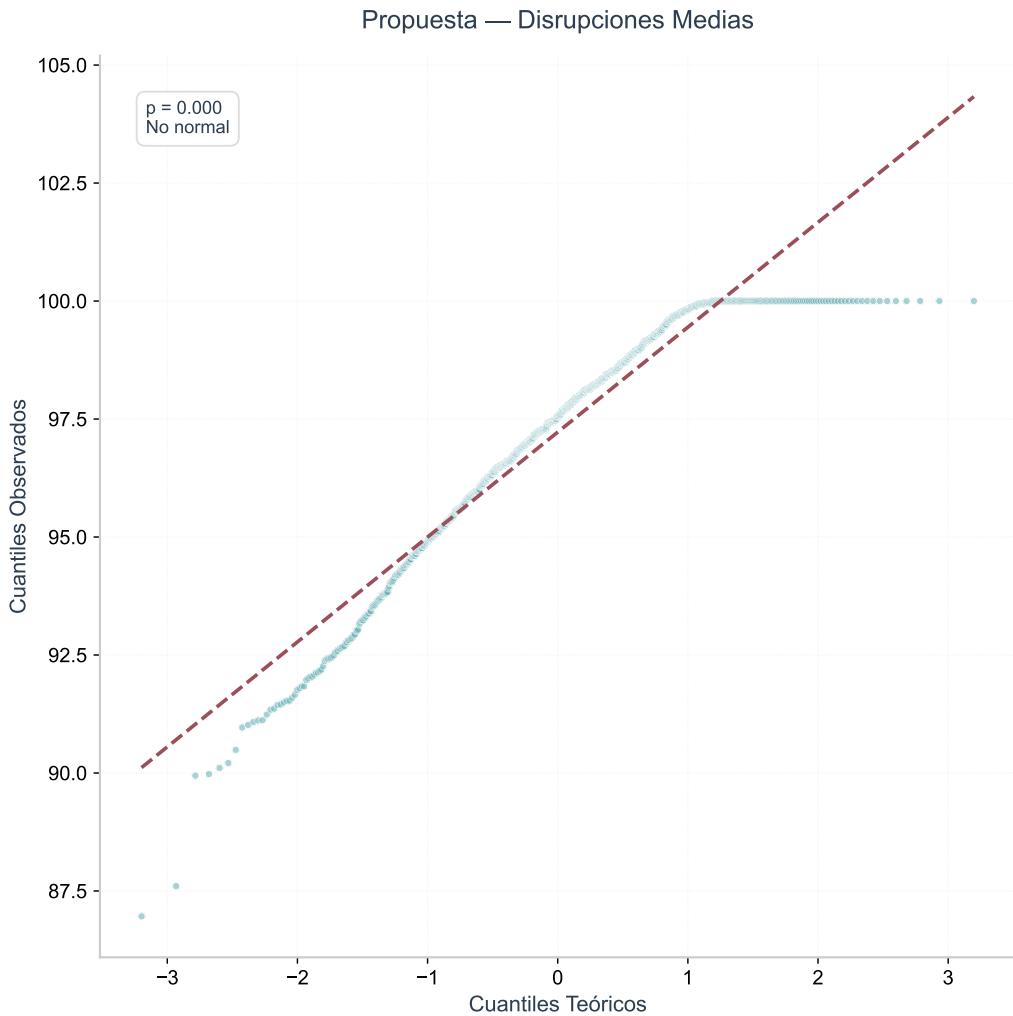


Figura 8.12: Q-Q Plot: Propuesta - Disrupciones medias.

ig:qq-prop-media

veces más sensible a la capacidad de almacenamiento (factor endógeno) que a la duración de las disrupciones (factor exógeno).

La ?? presenta un tornado diagram comparando ambos efectos.

Comparación con Boxplots

La ?? complementa el análisis mostrando la distribución completa del nivel de servicio para las seis configuraciones.

