



UNIVERSIDAD DE CONCEPCIÓN  
FACULTAD DE INGENIERÍA  
DEPARTAMENTO DE INGENIERÍA INDUSTRIAL



**UN MODELO COMPACTO DE PROGRAMACIÓN LINEAL  
ENTERA MIXTA PARA EL  
LOCATION-OR-ROUTING PROBLEM**

POR

**Matías Sebastián Barrientos Zagal**

Memoria de Título presentada a la Facultad de Ingeniería de la Universidad de Concepción para optar al título profesional de Ingeniero Civil Industrial

Profesor Guía

Carlos Contreras-Bolton

Septiembre 2023

Concepción (Chile)

© Matías Sebastián Barrientos Zagal 2023

© 2023, Matías Sebastián Barrientos Zagal

Se autoriza la reproducción total o parcial, con fines académicos, por cualquier medio o procedimiento, incluyendo la cita bibliográfica del documento

Para mis seres queridos que, con paciencia y resiliencia siempre me han apoyado.

## Resumen

El *Location-or-Routing Problem (LoRP)* es una generalización del problema de localización y ruteo de vehículos (*Location-Routing Problem, LRP*). El objetivo del LRP es minimizar el costo de abrir un subconjunto de depósitos candidatos y el costo de las rutas realizadas desde estos depósitos, con el fin de cubrir a todos los clientes mediante rutas. En el LoRP, la cobertura se puede realizar con un depósito o con una ruta. Así, la función objetivo es la suma del costo de abrir un subconjunto de depósitos candidatos, el costo de las rutas realizadas desde estos depósitos y el costo de cubrir a los clientes con cierto rango de cobertura. La investigación de este problema tiene una gran relevancia, ya que se sitúa en la última milla y es uno de los principales costos dentro de una empresa. Por tanto, este trabajo propone un modelo compacto de programación lineal entera mixta para el LoRP y también, restricciones de refuerzo para mejorar el modelo propuesto. Además, se presenta un *Branch & Cut (B&C)* que considera desigualdades válidas relacionadas con las subrutas, basadas en la capacidad y la distancia máxima de los vehículos. El rendimiento del modelo y algoritmo propuesto es validado sobre instancias de la literatura. Los resultados muestran que el modelo con restricciones de refuerzo obtiene el mejor rendimiento promedio entre los algoritmos y modelos propuestos. Finalmente, el B&C que sólo considera las restricciones de distancia de subrutas y desigualdades válidas de demanda, obtiene los mejores resultados entre los algoritmos propuestos en promedio.

**Palabras clave** – ruteo de vehículos, localización, LoRP, programación lineal entera mixta

## Abstract

The Location-or-Routing Problem (LoRP) is a generalization of the Location-Routing Problem (LRP). The LRP objective is to minimize the cost of opening a subset of candidate depots and the routes' cost performed from these depots in order to cover all customers via routes. In LoRP, coverage can be achieved with either a depot or a route. Thus, the objective function is the cost sum of opening a subset of candidate depots, the routes' cost performed from these depots, and the cost of covering customers within a certain coverage range. Research on this problem is highly relevant as it pertains to the last mile and is one of the major costs within a company. Therefore, this work proposes a compact mixed-integer linear programming model for LoRP and also strength constraints to enhance the proposed model. Additionally, a Branch & Cut (B&C) algorithm is presented that considers valid inequalities related to subroutes based on vehicle capacity and maximum distance. The performance of the proposed model and algorithm is validated on literature instances. The results indicate that the model with strength constraints achieves the best average performance among the proposed algorithms and models. Finally, the B&C algorithm that only considers subroute distance constraints and valid demand inequalities achieves the best average results among the proposed algorithms.

*Keywords* – vehicle routing, location, LoRP, mixed-integer linear programming.

# Índice General

<b>Resumen</b>	<b>I</b>
<b>Abstract</b>	<b>II</b>
<b>1. Introducción</b>	<b>1</b>
1.1. Motivación . . . . .	1
1.2. Objetivos . . . . .	2
1.2.1. Objetivo General . . . . .	2
1.2.2. Objetivos específicos . . . . .	3
1.3. Estructura del documento . . . . .	3
<b>2. Location-or-Routing Problem</b>	<b>4</b>
2.1. Descripción del problema . . . . .	4
2.1.1. Modelo de programación lineal entera mixta existente . . . . .	5
2.2. Revisión de la literatura . . . . .	7
<b>3. Modelo compacto de programación lineal entera mixta propuesto</b>	<b>9</b>
3.1. Modelo propuesto para el LoRP . . . . .	9
3.2. Imposición de subrutas y formulación de la corriente en nodos . . . . .	11
3.3. Formulación de subrutas . . . . .	12
3.4. Restricciones de reforzamiento . . . . .	14
3.5. Complejidad computacional del modelo propuesto . . . . .	16
3.6. Desigualdades válidas . . . . .	16
3.6.1. <i>Rounded capacity inequality (RCI)</i> . . . . .	17
3.6.2. <i>Rounded distance inequality (RDI)</i> . . . . .	17
3.6.3. <i>Laporte's inequality (LI)</i> . . . . .	18
3.6.4. <i>Rounded distance subtour inequality (RDSI)</i> . . . . .	18
3.6.5. <i>Subtour elimination inequality (SEI)</i> . . . . .	18
3.6.6. <i>Projection inequality (PI)</i> . . . . .	19
3.6.7. <i>Depot degree inequality (DDI)</i> . . . . .	19
3.7. Branch and Cut . . . . .	19
3.7.1. Solución inicial . . . . .	20
3.7.1.1. Heurística constructiva propuesta . . . . .	21
3.7.2. Algoritmo de separación . . . . .	22
<b>4. Resultados y Análisis</b>	<b>24</b>
4.1. Configuración de la experimentación . . . . .	24
4.2. Instancias de prueba . . . . .	24
4.3. Resultados computacionales . . . . .	25

