



UNIVERSIDAD
DE LA REPUBLICA
URUGUAY

Planificación conjunta de la producción, la remanufacturación y el transporte bajo objetivos de recuperación

Luciana Vidal

Programa de Posgrado en Investigación de Operaciones
Facultad de Ingeniería
Universidad de la República

Montevideo – Uruguay
Octubre de 2025



UNIVERSIDAD
DE LA REPUBLICA
URUGUAY

Planificación conjunta de la producción, la remanufacturación y el transporte bajo objetivos de recuperación

Luciana Vidal

Tesis de Maestría presentada al Programa de Posgrado en Investigación de Operaciones, Facultad de Ingeniería de la Universidad de la República, como parte de los requisitos necesarios para la obtención del título de Magíster en Investigación de Operaciones.

Director:

Dr. Ing. Pedro Piñeyro

Director académico:

Prof. Omar Viera

Montevideo – Uruguay

Octubre de 2025

Vidal, Luciana

Planificación conjunta de la producción, la remanufacturación y el transporte bajo objetivos de recuperación / Luciana Vidal. - Montevideo: Universidad de la República, Facultad de Ingeniería, 2025.

X, 78 p.: il.; 29, 7cm.

Director:

Pedro Piñeyro

Director académico:

Omar Viera

Tesis de Maestría – Universidad de la República, Programa en Investigación de Operaciones, 2025.

Referencias bibliográficas: p. 60 – 66.

1. Dimensionamiento del Lote, 2. Remanufacturación, 3. Transporte, 4. Objetivos de recuperación, 5. Tabu Search. I. Piñeyro, Pedro, . II. Universidad de la República, Programa de Posgrado en Investigación de Operaciones. III. Título.

INTEGRANTES DEL TRIBUNAL DE DEFENSA DE TESIS

Dra. Ing Soledad Gutiérrez

Dr. Ing Claudio Risso

Dr. Carlos Testuri

Montevideo – Uruguay
Octubre de 2025

RESUMEN

Esta tesis presenta un sistema híbrido de producción y remanufacturación de un único producto, conformado por un fabricante original (OEM) y un conjunto de clientes, en el que se establecen objetivos de recolección y remanufacturación. El problema consiste en determinar, de forma simultánea, los períodos y las cantidades a producir, remanufacturar y transportar tanto de productos finales como de retornos, con el fin de satisfacer la demanda a tiempo y cumplir con los objetivos de recuperación. En este sistema se incurre en costos fijos y variables asociados a cada una de las tres actividades, además de costos por mantener inventario de retornos y de productos finales, tanto en el OEM como en los clientes. Los objetivos de recuperación se plantean como límites inferiores sobre la cantidad de productos usados que deben ser recogidos y remanufacturados por el OEM. Se propone una formulación de programación lineal entera mixta (MILP) para el problema, y se diseñan procedimientos de resolución basados en la metaheurística de Tabu Search. Estos procedimientos se evalúan ampliando un conjunto de instancias de la literatura, y se comparan con un solver de optimización del estado del arte, considerando diversas configuraciones de parámetros del problema. Los resultados de la experimentación numérica permiten concluir que, al menos, una de las variantes del procedimiento de resolución resulta efectiva tanto en términos de costos como de tiempos de ejecución, llegando incluso a superar el desempeño del solver en algunas instancias de gran tamaño.

Palabras claves:

Dimensionamiento del Lote, Remanufacturación, Transporte, Objetivos de recuperación, Tabu Search.

Publicaciones relacionadas

La realización de esta tesis de maestría dio origen a los siguientes trabajos y presentaciones en conferencias:

- Vidal, L., Piñeyro, P., y Viera, O. (2025). The integrated lot-sizing problem of production, remanufacturing and transportation, considering recovery targets. *Journal of Remanufacturing*, 15, 127-149. <http://dx.doi.org/10.1007/s13243-025-00150-1>.
- Vidal, L., Piñeyro, P., y Viera, O. (2023b). The production-remanufacturing-transportation lot-sizing problem with recovery targets: formulation and solution procedures. *Proceedings of the 23rd International Conference of the International Federation of Operational Research Societies, IFORS 2023*, 1-4 August 2023, Santiago, Chile, 73.
- Vidal, L., Piñeyro, P., y Viera, O. (2023a). Hybrid production remanufacturing planning problem with bidirectional flows of items and recovery targets. *Proceedings of the 5th International Conference on Remanufacturing, ICoR 2023*, 27-29 June 2023, Amsterdam, Netherlands, 51-65.

Lista de tablas

4.1	Nomenclatura de los parámetros.	27
4.2	Nomenclatura de las variables de decisión.	28
6.1	Configuraciones para los conjuntos de datos.	40
6.2	Resultados de gap de costos y tiempo de ejecución de instancias chicas para $\alpha = \beta = 0$	43
6.3	Resultados de gap de costos y tiempo de ejecución de instancias chicas para $\alpha = \beta = 0.5$	44
6.4	Resultados de gap de costos y tiempo de ejecución de instancias chicas para $\alpha = \beta = 1$	45
6.5	Resultados de gap de costos y tiempo de ejecución de instancias grandes para $\alpha = \beta = 0$	48
6.6	Resultados de gap de costos y tiempo de ejecución de instancias grandes para $\alpha = \beta = 0.5$	49
6.7	Resultados de gap de costos y tiempo de ejecución de instancias grandes para $\alpha = \beta = 1$	50
6.8	Resultados de gap de costo promedio para la variante TSv3-50-E3 para el juego de datos nT24-nL3.	56
1.1	Descripción de los endpoints del proyecto.	73
2.1	Análisis de rango de iteraciones para instancias chicas utilizando la variante TSv3.	78

Lista de figuras

3.1	Sistema de inventario con producción y remanufacturación, basado en el diagrama presentado en Teunter et al. (2006).	13
4.1	Esquema del Problema de Dimensionamiento del Lote Económico con Manufacturación, Remanufacturación y Transporte. . . .	24
5.1	Flujo del procedimiento de resolución base TSv1.	35
6.1	Performance de los procedimientos de resolución para instancias pequeñas. $\alpha = \beta = 0$ se representa con 1, $\alpha = \beta = 0.5$ con 2 y $\alpha = \beta = 1$ con 3.	42
6.2	Performance de los procedimientos de resolución para instancias grandes. $\alpha = \beta = 0$ se representa con 1, $\alpha = \beta = 0.5$ con 2 y $\alpha = \beta = 1$ con 3.	51
6.3	Distribución del tiempo en segundos requerido por CPLEX para llegar a la mejor solución factible que pudo encontrar para el juego de datos nT24-nL3.	52
6.4	Comparación entre los tiempos de ejecución de TSv3-50-E3 contra CPLEX para el juego de datos nT24-nL3.	53
1.1	Arquitectura en alto nivel de la solución.	69
1.2	Arquitectura en alto nivel de la solución.	69
1.3	Arquitectura en alto nivel de la solución.	74
1.4	Vista de resultados mostrando los detalles de un archivo específico.	74
1.5	Vista de resultados mostrando los detalles de un archivo que fue ejecutado de forma masiva.	75
1.6	Vista de ejecución.	76

Tabla de contenidos

Publicaciones relacionadas	VI
Lista de tablas	VII
Lista de figuras	VIII
1 Introducción	1
2 Conceptos claves	4
2.1 Producción y recursos	4
2.2 Estrategias para reducir el impacto ambiental de la producción .	6
3 Revisión de la literatura	9
3.1 Problemas de Dimensionamiento del Lote Económico	9
3.2 Problema de Dimensionamiento del Lote Económico con Rema- nufacturación	13
3.2.1 Consideraciones sobre el transporte	17
3.2.2 Objetivos de recuperación	19
4 Definición y modelado del problema	23
4.1 Nomenclatura para el modelo	25
4.2 Formulación del modelo	28
5 Procedimiento de resolución	31
5.1 Representación de una solución factible	32
5.2 Procedimiento de resolución base	32
5.3 Variantes del procedimiento de resolución base	35
5.4 Estrategia de post optimización	36

6	Experimentación numérica	38
6.1	Instancias pequeñas nL3-nT12	41
6.2	Instancias grandes nL3-nT24	46
6.3	Interpretación de los resultados para la toma de decisiones . . .	54
7	Conclusiones	57
	Referencias	60
	Apéndices	67
	Apéndice 1 Implementación	68
	Apéndice 2 Evaluación de cantidad de iteraciones	77

Capítulo 1

Introducción

La industria manufacturera genera un impacto negativo en el medio ambiente debido a sus altas emisiones de carbono, consumo energético y desechos plásticos y electrónicos. Las emisiones y el consumo de energía son dos de los principales factores del cambio climático (Qiao et al., [2021](#)). Greenpeace ([2016](#)) reporta que alrededor de 8 millones de toneladas de plástico terminan en los océanos cada año, afectando gravemente a la fauna marina, con la muerte de más de 1 millón de aves y 100.000 mamíferos. En 2016, se generaron 44.7 millones de toneladas de desechos electrónicos, con un crecimiento anual estimado del 3 al 4 por ciento (Sahajwalla y Gaikwad, [2018](#)). Este escenario subraya la necesidad de adoptar modelos de producción alternativos.

La economía circular, promovida por la fundación Ellen MacArthur, propone reemplazar el modelo de "producir, usar y desechar" por uno que facilite la recuperación de productos usados mediante distintas estrategias (Ellen Macarthur Foundation, [s.f.](#)). Las estrategias de recuperación de productos al final de su vida útil incluyen la reparación, refurbishing, reacondicionamiento, remanufacturación, canibalización y reciclaje (Alamerew y Brissaud, [2019](#)); se diferencian según el grado de intervención en el producto y el valor que logran recuperar:

- Reparación: devuelve el producto a un estado funcional corrigiendo fallas específicas.
- Refurbishing: restaura el producto a un nivel aceptable, sin alcanzar la calidad de uno nuevo.
- Reacondicionamiento: implica un proceso más riguroso sobre el producto, que incluye pruebas extensas, aunque sigue ofreciendo una calidad

inferior a la de un producto nuevo.

- Canibalización: recupera componentes útiles de productos usados para ser reutilizados en otros procesos.
- Reciclaje: transforma materiales descartados en materias primas para nuevos productos, perdiendo su funcionalidad inicial.
- Remanufacturaación: restaura los productos usados a un estado equivalente o superior al de un producto nuevo.

La remanufacturaación ofrece beneficios tanto ambientales como económicos. Esta técnica puede reducir el consumo de energía hasta en un 85 %, disminuir las emisiones de carbono y optimizar el uso de materias primas críticas (European Remanufacturing Network, [s.f.](#)). Además, proporciona productos más accesibles y con mayor disponibilidad para los consumidores.

El aporte de este trabajo es investigar la integración de manufacturaación, remanufacturaación y transporte, incorporando objetivos de recuperación. Este enfoque busca optimizar recursos, reducir costos operativos y mejorar la gestión de productos al final de su vida útil. Para ello, se diseña un modelo matemático basado en programación lineal entera mixta (MILP) que extiende el Problema de Dimensionamiento del Lote Económico con Remanufacturaación (Teunter et al., [2006](#)), incorporando múltiples clientes, decisiones de transporte y objetivos de recuperación. Esta extensión del problema que incluye decisiones de producción, remanufacturaación y transporte con objetivos de recuperación, se la denomina Problema de Dimensionamiento del Lote Integrado de Producción, Remanufacturaación y Transporte con Objetivos de Recuperación (PRTLSP-RT). Se desarrollan procedimientos de resolución utilizando la metaheurística Tabu Search, los cuales demuestran ser más eficientes en tiempo y costo que un solver de optimización en instancias grandes. En instancias pequeñas, es viable manejar decisiones de producción y transporte por separado, pero en instancias grandes, es preferible integrarlas. Además, el aumento en las tasas de recolección y remanufacturaación incrementa la complejidad del problema.

El resto del documento está organizado de la siguiente manera: en el Capítulo [2](#) se describen conceptos fundamentales sobre los problemas de planificación de la producción con opciones de retornos. En el Capítulo [3](#) se realiza una revisión de los trabajos de la literatura relacionados al problema. Posteriormente, en los Capítulos [4](#) y [5](#) se define el problema y se lo modela matemáticamente, para luego presentar los procedimientos de resolución propuestos para el

problema. La experimentación numérica de los procedimientos de resolución propuestos se presenta en el Capítulo 6. Finalmente, en el Capítulo 7 se exponen las conclusiones y posibles líneas para trabajo futuro.

Capítulo 2

Conceptos claves

En este capítulo se describen los conceptos fundamentales de la planificación de la producción y las estrategias para reducir el impacto ambiental asociado, haciendo énfasis en la remanufacturación como una alternativa.

2.1. Producción y recursos

Como se explica en Karimi et al. (2003), la planificación de la producción decide el uso de los recursos con el fin de satisfacer la demanda en cierto período de tiempo denominado horizonte temporal. El foco del presente documento son los problemas de producción de término medio, en donde se busca tomar decisiones sobre las cantidades a producir a lo largo del horizonte temporal buscando optimizar ciertas métricas, como lo puede ser la minimización del costo económico en la producción.

Los problemas de producción son estudiados en un intervalo de tiempo denominado horizonte temporal, el cual puede ser clasificado según su cantidad de períodos. Un horizonte temporal finito es aquel en donde la cantidad períodos es limitada, en caso contrario sería un horizonte temporal infinito. El primer tipo es en el que se hace foco en este trabajo.

Tal como se comenta al principio del capítulo, los problemas de producción buscan satisfacer la demanda, entendida como la cantidad a producir o comprar para satisfacer una exigencia interna o externa de dicho artículo en un período de tiempo. Los trabajos de Ferrer et al. (2002) y Gungor y Gupta (1999) clasifican a la demanda según distintos criterios:

- Conocimiento sobre su valor:

- Determinística: la demanda se asume conocida en todo momento.
- Estocástica: problemas con incertidumbre en donde típicamente la demanda es definida como una distribución de probabilidad.
- Variabilidad en su valor:
 - Dinámica: la demanda cambia en el tiempo.
 - Estacionaria: la demanda se mantiene constante en todo el horizonte temporal.

Gungor y Gupta (1999), explican que existen problemas que buscan satisfacer la demanda por completo, pero otros permiten tener faltas en la misma incurriendo por ejemplo en una penalización económica. Este último caso, se divide entre aquellos en donde la demanda puede ser satisfecha total o parcialmente con retraso (problemas con *backlogging*) y aquellos en donde se aceptan las pérdidas.

En el trabajo de Ferrer et al. (2002), definen el inventario como el almacenamiento de artículos a utilizarse a lo largo del horizonte temporal. Entonces, si en un período se almacena, en el siguiente se contará con lo producido y aquello que se almacenó para satisfacer la demanda. El inventario se puede clasificar según su revisión:

- Revisión continua: el nivel de inventario se puede medir y cambiar en cualquier momento.
- Revisión periódica: el nivel de inventario es fijo para un intervalo de tiempo de longitud conocida.

Otro concepto importante, es el de los recursos. En Karimi et al. (2003) se explica que los recursos de un sistema productivo incluyen el equipamiento y las máquinas entre otros. Los problemas capacitados son aquellos en donde los recursos tienen limitaciones, como por ejemplo la capacidad de una máquina de producir un producto dado en un período de tiempo. Por otro lado, los problemas no capacitados tienen recursos infinitos.

Finalmente, en los trabajos de Ferrer et al. (2002) y Karimi et al. (2003) se definen distintos tipos de costos:

- Costos de producción: costo asociado a la fabricación de un artículo.
- Costos de inventario: costo de almacenar en inventario un artículo para utilizarlo en períodos futuros. Estos costos incluyen los costos de alma-

cenamiento (renta y pérdida del valor del producto), impuestos, entre otros. También denominado costo de stock.

- Costos de configuración (setup): previo al comienzo de un nuevo lote, se debe generar una configuración que permita producir, por lo que en los modelos debe ser agregado el costo del setup y/o el tiempo incurrido en el mismo.

Los conceptos trabajados en esta sección constituyen la base para el desarrollo de modelos y estrategias que permitan optimizar la producción, diseñando soluciones que satisfagan la demanda y, al mismo tiempo, sean eficientes en términos económicos y ambientales.

2.2. Estrategias para reducir el impacto ambiental de la producción

La industria manufacturera tiene un impacto negativo en la ecología del planeta debido a sus emisiones de carbono, altos consumos de energía, uso intensivo de agua y los desechos que genera. Por ejemplo, la industria textil es una de las más contaminantes, siendo responsable de una parte significativa de la huella de carbono, el consumo excesivo de agua y el desecho de residuos químicos. Esto impacta tanto a nivel social como ambiental, pudiendo considerarse como uno de los factores que generó la crisis de biodiversidad y del clima (Sandin y Peters, 2020). Además, son productos que luego son desechados por los consumidores, contribuyendo a la acumulación de residuos textiles en vertederos y afectando negativamente al medio ambiente.

En línea con la problemática anterior, la fundación Ellen MacArthur es una organización que promueve el uso de la economía circular, la cual plantea cambiar el modelo tradicional de producción (producir, usar y tirar), por uno en donde los productos vuelvan a incorporarse en la primera fase del ciclo (Ellen Macarthur Foundation, [s.f.](#)).

El Acuerdo de París es un tratado internacional impulsado en 2015 que busca mitigar el cambio climático, reduciendo las emisiones de gases de efecto invernadero (United Nations, 2015). En base a dicho acuerdo, el Gobierno de España impulsó un proyecto de Ley de Cambio Climático y Transición Energética (Congreso de Diputados España, 2020) que está en línea con la estrategia de la fundación Ellen MacArthur dado que pretenden una transición a

un modelo circular en donde consigan un mejor uso de los recursos y minimizar la huella ecológica.

Una de las estrategias en la economía circular es la remanufacturación, la cual consiste en convertir un producto usado en uno que tiene las mismas prestaciones (o incluso más) que uno nuevo (Alamerew y Brissaud, 2019). Para ello, el producto usado, denominado retorno, se somete a una serie de etapas tales como limpieza y testeado (European Remanufacturing Network, s.f.). Los retornos pueden presentar variabilidad en su estado y composición, lo que impacta en el proceso de remanufacturación. Sin embargo, cuando los retornos presentan características similares se denominan retornos de calidad homogénea, lo que facilita su procesamiento y optimiza los costos de remanufacturación.

Las ventajas de la remanufacturación para los consumidores son diversas (European Remanufacturing Network, s.f.). Los productos remanufacturados son entre un 60 % y 80 % más económicos que uno nuevo debido a los ahorros en la recuperación de los materiales. Además, estos productos tienden a tener un menor tiempo de espera (tiempo desde que se genera una orden a un proveedor hasta que se realiza la entrega al cliente), lo que significa que el tiempo de entrega, entendido como el intervalo entre la realización del pedido y la llegada del producto al consumidor final, se reduce, lo que mejora la disponibilidad del producto. Finalmente, se le otorga al consumidor flexibilidades a la hora de la compra tales como arrendamiento y devolución dado que las empresas están interesadas en la recuperación del producto para así volver a incorporarlo en el ciclo productivo.

Desde el punto de vista del remanufacturador, también se encuentran ventajas (European Remanufacturing Network, s.f.). La remanufacturación es una fuente de empleo, así como generadora de mejores márgenes de beneficios que la manufacturación. Por otro lado, las empresas remanufacturadoras suelen tener mejores vínculos con sus consumidores, dado que se preocupan en conocer el uso del producto para luego recuperarlo y volver a incorporarlo en el proceso. En este sentido, una cadena de suministro cerrada implica un ciclo continuo donde los productos, una vez al final de su vida útil, son recolectados, remanufacturados y reincorporados al ciclo de producción, reduciendo la necesidad de nuevos recursos y minimizando los residuos generados.