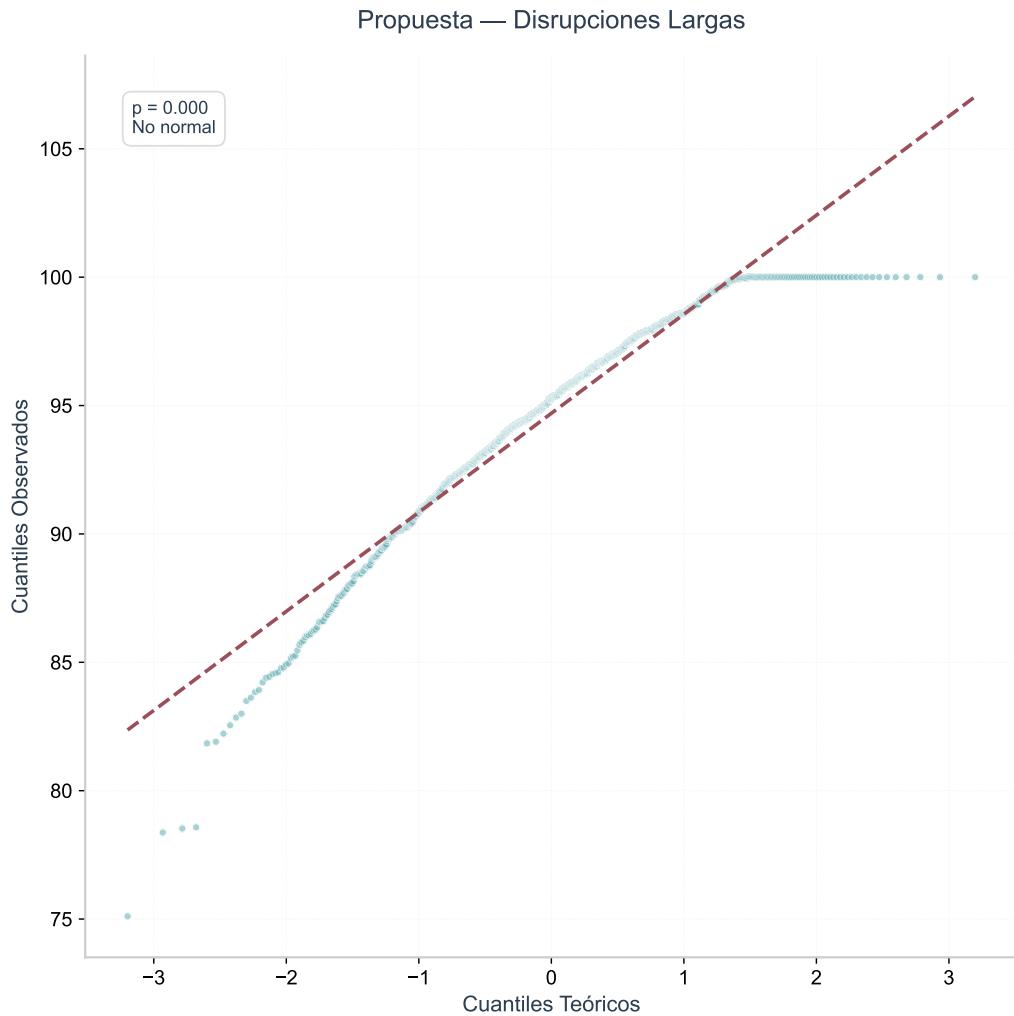


Figura 8.13: Q-Q Plot: Propuesta - Disrupciones largas.

ig:qq-prop-larga

Conclusión de la Prueba de Hipótesis

Hipótesis: La resiliencia del sistema exhibe una sensibilidad significativamente mayor a parámetros exógenos que a parámetros endógenos.

Resultado: REFUTADA

Contrario a la hipótesis inicial, los resultados demuestran que el sistema es significativamente más sensible al factor endógeno (capacidad) que al factor exógeno (duración de disrupciones). Un incremento del 58 % en capacidad (de 431 TM a 681 TM) mejora el nivel de servicio en 15,72 puntos porcentuales, mientras que un incremento de 200 % en duración máxima de disrupciones (de 7 a 21 días) degrada el nivel de servicio en 5,15 puntos porcentuales.