En cuanto al cuidado medioambiental, la remanufacturación también tiene beneficios. En X. Zhang et al. (2020) la resaltan como una técnica efectiva tanto para reducir las emisiones de carbono como para mejorar la utilización de la

energía y los desechos, en donde el consumo de energía en la remanufacturación puede ser reducido en un 85 %. Además de disminuir el consumo de energía y las emisiones de carbono, permite reducir el consumo de materia prima, siendo esto un punto vital cuando el producto tiene materias primas críticas cuya oferta es escasa (European Remanufacturing Network, [s.f.](#)).

La Responsabilidad Extendida del Productor (EPR, por sus siglas en inglés) fue definida en Lindhqvist ([2000](#)), y hace referencia a la gestión apropiada de los productos y los desechos por parte de los productores, quienes toman la responsabilidad del ciclo de vida completo del producto. Esto implica que los fabricantes deben tomar acción cuando el producto llega al final de su vida útil, incentivando la producción de bienes que sean más fáciles de reutilizar o desechar de manera segura, promoviendo la sustentabilidad ambiental y la reducción de residuos.

En este contexto, en el Problema de Dimensionamiento del Lote Económico con Remanufactura se determinan las cantidades a producir y remanufacturar. Está relacionado con la economía circular ya que además de la producción, incorpora la remanufacturación de retornos. En particular, la combinación de este problema con el EPR permite definir políticas eficientes en las que los fabricantes deben planificar cuándo y cuánto producir, remanufacturar y recolectar productos para garantizar el cumplimiento de sus responsabilidades ambientales.

La gestión eficaz del flujo de productos desde los consumidores de vuelta al productor o a los puntos de reciclaje o de disposición es esencial para promover la sustentabilidad y la economía circular. Este proceso es el de Logística Inversa (Guide y Luk Van Wassenhove, [2002](#)), el cual involucra la recolección, clasificación, transporte y procesamiento de productos usados con el objetivo de reducir el impacto ambiental y mejorar el uso de los recursos.

En este marco, se definen dos actores esenciales: los fabricantes de equipos originales (OEMs) y los remanufacturadores independientes (IRs). Los OEMs remanufacturan sus propios productos, mientras que los IRs remanufacturan productos sin ser necesariamente los fabricantes originales, alargando el ciclo de vida de los productos (S. Zhang et al., [2023](#)).

Capítulo 3

Revisión de la literatura

En este capítulo se presenta una revisión de la literatura para el problema abordado en este trabajo, destacando los principales estudios que sirvieron como base para el desarrollo del mismo. Se comienza con una introducción a los problemas de dimensionamiento del lote, estableciendo el marco teórico necesario. A continuación, se exploran estudios que incorporan remanufacturación como una estrategia para producir, subdividiendo estos trabajos en aquellos que consideran el transporte y los que incorporan objetivos de recuperación.

3.1. Problemas de Dimensionamiento del Lote Económico

El Problema de Dimensionamiento del Lote Económico (ELSP por sus siglas en inglés), se define como el problema de determinar las cantidades a producir para satisfacer la demanda a tiempo de un conjunto de períodos, minimizando la suma de los costos de producir y mantener en inventario, sin restricciones sobre los recursos (Karimi et al., 2003). En este problema, el horizonte temporal es finito y la demanda es dinámica, y se asume conocida en cada período. El objetivo es satisfacer la demanda de un único artículo minimizando la suma de los costos de setup, el costo de inventario por unidad (lineal y acreditado al final del período) y el costo unitario de manufacturación.

El ELSP puede modelarse como un problema de programación lineal entera mixta (MILP por sus siglas en inglés) de (3.1) a (3.5). Los conjuntos y parámetros del modelo son:

- T : conjunto de períodos.

- M : número grande.
- K_t : costo de setup en el período $t \in T$.
- c_t : costo unitario de producción en el período $t \in T$.
- h_t : costo de inventario por unidad en el período $t \in T$.
- D_t : demanda en el período $t \in T$.

Y las variables de decisión son:

- x_t : cantidad a producir en el período $t \in T$.
- I_t : inventario al final del período $t \in T$.
- y_t : variable binaria que indica si hay producción en el período $t \in T$, $y_t = 1$ si se produce, $y_t = 0$ en caso contrario.

$$\min \sum_{t \in T} (K_t y_t + c_t x_t + h_t I_t) \quad (3.1)$$

s.a:

$$I_{t-1} + x_t = D_t + I_t, \quad t \in T \quad (3.2)$$

$$x_t \leq M y_t, \quad t \in T \quad (3.3)$$

$$I_0 = 0 \quad (3.4)$$

$$x_t \geq 0, \quad I_t \geq 0, \quad y_t \in \{0, 1\}, \quad \forall t \in T \quad (3.5)$$

La función objetivo (3.1) es para minimizar la suma de los costos de setup y costos unitarios de producción y de inventario. Las ecuaciones (3.2) representan el balance de inventario, en donde se establece que el inventario del período anterior, sumado a la cantidad producida en el período actual, debe ser igual a la demanda del período actual más el inventario final del período. La familia de restricciones (3.3) es para la activación de la producción, garantizando que la producción en un período solo ocurra si se activa mediante la variable binaria y_t . En (3.4) se impone que el nivel de inventario inicial debe ser cero, y finalmente, la familia (3.5) define el dominio de las variables de decisión, estableciendo la no negatividad de las cantidades producidas e inventarios y que las variables de activación sean binarias.

Wagner y Whitin (1958) resolvieron el problema utilizando un algoritmo de programación dinámica con una complejidad computacional de $O(T^2)$. En su trabajo, los autores demostraron varios teoremas, incluyendo la propiedad

de inventario cero utilizada para desarrollar su algoritmo: existe una solución óptima en la que no puede haber simultáneamente producción en el período actual e inventario positivo del período anterior.

Varios autores trabajaron con el problema ELSP y desarrollaron algoritmos que mejoraron la eficiencia computacional en comparación con el algoritmo de Wagner y Whitin:

- Federgruen y Tzur (1991) diseñaron dos algoritmos. El primero, con complejidad $O(T \log T)$, se aplica al caso general con costos arbitrarios de producción e inventario. Mientras que el segundo, de complejidad $O(T)$, es aplicable cuando los costos de inventario no decrecen a lo largo del tiempo.
- Wagelmans et al. (1992) propusieron un algoritmo con complejidad $O(T \log T)$.
- Aggarwal y Park (1993) también desarrollaron un algoritmo con complejidad $O(T \log T)$.

Más recientemente, se han propuesto variantes y extensiones del problema ELSP que incorporan nuevas condiciones operativas y características dinámicas de la demanda:

- Pervin et al. (2018): consideraron demanda variable, deterioro de productos y condiciones de crédito. Integraron enfoques estocásticos con heurísticas para modelar la incertidumbre en la demanda y el deterioro de los productos.
- Önal y Albey (2020): estudiaron el ELSP con demanda dependiente al nivel de inventario, por lo que la demanda de productos se ve alterada según la cantidad de inventario. Propusieron un algoritmo basado en programación dinámica que resuelve el problema en tiempo polinomial.
- Forel y Grunow (2023): abordaron una extensión del ELSP bajo demanda incierta y evolución de pronósticos, aplicando una heurística de horizonte rodante que ajusta iterativamente las decisiones conforme se actualiza la información.
- Papadopoulos (2024): abordaron el ELSP con costos de producción definidos por tramos, pudiendo así modelar estructuras de costos más realistas. Propusieron un algoritmo exacto, resolviendo el problema en tiempo polinomial para el número de tramos.

Existen diversas variantes del problema ELSP, si bien se mencionarán algunos de ellos, no se profundizarán dado que no son el foco del presente trabajo. El Problema Dimensionamiento del Lote Capacitado (CLSP por sus siglas en inglés), para el cual Florian y Klein (1971) introdujeron un algoritmo de complejidad $O(T^4)$, comparte el mismo objetivo que el ELSP pero el CLSP introduce restricciones de capacidad en la producción. Otra variante es la de backlogging en donde se permite la posibilidad de satisfacer la demanda con retrasos, abordada inicialmente en Zangwill (1969). En el mismo trabajo, se estudia un problema multi nivel, en donde los productos están formados por componentes por lo que se producen en múltiples niveles o etapas. Finalmente, otra variante es la de múltiples productos, un acercamiento a este problema es el trabajo de Carreno (1990).

3.2. Problema de Dimensionamiento del Lote Económico con Remanufactura

El Problema de Dimensionamiento del Lote Económico con Remanufactura (ELSR por sus siglas en inglés) es una extensión del ELSP que incorpora la remanufactura de productos usados para satisfacer la demanda. El objetivo es determinar la cantidad y los períodos a manufacturar y remanufacturar, minimizando los costos asociados de setup, almacenamiento, manufactura y remanufactura. En la Figura 3.1 se muestra un sistema que permite tanto la manufactura como la remanufactura, distinguiéndose dos tipos de inventario:

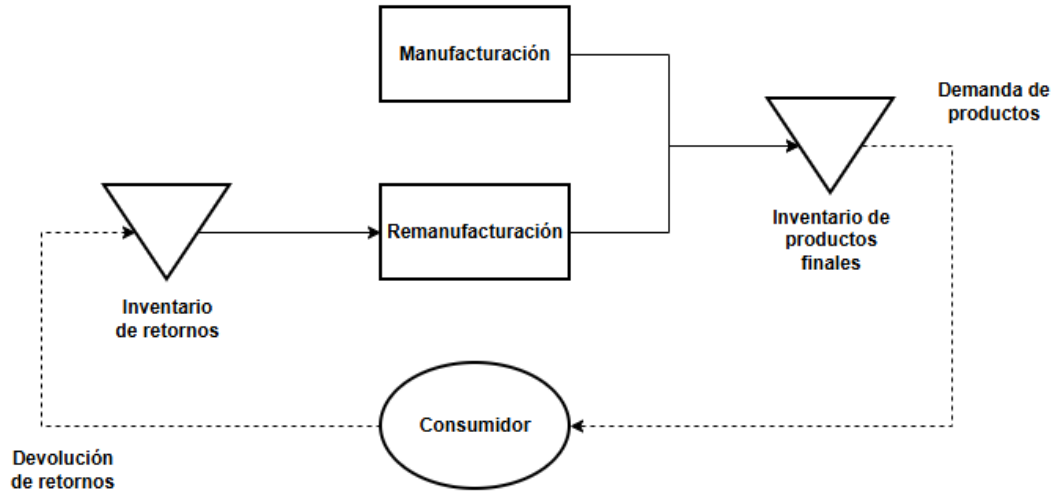


Figura 3.1: Sistema de inventario con producción y remanufactura, basado en el diagrama presentado en Teunter et al. (2006).

- Inventario de retornos: productos usados provenientes de los consumidores.
- Inventario de productos finales: productos nuevos o remanufacturados demandados por los consumidores.

Teunter et al. (2006) estudió el problema ELSR bajo dos esquemas: el de costos de setup unificados (ELSRj por ELSR with join setups) y el de costos de setup separados (ELSRs por ELSR with separate setups). En el ELSRj se utiliza una única línea para manufactura y remanufactura, de modo

que el costo de setup es el mismo para ambos procesos. En contraste, en ELSRs tiene dos líneas separadas y dedicadas a cada proceso, por lo que se asigna un costo de setup para la manufacturación y otro para la remanufacturación.

Ambos enfoques se basan en los siguientes supuestos comunes: la demanda y la cantidad de retornos son conocidas de antemano, el horizonte temporal es finito, y el tiempo de entrega para productos manufacturados y remanufacturados es cero. Además, se asume que los retornos tienen calidad homogénea, el inventario inicial es cero, la demanda en el período inicial es positiva, y no hay costo unitario de producción.

Teunter et al. (2006) modeló el ELSRj como un MILP y lo resolvió con un algoritmo de programación dinámica con complejidad $O(T^4)$, que generaliza el algoritmo propuesto por Wagner y Whitin (Wagner y Whitin, 1958) para el ELSP.

Para el ELSRs se modificó el MILP propuesto en ELSRj para reflejar las líneas separadas de manufacturación y remanufacturación. Se modela en (3.6) a (3.11), los conjuntos y parámetros son:

- T : conjunto de períodos.
- K^r : costo de setup de remanufacturación.
- K^m : costo de setup de manufacturación.
- h^s : costo de inventario de productos finales por unidad.
- h^r : costo de inventario de retornos por unidad.
- D_t : demanda en el período $t \in T$.
- R_t : cantidad de retornos en el período $t \in T$.
- M_t : suma de las demandas del período 1 al período $t \in T$.

Y las variables de decisión son:

- x_t^m : cantidad a manufacturar en el período $t \in T$.
- x_t^r : cantidad a remanufacturar en el período $t \in T$.
- I_t^s : inventario de productos finales al final del período $t \in T$.
- I_t^r : inventario de retornos al final del período $t \in T$.
- δ_t^m : variable binaria que indica si hay manufacturación en el período $t \in T$.
- δ_t^r : variable binaria que indica si hay remanufacturación en el período $t \in T$.

$$\min \sum_{t \in T} (K^m \delta_t^m + K_t^r \delta_t^r + h^r I_t^r + h^s I_t^s) \quad (3.6)$$

s.a :

$$I_{t-1}^r + R_t - x_t^r = I_t^r \quad \forall t \in T \quad (3.7)$$

$$I_{t-1}^s + x_t^r + x_t^m - D_t = I_t^s \quad \forall t \in T \quad (3.8)$$

$$x_t^m \leq M_t \delta_t^m \quad \forall t \in T \quad (3.9)$$

$$x_t^r \leq M_t \delta_t^r \quad \forall t \in T \quad (3.10)$$

$$\delta_t^m, \delta_t^r \in \{0, 1\} \quad \forall t \in T \quad (3.11)$$

$$x_t^r, x_t^m, I_t^r, I_t^s \geq 0 \quad \forall t \in T \quad (3.12)$$

La función objetivo (3.6) es para minimizar la suma de los costos de setup y los costos de almacenar productos finales y retornos sujeto a ciertas restricciones. Las restricciones definidas en (3.7) y (3.8) son las ecuaciones de balance de inventario para los retornos y productos finales, respectivamente. En (3.9) y (3.10) se determinan la activación del setup de manufacturación y remanufacturación respectivamente. Por último, en (3.11) se define el dominio de las variables de decisión binarias y en (3.12) las continuas.

Para resolver el ELSRs, Teunter et al. (2006) adaptaron tres heurísticas de la literatura de control de inventario: Silver Meal (SM), Costo Unitario Mínimo (LUC) y Balanceo de Período Fragmentado (PPB). Estas heurísticas se centran en la demanda inmediata y satisfacen la propiedad de inventario cero (Bañol y Restrepo, 2010). La heurística SM busca el costo promedio mínimo por período, LUC minimiza el costo por unidad de artículo ordenada, y PPB minimiza la diferencia entre el costo de configuración y el costo total de inventario.

Retel Helmrich et al. (2014) abordó el ELSR con costos dinámicos en vez de estacionarios, proponiendo variantes a la formulación de Teunter et al. (2006) a través de diversas técnicas:

- Formulación basada en caminos más cortos: adaptaron la formulación desarrollada por Eppen y Martin (1987) para el CLSP al ELSR. Modelaron el ELSR como un problema de encontrar el camino más corto en un grafo, logrando límites inferiores más precisos en la relajación de programación lineal.
- Formulación con desigualdades: incorporaron las

desigualdades (l, S, W, W) propuestas por Pochet y Wolsey (2006) para el CLSP, mejorando la calidad de la relajación de programación lineal y reduciendo el espacio de búsqueda de soluciones.

- Formulación con restricciones de validez general: En lugar de usar solo las restricciones clásicas del problema, se integraron restricciones adicionales que son siempre válidas para el modelo. Estas restricciones permiten reducir el espacio de búsqueda al excluir soluciones no factibles desde el principio, y al mismo tiempo mejoran el límite inferior (más cercano a la solución óptima). Dichas restricciones fueron adaptadas de las propuestas en Pochet y Wolsey (2006) para el CLSP. Estas restricciones adicionales se basan en inecuaciones válidas que son más generales que las restricciones estándar (como las restricciones de tipo (l, S, W, W) , que son específicas a ciertos Problemas de Dimensionamiento del Lote).

El ELSRs fue abordado por varios autores:

- Yang et al. (2005), Retel Helmrich et al. (2014) y Baki et al. (2014): demostraron que el ELSR es un problema NP-hard en general, incluso bajo funciones de costos restrictivas.
- Teunter et al. (2006), Piñeyro y Viera (2009), Schulz (2011), Piñeyro y Viera (2015) y Sifaleras et al. (2015): propusieron procedimientos heurísticos para resolver el problema.
- Pan et al. (2009): estudiaron el ELSR con restricciones de capacidad en la producción, remanufacturación y desechos. Propusieron algoritmos basados en programación dinámica sobre distintos escenarios de capacidad.
- Sifaleras y Konstantaras (2017): estudiaron el problema con múltiples artículos en logística inversa, y propusieron una heurística basada en Variable Neighborhood Descent (Mladenovic y Hansen, 1997) para mejorar la exploración de las vecindades.
- Macedo et al. (2016), Kilic et al. (2018) y Quezada et al. (2020): abordaron el problema incorporando incertidumbre desde diferentes perspectivas. En Macedo et al. (2016) consideraron la incertidumbre en la demanda y los retornos, junto con costos de *setup* aleatorios que reflejan su variabilidad; Kilic et al. (2018) centraron la incertidumbre en la demanda, proponiendo métodos que equilibran la calidad de la solución con la velocidad de cómputo; y Quezada et al. (2020) añadieron la posibilidad de ventas perdidas, desarrollando un modelo de programación

entera estocástica multi-etapa para coordinar la red bajo incertidumbre.

- Doh y Lee (2024): estudiaron el problema con operaciones de remanufactura, reprocesamiento y desensamblaje/reensamblaje para productos estructurados de varios niveles.

3.2.1. Consideraciones sobre el transporte

Una extensión del ELSR es incluir decisiones de transporte, en donde se debe coordinar no solo la producción y remanufacturación, sino también la entrega y recolección de productos. Zouadi et al. (2018) estudiaron un problema de producción con remanufacturación y selección de proveedores para retornos, considerando costos de transporte y restricciones de emisiones de carbono. Modelaron el problema como un MILP y propusieron dos métodos híbridos para encontrar una solución aproximada:

- Método HM1: Este método se basa en una estrategia en dos fases. Inicialmente, utiliza un algoritmo genético (Goldberg et al., 1992) para explorar el espacio de soluciones y definir eficientemente las decisiones clave (por ejemplo, la selección de proveedores o la activación de rutas de transporte). Una vez fijadas estas decisiones, se aplica una técnica de resolución exacta que optimiza las variables enteras, tales como los volúmenes de producción y remanufactura. De esta forma, se descompone el problema en etapas complementarias que facilitan la búsqueda de una solución globalmente satisfactoria.
- Método HM2: aunque sigue una estructura similar a la del HM1, en este enfoque se implementa una codificación más específica para las variables de decisión dentro del algoritmo genético. Esta mejora en la representación permite una exploración más precisa del espacio de soluciones, lo que conduce a una selección inicial más afinada. Como resultado, la fase de resolución exacta se beneficia de decisiones más cercanas a la óptima, mejorando así la calidad de la solución final devuelta.

Qiu et al. (2018) abordaron un problema de ruteo de producción con remanufacturación en logística inversa, que incluye recolección y entrega simultáneas, sin límites en la cantidad de vehículos y con igual capacidad, así como centros de producción y remanufacturación separados con capacidad limitada. Formularon el problema como un MILP y lo resolvieron utilizando

un algoritmo Branch and Cut (Padberg y Rinaldi, 1991). Primero, se utiliza una heurística constructiva para generar una solución inicial factible, que sirve como punto de partida y proporciona un límite superior inicial al costo total del sistema. Durante el proceso de branching, se aplican heurísticas para seleccionar las variables de decisión que dividen el problema en subproblemas, y se utilizan técnicas de búsqueda local para explorar los subproblemas prometedores antes de aplicar el método exacto.

En un enfoque híbrido, Parchami Afra y Behnamian (2021) analizaron problemas de ruteo y producción combinando también un método exacto con un heurístico. Consideraron múltiples tipos de productos con diferentes demandas, centros de producción y remanufacturación con capacidad infinita, y restricciones ambientales relacionadas con costos de emisiones de carbono. Formularon el problema como un MILP y lo resolvieron mediante una combinación de la Relajación Lagrangiana y un heurístico. La Relajación Lagrangiana relaja algunas restricciones del problema original, convirtiéndolas en penalizaciones con multiplicadores de Lagrange, y ajusta estos multiplicadores de manera iterativa para aproximarse a la solución óptima. La heurística, por su parte, ajusta las soluciones obtenidas por el método exacto para garantizar su factibilidad en el problema original.

Chekoubi et al. (2022) analizaron un problema de ruteo de producción considerando inventario y remanufacturación. Su objetivo era optimizar las decisiones de producción, gestión de inventarios y distribución en cadenas de suministro cerradas, incluyendo la recolección y remanufacturación de productos al final de su vida útil (EOL). Para ello, utilizaron una heurística de descomposición en dos fases, resolviendo iterativamente los siguientes problemas:

- Fase 1: Problema de tamaño de lote capacitado con remanufacturación.
- Fase 2: Problema de ruteo de vehículos con recogida y entrega simultánea, gestionando la distribución y recolección de productos EOL.

Recientemente, Ullah (2023) considera un sistema híbrido de producción y remanufacturación con decisiones de transporte y restricciones de emisiones de carbono. Su estudio asume una demanda constante y flujos de retorno, y desarrollan modelos matemáticos para determinar la tasa óptima de remanufacturación y el tiempo óptimo de ciclo de reposición para tres configuraciones

de bucles cerrados que incluyen al fabricante/remanufacturador, un recolector y múltiples minoristas:

- Configuración 1: Los productos usados son recolectados por los minoristas, transportados por una empresa externa de logística y enviados al fabricante/remanufacturador.
- Configuración 2: Los minoristas recolectan y envían directamente al fabricante/remanufacturador, eliminando la necesidad de una empresa externa.
- Configuración 3: La recolección es manejada tanto por los minoristas como por la empresa externa de logística, optimizando la tasa de recolección basada en las distancias de transporte.

Ullah (2023) concluye que, cuando las distancias de transporte de los productos usados hacia las instalaciones de manufacturación son largas, la cantidad óptima de productos a remanufacturar disminuye. Esto se debe a que es más rentable remanufacturar menos productos cuando los costos y emisiones asociados al transporte son altos. Por otro lado, la distancia entre el fabricante y el consumidor final no tiene un impacto significativo en la cantidad óptima de productos a remanufacturar. En resumen, considerar las distancias de transporte en la logística inversa es importante para la toma de decisiones en sistemas híbridos de producción y remanufacturación.

A diferencia de los trabajos analizados, en esta tesis se contempla al mismo tiempo los inventarios de los clientes y el del productor/remanufacturador.

3.2.2. Objetivos de recuperación

Los objetivos de recuperación son metas establecidas para la recolección y el aprovechamiento de productos al final de su vida útil (Karakayali et al., 2011). Estos pueden incluir tanto la recolección de los productos como su remanufacturación, permitiendo reincorporarlos al ciclo productivo. Su importancia radica en que contribuyen a reducir la cantidad de residuos generados y a disminuir el uso de materiales vírgenes, al aprovechar los recursos previamente utilizados.

Estas metas son fijadas por autoridades o legisladores, mientras que el fabricante original es responsable de cumplirlas. Un ejemplo de ello es la Directiva 2012/19/UE del Parlamento Europeo y del Consejo (Unión Europea,

2012), que establece una regulación del tipo Responsabilidad Extendida del Productor (EPR por sus siglas en inglés) introducida en Lindhqvist (2000). Esta normativa obliga a fabricantes y distribuidores de equipos electrónicos a gestionar la recolección y el tratamiento de los productos al final de su vida útil. Además, fija metas de recolección y recuperación progresivas: desde 2019, los Estados miembros deben recolectar al menos el 65 % del peso promedio de los equipos vendidos en los tres años anteriores o el 85 % de los residuos de aparatos eléctricos y electrónicos. Estas y otras medidas buscan aumentar la recuperación de materiales y reducir el impacto ambiental (Karakayali et al., 2011).

Kwak (2015) aborda un sistema de manufacturación con enfoque en los objetivos de recuperación, que derivan tanto de las demandas del mercado como de las exigencias legales. El problema se formula como un MILP bi-objetivo, con el propósito de maximizar el beneficio económico y minimizar el impacto ambiental. Se establece que debe recolectarse al menos un peso total de productos al final de su vida útil, de los cuales se debe recuperar una fracción específica. Para resolver este problema, Kwak (2015) emplea el método ϵ -constrained (Andersson, 2000). En este enfoque, uno de los objetivos se optimiza mientras que el otro se considera como una restricción con un lado derecho ϵ . Específicamente, el modelo maximiza el beneficio económico y establece el impacto ambiental como una restricción que debe ser al menos igual a ϵ . Variando el valor de ϵ , se obtienen diversas soluciones Pareto óptimas que muestran diferentes combinaciones de desempeño económico y ambiental, permitiendo a las empresas elegir planes de remanufacturación que mejor se adapten a sus objetivos económicos y de sostenibilidad.

Piñeyro y Viera (2018) estudian el Problema de Dimensionamiento del Lote Económico con Remanufacturación y Objetivos de Recuperación (ELSR-RT) por sus siglas en inglés, centrándose en un único producto y representando los objetivos de recuperación como un límite inferior en la cantidad de productos que deben ser remanufacturados, ya sea por exigencias legales o del mercado. Proponen un procedimiento heurístico que generaliza el problema tradicional sin objetivos de recuperación, el cual es NP-hard. La heurística sugiere criterios para determinar los períodos con remanufacturación positiva, resultando en dos variantes del método. Los resultados indican que la heurística es efectiva en términos de costo y tiempo en muchos escenarios, aunque su desempeño disminuye a medida que aumentan los objetivos de recuperación.

Esenduran et al. (2016) se enfocan en cómo los objetivos de recolección y remanufacturación impuestos por la legislación afectan a los productores. Modelan el problema considerando diferentes escenarios regulatorios y utilizan las condiciones de optimalidad de Karush-Kuhn-Tucker para resolver el problema de optimización bajo restricciones, obteniendo soluciones óptimas que dependen de las condiciones impuestas por la legislación, como niveles mínimos de remanufacturación y recolección. Analizan tres escenarios:

- Sin legislación: No se imponen obligaciones a los productores, lo que puede llevar a una menor tasa de recolección y, en consecuencia, a un mayor impacto ambiental debido a la acumulación de desechos.
- Legislación con objetivos de recolección: El productor es responsable de recolectar una fracción de los productos vendidos. Por ejemplo, una directiva de la Unión Europea exige que los estados miembros recojan el 45 % de los bienes electrónicos puestos en el mercado en 2016, aumentando al 65 % en 2019. Este escenario puede aumentar la remanufacturación y, en general, reducir el impacto ambiental.
- Legislación con objetivos de recolección y remanufacturación: Además de los objetivos de recolección, se impone un objetivo para la cantidad de remanufacturación. Este escenario puede incrementar significativamente la remanufacturación, aunque también puede aumentar los costos para los productores. Sin embargo, si se implementa de manera eficiente, puede maximizar la reutilización de productos, reduciendo los residuos y el impacto ambiental.

Un estudio similar al de Esenduran et al. (2016) es el de S. Zhang et al. (2023), que examina la cooperación entre fabricantes de productos nuevos (OEMs) y remanufacturadores (IRs). Por otro lado, S. Zhang et al. (2023) se centra en cómo los subsidios gubernamentales afectan la cooperación entre OEMs e IRs. Modelaron el problema bajo diferentes escenarios utilizando un enfoque de Teoría de Juegos y condiciones de optimalidad para determinar los equilibrios en los distintos escenarios:

- Sin subsidio gubernamental: Los OEMs y los IRs operan de manera independiente sin apoyo financiero del gobierno. En este escenario, los costos y riesgos asociados a la remanufacturación son elevados, lo que puede llevar a una menor cooperación técnica y una menor tasa de remanufacturación.

- Subsidio gubernamental moderado: Se proporcionan subsidios a un nivel moderado, incentivando la cooperación entre OEMs e IRs. Este escenario fomenta la creación de empresas conjuntas dedicadas a la innovación y mejora de procesos, donde los costos y riesgos compartidos promueven la innovación y la eficiencia en la remanufacturación. En general, este nivel de subsidio puede mejorar tanto el desempeño económico como la sostenibilidad ambiental al reducir desechos y promover la reutilización de recursos.
- Subsidio gubernamental alto: Se otorgan subsidios elevados, promoviendo una mayor cooperación entre los distintos actores. Este escenario maximiza la manufacturación al compensar los altos costos, aunque puede incrementar los costos para los productores. Sin embargo, probablemente reduce significativamente los desechos y el impacto ambiental, logrando un equilibrio entre consideraciones económicas y ambientales.

Es importante destacar que, durante el análisis de la literatura no se encontraron trabajos que integren de forma simultánea decisiones de producción, remanufacturación y transporte, junto con objetivos de recuperación. Este trabajo estudia este tipo de problema y propone un modelo MILP del mismo, y diseña procedimientos de resolución basados en la metaheurística Tabu Search para resolverlo, evaluando su performance en términos de costos y tiempo de ejecución contra un solver de optimización.

Capítulo 4

Definición y modelado del problema

El objetivo de este Capítulo es presentar la descripción detallada del problema de optimización que se aborda en la tesis, y el modelo matemático propuesto para el mismo.

Se considera un Problema de Dimensionamiento del Lote Económico que hace foco en la integración de las decisiones de manufacturación, remanufacturación y transporte; agregando además objetivos de recuperación. Se hace presente el concepto de Responsabilidad Extendida del Productor (EPR) en donde el productor no solamente tiene la responsabilidad de manufacturar y/o remanufacturar, sino que también debe gestionar la recolección de retornos. El problema estudiado se representa de forma esquemática en la Figura 4.1 en donde se tienen dos tipos de actores:

- Productor: es el responsable de manufacturar y/o remanufacturar productos, almacenar retornos y productos finales, transportar productos finales a los clientes y recolectar retornos.
- Conjunto de Clientes: cada cliente tiene su propia demanda y almacena retornos y productos finales. Puede representar a una única entidad o a un conjunto que se encarga de la distribución de los productos finales a partir de las entregas realizadas por el productor.

Con el fin de representar el concepto de EPR, el productor debe cumplir con objetivos de recuperación (recolección y remanufacturación) mediante tasas mínimas. Por un lado, la cantidad de retornos recolectados debe representar al menos una fracción $\beta \in [0, 1]$ del total de retornos disponibles, asegurando

un nivel mínimo de recolección. Por otro, al menos una fracción $\alpha \in [0, 1]$ de los retornos recolectados debe ser remanufacturada, garantizando así un mínimo de remanufacturación. Cabe destacar que, una vez recolectados, los retornos deben pasar por un proceso de preparación antes de ser utilizados en la remanufacturación. En Guide y Van Wassenhove (2001) se explica que esta preparación es necesaria para controlar su calidad, e incluye tareas como la limpieza y la clasificación.

El objetivo es determinar la cantidad a manufacturar, remanufacturar y transportar en cada período, desde y hacia los clientes de un único producto dentro de un horizonte temporal finito con el fin de satisfacer la demanda sin retrasos y los objetivos de recuperación, minimizando la suma total de los costos de manufacturación y remanufacturación, inventario de productos finales y retornos tanto en el productor como en los clientes, transporte de productos finales y recolección de retornos.

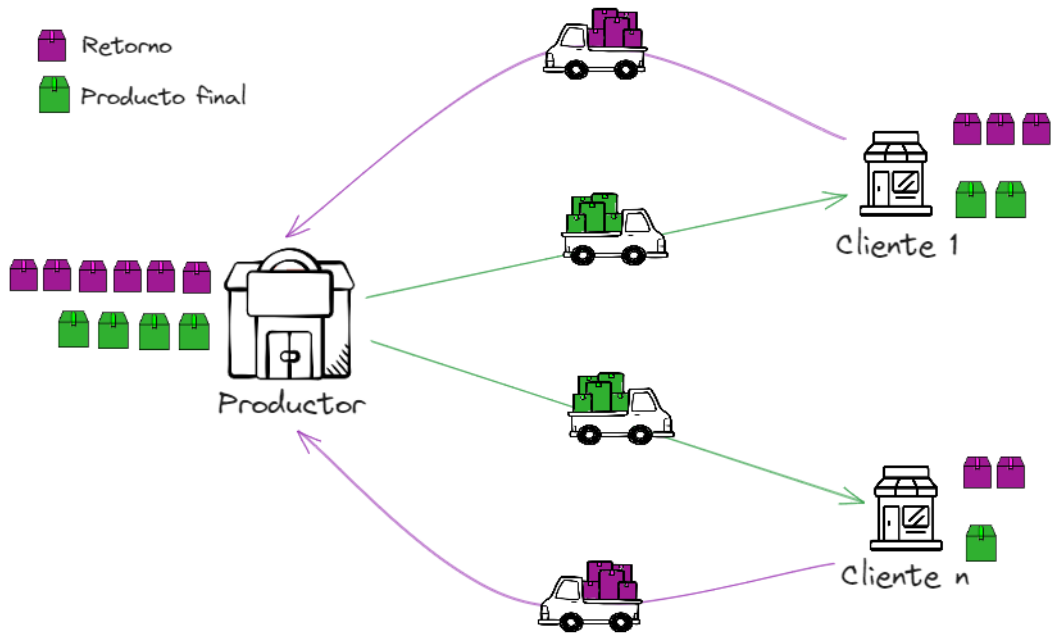


Figura 4.1: Esquema del Problema de Dimensionamiento del Lote Económico con Manufacturación, Remanufacturación y Transporte.

Se asume que todos los retornos son homogéneos, por lo que cualquiera de ellos puede ser remanufacturado, y la remanufactura puede realizarse en el período siguiente a su recolección. Además, se considera que el tiempo de viaje, tanto de los retornos como de los productos finales, es insignificante en comparación con la duración de un período.

Los valores de demanda y retornos se asumen conocidos y dinámicos a lo largo del horizonte de planificación. Los costos considerados son los siguientes:

- Costos de producción y remanufacturación: costos asociados al setup de los procesos y costos unitarios de producción o remanufacturación.
- Costos de transporte: costos de setup por realizar un viaje (independe de la cantidad de clientes a visitar), así como costos unitarios por el envío de productos finales y recolección de retornos.
- Costos de inventario: costos unitarios por almacenar productos finales y retornos, tanto en el productor como en los clientes.

Desde el punto de vista de la demanda, los productos nuevos y remanufacturados son equivalentes, por lo que pueden utilizarse indistintamente para satisfacer los requerimientos de los clientes. En cuanto al inventario inicial, se considera que el productor no tiene inventario de productos finales, pero sí puede tener una cantidad positiva de retornos, habilitando la posibilidad de remanufacturación en el primer período. En el caso de los clientes, se asume que no cuentan con inventario inicial, ni de productos finales ni de retornos.

En cuanto al orden de los eventos dentro de un mismo período, se establece que la remanufactura se lleva a cabo antes que la producción, y ambas ocurren antes que la entrega y la recolección. Los niveles de inventario se calculan al final, una vez que se han determinado las cantidades correspondientes a las actividades de producción y transporte.

4.1. Nomenclatura para el modelo

Con el fin de formular el modelo para el problema descrito en la sección anterior, se definen los siguientes conjuntos:

- T : conjunto de períodos, $T = \{1, \dots, nT\}$ en donde nT es la cantidad de períodos.
- L : conjunto de clientes, $L = \{1, \dots, nL\}$ en donde nL es la cantidad de clientes.

Además, se definen parámetros y variables de decisión que permiten modelar el problema. Los mismos están relacionados al productor y a los clientes según corresponda, por lo que están indexados de la siguiente manera:

- Supraíndices:
 - m : parámetros y variables de decisión relacionados al proceso de manufacturación llevado a cabo por el productor.
 - r : parámetros y variables de decisión relacionados al proceso de remanufacturación llevado a cabo por el productor.
 - s : parámetros y variables de decisión relacionados a productos finales.
 - u : parámetros y variables de decisión relacionados a retornos.
- Subíndices:
 - t : indica el período analizado, con $t \in T$.
 - ℓ : indica el cliente, con $\ell \in L$.

En las Tablas 4.1 y 4.2 se presentan los parámetros y las variables empleadas, las cuales siguen las reglas de indexación mencionadas con anterioridad. En la primera columna se realiza una agrupación según la naturaleza de los parámetros o variables de decisión, en la segunda la nomenclatura de la misma y, finalmente, en la tercera se brinda su descripción.

Tabla 4.1: Nomenclatura de los parámetros.

Tipo	Not.	Descripción
General	$D_{\ell,t}$	Demanda del cliente $\ell \in L$ en el período $t \in T$.
	M	Número grande.
	U_0	Cantidad de retornos del productor en el período inicial.
	$U_{\ell,t}$	Cantidad de retornos del cliente $\ell \in L$ en el período $t \in T$.
	α	Tasa mínima de remanufacturación expresada como una fracción del total de retornos recolectados. $\alpha \in [0, 1]$
	β	Tasa mínima de recolección expresada como una fracción del total de retornos disponibles. $\beta \in [0, 1]$
Producción	c_t^m	Costo de manufacturar una unidad en el período $t \in T$.
	K_t^m	Costo de setup de manufacturar en el período $t \in T$.
Remanu- facturación	c_t^r	Costo de remanufacturar una unidad en el período $t \in T$.
	K_t^r	Costo de setup de remanufacturar en el período $t \in T$.
Transporte	$K_{l,t}^v$	Costo de visitar en el período $t \in T$ al cliente $\ell \in L$.
	$cv_{l,t}^s$	Costo unitario de transportar productos finales en el período $t \in T$ hacia al cliente $\ell \in L$.
	$cv_{l,t}^u$	Costo unitario de transportar retornos en el período $t \in T$ desde el cliente $\ell \in L$.
Inventario	h_t^s	Costo de almacenar un producto final en el período $t \in T$.
	h_t^u	Costo de almacenar un retorno en el período $t \in T$.
	$h_{\ell,t}^s$	Costo de almacenar un producto final en el período $t \in T$ el cliente $\ell \in L$.
	$h_{\ell,t}^u$	Costo de almacenar un retorno en el período $t \in T$ en el cliente $\ell \in L$.

Tabla 4.2: Nomenclatura de las variables de decisión.