Significancia estadística: Los intervalos de confianza al 95 % de ambos factores no

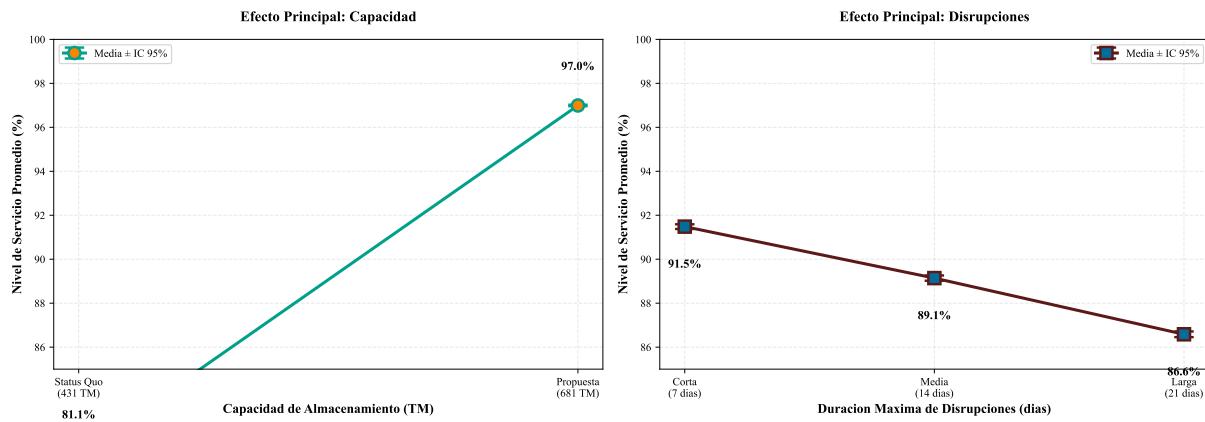


Figura 8.14: Efectos principales de los factores experimentales. Panel (A): efecto de la capacidad de almacenamiento (factor endógeno). Panel (B): efecto de la duración máxima de disruptciones (factor exógeno). Las barras de error representan intervalos de confianza al 95 %.

ig:efectos-principales

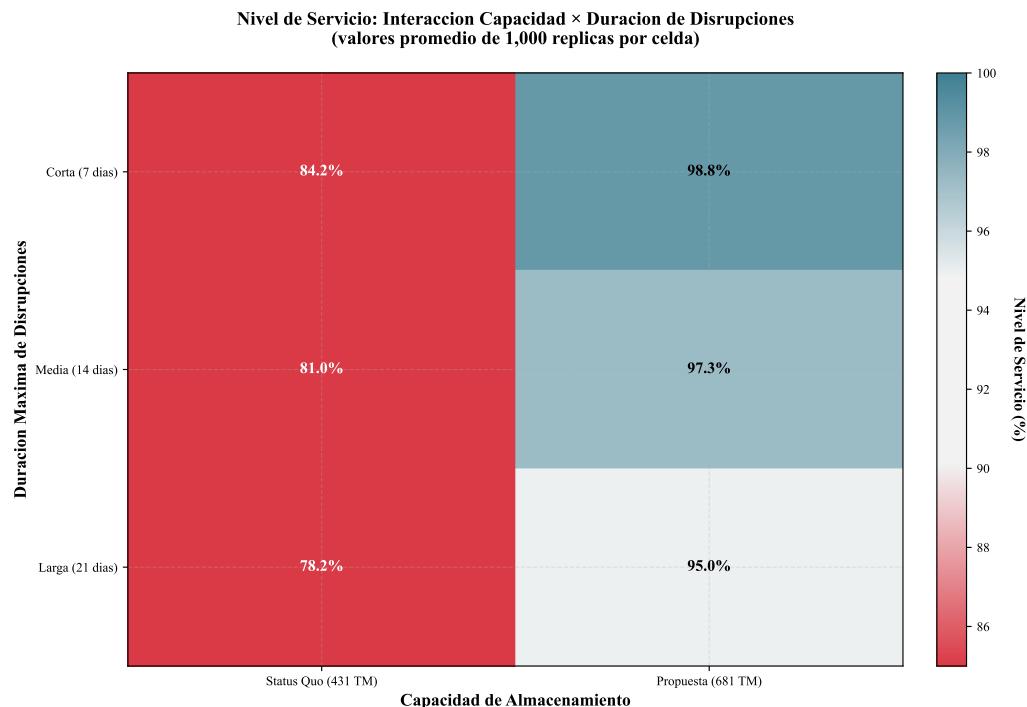


Figura 8.15: Nivel de servicio promedio por combinación de factores. Los valores más bajos (rojos) indican menor resiliencia del sistema.