Tipo	Not.	Descripción
Producción	y_t^m	1 si en el período $t \in T$ se manufactura.
	x_t^m	Cantidad de productos manufacturados en el período $t \in T$.
	y_t^r	1 si en el período $t \in T$ se remanufactura.
	x_t^r	Cantidad de productos remanufacturados en el período $t \in T$.
Productos finales	$x_{\ell,t}^s$	Cantidad de productos finales entregados en el período $t \in T$ al cliente $\ell \in L$.
	I_t^s	Cantidad de productos finales a almacenar en el período $t \in T$.
	$I_{\ell,t}^s$	Cantidad de productos finales a almacenar en el período $t \in T$ en el cliente $\ell \in L$.
Retornos	$y_{\ell,t}^v$	1 si en el período $t \in T$ se viaja al cliente $\ell \in L$.
	$x_{\ell,t}^u$	Cantidad de retornos que se traen en el período $t \in T$ del cliente $\ell \in L$.
	I_t^u	Cantidad de retornos a almacenar en el período $t \in T$.
	$I_{\ell,t}^u$	Cantidad de retornos a almacenar en el período $t \in T$ en el cliente $\ell \in L$.

4.2. Formulación del modelo

En (4.1) a (4.16), se presenta el modelo para el problema descrito al inicio del Capítulo 4 como un problema de programación lineal entera mixta denominado a partir de ahora como Problema de Dimensionamiento del Lote Integrado de Producción, Remanufacturación y Transporte con Objetivos de Recuperación (PRTLSP-RT).

Con el fin de representar la responsabilidad extendida del productor se definieron dos objetivos de recuperación: remanufacturación y recolección. El objetivo de remanufacturación determina que la cantidad a remanufacturar debe ser una fracción de la cantidad de retornos que se traen de los clientes dentro del horizonte temporal, mientras que el objetivo de recolección establece que la cantidad de retornos a recolectar en cada cliente debe ser una fracción de la cantidad total de retornos a recolectar.

$$\min \tag{4.1}$$

$$\begin{aligned} & \sum_{t \in T} (K_t^m y_t^m + c_t^m x_t^m) \\ & + \sum_{t \in T} (K_t^r y_t^r + c_t^r x_t^r) \\ & + \sum_{t \in T} \left(h_t^u I_t^u + h_t^s I_t^s + \sum_{\ell \in L} (h_{\ell,t}^u I_{\ell,t}^u + h_{\ell,t}^s I_{\ell,t}^s) \right) \\ & + \sum_{t \in T} \sum_{\ell \in L} (K_{\ell,t}^v y_{\ell,t}^v + cv_{\ell}^s x_{\ell,t}^s + cv_{\ell}^u x_{\ell,t}^u) \end{aligned}$$

s.a:

$$I_t^s = I_{t-1}^s + x_t^m + x_t^r - \sum_{\ell \in L} x_{\ell,t}^s, \quad \forall t \in T \tag{4.2}$$

$$I_t^u = I_{t-1}^u + \sum_{\ell \in L} x_{\ell,t}^u - x_t^r, \quad \forall t \in T \tag{4.3}$$

$$I_{\ell,t}^s = I_{\ell,t-1}^s + x_{\ell,t}^s - D_{\ell,t}, \quad \forall \ell \in L, \forall t \in T \tag{4.4}$$

$$I_{\ell,t}^u = I_{\ell,t-1}^u + U_{\ell,t} - x_{\ell,t}^u, \quad \forall \ell \in L, \forall t \in T \tag{4.5}$$

$$x_t^m \leq M y_t^m, \quad \forall t \in T \tag{4.6}$$

$$x_t^r \leq M y_t^r, \quad \forall t \in T \tag{4.7}$$

$$x_{\ell,t}^u + x_{\ell,t}^s \leq M y_{\ell,t}^v, \quad \forall \ell \in L, \forall t \in T \tag{4.8}$$

$$x_t^r \leq I_{t-1}^u, \quad \forall t \in T \tag{4.9}$$

$$I_0^s = 0, \tag{4.10}$$

$$I_0^u = U_0, \tag{4.11}$$

$$I_{\ell,0}^s = I_{\ell,0}^u = 0, \quad \forall \ell \in L \tag{4.12}$$

$$\sum_{t \in \{1, \dots, nT-1\}} \sum_{\ell \in L} x_{\ell,t}^u \geq \beta \sum_{t \in \{1, \dots, nT-1\}} \sum_{\ell \in L} U_{\ell,t}, \tag{4.13}$$

$$\sum_{t \in T} x_t^r \geq \alpha \sum_{t \in \{1, \dots, nT-1\}} \sum_{\ell \in L} x_{\ell,t}^u, \tag{4.14}$$

$$y_t^m, y_t^r, y_{\ell,t}^v \in \{0, 1\}, \quad \forall \ell \in L, \forall t \in T \tag{4.15}$$

$$x_t^m, x_t^r, x_{\ell,t}^s, x_{\ell,t}^u, I_t^s, I_t^u, I_{\ell,t}^s, I_{\ell,t}^u \geq 0, \quad \forall \ell \in L, \forall t \in T \tag{4.16}$$

La función objetivo (4.1) busca minimizar la suma de los costos de manufacturación, que incluye tanto el costo de setup como el costo unitario de producir nuevos productos; el costo de remanufacturación, que es análogo al

de manufacturación pero aplicado al proceso de remanufacturación; el costo de inventario, que abarca el costo unitario de almacenar productos finales y retornos, tanto en las instalaciones del productor como en las del cliente; y finalmente, el costo de transporte, que considera tanto el costo de setup de realizar entregas a los clientes como el costo unitario de transportar productos finales y retornos entre el productor y los clientes.

En (4.2) se define el balance de productos finales en el productor: la cantidad almacenada al final del período $t - 1$ más la producción (nueva y remanufacturada) en el período t debe igualar la cantidad almacenada en inventario en t y la cantidad enviada a los clientes en t . De forma similar, las restricciones (4.3) son para los retornos en el productor, donde se equilibra la cantidad almacenada de retornos en $t - 1$ con los retornos recibidos de los clientes y lo remanufacturado en t .

En cuanto a los clientes, las ecuaciones (4.4) y (4.5) definen el balance de flujo de productos finales y retornos, respectivamente. Las restricciones (4.6), (4.7) y (4.8) aseguran que el setup de la manufacturación, remanufacturación y las entregas se activen en los momentos adecuados.

Las restricciones de (4.9) limitan la cantidad de remanufacturación al inventario disponible de retornos en el período anterior, ya que se asume que los retornos recolectados en un período están disponibles a partir del siguiente. Las restricciones (4.10) a (4.12) definen las condiciones iniciales del inventario para el productor y los clientes, siendo nulas salvo para los retornos en el productor, habilitando así la remanufacturación en el primer período.

Las restricciones (4.13) y (4.14) definen los objetivos mínimos de recolección y remanufactura de retornos a lo largo del horizonte temporal. La restricción (4.13) establece que la cantidad de retornos recolectados debe ser al menos β veces el total de los retornos disponibles. Análogamente, la restricción (4.14) asegura que la cantidad de retornos remanufacturados debe ser al menos α veces la cantidad de retornos recolectados. Ambas restricciones garantizan que se cumplan las tasas mínimas de recolección y remanufactura, las cuales están controladas por los parámetros α y β .

Finalmente, las restricciones (4.15) y (4.16) especifican el dominio de las variables de decisión.

Capítulo 5

Procedimiento de resolución

En este capítulo se describen los procedimientos de resolución propuestos para el PRTLSP-RT. Se comienza definiendo la representación utilizada de una solución factible y, posteriormente, se describe el procedimiento para resolver el problema junto con sus variantes.

El problema abordado puede considerarse una extensión del ELSR-RT introducido en Piñeyro y Viera (2018), el cual, a su vez, es una extensión del ELSR presentado en Teunter et al. (2006). En Baki et al. (2014) se demuestra que el problema ELSR es NP-Hard incluso bajo costos estacionarios, por lo que se puede inferir que el PRTLSP-RT tiene la misma complejidad.

Teniendo en cuenta la complejidad de resolución del problema, se decidió basar el procedimiento de resolución en la metaheurística Tabu Search ya que es una estrategia eficaz para resolver problemas de optimización combinatoria que se puede aplicar tanto en problemas de programación entera mixta como en otros (Glover et al., 2018). Además, se comprobó en la literatura que es una técnica eficiente para este tipo de problemas. El procedimiento desarrollado en este trabajo es una extensión del sugerido en Piñeyro y Viera (2009) para el problema ELSR con disposición final de retornos.

La metaheurística Tabu Search es un método de búsqueda que minimiza una función objetivo mediante iteraciones sucesivas en donde se enfoca en explorar y explotar soluciones cercanas a la solución actual mediante el concepto de vecindad de la solución actual, esto se repite hasta que se cumple la condición de parada (Glover et al., 2018). Utiliza una estructura denominada Lista Tabú en donde se almacenan las soluciones analizadas, para así no repetirlas y evitar caer en óptimos locales.

5.1. Representación de una solución factible

Los períodos de una solución factible del problema PRTLSP-RT se representan con el par $s = (v, r)$, donde:

- v : representa las visitas realizadas a los clientes, ya sea para entregarles productos finales así como para recolectar retornos. Es una matriz $v \in \{0, 1\}^{nL \times nT}$ con una fila para cada cliente y una columna para cada período, con $v[\ell, t] = 1$ si el cliente $\ell \in L$ es visitado en el período $t \in T$; $v[\ell, t] = 0$ en caso contrario.
- r : representa si se remanufactura en cada período. Es un vector $r \in \{0, 1\}^{nT}$ con una posición para cada período, con $r[t] = 1$ si en el período $t \in T$ se remanufactura; $r[t] = 0$ en caso contrario.

A partir del par s se determinan las cantidades a producir, remanufacturar, entregar y recolectar, así como los niveles de inventario de retornos y productos finales en cada período, para obtener una solución factible para el PRTLSP-RT.

5.2. Procedimiento de resolución base

En la Figura 5.1 se presenta el flujo del procedimiento de resolución base denominado **TSv1**, el cual recibe los datos de la instancia del problema y devuelve la mejor solución que pueda encontrar.

El procedimiento de resolución inicia con la *fase de inicialización* construyendo una solución factible. Se decidió visitar a todos los clientes en el primer y penúltimo período para asegurarse el cumplimiento de la demanda y de los objetivos de recolección. En el último período se remanufactura para que así todos los retornos recogidos sean remanufacturados cumpliendo con el objetivo de remanufacturación establecido.

Luego, se da paso a la *fase de búsqueda*, en donde se itera en busca de una mejor solución. En cada iteración, se identifica el vecindario de la solución actual $s = (v, r)$ intercambiando los 1s por 0s, y viceversa, de la matriz de visitas v y el vector de remanufacturación r , obteniendo así $nT \times nL$ tuplas vecinas $s' = (v', r')$. Para cada una de las tuplas s' , si no se encuentra en la lista tabú, se genera su solución mediante la fase de generación. Se genera una nueva iteración con la tupla vecina cuya solución es de menor costo, y si

es de menor costo que la mejor solución encontrada hasta el momento, se la reemplaza. El procedimiento finaliza cuando se alcanza el número máximo de iteraciones establecido como parámetro de entrada.

La lista tabú se diseñó como una cola FIFO (first in, first out) con un tamaño fijo definido como parámetro de entrada. Al alcanzar su capacidad máxima, se elimina el elemento más antiguo, lo que implica que una solución podría ser considerada nuevamente. Este enfoque presenta un trade-off, ya que una lista tabú muy pequeña puede llevar a la reconsideración frecuente de soluciones previamente analizadas, mientras que una lista demasiado grande puede resultar en un consumo elevado de recursos. Es importante entonces ajustar el tamaño de la lista tabú para equilibrar la exploración del espacio de soluciones y la eficiencia computacional.

La *fase de generación* recibe una tupla s y genera una solución factible del PRTLSP-RT a través de cuatro etapas en las que se determinan las cantidades a producir, remanufacturar, entregar y recolectar, para cada período y cliente. Además, se verifica que la solución generada sea factible, con respecto a los objetivos de recuperación tanto de recolección como de remanufacturación. Las reglas utilizadas en esta fase están basadas en las utilizadas por Piñeyro y Viera (2009) para el ELSR.

La primera etapa, en la fase de generación, es el plan de transporte que consiste en hallar cuántos productos finales entregar a los clientes y cuántos retornos recolectar en función de los períodos indicados en la matriz v . En (5.1) se define que la cantidad de productos finales a entregar a un cliente en un período j , es la suma de las demandas desde ese período hasta el anterior periodo a la última visita k ; en otras palabras en el período j se entrega una cantidad suficiente para satisfacer toda la demanda hasta la próxima visita. Y en (5.2) se define que la cantidad de retornos a recolectar en el período j , es la suma de los retornos generados desde el período posterior a la última visita i hasta el período actual, por lo que se recogen todos los retornos acumulados desde la visita anterior.

Sea $\ell \in L$, $0 \leq i < j < k \leq nT + 1$, con $v[\ell, i] = v[\ell, j] = v[\ell, k] = 1$

y $v[\ell, t] = 0 \quad \forall t \in \{i < t < j\} \cup \{j < t < k\}$:

$$x_{\ell, j}^s = D_{\ell, j} + \dots + D_{\ell, k-1} \quad (5.1)$$

$$x_{\ell, j}^u = U_{\ell, i+1} + \dots + U_{\ell, j} \quad (5.2)$$

A continuación, en el plan de remanufacturación se determina cuántos retornos remanufacturar en función del vector r . La cantidad a remanufacturar en el período j se define en (5.3) como el mínimo entre los retornos disponibles en el período anterior $j - 1$ y la demanda total a entregar para todos los clientes en el período j .

Sea $0 \leq i < j < k \leq nT$ / $r[i] = r[j] = r[k] = 1$:

$$x_j^r = \min \left\{ I_{j-1}^u, \sum_{\ell \in L} x_{\ell, j}^s \right\} \quad (5.3)$$

En la tercera etapa, se determina el plan de manufacturación mediante el algoritmo de Wagner y Whitin (1958), en donde se hallan las cantidades a manufacturar para cada período, para satisfacer la demanda restante, es decir la porción de la demanda que no pudo satisfacerse a través de la remanufacturación.

Finalmente, se verifica la factibilidad de la solución obtenida, comprobando que se cumplen las restricciones de la tasa de remanufacturación (4.13) y la de recolección (4.14).

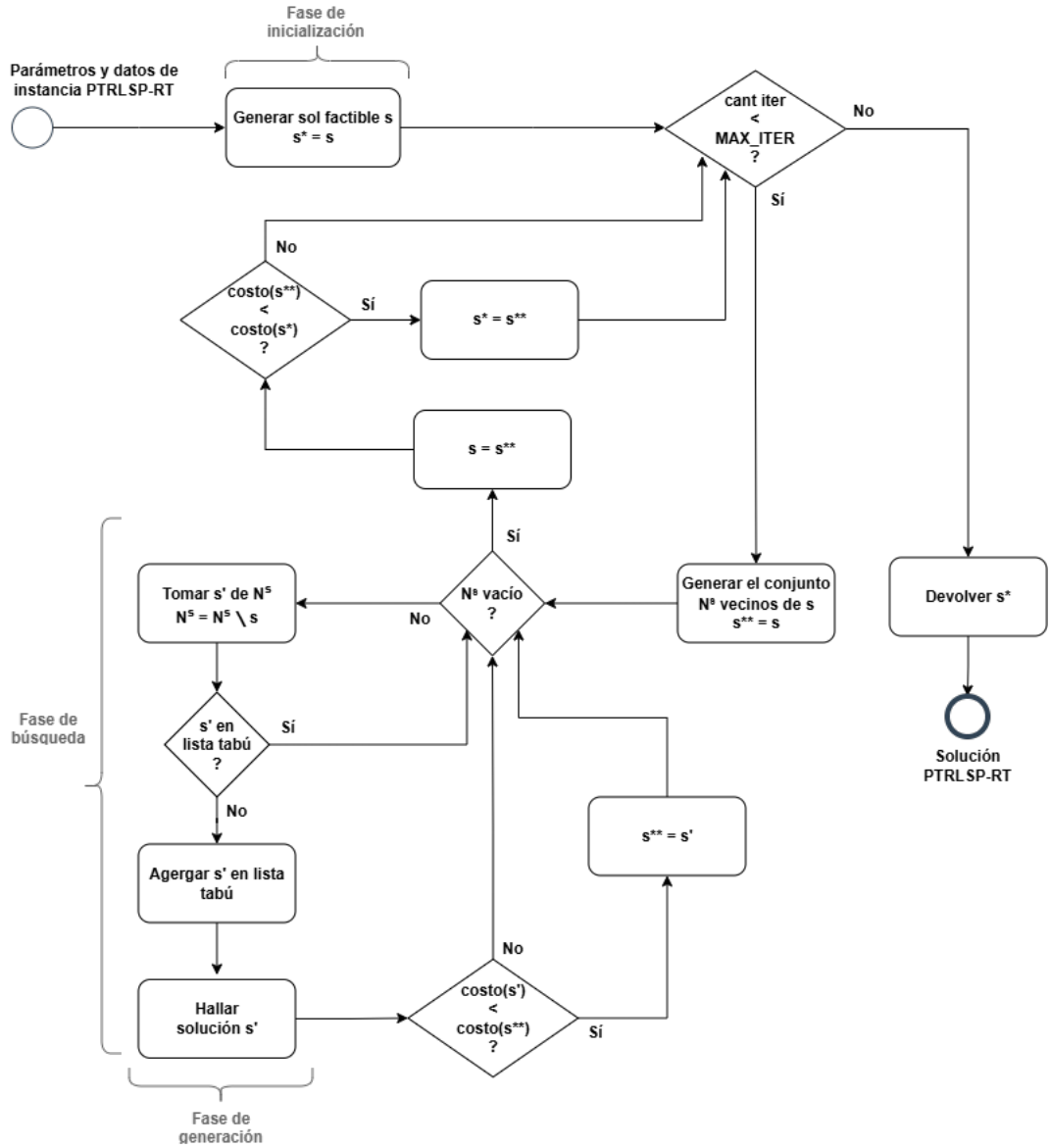


Figura 5.1: Flujo del procedimiento de resolución base TSv1.

5.3. Variantes del procedimiento de resolución base

Por cada nueva solución en la fase de exploración, se evalúan $nL \times nT^2$ soluciones en TSv1. Esto se debe a que, por cada cliente, se consideran todas las combinaciones posibles de períodos de recolección y entrega. Como resultado, la complejidad es cuadrática respecto al número de períodos (nT) y lineal

respecto al número de clientes (nL), lo que genera un gran número de soluciones a evaluar e incrementa significativamente el tiempo computacional.

La variante **TSv2** simplifica el procedimiento TSv1 reduciendo la cantidad de soluciones a explorar en cada paso de $nL \times nT^2$ a nT . Para cada cliente se fija el plan de transporte al inicio, determinando los períodos y las cantidades de recolección y entrega mediante el algoritmo de Wagner y Whitin (1958), el cual minimiza costos de entrega e inventario. Así, para cada cliente se resuelve un problema ELSP, donde los costos de producción corresponden a los costos de transporte del problema PRTLSP-RT, ignorando los costos de inventario de los retornos. La recolección de retornos sigue el esquema de TSv1 y, si no se cumple el objetivo de recolección, se añade un viaje final hacia todos los clientes. Entonces, la exploración en esta variante solo considera el vector de remanufacturación r , consiguiendo una reducción significativa en la cantidad de soluciones a explorar en el peor caso.

Con el objetivo de ampliar la búsqueda de soluciones, se desarrolló la variante **TSv3**, que sigue el mismo procedimiento que TSv1, pero con la solución inicial obtenida con TSv2.

La variante **TSv4** es una extensión de TSv2, en donde se modifica el plan de transporte para que no solamente tome en cuenta los costos de visita y el costo de inventario de productos finales en el cliente, sino que también considere el costo de inventario de los retornos.