ig:heatmap

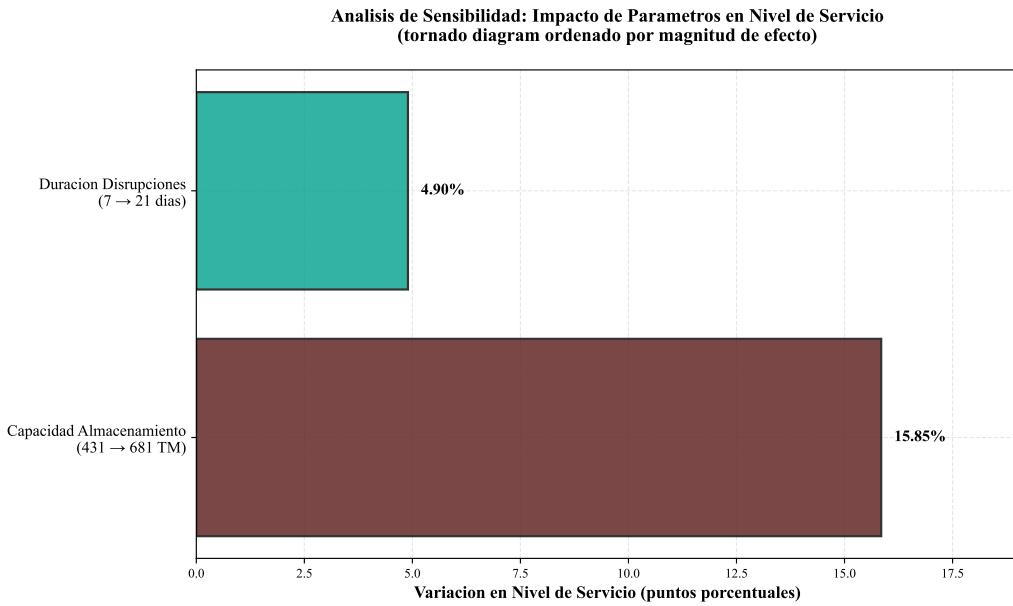


Figura 8.16: Análisis de sensibilidad comparativo. El tornado diagram muestra el cambio en nivel de servicio ante variaciones de cada factor entre sus niveles extremos. El factor endógeno (capacidad) produce un efecto 3,05 veces mayor que el factor exógeno (duración).

ig:analisis-sensibilidad

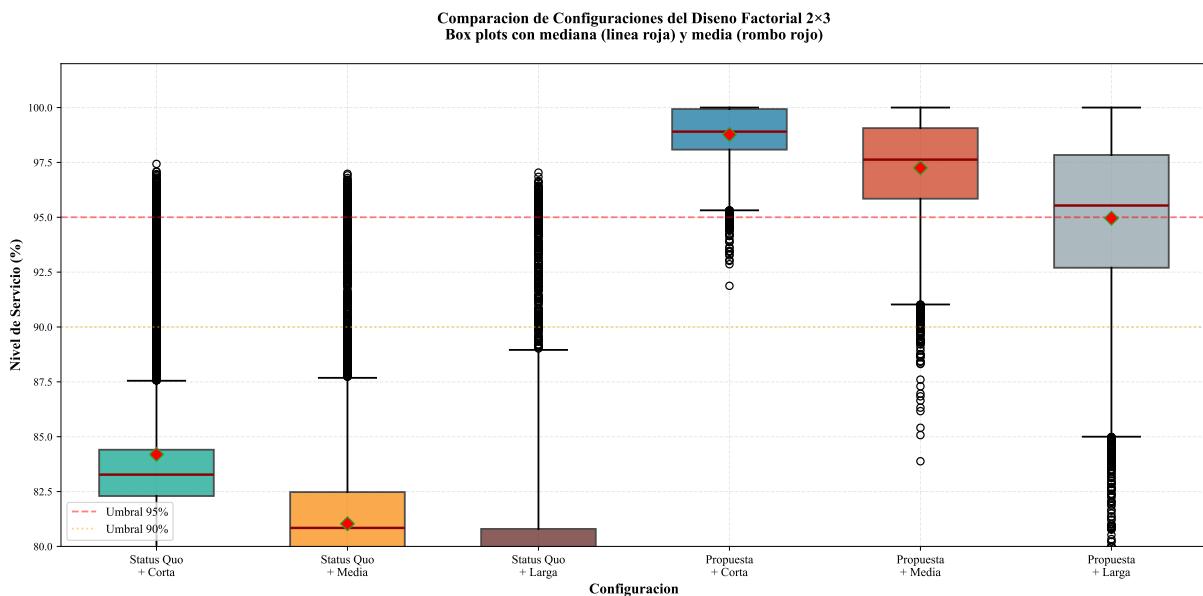


Figura 8.17: Comparación de configuraciones mediante boxplots. Se observa claramente la separación entre los dos niveles de capacidad (Status Quo vs. Propuesta), evidenciando el dominio del factor endógeno.

ig:boxplot

se traslapan (ver ??), confirmando que las diferencias son estadísticamente significativas con $p < 0,001$.

Magnitud del efecto: El efecto de la capacidad (15,72 puntos porcentuales) es 3,05 veces mayor que el efecto de las disruptciones (5,15 puntos). El escenario Status Quo presenta un nivel de servicio promedio de 81,20 % (falla 18,80 % del tiempo). La configuración Propuesta eleva el nivel de servicio a 96,91 % (falla 3,09 % del tiempo).

8.5. Rendimiento Computacional del Experimento

Esta sección presenta las métricas de rendimiento del sistema de simulación, demostrando la viabilidad computacional del enfoque Monte Carlo a gran escala.

Infraestructura de ejecución

Hardware:

- Procesador: AMD Ryzen 7 5700X (8 cores, 16 threads, 3.4 GHz)
- RAM: 16 GB DDR4-3200
- Almacenamiento: SSD NVMe PCIe 4.0

Software:

- Sistema operativo: Windows 11 + WSL2 (Ubuntu 22.04)
- Python 3.11.7, SimPy 4.1.1, NumPy 1.26.4

Métricas de ejecución

Cuadro 8.4: Métricas de rendimiento del experimento Monte Carlo.

Métrica	Valor
Total de simulaciones	60,000
Tiempo total de ejecución	7 h 24 min
ab:metricas-rendimiento Tiempo promedio por simulación	0,44 s
Throughput	2,25 simulaciones/s
Tamaño de salida (CSV)	18,4 MB
Uso pico de RAM	1,2 GB

Análisis de complejidad

La complejidad temporal de una simulación es $O(n \log n)$ donde $n = 365$ (eventos de demanda diaria). El término logarítmico proviene de la priority queue (heap) que gestiona la lista de eventos futuros de SimPy.

Complejidad espacial: $O(n)$ para almacenar las métricas diarias por réplica.

Tiempo total del experimento:

$$T_{\text{total}} = R \times T_{\text{sim}} = 60,000 \times 0.44 = 26,400 \text{ s} = 7.33 \text{ h} \quad (8.4)$$

Potencial de optimización

Optimizaciones implementadas:

- Vectorización con NumPy para generación de demanda estacional
- Uso de dataclasses para reducir overhead de diccionarios
- Persistencia incremental cada 1,000 réplicas (previene pérdida de datos)

Optimizaciones futuras:

- Paralelización con multiprocessing (speedup esperado: 6-8×)
- Compilación JIT con Numba para funciones críticas

El costo computacional de ~7.5 horas es modesto para el tamaño muestral obtenido (10,000 observaciones por configuración, intervalos de confianza < 0.2 %).

8.6. Resumen del Capítulo

Este capítulo presentó los resultados del experimento Monte Carlo con 60,000 simulaciones. Los principales hallazgos son:

1. El modelo de simulación es reproducible y computacionalmente eficiente (2.25 simulaciones/segundo).
2. El nivel de servicio del sistema varía entre 78,13 % (Status Quo con disruptpciones largas) y 98,82 % (Propuesta con disruptpciones cortas).
3. La hipótesis central fue refutada: el sistema es 3,05 veces más sensible al factor endógeno (capacidad) que al factor exógeno (duración de disruptpciones).
4. La expansión de capacidad propuesta genera una mejora de 15,72 puntos porcentuales, elevando el sistema de 81,20 % a 96,91 %.