Análogo a TSv3, **TSv5** es la combinación de TSv4 y TSv1. Por lo tanto, el procedimiento de resolución TSv1 toma como solución inicial la hallada por TSv4. Así como TSv3, esta variante tiene como objetivo explorar soluciones partiendo de una solución distinta que considera también el costo de inventario de los retornos.

5.4. Estrategia de post optimización

Con el fin de mejorar las soluciones, se definió una estrategia de post optimización denominada "saltos" la cual permite explorar una mayor cantidad de soluciones, así como soluciones con características más diversas.

Dada una solución $s = (v, r)$ del problema, se aplica una transformación al vector de remanufacturación r según dos estrategias: la estrategia par setea en 1 las posiciones pares del vector y en 0 las impares, al contrario la estrategia impar setea en 1 las posiciones impares y en 0 las pares. Se decidió realizar

la transformación únicamente sobre el vector r ya que de haberlo hecho también para la matriz v el costo computacional hubiese sido mayor dado sus dimensiones.

Este procedimiento se repite una cantidad de veces denominada "cantidad de saltos". Es necesario aclarar que, en un estudio preliminar, se obtuvieron mejores resultados utilizando estas estrategias versus las de intercambiar el valor de cada posición del vector r (1 por 0, y 0 por 1).

Capítulo 6

Experimentación numérica

En el presente capítulo se evalúa la performance de los procedimientos de resolución propuestos en el Capítulo 5 para el PRTLSP-RT contra la ejecución del solver CPLEX. Un procedimiento de resolución no tiene buena performance únicamente por conseguir una solución cercana a la óptima, sino que también es importante el tiempo de ejecución, por lo que para medir la eficiencia de los procedimientos de resolución se tomaron en cuenta ambas métricas. Para esto fueron generados un conjunto de instancias que fueron extendidas de problemas de ELSR agregando costos de transporte, multiplicidad de clientes y objetivos de recuperación.

Se decidió utilizar un conjunto de tres clientes para facilitar el análisis de los resultados y no generar instancias demasiados grandes, teniendo en cuenta la complejidad computacional del problema. Además, se puede considerar que cada cliente representa en la realidad un grupo de clientes lo suficientemente cercano uno del otro, por lo cual puede tener sentido visitarlos en forma simultánea. Para el resto de los parámetros se decidió extender el marco de instancias de Schulz (2011) para el ELSR con costos estacionarios y que han sido utilizadas en trabajos de la literatura, como por ejemplo Piñeyro y Viera (2015), Piñeyro y Viera (2018) y Sifaleras et al. (2015).

Se generaron dos tipos de conjuntos de datos: uno de instancias pequeñas con 12 períodos y tres clientes (denominado $nT12-nL3$), y uno de instancias grandes con 24 períodos y tres clientes (denominado $nT24-nL3$), ambos con clientes idénticos desde el punto de vista de los costos y del comportamiento de la demanda y los retornos. Se utilizaron distintos escenarios para los valores de la cantidad de retornos, costos de setup e inventario de retornos con el fin

de experimentar con diferentes situaciones. Los costos unitarios fueron establecidos en cero como una simplificación para el análisis. Según lo explicado en Teunter et al. (2006), en el caso de costos estacionarios, las decisiones sobre la cantidad a producir a largo plazo están principalmente influenciadas por los costos de setup y los de inventario. Sin embargo, cabe la aclaración que esta simplificación no aplica en escenarios donde se considera la remanufacturación, ya que Teunter et al. (2006) presentan un ejemplo en el que los costos unitarios sí afectan dichas decisiones.

En la Tabla 6.1 se presentan los valores utilizados para cada parámetro así como los escenarios si corresponde. A partir de estas configuraciones, se generaron $3^4 \times 2 = 162$ combinaciones únicas, y para cada una de ellas se crearon 10 instancias, totalizando así 1620 instancias. Se evaluaron estas instancias bajo tres escenarios de objetivos de recuperación:

- Escenario sin objetivos de recuperación ($\alpha = \beta = 0$): representa una situación en donde el marco regulatorio no exige ni recolección ni remanufacturación, por lo que no hay objetivos de recuperación.
- Escenario intermedio ($\alpha = \beta = 0.5$): representa una situación en donde se exige recolectar el 50 % de los retornos totales y remanufacturar el 50 % de los retornos recolectados.
- Escenario con recuperación total ($\alpha = \beta = 1$): representa una situación en donde se exige recolectar el 100 % de los retornos totales y remanufacturar todos ellos.

En consecuencia, se obtuvieron $1620 \times 3 = 4860$ instancias para cada conjunto de instancias pequeñas y grandes.

En cuanto a la paramétrica de los procedimientos de resolución, se definió el valor de la lista tabú en 100, dado que en experimentos preliminares con un valor mayor no se encontraron mejoras significativas en la calidad de las soluciones halladas, y con valores menores aumentaban los tiempos de cómputo dado que se volvían a considerar soluciones ya evaluadas sin obtener mejoras significativas. Por otro lado, para la cantidad de iteraciones de la fase de exploración del procedimiento, se utilizaron valores de 15, 25, 50 y 200, ya que con valores mayores no se encontraron tampoco mejoras significativas (más información en Apéndice 2).

Se evaluó el rendimiento de los procedimientos de resolución desarrollados utilizando dos métricas: el gap en el costo definido como la diferencia entre el

PARÁMETROS	ESCENARIOS	VALORES
Períodos	-	12
(nT)	-	24
Clientes	-	3
(nL)	-	3
Retornos	Bajo	N(30,6)
$(U_{\ell,t})$	Alto	N(70,14)
Setup	Bajo	200
(K_t^m, K_t^r)	Medio	500
	Alto	2000
Stock prod finales	-	1
(h_t^s)	-	1
Stock retornos	Bajo	0.2
(h_t^u)	Medio	0.5
	Alto	0.8

Tabla 6.1: Configuraciones para los conjuntos de datos.

costo de la solución encontrada y el costo de la solución del solver, expresado de acuerdo a (6.1); y el tiempo de ejecución en comparación con el tiempo del solver. Para medir el tiempo de ejecución de TSv3 (TSv5), no se considera el tiempo empleado en TSv2 (TSv4), ya que esta se asume como una solución previamente dada.

Dado que las variantes TSv3 y TSv5 mostraron ser las más efectivas, se llevaron a cabo evaluaciones exhaustivas de ambas, abarcando todos los valores predefinidos para la cantidad de iteraciones. Además, se implementó la estrategia de saltos para la cantidad de iteraciones 25 y 50 dado que fueron las que obtuvieron mejores resultados en las pruebas iniciales. Se optó por explorar únicamente 3 saltos, porque al probar con 5 no se observó una mejora significativa en el gap de costos, pero sí se evidenció un aumento considerable en el tiempo de ejecución.

$$\text{gap_costo} = \frac{\text{costo_procResol} - \text{costo_solver}}{\text{costo_solver}} \quad (6.1)$$

Las ejecuciones de los procedimientos de resolución se llevaron a cabo en una computadora personal con un procesador Intel (R) Core (TM) i5/9300H a 2.40 GHz y 16 GB de RAM, con el sistema operativo Windows Home 10 de 64 bits. La formulación del modelo (4.1) - (4.16) se codificó en AMPL y fue resuelto con CPLEX 20.1.0.0. Para el conjunto $nT12-nL3$ las ejecuciones se realizaron en el mismo entorno pero las instancias grandes ($nT24-nL3$) requirieron un ambiente más robusto, utilizando 16 CPU Intel Core i9-9900K a 3.60 GHz y 64 GB de RAM, con el sistema operativo Rocky Linux 8.4. Esto se debió a que, al tratarse de un problema con mayores dimensiones, demandó mayores recursos computacionales.

Para el reporte de los resultados se sigue la siguiente codificación: **versión-cant_iteraciones-estrategia_saltos**, donde **versión** indica la versión del procedimiento de resolución (TSv1, TSv2, TSv3 o TSv4), **cant_iteraciones** la cantidad de iteraciones y **estrategia_saltos** se emplea únicamente si se utiliza una estrategia de saltos, primero se indica el tipo de estrategia (O si es impar y E si es par) y luego la cantidad de saltos. Por ejemplo, TSv5-50-03 representa la versión TSv5 con 50 iteraciones, una estrategia de saltos impar con 3 saltos, mientras que TSv3-15 hace referencia a la versión TSv3 con 15 iteraciones sin estrategia de saltos.

6.1. Instancias pequeñas nL3-nT12

En la Tabla 6.2 se presentan los resultados para el conjunto de instancias $nT12-nL3$ para el escenario sin objetivos de recuperación ($\alpha = \beta = 0$), mostrándose en la primera columna el nombre del procedimiento de resolución, luego los valores promedios del gap de costo y tiempo de ejecución, los valores mínimos y los máximos. Para los valores de los procedimientos de resolución se destacaron en rojo los peores valores alcanzados y en verde los mejores. De forma análoga se presentan los resultados en las Tablas 6.3 y 6.4 para los escenarios intermedio ($\alpha = \beta = 0.5$) y con recuperación total ($\alpha = \beta = 1$), respectivamente. En la Figura 6.1 se resumen los resultados obtenidos para las mejores variantes en una gráfica.

Al analizar los resultados, se observa que en los escenarios sin objetivo

de recuperación e intermedio, los procedimientos de resolución alcanzaron al menos una vez la solución óptima. Sin embargo, cuando la recuperación es total, no fue siempre posible probablemente debido a que el problema es más restrictivo, aún así los gaps mínimos fueron inferiores al 1 %.

En todos los escenarios evaluados, los procedimientos de resolución proporcionaron soluciones en un tiempo de ejecución inferior al requerido por CPLEX, aunque CPLEX alcanzó la solución óptima en todos los casos. Además, en las familias de variantes de TSv3 y TSv5, se observaron gaps de costos inferiores al 3 %, destacándose especialmente las variantes que implementaron saltos en 50 iteraciones, consiguiendo alcanzar la solución óptima para algunas instancias.

Como era de esperar, las variantes TSv4-50 y TSv2-50 fueron las más rápidas en ejecutar, ya que el plan de entrega fijo reduce el espacio de búsqueda. Sin embargo, también fueron las que obtuvieron soluciones más alejadas del óptimo, dado que solo se exploraron las decisiones de producción y manufactura.

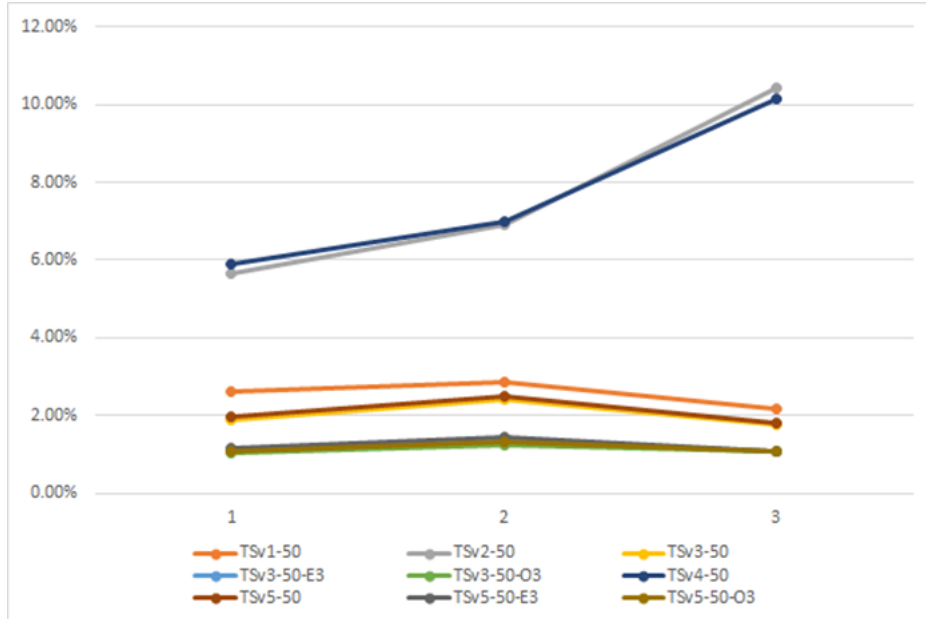


Figura 6.1: Performance de los procedimientos de resolución para instancias pequeñas. $\alpha = \beta = 0$ se representa con 1, $\alpha = \beta = 0.5$ con 2 y $\alpha = \beta = 1$ con 3.

	Promedio		Mínimo		Máximo	
	Gap costo (%)	Tiempo (ms)	Gap costo (%)	Tiempo (ms)	Gap costo (%)	Tiempo (ms)
CLPEX	-	2574.09	-	297	-	3578
TSv1-50	2.62	517.94	0	243	16.73	1065
TSv2-50	5.66	5.94	0	4	22.83	45
TSv3-15	2.60	121.47	0	70	18.85	383
TSv3-25	2.28	194.46	0	118	15.99	671
TSv3-25-E3	1.55	554.57	0	464	11.32	1072
TSv3-25-O3	1.47	460.70	0	448	11.32	642
TSv3-50	1.90	233.47	0	220	14.21	580
TSv3-50-E3	1.09	925.77	0	906	9.89	1172
TSv3-50-O3	1.04	1015.30	0	982	9.28	1265
TSv3-200	1.26	945.03	0	905	11.33	1314
TSv4-50	5.91	10.28	0	5	21.19	44
TSv5-15	2.62	69.34	0	65	15.99	158
TSv5-25	2.32	115.02	0	110	15.99	203
TSv5-25-E3	1.58	457.01	0	444	11.32	598
TSv5-25-O3	1.51	457.21	0	444	11.32	668
TSv5-50	1.95	233.44	0	220	14.21	1068
TSv5-50-E3	1.14	953.81	0	902	9.89	2055
TSv5-50-O3	1.08	928.87	0	906	9.28	1140
TSv5-200	1.30	925.39	0	903	11.33	1119

Tabla 6.2: Resultados de gap de costos y tiempo de ejecución de instancias chicas para $\alpha = \beta = 0$.

	Promedio		Mínimo		Máximo	
	Gap costo (%)	Tiempo (ms)	Gap costo (%)	Tiempo (ms)	Gap costo (%)	Tiempo (ms)
CPLEX	-	1321.67	-	234	-	11984
TSv1-50	2.87	488.51	0	232	17.74	1009
TSv2-50	6.90	5.85	0	4	22.83	27
TSv3-15	3.24	79.46	0	61	18.85	267
TSv3-25	2.88	199.38	0	110	15.99	288
TSv3-25-E3	1.88	537.82	0	461	12.28	1019
TSv3-25-O3	1.86	459.31	0	446	11.44	643
TSv3-50	2.43	227.54	0	220	14.65	356
TSv3-50-E3	1.37	954.20	0	907	10.82	1423
TSv3-50-O3	1.25	970.46	0	901	9.06	1912
TSv3-200	1.70	1088.94	0	899	11.47	3616
TSv4-50	7	5.94	0	4	25.45	28
TSv5-15	3.34	69.93	0	65	18.27	156
TSv5-25	2.96	114.93	0	109	15.99	291
TSv5-25-E3	1.93	461.36	0	448	11.47	676
TSv5-25-O3	1.88	460.75	0	448	11.44	658
TSv5-50	2.48	234.33	0	221	14.65	407
TSv5-50-E3	1.44	927.77	0	901	9.89	1177
TSv5-50-O3	1.31	936.12	0	904	11.84	3707
TSv5-200	1.63	942.51	0	906	11.84	1363

Tabla 6.3: Resultados de gap de costos y tiempo de ejecución de instancias chicas para $\alpha = \beta = 0.5$.

	Promedio		Mínimo		Máximo	
	Gap costo (%)	Tiempo (ms)	Gap costo (%)	Tiempo (ms)	Gap costo (%)	Tiempo (ms)
CPLEX	-	1216.79	-	125	-	11188
TSv1-50	2.15	223.23	0.01	195	13.42	556
TSv2-50	10.43	5.83	0.37	4	26.85	38
TSv3-15	2.46	71.42	0.01	60	17.92	270
TSv3-25	2.11	112.77	0.01	102	16.38	257
TSv3-25-E3	1.31	753.38	0.01	421	9.08	2152
TSv3-25-O3	1.38	417.17	0	404	7.66	573
TSv3-50	1.77	211.56	0.01	200	11.77	488
TSv3-50-E3	1.07	984.05	0	816	9.63	2594
TSv3-50-O3	1.06	836.84	0	815	8.89	2014
TSv3-200	1.27	859.48	0	821	9.08	1231
TSv4-50	10.14	5.80	0.37	4	26.69	37
TSv5-15	2.47	65.84	0.02	59	17.53	166
TSv5-25	2.15	106.55	0.02	99	12.45	329
TSv5-25-E3	1.35	420.54	0	407	9.64	661
TSv5-25-O3	1.43	426.06	0	405	8.67	772
TSv5-50	1.80	209.83	0.01	199	11.77	421
TSv5-50-E3	1.09	849.82	0	824	9.63	1048
TSv5-50-O3	1.08	846.97	0.01	821	8.89	1053
TSv5-200	1.30	1115.05	0.01	852	9.94	2640

Tabla 6.4: Resultados de gap de costos y tiempo de ejecución de instancias chicas para $\alpha = \beta = 1$.

6.2. Instancias grandes nL3-nT24

En esta sección se presentan y analizan los resultados obtenidos para el procedimiento de resolución propuesto considerando el conjunto de instancias grandes nL3-nT24. Debido a la complejidad de resolución de estas instancias, se estableció un límite de 900 segundos (15 minutos) para la ejecución de CPLEX. Es importante destacar que estas ejecuciones se llevaron a cabo en un entorno más robusto que las de los procedimientos de resolución. Se utilizó 16 CPU Intel Core i9-9900K a 3.60 GHz y 64 GB de RAM, con el sistema operativo Rocky Linux 8.4. Por lo tanto, se espera que los tiempos de ejecución de los procedimientos fueran considerablemente menores si se hubiesen realizado en el mismo entorno.

Para analizar la performance de los procedimientos de resolución se presentan tres tablas: Tabla 6.5 para objetivos de recuperación α y β de valor 0, Tabla 6.6 para 0.5 y Tabla 6.7 para 1. Los tiempos de ejecución reportados para las instancias grandes están expresados en segundos, a diferencia de los de las instancias pequeñas que se presentan en milisegundos. Al analizar las tablas se pueden observar valores negativos en los gaps de costos mínimos, esto se debe a que con la limitación de tiempo establecido para CPLEX no consiguió una solución óptima y el procedimiento de resolución logró encontrar una mejor solución. CPLEX no pudo encontrar la solución óptima para el 20.25 % de las instancias para tasas 0, 21.23 % para tasas 0.5 y 18.09 % para tasas 1.

Tal como se puede deducir de las tablas, en general el comportamiento de las variantes es similar al observado para las instancias pequeñas. En promedio, casi todas las variantes son capaces de obtener soluciones de muy buena calidad en aproximadamente 11 veces menos tiempo que CPLEX. Esto se mantiene incluso considerando que CPLEX fue ejecutado en un ambiente con más recursos. Además, algunas las variantes consiguen encontrar soluciones de menor costo que las de CPLEX.

Al igual que con las instancias pequeñas, los procedimientos de resolución TSv3 y TSv5 con 50 iteraciones y estrategia de saltos destacan como los más eficientes logrando soluciones con un gap de costo menor al 3 % en promedio, con un rango promedio del 2.16 % al 2.29 % considerando todos los escenarios de objetivos de recuperación. Además, nuevamente las variantes más eficientes en tiempo son TSv2 y TSv4, pero con estas instancias se puede observar que TSv4 supera de manera más significativa a TSv3 ya que para instancias grandes

es importante considerar el costo de inventario de los retornos.