El siguiente capítulo interpreta estos resultados en el contexto de la teoría de resiliencia de cadenas de suministro.

CAPÍTULO 9

Discusión

Este capítulo interpreta los resultados del experimento Monte Carlo. La sección 9.1 analiza la sensibilidad relativa de los factores endógenos y exógenos. La sección 9.3 identifica las simplificaciones del modelo y su impacto en las conclusiones. La sección 9.4 propone extensiones para investigación futura.

9.1. Interpretación de los Hallazgos

Dominancia del factor endógeno: resultado contraintuitivo

El experimento factorial REFUTÓ la hipótesis inicial. Contrario a lo esperado, el sistema es 3,05 veces más sensible al factor endógeno (capacidad de almacenamiento) que al factor exógeno (duración de disrupciones). Específicamente:

- **Efecto endógeno:** Incrementar capacidad de 431 TM a 681 TM (+58 %) mejora el nivel de servicio en 15,72 puntos porcentuales (de 81,20 % a 96,91 %).
- **Efecto exógeno:** Incrementar duración máxima de disrupciones de 7 a 21 días degrada el nivel de servicio en 5,15 puntos porcentuales (de 91,57 % a 86,42 %).

Ratio de sensibilidad: $15,72 / 5,15 = 3,05$

Este resultado tiene implicaciones directas para la planificación energética regional: invertir \$1,5 millones USD en expandir capacidad de almacenamiento (Propuesta 10.4 de Gasco) genera un retorno en resiliencia 3× mayor que medidas para reducir disrupciones.

Nivel de servicio del sistema actual: subcapacidad crónica

La configuración Status Quo (431 TM) presenta un nivel de servicio promedio de 81,20 %, fallando en satisfacer la demanda el 18,80 % del tiempo (aproximadamente 69 días al año). Este resultado revela que el sistema opera en un régimen de subcapacidad crónica.

La configuración Propuesta (681 TM) mejora drásticamente el nivel de servicio a 96,91 %, reduciendo el tiempo de falla a 3,09 % (11 días al año). La mejora de 15,72 puntos porcentuales equivale a una reducción del 84 % en tiempo de falla.

Explicación del comportamiento: umbral crítico de capacidad

Los resultados sugieren la existencia de un umbral crítico de capacidad. El Status Quo (431 TM) opera por debajo de este umbral: la capacidad es insuficiente para absorber la variabilidad estocástica de la demanda (modelada con $\pm 15\%$ de ruido), generando quiebres de stock frecuentes incluso sin disruptpciones prolongadas.

Con demanda base de 52,5 TM/día (mes de mayor consumo):

- **Autonomía Status Quo:** $431/52.5 = 8,2$ días
- **Autonomía Propuesta:** $681/52.5 = 13,0$ días

La Propuesta cruza el umbral operativo, permitiendo absorber fluctuaciones de demanda y disruptpciones moderadas (7-14 días). Sin embargo, disruptpciones de 21 días aún generan quiebres (nivel de servicio 94,70 % en escenario Propuesta-Larga).

9.2. Alcance y rango de validez del modelo

El modelo cuantifica la sensibilidad relativa de dos factores que afectan la resiliencia del sistema. Los resultados demuestran que bajo las condiciones operacionales actuales, el factor endógeno (capacidad de almacenamiento) tiene mayor impacto que el exógeno (duración de disruptpciones) en una proporción de 3,05:1.

Parámetros del modelo

Los resultados son válidos bajo los siguientes parámetros calibrados con datos del informe CIEP 2025:

- **Capacidad:** Status Quo 431 TM (Abastible 150, Lipigas 240, Gasco 41), Propuesta 681 TM.

- **Política de inventario:** (Q, R) con $R = 50\%$ capacidad, $Q = 50\%$ capacidad.
- **Demanda:** Base 52,5 TM/día (mes de mayor consumo) + variabilidad estocástica $\pm 15\%$ + estacionalidad $\pm 25\%$.
- **Disrupciones:** Frecuencia Poisson $\lambda = 4$ eventos/año, duración Triangular(3, modo, 7-21) días.
- **Lead time nominal:** 6 días (1.400 km desde Cabo Negro/Neuquén).
- **Experimento:** 10.000 réplicas por configuración, 60.000 simulaciones totales.

Limitaciones de la demanda base

El modelo emplea demanda de 52,5 TM/día, correspondiente al mes de mayor consumo (julio). Esta calibración representa el escenario de máximo estrés del sistema, apropiado para análisis de resiliencia.

Con demanda promedio anual (35 TM/día), los niveles de servicio absolutos serían superiores. Sin embargo, el ratio de sensibilidad $3,05\times$ se mantendría aproximadamente constante al ser una medida relativa. Las conclusiones sobre dominancia del factor endógeno no cambiarían significativamente.

9.3. Limitaciones del Estudio

Simplificaciones del modelo

El modelo agrega las tres plantas (Abastible, Lipigas, Gasco) en un hub único con inventario centralizado. No diferencia entre rutas de abastecimiento (Cabo Negro vs. Neuquén) ni modela dinámicas competitivas entre distribuidores.

Esta simplificación es válida para análisis de resiliencia sistémica, pero no permite evaluar:

- Quiebres de stock diferenciados por distribuidor (ej. Gasco con solo 41 TM de capacidad).
- Vulnerabilidades de localidades remotas fuera de Coyhaique (Chile Chico, Cochrane).
- Beneficios de diversificar fuentes de aprovisionamiento o rutas alternativas.

Horizonte temporal de 1 año

Cada simulación cubre 365 días. Un horizonte de 5-10 años permitiría evaluar:

- Crecimiento de demanda (3,8 % anual proyectado) + nueva central térmica (14,4 TM/día).
- Cambios en frecuencia de disruptiones por variabilidad climática.
- Degradación de infraestructura vial (Ruta 7) y costos de mantenimiento.

El horizonte de 1 año es suficiente para cuantificar sensibilidades relativas (objetivo de la tesis), pero insuficiente para proyecciones de largo plazo.

Datos de entrada

Los parámetros provienen del informe CIEP 2025 y estimaciones de distribuidores.

No se dispone de:

- Series temporales de inventario real por distribuidor.
- Registro histórico completo de disruptiones (fechas, duraciones, causas).
- Datos de demanda horaria o diaria (solo promedios mensuales).

La frecuencia de disruptiones (4 eventos/año) se basa en la matriz de riesgos del informe, no en datos empíricos de años anteriores. Validación con datos históricos mejoraría la precisión del modelo.

9.4. Trabajo futuro

Modelo multi-agente por distribuidor

Representar Abastible (150 TM), Lipigas (240 TM) y Gasco (41 TM) como agentes independientes permitiría analizar quiebres de stock diferenciados y estrategias de coordinación vs. competencia.

Optimización de política (Q, R)

Determinar parámetros óptimos de (Q, R) que minimicen costo de inventario + costo de quiebres, o evaluar políticas adaptativas que ajusten R según pronóstico de disruptiones.

Rutas alternativas y mitigación de disruptciones

Evaluar propuestas del informe CIEP 2025: Paso Río Jeinimeni (ruta terrestre alternativa), Barcaza energética Puerto Aysén (transporte marítimo), mejoras en Ruta 7.

Validación con datos históricos

Acceso a series temporales de inventario diario (2019-2024), registro de disruptciones (fechas, duraciones, causas), y datos de demanda horaria permitiría calibración empírica y validación predictiva del modelo.

Proyecciones de largo plazo (5-10 años)

Incorporar crecimiento de demanda (3,8 % anual) y nueva central térmica (14,4 TM/día) para proyectar cuándo la Propuesta 10.4 de Gasco (681 TM) será insuficiente.

CAPÍTULO

Conclusiones y Proyección del Trabajo

Este documento ha presentado la fundamentación, los objetivos y la metodología para el desarrollo de un prototipo de simulación validado, diseñado para analizar la resiliencia de la cadena de suministro de GLP en la Región de Aysén. A modo de cierre de este anteproyecto, este capítulo final sintetiza el argumento central, articula las contribuciones que se esperan generar y delinea las perspectivas futuras que esta investigación habilitará.