	Promedio		Mínimo		Máximo	
	Gap costo (%)	Tiempo (s)	Gap costo (%)	Tiempo (s)	Gap costo (%)	Tiempo (s)
CPLEX	-	295.36	-	0.50	-	903.86
TSv1-50	4.38	5.70	0.00	4.97	20.21	9.07
TSv2-50	7.38	0.08	0.60	0.07	21.99	0.16
TSv3-15	4.01	1.83	0.00	1.77	16.60	2.08
TSv3-25	3.53	3.04	0.00	2.73	16.60	4.22
TSv3-25-E3	2.69	12.17	0.00	10.93	11.33	33.28
TSv3-25-O3	2.73	13.71	-0.06	10.99	12.70	43.98
TSv3-50	2.99	7.14	0.00	5.44	13.08	17.63
TSv3-50-E3	2.16	25.13	-0.03	21.78	9.70	68.26
TSv3-50-O3	2.18	23.88	-0.33	21.80	10.99	59.82
TSv3-200	2.21	24.60	0.00	21.86	10.42	67.78
TSv4-50	7.10	0.08	0.43	0.07	20.32	0.16
TSv5-15	4.04	1.85	0.00	1.78	14.30	5.30
TSv5-25	3.50	3.03	0.00	2.74	13.63	8.52
TSv5-25-E3	2.65	11.30	-0.15	11.04	10.92	12.27
TSv5-25-O3	2.71	12.85	0.00	11.02	12.70	41.80
TSv5-50	2.97	6.32	-0.18	5.44	12.61	15.86
TSv5-50-E3	2.18	22.66	-0.18	21.12	9.47	36.74
TSv5-50-O3	2.17	22.79	-0.18	21.31	9.62	28.23
TSv5-200	2.21	23.41	-0.18	21.86	10.26	54.16

Tabla 6.5: Resultados de gap de costos y tiempo de ejecución de instancias grandes para $\alpha = \beta = 0$.

	Promedio		Mínimo		Máximo	
	Gap costo (%)	Tiempo (s)	Gap costo (%)	Tiempo (s)	Gap costo (%)	Tiempo (s)
CPLEX	-	296.04	-	0.63	-	904.11
TSv1-50	4.56	5.90	0.19	5.51	28.54	9.74
TSv2-50	7.48	0.08	0.55	0.07	21.99	0.15
TSv3-15	4.14	1.78	0.00	1.62	16.60	4.78
TSv3-25	3.65	3.65	0.00	2.72	16.60	7.70
TSv3-25-E3	2.79	13.97	0.00	10.85	11.33	36.07
TSv3-25-O3	2.84	14.63	0.00	10.96	12.70	47.01
TSv3-50	3.09	9.35	0.00	5.55	13.08	22.67
TSv3-50-E3	2.24	25.95	0.00	22.10	9.70	63.51
TSv3-50-O3	2.27	24.82	0.00	22.04	10.99	79.26
TSv3-200	2.28	30.40	0.00	21.75	10.42	80.85
TSv4-50	7.20	0.09	0.43	0.07	20.32	0.48
TSv5-15	4.16	1.68	0.00	1.63	14.30	1.86
TSv5-25	3.62	3.35	0.00	2.73	13.63	10.26
TSv5-25-E3	2.74	11.35	0.00	10.98	10.92	22.45
TSv5-25-O3	2.81	11.26	0.00	11.01	12.70	22.57
TSv5-50	3.06	7.92	0.00	5.44	12.61	20.52
TSv5-50-E3	2.25	24.60	-0.04	20.27	9.47	75.60
TSv5-50-O3	2.25	25.51	0.00	22.15	9.62	87.87
TSv5-200	2.28	26.12	-0.08	22.02	10.26	73.02

Tabla 6.6: Resultados de gap de costos y tiempo de ejecución de instancias grandes para $\alpha = \beta = 0.5$.

	Promedio		Mínimo		Máximo	
	Gap costo (%)	Tiempo (s)	Gap costo (%)	Tiempo (s)	Gap costo (%)	Tiempo (s)
CPLEX	-	261.63	-	0.77	-	904.05
TSv1-50	4.06	5.52	0.10	5.34	28.49	8.43
TSv2-50	9.60	0.08	2.37	0.07	25.47	0.15
TSv3-15	4.08	1.77	0.13	1.57	14.52	3.53
TSv3-25	2.79	5.95	-0.09	2.91	10.41	10.43
TSv3-25-E3	2.64	11.37	-0.04	10.02	12.24	25.70
TSv3-25-O3	2.79	11.15	-0.09	10.53	10.41	15.87
TSv3-50	3.09	7.90	0.03	4.60	14.14	15.53
TSv3-50-E3	2.20	22.90	-0.20	20.82	10.68	51.78
TSv3-50-O3	2.29	23.34	-0.20	21.07	9.45	59.74
TSv3-200	2.32	32.23	-0.11	21.03	11.77	75.96
TSv4-50	9.10	0.08	1.00	0.07	22.30	0.22
TSv5-15	4.06	1.62	0.06	1.57	16.54	2.56
TSv5-25	3.57	3.40	-0.11	2.63	14.31	5.24
TSv5-25-E3	2.67	10.80	-0.15	10.52	10.27	11.83
TSv5-25-O3	2.75	10.71	-0.11	10.47	12.93	13.53
TSv5-50	3.04	6.47	-0.11	5.03	14.14	18.26
TSv5-50-E3	2.20	21.59	-0.15	21.11	9.92	25.30
TSv5-50-O3	2.26	22.84	-0.11	21.21	9.77	53.33
TSv5-200	2.29	24.04	-0.11	18.85	11.07	65.34

Tabla 6.7: Resultados de gap de costos y tiempo de ejecución de instancias grandes para $\alpha = \beta = 1$.

La Figura 6.2 resume el gap de costos promedio para las mejores variantes. Como era de esperar, el rendimiento disminuye en comparación con las instancias pequeñas debido a que la cantidad de soluciones aumenta exponencialmente. Aún así, se concluye que los procedimientos de resolución propuestos para el PRTLSP-RT son capaces de obtener soluciones cercanas a la óptima, e incluso óptimas o mejores que aquellas obtenidas con un solver de optimización del estado del arte, en un tiempo de cómputo mucho menor. De hecho, se puede observar en las Tablas 6.5, 6.6 y 6.7 que, en el caso de las instancias grandes consideradas, las variantes heurísticas nunca superaron los 88 segundos de cálculo como máximo, o 26 segundos en promedio. Por otro lado, los tiempos de ejecución de CPLEX fueron, en promedio, superiores a 260 segundos en un entorno computacional más potente, alcanzando el límite de tiempo establecido de 900 segundos sin determinar la optimalidad de la solución encontrada para algunas de las instancias (alrededor del 20 %).

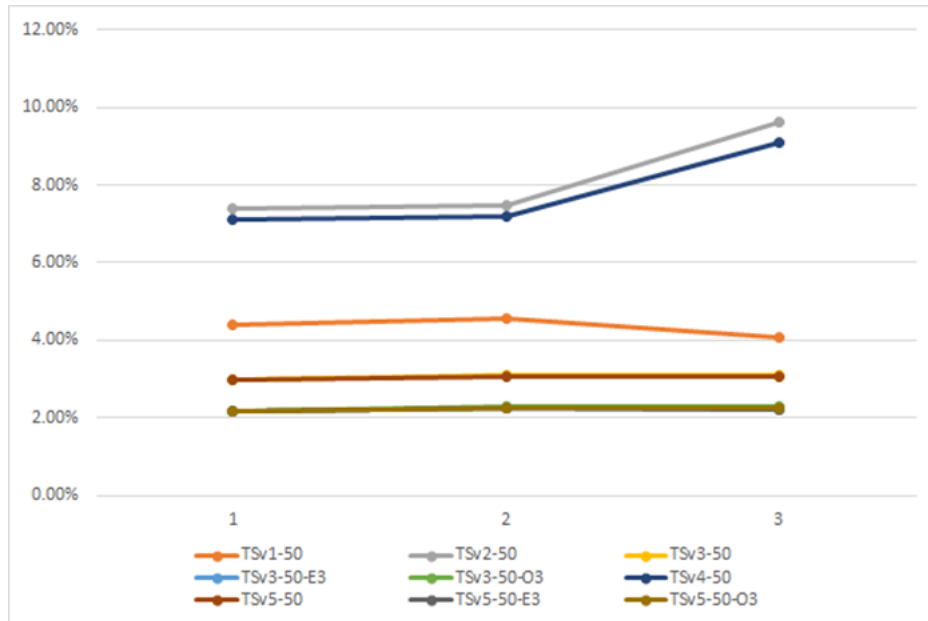


Figura 6.2: Performance de los procedimientos de resolución para instancias grandes. $\alpha = \beta = 0$ se representa con 1, $\alpha = \beta = 0.5$ con 2 y $\alpha = \beta = 1$ con 3.

Respecto a los tiempos de ejecución, se puede concluir de las Tablas 6.5, 6.6 y 6.7 que los procedimientos de resolución requieren de menos tiempo de ejecución para llegar a soluciones de buena calidad. De forma gráfica, en la Figura 6.3 se agrupan los tiempos de ejecución, medidos en segundo, que emplea CPLEX para encontrar la mejor solución factible para las instancias grandes.

Para el 60 % de las instancias consigue llegar a la mejor solución hallada en los primeros 100 segundos de ejecución, mientras que un 12 % consigue la solución en los últimos 100 segundos del tiempo propuesto (15 minutos), el resto se distribuyen de manera similar.

En la Figura 6.4 se comparan los tiempos de ejecución totales de CPLEX con los de la variante heurística TSv3-50-E3, la más eficiente para instancias grandes. Se observa que TSv3-50-E3 mantiene tiempos significativamente menores y, según el análisis de los datos, más estables entre instancias, incluso ejecutándose en un entorno con menor capacidad de cómputo. En contraste, los tiempos de CPLEX muestran una mayor dispersión, alcanzando en algunos casos el límite de 900 segundos.

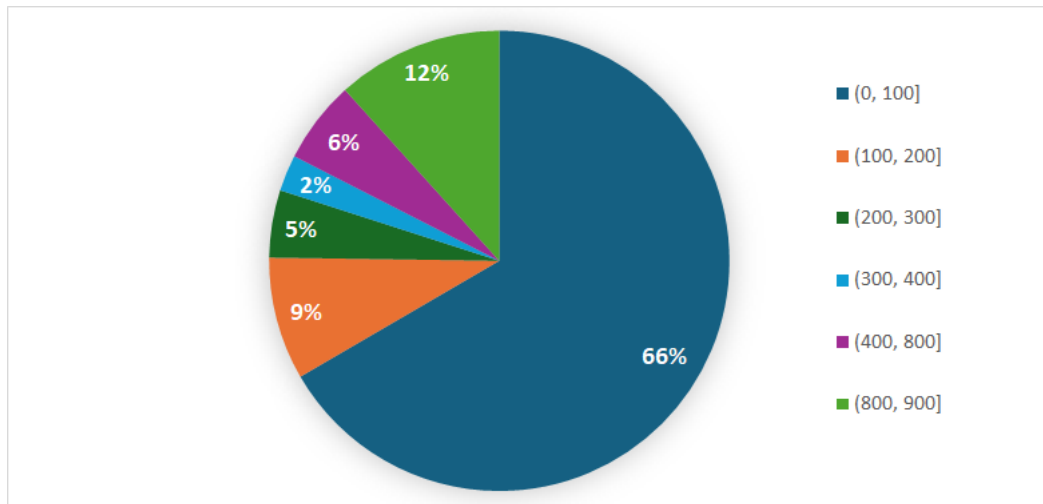


Figura 6.3: Distribución del tiempo en segundos requerido por CPLEX para llegar a la mejor solución factible que pudo encontrar para el juego de datos nT24-nL3.

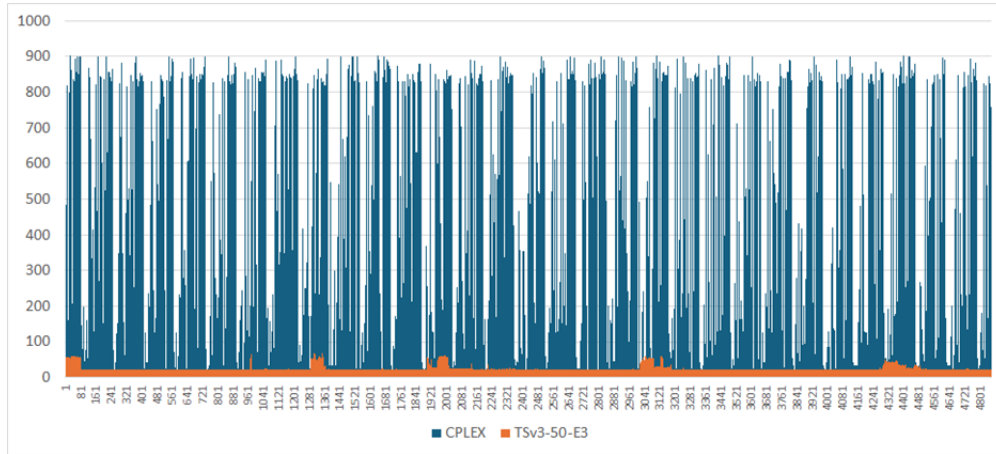


Figura 6.4: Comparación entre los tiempos de ejecución de TSv3-50-E3 contra CPLEX para el juego de datos nT24-nL3.

6.3. Interpretación de los resultados para la toma de decisiones

El análisis de los resultados revela consideraciones importantes para la toma de decisiones en este problema, al tiempo que sugiere posibles explicaciones para los comportamientos observados. En líneas generales, la integración simultánea de las decisiones de manufacturación, remanufacturación y transporte resulta crucial para obtener soluciones de calidad. Sin embargo, para TSv2 y TSv4 con instancias pequeñas se evidenció en las Tablas 6.2, 6.3 y 6.4 que la descomposición del problema en dos subproblemas (uno que agrupa las decisiones sobre las visitas a los clientes y otro que centraliza las decisiones de producción y remanufacturación) puede generar buenos resultados. Este comportamiento podría deberse a que, en escenarios de menor tamaño, la reducción de la dimensionalidad y la complejidad facilita la búsqueda de soluciones óptimas sin comprometer significativamente la calidad.

En relación con los objetivos de recuperación, se observa que el aumento en las tasas de recolección y remanufacturación tiende a incrementar la dificultad de resolución del problema. Aunque cuando estas tasas son completas ($\alpha = \beta = 1$), la dificultad se suaviza. Una posible explicación es que la imposición de tasas completas reduce significativamente el conjunto de soluciones factibles, simplificando la exploración del espacio de búsqueda. Este comportamiento se observa en mayor medida en los procedimientos de resolución, pero también se puede ver el patrón en las soluciones de CPLEX.

Para evaluar el efecto de los distintos parámetros en los gaps de costos, se analizó el conjunto de instancias grandes nT24-nL3, porque representan condiciones más complejas. El estudio se centró en la versión TSv3-50-E3, dado que fue la que obtuvo los mejores resultados, aunque en las demás versiones se observó un comportamiento similar. En la Tabla 6.8 se presentan los gaps de costos agrupados según parámetros como la cantidad de retornos por período y cliente ($U_{\ell,t}$), los costos de setup para manufacturar (K_t^m) y remanufacturar (K_t^r), el costo de visitar un cliente ($K_{\ell,t}^v$) por período, y el costo de almacenar retornos (h_t^u). Los resultados indican que los problemas con mayores cantidades de retornos disponibles son más desafiantes, lo cual puede atribuirse a una mayor interdependencia entre las decisiones y a un incremento en la complejidad del problema. Por otro lado, las instancias con altos costos de setup en remanufactura, transporte e inventario resultan más fáciles de resolver que

aquellas con elevados costos de manufacturación, sugiriendo que estos últimos tienen un impacto considerable en la dificultad del problema.

Los costos de setup para manufacturar parecen influir de manera significativa en el desempeño de los procedimientos de resolución, ya que se encuentran asociados tanto con los mejores como con los peores resultados (resaltados en verde y rojo). Este hallazgo es relevante, pues en la práctica dichos costos suelen ser superiores a los de remanufacturación, tal como se señala en Souza (2014). En consecuencia, cualquier variación en los costos de manufacturación puede amplificar las diferencias en la calidad de la solución, afectando de manera determinante la eficiencia de un procedimiento de resolución.

		$\alpha = \beta = 0$	$\alpha = \beta = .5$	$\alpha = \beta = 1$
Parámetros		Gap costo (%)	Gap costo (%)	Gap costo (%)
Todos		2.16	2.24	2.20
U_t^ℓ	N(70,14)	2.66	2.71	2.66
	N(30,6)	1.66	1.77	1.75
K_t^v	2000	1.68	1.73	1.77
	500	2.28	2.34	2.23
	200	2.53	2.65	2.62
K_t^m	2000	2.92	2.96	2.93
	500	2.04	2.12	2.03
	200	1.53	1.64	1.66
K_t^r	2000	1.98	2.23	1.99
	500	2.17	2.16	2.23
	200	2.34	2.34	2.40
h_t^u	2.01	2.00	2.23	1.93
	0.5	2.07	2.08	2.11
	0.2	2.41	2.64	2.58

Tabla 6.8: Resultados de gap de costo promedio para la variante TSv3-50-E3 para el juego de datos nT24-nL3.

Capítulo 7

Conclusiones

En el trabajo de tesis presentado en este informe se estudió un Problema de Dimensionamiento del Lote Económico que integra las decisiones de Manufacturación, Remanufacturación y Transporte, considerando Objetivos de Recuperación, al cual se lo denominó PRTLSP-RT por las siglas de su nombre en inglés. El problema consiste en determinar cuándo y cuánto producir y/o remanufacturar de un único producto, así como los períodos y cantidades a recolectar y entregar de retornos y productos finales desde y hacia un conjunto de clientes, con el objetivo de satisfacer la demanda a tiempo al menor costo posible, cumpliendo con objetivos de recolección y remanufacturación.

Se desarrolló un modelo matemático mediante una formulación MILP que extiende el problema ELSR incluyendo múltiples clientes, decisiones de transporte y objetivos de recuperación. Para solucionar el PRTLSP-RT se propuso un procedimiento de solución con variantes, basado en la metaheurística de Tabu Search, tomando como base el trabajo de Piñeyro y Viera (2009) para el ELSR con disposición final. Para la evaluación del procedimiento se extendió el marco de referencia de Schulz (2011) para considerar costos de transporte, múltiples clientes y objetivos de recuperación. Se utilizaron dos conjuntos de datos, uno de ellos con instancias pequeñas de 12 períodos y otro de instancias grandes con 24 períodos. En ambos casos se consideró una cantidad de tres clientes idénticos desde el punto de vista de los costos y para los valores de demanda y retornos.

Al evaluar la performance de los procedimientos de resolución en comparación con la del solver CPLEX, se comprobó que las variantes TSv3 y TSv5 son capaces de obtener soluciones en menor tiempo y de buena calidad. En algunas

de las instancias grandes, estas variantes incluso lograron soluciones de menor costo que las del solver CPLEX dentro del límite máximo de 15 minutos de ejecución. En particular, en los casos de instancias grandes, los procedimientos de resolución se destacaron por obtener soluciones en un tiempo de ejecución significativamente menor que el solver, a pesar de ejecutarse en un ambiente con mucho menos recursos.

A partir del análisis de interpretación de resultados, se concluye que para instancias pequeñas, se pueden realizar decisiones de producción y remanufacturación según un plan de transporte previamente determinado, pero ante instancias grandes es necesaria una adecuada coordinación de las decisiones de producción, remanufactura y transporte. En cuanto a los objetivos de recuperación, al aumentar las tasas de recolección y remanufacturación, el problema se vuelve más difícil de resolver, aunque si ambas tasas son 1 no es tan complejo probablemente por la reducción en la cantidad de soluciones factibles.

Respecto a los resultados numéricos obtenidos:

- En instancias pequeñas para las tasas de recuperación de 0 y 0.5 todas las variantes alcanzaron al menos una vez la solución óptima, o con diferencias menores al 1 %.
- En instancias grandes, las variantes TSv3 y TSv5 mostraron márgenes de error de costos inferiores al 3 % en la mayoría de los escenarios. Además, los tiempos de ejecución fueron significativamente más bajos que los del solver CPLEX, y en algunos casos se obtuvieron mejores resultados que con el solver.

En el presente trabajo se realizaron varias suposiciones: homogeneidad de los retornos recolectados, lo que implica que los retornos pueden ser remanufacturados de la misma forma; los retornos que llegan en un período están listos para usarse en el siguiente período; tiempo de viaje insignificante respecto a la duración de un período; entre otras. El trabajo se centró en analizar la performance de los procedimientos generados, para eso se crearon 162 configuraciones (variando los parámetros del problema), y para cada una se crearon 10 instancias, por lo que se generaron 1620 juegos de datos para instancias pequeñas y 1620 para instancias grandes.

Existen diferentes direcciones para futuros trabajos. Se podría relajar las suposiciones descritas para hacer más realista el trabajo, mejoras en los procedimientos de resolución y mejoras en la codificación de la solución.

En primera instancia, relajar las suposiciones sería interesante para hacer más realista el problema, permitiendo explorar instancias más complejas en las que los clientes tengan diferentes configuraciones de costo, demanda y retornos; por ejemplo se podría estudiar las causas del impacto de los objetivos de recuperación en las decisiones de producción, remanufacturación y transporte. Además, en el modelo actual no se incluyó la optimización en el ruteo de vehículos, por lo que se podría incorporar un plan de transporte más rentable para cada cliente, considerando también la carga de los vehículos. Asimismo, sería interesante agregar al problema la posibilidad de que los retornos no sean de calidad uniforme, lo que introduciría una característica más realista al problema.

En cuanto a mejoras en los procedimientos de resolución, se podría diseñar una estrategia más elaborada para explorar el espacio de soluciones del procedimiento TSv1. Actualmente se basa en múltiples ejecuciones con distintas soluciones iniciales. Como alternativa, podrían considerarse estrategias de búsqueda más profundas o híbridos con otros métodos heurísticos, que permitan una exploración más eficiente y estructurada del espacio de soluciones.

Por otro lado, sería interesante analizar en más detalle el estudio del subproblema de hallar un plan de transporte adecuado para cada cliente, como lo abordan TSv2 y TSv4, incluyendo también los objetivos de recuperación.

Finalmente, se podrían realizar mejoras en la forma en que se programaron los procedimientos de resolución aplicando técnicas de paralelización. En lugar de analizar las soluciones vecinas de manera secuencial, se podrían implementar métodos de análisis paralelo para reducir los tiempos de cómputo. Esto permitiría evaluar un mayor número de iteraciones y de soluciones, mejorando así la calidad de las soluciones encontradas.

Referencias

- Aggarwal, A., y Park, J. (1993). Improved Algorithms for Economic Lot Size Problems. *Operations Research*, 41, 549-571. <https://doi.org/10.1287/opre.41.3.549>
- Alamerew, Y., y Brissaud, D. (2019). Circular economy assessment tool for end of life product recovery strategies. *Journal of Remanufacturing*, 9, 169-185. <https://doi.org/10.1007/s13243-018-0064-8>
- Andersson, J. (2000). *A Survey of Multiobjective Optimization in Engineering Design* (Technical Report N.º LiTH-IKP-R-1097). Department of Mechanical Engineering, Linköping University.
- Baki, M., Chaouch, B., y Abdul-Kader, W. (2014). A heuristic solution procedure for the dynamic lot sizing problem with remanufacturing and product recovery. *Computers & Operations Research*, 43, 225-236. <https://doi.org/10.1016/j.cor.2013.10.001>
- Bañol, A., y Restrepo, J. (2010). Application of three methods for solution to problem of batch sizing and MRP. *Scientia Et Technica*, 16, 239-244. <https://www.redalyc.org/articulo.oa?id=84917249044>
- Carreno, J. J. (1990). Economic Lot Scheduling for Multiple Products on Parallel Identical Processors. *Management Science*, 36, 348-358. <http://www.jstor.org/stable/2631954>
- Chekoubi, Z., Trabelsi, W., Sauer, N., y Majdouline, I. (2022). The Integrated Production-Inventory-Routing Problem with Reverse Logistics and Remanufacturing: A Two-Phase Decomposition Heuristic. *Sustainability*, 14, 13563. <https://doi.org/10.3390/su142013563>
- Congreso de Diputados España. (2020). Enmiendas e índice de enmiendas al articulado: 121/000019 Proyecto de Ley de cambio climático y transición energética. *Boletín Oficial de las Cortes Generales*, 6-16. https://ecodes.org/images/que-hacemos/01.Cambio_Climatico/pdf/Propuestas_de_enmiendas_Ley_CC_y_TE.pdf

- Doh, H., y Lee, D. (2024). Dynamic lotsizing for remanufacturing systems with multilevel structured products and components commonality. *Applied Mathematical Modelling*, 125, 789-808. <https://doi.org/10.1016/j.apm.2023.09.020>
- Ellen Macarthur Foundation. (s.f.). What is a circular economy? [Último acceso: 20/05/2024]. <https://www.ellenmacarthurfoundation.org/circular-economy/concept>
- Eppen, G., y Martin, R. (1987). Solving Multi-Item Capacitated Lot-Sizing Problems Using Variable Redefinition. *Management Science*, 33, 1354-1369. <https://www.jstor.org/stable/171432>
- Esenduran, G., Subramanian, R., y Gupta, S. (2016). Take-Back Legislation: Consequences for Remanufacturing and Environment. *Decision Sciences*, 47, 219-256. <https://doi.org/10.1111/deci.12174>
- European Remanufacturing Network. (s.f.). What is Remanufacturing? [Último acceso: 24/05/2025]. <http://www.remanufacturing.eu/about-remanufacturing.php>
- Federgruen, A., y Tzur, M. (1991). A Simple Forward Algorithm to Solve General Dynamic Lot Sizing Model with n Periods in $O(n \ln n)$ or $O(n)$ Time. *Management Science*, 37, 909-925. <https://doi.org/10.1287/mnsc.37.8.909>
- Ferrer, E., Mouriz, Á., y Piñeyro, P. (2002). *Políticas de inventarios, mantenimiento y pronósticos* [Tesis de grado]. Facultad de Ingeniería UdelaR. <https://hdl.handle.net/20.500.12008/3048>
- Florian, M., y Klein, M. (1971). Deterministic Production Planning with Concave Costs and Capacity Constraints. *Management Science*, 18, 12-20. <https://doi.org/10.1287/mnsc.18.1.12>
- Forel, D., y Grunow, M. (2023). Dynamic Stochastic Lot Sizing with Forecast Evolution in Rolling-Horizon Planning. *Production and Operations Management*, 32, 449-468. <https://doi.org/10.1111/poms.13881>
- Glover, F., Laguna, M., y Martí, R. (2018). Principles and Strategies of Tabu Search. *Handbook of Approximation Algorithms and Metaheuristics, Second Edition*, 2, 20-25. <https://api.semanticscholar.org/CorpusID:155534779>
- Goldberg, D., Deb, K., y Clark, J. (1992). Genetic Algorithms, Noise, and the Sizing of Populations. *Complex System*, 6, 333-362. <https://content.wolfram.com/sites/13/2018/02/06-4-3.pdf>

- Greenpeace. (2016). Plásticos en los océanos: datos, comparativas e impactos [Último acceso: 25/05/2024]. http://archivo-es.greenpeace.org/espana/Global/espana/2016/report/plasticos/plasticos_en_los_oceanos_LR.pdf
- Guide, D., y Luk Van Wassenhove. (2002). The Reverse Supply Chain [Último acceso: 25/05/2024]. <https://hbr.org/2002/02/the-reverse-supply-chain>
- Guide, D., y Van Wassenhove, L. (2001). Managing Product Returns for Remanufacturing. *Production and Operations Management*, 10, 142-155. <https://doi.org/10.1111/j.1937-5956.2001.tb00075.x>
- Gungor, A., y Gupta, S. (1999). Issues in environmentally conscious manufacturing and product recovery: a survey. *Computers & Industrial Engineering*, 36, 811-853. [https://doi.org/10.1016/S0360-8352\(99\)00167-9](https://doi.org/10.1016/S0360-8352(99)00167-9)
- Karakayali, I., Boyacı, T., Verter, V., y Van Wassenhove, L. (2011). On the Incorporation of Remanufacturing in Recovery Targets. *Documento de trabajo*. https://www.researchgate.net/publication/266181939_On_the_Incorporation_of_Remanufacturing_in_Recovery_Targets
- Karimi, B., Fatemi Ghomi, S., y Wilson, J. (2003). The capacitated lot sizing problem: a review of models and algorithms. *Omega*, 31, 365-378. [https://doi.org/10.1016/S0305-0483\(03\)00059-8](https://doi.org/10.1016/S0305-0483(03)00059-8)
- Kilic, O., Tunc, H., y Tarim, S. (2018). Heuristic policies for the stochastic economic lot sizing problem with remanufacturing under service level constraints. *European Journal of Operational Research*, 267, 1102-1109. <https://doi.org/10.1016/j.ejor.2017.12.041>
- Kwak, M. (2015). Planning Demand- and Legislation-Driven Remanufacturing for a Product Family: A Model for Maximizing Economic and Environmental Potential. *Industrial Engineering & Management Systems*, 14, 159-174. <https://doi.org/10.7232/IEMS.2015.14.2.159>
- Lindhqvist, T. (2000). *Extended Producer Responsibility in Cleaner Production* [Tesis doctoral]. Lund University. <https://lucris.lub.lu.se/ws/portalfiles/portal/4433708/1002025.pdf>
- Macedo, P., Alem, D., Santos, M., Junior, M., y Moreno, A. (2016). Hybrid manufacturing and remanufacturing lot-sizing problem with stochastic demand, return, and setup costs. *The International Journal of Advanced Manufacturing Technology*, 82, 1241-1257. <https://doi.org/10.1007/s00170-015-7445-z>

- Mladenovic, N., y Hansen, P. (1997). Variable neighborhood search. *Computer & Operations research*, 24, 1097-1100. [https://doi.org/10.1016/S0305-0548\(97\)00031-2](https://doi.org/10.1016/S0305-0548(97)00031-2)
- Önal, M., y Albey, E. (2020). Economic lot sizing problem with inventory dependent demand. *Optimization Letters*, 14, 2087-2106. <https://doi.org/10.1111/poms.13881>
- Padberg, M., y Rinaldi, G. (1991). A Branch-and-Cut Algorithm for the Resolution of Large-Scale Symmetric Traveling Salesman Problems. *SIAM Review*, 33, 60-100. <https://doi.org/10.1137/1033004>
- Pan, Z., Tang, J., y Liu, O. (2009). Capacitated dynamic lot sizing problems in closed-loop supply chain. *European Journal of Operational Research*, 198, 810-821. <https://doi.org/10.1016/j.ejor.2008.10.018>
- Papadopoulos, K. (2024). A Novel Exact Algorithm for Economic Lot-Sizing with Piecewise Linear Production Costs [Último acceso: 15/07/2024]. <https://arxiv.org/abs/2403.16314>
- Parchami Afra, A., y Behnamian, J. (2021). Lagrangian heuristic algorithm for green multi-product production routing problem with reverse logistics and remanufacturing. *Journal of Manufacturing Systems*, 58, 33-43. <https://doi.org/10.1016/j.jmsy.2020.11.013>
- Pervin, M., Roy, S. K., y Weber, G. W. (2018). An integrated inventory model with variable holding cost under two levels of trade-credit policy. *Numerical Algebra, Control and Optimization*, 8, 169-191. <https://doi.org/10.3934/naco.2018010>
- Piñeyro, P., y Viera, O. (2009). Inventory policies for the economic lot-sizing problem with remanufacturing and final disposal options. *Journal of Industrial and Management Optimization*, 5, 217-238. <https://doi.org/10.3934/jimo.2009.5.217>
- Piñeyro, P., y Viera, O. (2015). The economic lot-sizing problem with remanufacturing: analysis and an improved algorithm. *International Journal of Production Research*, 53, 1-13. <https://doi.org/10.1186/s13243-015-0021-8>
- Piñeyro, P., y Viera, O. (2018). Heuristic procedure for the economic lot-sizing problem with remanufacturing and recovery targets. *International Journal of Production Research*, 56, 39-50. <https://doi.org/10.1007/s13243-018-0044-z>

- Pochet, Y., y Wolsey, L. (2006). *Production Planning by Mixed Integer Programming*. Springer. <https://doi.org/10.1007/0-387-33477-7>
- Qiao, A., Chobi, S., y Pan, Y. (2021). Multi-party coordination in sustainable supply chain under consumer green awareness. *Science of the Total Environment*, 777, 146043. <https://doi.org/10.1016/j.scitotenv.2021.146043>
- Qiu, Y., Nia, M., Wanga, L., Lia, Q., Fanga, X., y Pardalos, P. (2018). Production routing problems with reverse logistics and remanufacturing. *Transportation Research Part E*, 87-100. <https://doi.org/10.1016/j.tre.2018.01.009>
- Quezada, F., Gicquel, C., Kedad-Sidhoum, S., y Vu, D. (2020). A multi-stage stochastic integer programming approach for a multi-echelon lot-sizing problem with returns and lost sales. *Computers & Operations Research*, 116, 104865. <https://doi.org/10.1016/j.cor.2019.104865>
- Retel Helmrich, M., van den Heuvel, W., de Kok, T., y Jans, R. (2014). Economic lot-sizing with remanufacturing: complexity and efficient formulations. *IIE Transactions*, 46, 67-86. <https://doi.org/10.1080/0740817X.2013.802842>
- Sahajwalla, V., y Gaikwad, V. (2018). The present and future of e-waste plastics recycling. *Current Opinion in Green and Sustainable Chemistry*, 13, 102-107. <https://doi.org/10.1016/j.cogsc.2018.06.006>
- Sandin, G., y Peters, G. (2020). Environmental impact of textile reuse and recycling – A review. *Journal of Cleaner Production*, 184, 353-365. <https://doi.org/10.1016/j.jclepro.2018.02.266>
- Schulz, T. (2011). A new Silver–Meal based heuristic for the single-item dynamic lot sizing problem with returns and remanufacturing. *International Journal of Production Research*, 49, 2519-2533. <https://doi.org/10.1080/00207543.2010.532916>
- Sifaleras, A., y Konstantaras, I. (2017). Variable neighborhood descent heuristic for solving reverse logistics multiitem dynamic lotsizing problems. *Computers & Operations Research*, 78, 385-392. <https://doi.org/10.1016/j.cor.2015.10.004>
- Sifaleras, A., Konstantaras, I., y Mladenović, N. (2015). Variable neighborhood search for the economic lot sizing problem with product returns and recovery. *International Journal of Production Economics*, 160, 133-143. <https://doi.org/10.1016/j.ijpe.2014.10.003>

- Souza, G. (2014). Closed-Loop Supply Chains with Remanufacturing. *INFORMS Tutorials in Operations Research*, 130-153. <https://doi.org/10.1287/educ.1080.0040>
- Teunter, R., Bayindir, Z., y Heuvel, W. V. D. (2006). Dynamic lot sizing with product returns and remanufacturing. *International Journal of Production Research*, 44, 4377-4400. <https://doi.org/10.1080/00207540600693564>
- Ullah, M. (2023). Impact of transportation and carbon emissions on reverse channel selection in closed-loop supply chain management. *Journal of Cleaner Production*, 394, 136370. <https://doi.org/10.1016/j.jclepro.2023.136370>
- Unión Europea. (2012). Directiva 2012/19/UE del Parlamento Europeo y del Consejo, de 4 de julio de 2012, sobre residuos de aparatos eléctricos y electrónicos (RAEE) [Último acceso: 03/09/2025]. <https://eur-lex.europa.eu/legal-content/ES/TXT/?uri=celex%3A32012L0019>
- United Nations. (2015). The Paris Agreement [Último acceso: 22/10/2024]. <https://unfccc.int/process-and-meetings/the-paris-agreement>
- Wagelmans, A., Hoesel, S., y Kolen, A. (1992). Economic Lot Sizing: An $O(n \log n)$ Algorithm That Runs in Linear Time in the Wagner-Whitin Case. *Maastricht University, Open Access publications from Maastricht University*, 40, 909-925. <https://doi.org/https://www.jstor.org/stable/3840844>
- Wagner, H., y Whitin, T. (1958). Dynamic Version of the Economic Lot Size Model. *Management Science*, 5, 89-96. <https://doi.org/10.1287/mnsc.5.1.89>
- Yang, J., Golany, B., y Yu, G. (2005). A concave-cost production planning problem with remanufacturing options. *Nav Res Logist*, 52, 443-458. <https://doi.org/10.1002/nav.20089>
- Zangwill, W. (1969). A Backlogging Model and a Multi-Echelon Model of a Dynamic Economic Lot Size Production System-A Network Approach. *Management Science*, 15, 506-527. <http://www.jstor.org/stable/2628386>
- Zhang, S., Liu, W., Hou, P., y Wang, X. (2023). Impact of government subsidy on the remanufacturing industry regarding technology cooperation. *Industrial & Management Optimization*, 19, 4152-4179. <https://doi.org/10.3934/jimo.2022125>

- Zhang, X., Zhang, M., Zang, H., Jiang, Z., Liu, C., y Cai, W. (2020). A review on energy, environment and economic assessment in remanufacturing based on life cycle assessment method. *Journal of Cleaner Production*, 255, 120160. <https://doi.org/10.1016/j.jclepro.2020.120160>
- Zouadi, T., Yalaoui, A., y Reghioui, M. (2018). Hybrid manufacturing / remanufacturing lot-sizing and supplier selection with returns, under carbon emission constraint. *IFAC-PapersOnLine*, 56, 1233-1248. <https://doi.org/10.1080/00207543.2017.1412524>

APÉNDICES

Apéndice 1

Implementación

En este Apéndice se detalla la implementación técnica necesaria para llevar a cabo la solución del problema planteado. En la Figura 1.1 se representa la solución implementada, la misma se compone de:

- Generador de archivos DAT: proyecto desarrollado en Python que genera archivos con extensión DAT que contienen los datos de la realidad. Estos archivos son los de entrada para los procedimientos de resolución y el solver.
- Resolución del solver: contiene el código en AMPL para resolver un problema mediante el solver AMPL/CPLEX. Dado un archivo DAT, devuelve un archivo OUT con la solución encontrada.
- ELSREPR: proyecto desarrollado con Java que contiene la lógica de los procedimientos de resolución. Dado un archivo DAT, resuelve el problema de forma heurística según la versión seleccionada y devuelve un archivo OUT con la mejor solución encontrada.
- PANEL-ELSR-EPR: proyecto desarrollado en React JS que contiene la parte visual de la solución. Permite invocar al proyecto ELSREPR para la ejecución del procedimiento de resolución, y visualizar los resultados.
- Generador de reportes: proyecto desarrollado en Python que permite generar reportes a partir de los archivos OUT de soluciones, para así evaluar el desempeño de los procedimientos de resolución implementadas contra el solver AMPL/CPLEX.