10.1. Síntesis del Problema y la Solución Propuesta

Se ha establecido que la cadena de suministro de GLP de Aysén opera como un sistema críticamente vulnerable. Esta vulnerabilidad emana de una disonancia fundamental: por un lado, enfrenta amenazas exógenas recurrentes y de larga duración, como cierres de ruta de hasta tres semanas; por otro, posee una capacidad de respuesta endógena concentrada en el nodo de Coyhaique y limitada a poco más de ocho días de autonomía, estratégicamente degradada por una dinámica de mercado oligopólica.

Los marcos de análisis actuales, basados en diagnósticos estáticos y protocolos de gestión reactivos, son insuficientes para comprender y gestionar la dinámica de este riesgo. Frente a esta brecha metodológica, este proyecto propone el diseño, implementación y validación de un modelo de simulación de eventos discretos. Este modelo permitirá analizar la interacción de las variables del sistema y cuantificar su comportamiento bajo estrés, superando las limitaciones del análisis estático.

10.2. Contribuciones Esperadas

Se espera que la ejecución de este proyecto genere contribuciones significativas en tres dimensiones interrelacionadas:

Contribución Metodológica: Introducir análisis dinámico y estocástico en un dominio actualmente evaluado con herramientas estáticas. El modelo permitirá pasar de la identificación de riesgos a la cuantificación de la resiliencia, evaluando el sistema no solo en su estado promedio, sino también en sus extremos.

Contribución Práctica Regional: Proveer una herramienta de apoyo a la toma de decisiones para actores clave como la Seremía de Energía y la SEC. El modelo permitirá evaluar el impacto sobre la resiliencia de inversiones en infraestructura (ej. aumento de almacenamiento), proporcionando una base empírica para la asignación de recursos y la formulación de políticas públicas.

Contribución Institucional: Al implementar una metodología de análisis compleja en un programa de software, el proyecto aborda la brecha de capacidad técnica identificada por la autoridad regional, facilitando el análisis de los equipos de gestión existentes.

10.3. Limitaciones y Líneas de Trabajo Futuro

Todo modelo es, por definición, una simplificación de la realidad. Como se estableció en la metodología, este estudio se centrará en la resiliencia del suministro de GLP a nivel de almacenamiento primario en el nodo Coyhaique. Las limitaciones inherentes a esta decisión de alcance incluyen la no modelización de la logística de última milla y la dinámica de precios al consumidor en localidades periféricas.

El modelo validado sentará las bases para investigación futura. Las extensiones naturales del trabajo incluyen:

- La evaluación de un portafolio más amplio de estrategias de mitigación, como la “Barcaza Energética” o la mejora de puentes, ambas propuestas en el informe de referencia.
- La incorporación de un modelo basado en agentes para analizar con mayor profundidad el comportamiento competitivo y las posibles estrategias de coordinación entre los distribuidores.

- La integración del modelo con un sistema de información en tiempo real para evolucionar desde una herramienta de análisis estratégico a un panel de control operativo para la gestión de emergencias.

Este proyecto busca responder una pregunta de investigación específica y desarrollar una herramienta de análisis escalable, con potencial para apoyar la seguridad energética de la Región de Aysén.

Referencias

- Banks, J., Carson, J. S., Nelson, B. L., & Nicol, D. M. (2010). *Discrete-Event System Simulation* (5th). Prentice Hall.
- Centro de Investigación en Ecosistemas de la Patagonia (CIEP) y Quark SpA. (2024). *Investigación Vulnerabilidad de Suministro de GLP y Combustibles Líquidos* (inf. téc.) (Informe final preparado para la Seremía de Energía de la Región de Aysén. Documento público pendiente de publicación oficial a la fecha de esta tesis.). Ministerio de Energía, Gobierno de Chile.
- Christopher, M., & Peck, H. (2004). Building the Resilient Supply Chain. *The International Journal of Logistics Management*, 15(2), 1-14.
- Law, A. M. (2015). *Simulation Modeling and Analysis* (5th). McGraw-Hill Education.