Los componentes core de la implementación son ELSREPR y el PANEL-ELSR-ERP, dado que mediante la comunicación entre ambos se pueden ejecutar los procedimientos de resolución y obtener los resultados de las mismas,

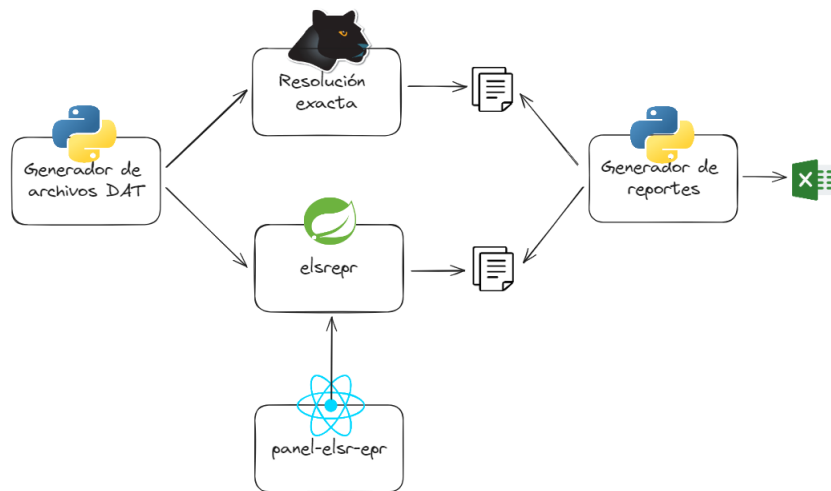


Figura 1.1: Arquitectura en alto nivel de la solución.

por lo que en esta sección se hará foco en ellos.

En la Figura 1.2 se presenta la comunicación entre ambos proyectos: EL-REPR actuando como Backend y PANEL-ELSR-EPR como Frontend. Se comunican mediante APIs Rest en donde ELSREPR es quien expone endpoints y PANEL-ELRS-EPR los invoca y muestra los resultados a través de una interfaz web.

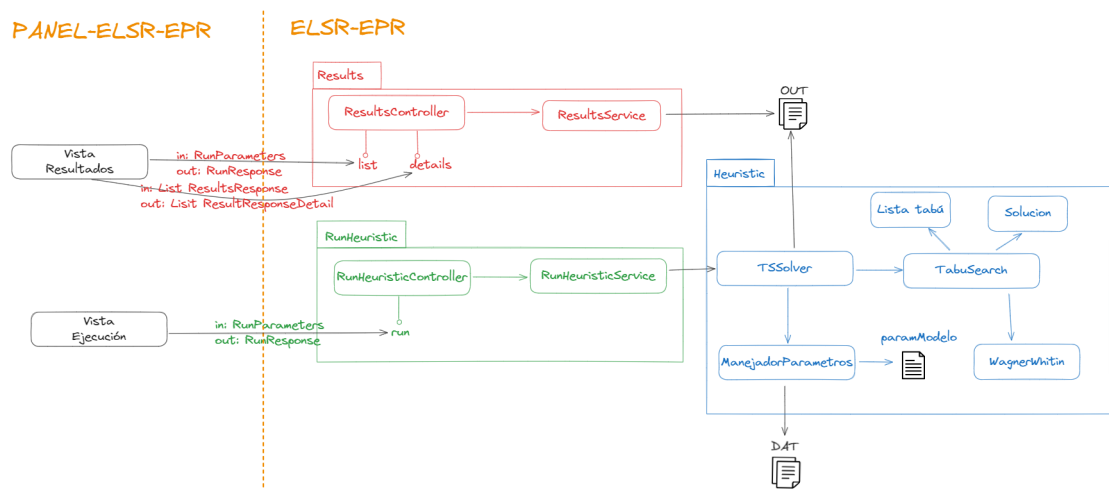


Figura 1.2: Arquitectura en alto nivel de la solución.

El proyecto sigue el patrón MVC (Model-View-Controller):

- Backend:
 - Model: Los paquetes Results, RunHeuristic, Data, y Heuristic están

involucrados en la representación y manipulación de datos, así como en la lógica de negocio.

- **Controller:** Los controladores en los paquetes `Results` y `RunHeuristic` gestionan las solicitudes HTTP relacionadas con los resultados y la ejecución del procedimiento de resolución. Estos controladores interactúan con los servicios que contienen la lógica de los procedimientos de resolución.
- **Frontend:**
 - **View:** La interfaz gráfica es la capa de vista, que presenta los datos y proporciona la interacción del usuario. Esta capa interactúa con los endpoints proporcionados por la aplicación del servidor para obtener y enviar datos.

Proyecto ELSREPR

El proyecto **ELSREPR** (repositorio git: <https://github.com/lupish/elsrepr>) actúa de Backend dado que tiene la lógica de los procedimientos de resolución. Fue desarrollado en Java con el framework Spring, al utilizar Java se consiguió una plataforma robusta y buena performance, y a través de Spring se consiguió un diseño estructurado y mantenible. Como se representa en la Figura 1.2 está diseñado en paquetes para separar las responsabilidades de los componentes:

- **Results:**
 - **Controller (ResultsController):** Este paquete contiene la lógica del controlador encargado de gestionar las solicitudes relacionadas con los resultados de los procedimientos de resolución. Aquí se definen los puntos de entrada HTTP y se manejan las interacciones con el servicio.
 - **Service (ResultsService):** Contiene la lógica de negocio relacionada con los resultados de los procedimientos de resolución. Aquí se aplican los procedimientos de resolución y se gestionan los resultados antes de ser devueltos al controlador.
 - **DTOs (ResultResponseDetail, ResultsParameters, ResultsResponse):** Definen objetos de transferencia de datos (DTO) utilizados para la comunicación entre el controlador y el servicio, y viceversa.

- RunHeuristic:
 - Controller (RunHeuristicController): Encargado de gestionar las solicitudes relacionadas con la ejecución de los procedimientos de resolución desde la interfaz gráfica. Define los endpoints necesarios para iniciar el proceso de ejecución de los procedimientos de resolución.
 - Service (RunHeuristicService): Contiene la lógica de negocio para ejecutar los procedimientos de resolución. Aquí se orquesta la ejecución de los procedimientos de resolución y se coordina con otras capas de la aplicación según sea necesario.
 - DTOs (RunParameters, RunResponse): Definen objetos de transferencia de datos para la entrada y salida de la ejecución de los procedimientos de resolución. Específicamente para los parámetros de entrada de los procedimientos de resolución y el mensaje de salida luego de la ejecución.
- Data:
 - ManejadorParametros: Una clase que se encarga de manejar los parámetros de entrada para los procedimientos de resolución. Aquí se puede definir la validación y procesamiento de los datos de entrada.
 - paramModelo: archivo de parámetros de las heurísticas por defecto, serán reemplazados por los valores de DTO RunParameters en caso de existir.
- Heuristic: Este paquete contiene la lógica principal de los procedimientos de resolución, con clases separadas según sus responsabilidades específicas. Podrían incluirse clases para la generación de soluciones, evaluación de calidad, entre otras.
 - Parametros: DTO que contiene la paramétrica del procedimiento de resolución.
 - Solucion: DTO que representa la estructura de la solución del procedimiento de resolución.
 - TabuSearch: contiene la lógica de cada corrida del Tabu Search, es invocada por TSSolver.
 - TSExcepcion: define las posibles excepciones al ejecutar el procedimiento de resolución.

- TSSolver: contiene la lógica de las heurísticas. Se encarga de controlar las ejecuciones y el control de parada del procedimiento de resolución.
- WagnerWhitin: lógica del algoritmo de Wagner y Whitin.
- ElsreprApplication: La clase principal que inicia la aplicación Spring Boot. Aquí se configuran los componentes principales y se arranca el servidor web.

Con el fin de reducir la verbosidad del código y mejorar la legibilidad, se utilizó la librería Lombok y Sprint, entre otras:

- Anotaciones de Lombok:
 - Data, NoArgsConstructor, AllArgsConstructor, RequiredArgsConstructor : combinada con otras anotaciones, simplifica la creación de clases generando automáticamente métodos comunes como lo son constructores, getters y setters.
- Anotaciones de Spring:
 - SpringBootApplication: marca la clase principal.
 - RestController: marca la clase como controlador en una arquitectura REST, por lo que indica que los métodos manejan solicitudes HTTP y que devuelven respuestas del tipo JSON.
 - RequestMapping: asigna rutas a los métodos de los controladores.
 - Service: marcar una clase como componente de servicio, por lo que indica que contiene la lógica de las heurísticas.
- Otras:
 - Override: marca un método que va a sobrescribir un método de una superclase.
 - CrossOrigin: configura el CORS (política de intercambio de recursos entre orígenes) para permitir solicitudes desde otros dominios.

En la tabla [1.1](#) se presentan los endpoints expuestos, en donde primero se detalla el nombre del endpoint, luego el método HTTP y finalmente una descripción.

Endpoint	Método	Descripción
/run	POST	Recibe un objeto de tipo RunParameters como entrada y devuelve un objeto de tipo RunResponse. La funcionalidad principal consiste en ejecutar un procedimiento de resolución utilizando el paquete Heuristic. El procedimiento de resolución ejecutada es la versión definida en el objeto de entrada, junto con otros parámetros relevantes. Al finalizar la ejecución, la solución generada se guarda en un archivo de salida (OUT).
/list	POST	Recibe un objeto de tipo ResultParameters, indicando qué tipos de resultados se desean listar (por ejemplo, la versión del procedimiento de resolución). La respuesta consiste en una lista de objetos ResultResponse que contienen información detallada sobre los archivos que coinciden con los datos de entrada. La información incluye el nombre del archivo, la fecha de creación, la versión y la tasa de recuperación.
/details	POST	Recibe un objeto de tipo ResultParameters que especifica el archivo de entrada. La respuesta incluye datos detallados sobre el costo y tiempo de ejecución del resultado correspondiente al archivo especificado. También es capaz de recibir una carpeta como entrada, devolviendo una lista que contiene información sobre el costo y tiempo de ejecución de cada uno de los resultados en dicha carpeta.

Tabla 1.1: Descripción de los endpoints del proyecto.

Proyecto PANEL-ELSR-EPR

Por otro lado, el proyecto **PANEL-ELSR-EPR** (repositorio git: <https://github.com/lupish/panel-elsr-epr>) es una aplicación web desarrollada utilizando React con NextJS 13 como framework principal y Tailwind CSS para el diseño. El proyecto es el frontend de la solución, e interactúa con los endpoints proporcionados por el proyecto ELSREPR.

La vista de resultados se presenta en la Figura 1.3 la cual es accedida a través del header y seleccionando Resultados. Contiene distintos filtros que permiten al usuario seleccionar qué tipos de resultados desea ver: carpeta, nombre de archivo específico, configuración (grupo de archivos de entrada), cantidad de períodos y cantidad de clientes. Al presionar el botón buscar se invocará al endpoint /list del proyecto ELSREPR, recibiendo así los archivos

que macheen con los filtros, y estos se mostrarán debajo de los filtros mostrando la fecha de creación, el nombre del archivo, la versión de la huerística, la tasa y una lupa que le permitirá acceder a más información.

Resultados de la heurística

Carpeta: C:\proy_io\Codigo\archivosDAT\ Archivo: Config: Config 1

Cantidad de periodos: 12 Cantidad de clientes: 3

Patrón de búsqueda: C:\proy_io\Codigo\archivosDAT\nt12-nl3(config1)\OUT\

BUSCAR

Fecha	Archivo	Versión	Tasa	Ver más
2023-10-08 21:06:01	api_20231008210601_v1_it50_datos_nT12_nL3_BBBAM.out	TSv1	0	🔍
2023-10-11 22:55:22	api_20231011225443_v1_it50_datos_nT12_nL3.out	TSv1	0	🔍
2023-10-12 21:37:20	api_20231012213642_tasas05_v1_it50_datos_nT12_nL3.out	TSv1	0.5	🔍
2023-10-12 23:02:31	api_20231012230227_v1_it50_datos_nT12_nL3.out	TSv1	0	🔍

Luciana Vidal Tesis de Maestría de Investigación Operativa 2023

Figura 1.3: Arquitectura en alto nivel de la solución.

En el caso de que el usuario seleccione la lupa junto a un archivo en la vista de resultados, se invoca el endpoint /details para obtener información detallada sobre el resultado contenido en ese archivo. Debajo de la lista de resultados, se despliega una sección que proporciona información específica sobre el resultado seleccionado.

Si el archivo seleccionado representa la ejecución de un archivo DAT específico (como se muestra en la Figura 1.4), se presentarán detalles como el costo total, costo de entrega, costo de remanufacturación, costo de manufacturación y tiempo de ejecución asociados con esa ejecución en particular.

DETALLES DEL ARCHIVO api_20231008210601_v1_it50_datos_nT12_nL3_BBBAM.out:

[BBBAM](#)

Costo total: 10449.50

Entrega: 6150.00

Remanu: 3299.50

Manu: 1000.00

Runtime: 368.00 ms

Figura 1.4: Vista de resultados mostrando los detalles de un archivo específico.

En caso contrario (como se muestra en la Figura 1.5), si el archivo contiene resultados masivos, se mostrarán los mismos detalles (costos y tiempo de ejecución) para cada uno de los resultados contenidos en el archivo. Esta organización garantiza una presentación clara y detallada de la información,

permitiendo al usuario comprender fácilmente los resultados correspondientes a la selección realizada.

DETALLES DEL ARCHIVO api_20231012230227_v1_it2_datos_nT12_nL3.out:			
BBBBA	AMBBM	BABMM	BBBBB
Costo total: 16177.40	Costo total: 20671.00	Costo total: 28411.50	Costo total: 12371.60
Entrega: 12578.20	Entrega: 15765.00	Entrega: 25783.00	Entrega: 10769.80
Remanu: 2799.20	Remanu: 4061.00	Remanu: 2028.50	Remanu: 1001.80
Manu: 800.00	Manu: 845.00	Manu: 600.00	Manu: 600.00
Runtime: 32.00 ms	Runtime: 15.00 ms	Runtime: 13.00 ms	Runtime: 16.00 ms
BMBAM	BBBBM	BMBBA	BMBBB
Costo total: 18323.50	Costo total: 14123.50	Costo total: 18577.40	Costo total: 14771.60
Entrega: 14056.00	Entrega: 11656.00	Entrega: 14978.20	Entrega: 13169.80
Remanu: 3554.50	Remanu: 1754.50	Remanu: 2799.20	Remanu: 1001.80
Manu: 713.00	Manu: 713.00	Manu: 800.00	Manu: 600.00
Runtime: 20.00 ms	Runtime: 16.00 ms	Runtime: 13.00 ms	Runtime: 10.00 ms

Figura 1.5: Vista de resultados mostrando los detalles de un archivo que fue ejecutado de forma masiva.

Finalmente la vista de ejecución de una huerística se accede desde el header seleccionando la opción Ejecución (1.6), y permite al usuario ejecutar la huerística, eligiendo la versión que desee correr. Se tiene dos tipos de filtros: generales y específicos de la huerística. Los filtros generales permiten elegir la cantidad de clientes, períodos, tasa de recovery, carpeta y si se desea hacer una ejecución masiva o específica. Por otro lado, los filtros específicos de la huerística permiten elegir la versión a ejecutar, así como la cantidad de iteraciones y los saltos, y en caso de elegir saltos se debe seleccionar la estrategia de los mismos. Al seleccionar el botón de ejecutar se invoca al endpoint de /run y así se ejecuta el procedimiento de resolución seleccionado. En caso de éxito, se mostrará un mensaje que indique el nombre del archivo OUT generado, y en caso de haber elegido una ejecución masiva será una lista de los mismos.

La vista de ejecución de un procedimiento de resolución se encuentra disponible desde el header al seleccionar la opción 'Ejecución' (como se ilustra en la Figura 1.6). Esta vista brinda al usuario la capacidad de ejecutar el procedimiento de resolución, permitiéndole elegir la versión específica que desea correr.

La interfaz ofrece dos conjuntos de filtros: filtros generales y filtros específicos del procedimiento de resolución. Los filtros generales incluyen opciones para seleccionar la cantidad de clientes, períodos, tasa de recovery, carpeta y la elección entre una ejecución masiva o específica. Por otro lado, los filtros específicos del procedimiento de resolución permiten al usuario elegir la versión que se va a ejecutar, así como la cantidad de iteraciones y los saltos. En caso de seleccionar saltos, se proporciona la opción de elegir la estrategia de los mismos.

Al hacer clic en el botón 'Ejecutar Heurística', se invoca el endpoint /run del proyecto. Este proceso ejecuta el procedimiento de resolución seleccionado según los parámetros configurados en los filtros. Al finalizar la ejecución, se mostrará un mensaje que indicará el nombre del archivo OUT generado. Si la ejecución es masiva, se presentará una lista de los archivos OUT generados durante el proceso.

Este enfoque proporciona al usuario una experiencia intuitiva y personalizable para la ejecución del procedimiento de resolución, ofreciendo flexibilidad a través de filtros generales y específicos, y brindando retroalimentación clara sobre los resultados obtenidos.

Ejercicio de la heurística

GENERALES		HEURÍSTICA	
Cantidad de clientes:	3	Nro versión:	TSv1
Cantidad de períodos:	12	Cantidad de iteraciones:	50
Tasa de recovery:	0	Cantidad de saltos:	01
Carpeta:	C:\proy_io\Codigo\archivosDAT\T12-nL3\	Estrategia de saltos:	Par
Masivo:	☐ Config 1 ☒ BBBAM		

EJECUTAR HEURÍSTICA

La salida se guardó en:
 • C:\proy_io\Codigo\archivosDAT\T12-nL3\config1\OUT\api_20240106212650_v1_it50_saltos1_datos_nT12_nL3_BBBAM.out

Figura 1.6: Vista de ejecución.

Apéndice 2

Evaluación de cantidad de iteraciones

En este Apéndice se justifica el uso de los rangos de cantidad de iteraciones utilizados en las pruebas.

Con el fin de determinar el rango adecuado de iteraciones a considerar en la experimentación numérica, se realizaron pruebas variando este parámetro en una de las variantes del procedimiento de resolución. La variante TSv3 fue seleccionada debido a su rendimiento en las pruebas preliminares. Además, las pruebas se llevaron a cabo utilizando el conjunto de datos correspondientes a las instancias pequeñas (nT12-nL3), lo que permitió evaluar el impacto del número de iteraciones en el desempeño del algoritmo.

En la Tabla 2.1 se presentan los resultados obtenidos al comparar el rendimiento de las variantes al modificar la cantidad de iteraciones. Aunque las pruebas realizadas con 500 iteraciones lograron reducir el gap de costo en todos los escenarios evaluados, el tiempo de ejecución aumentó considerablemente. De hecho, en ciertos casos, el tiempo de ejecución con 500 iteraciones se iguala o incluso supera al del solver, haciendo que esta configuración sea computacionalmente ineficiente. Por lo tanto, para las pruebas exhaustivas que se presentan en el Capítulo 6, se decidió limitar el número de iteraciones a un máximo de 200, logrando un equilibrio adecuado entre precisión y eficiencia.

	Tasas 0		Tasas 0.5		Tasas 1	
	Gap costo (%)	Tiempo (ms)	Gap costo (%)	Tiempo (ms)	Gap costo (%)	Tiempo (ms)
CLPEX	0	2574.09	0	1321.67	0	1216.79
TSv3-15	3.24	79.46	0	61	18.85	267
TSv3-50	1.90	233.47	2.43	227.54	1.77	211.56
TSv3-200	1.26	945.03	1.70	1088.94	1.27	859.48
TSv3-500	1.08	2446.93	1.31	2879.58	1.05	8135.27

Tabla 2.1: Análisis de rango de iteraciones para instancias chicas utilizando la variante TSv3.