---

4.3.1. MB vs MR . . . . .	25
4.3.2. MR vs B&C's . . . . .	26
4.3.3. MR vs B&P . . . . .	26
4.3.4. Análisis respecto al rango de cobertura . . . . .	27
4.3.5. Instancias con nueva mejor solución . . . . .	28
<b>5. Conclusiones</b>	<b>30</b>
<b>Referencias</b>	<b>32</b>
<b>Anexo</b>	<b>34</b>

# Índice de Tablas

1.	Nomenclatura usada en la definición del modelo existente. . . . .	6
2.	Nomenclatura usada en la definición del modelo. . . . .	9
3.	Complejidad computacional de los modelos MB y MR. . . . .	16
4.	Desigualdades válidas y formulación GG para cada B&C. . . . .	20
5.	Desglose sobre las instancias de Akca <i>et al.</i> (2009). . . . .	25
6.	Comparación resultados promedios de MB y MR. . . . .	25
7.	Comparación promedios entre MR y B&C's. . . . .	26
8.	Comparación promedios entre MR y B&P. . . . .	27
9.	Comparación de los promedios con $R \in \{0, 10, 20\}$ . . . . .	27
10.	Comparación de los promedios con $R \in \{30, 40, 50\}$ . . . . .	28
11.	Instancias con óptimos distintos de Arslan (2021). . . . .	29
A.12.	Resultados promedios de B&P con respecto a $R$ . . . . .	34
A.13.	Resultados promedios de B&C-Q con respecto a $R$ . . . . .	34
A.14.	Resultados promedios de B&C-QT con respecto a $R$ . . . . .	35
A.15.	Resultados promedios de B&C-T con respecto a $R$ . . . . .	35
A.16.	Resultados promedios de MB con respecto a $R$ . . . . .	35
A.17.	Resultados promedios de MR con respecto a $R$ . . . . .	35

# Índice de Figuras

1. Ejemplos de una instancia con 30 clientes y cinco depósitos. . . . . 5
2. Ejemplo de tres depósitos y seis clientes con la variable  $d_i$  respectiva (Burger *et al.*, 2018). . . . . 12

# Capítulo 1

## Introducción

En este capítulo se presenta el *Location-or-Routing Problem* y algunas de sus aplicaciones son expuestas. Además, se abordan la motivación, el objetivo general y los objetivos específicos de este trabajo.

### 1.1. Motivación

En diversas empresas existe el área de la cadena de suministro o *supply chain* en inglés. Esta área considera desde la realización de un pedido por parte de un cliente hasta que el producto o servicio es entregado. Una subárea es la logística que contiene al transporte. En este último, se encuentra la última milla que es el proceso entre el último centro de distribución hasta que el producto o servicio se le entrega al cliente. En esta área, uno de los desafíos es reducir los costos de transporte, donde se encuentran los ruteos de vehículos como uno de los factores que más contribuyen a los costos de una empresa. La última milla ha sido relevante en los últimos años, buscando reducir los tiempos y costos con el fin de generar una ventaja competitiva dentro del mercado (Demir *et al.*, 2022).

El *Location-or-Routing Problem* (LoRP) consiste en cubrir un conjunto de clientes mediante rutas de vehículos o por medio de un rango de cobertura asociado a cada depósito. Esto implica que el LoRP se sitúa en la última milla. Por tanto, es relevante tanto su modelado como la contribución de nuevos algoritmos para el LoRP. Además, aparece en diversos contextos en la vida real tales como, la ubicación de escuelas y rutas de autobuses. Esta aplicación considera que los estudiantes están cubiertos dentro de una cierta distancia de la escuela y los autobuses se utilizan para transportar a los estudiantes que se encuentran fuera del rango de cobertura (Arslan, 2021).

El LoRP puede ser considerado en la localización de centros de entrega urbanos con operaciones de drones, vehículos, camiones, entre otros. Por tanto, la localización de torres de control con operaciones de drones, como la vigilancia de incendios forestales y

la vigilancia de la frontera de un territorio en particular (De la Fuente *et al.*, 2023). En general, es un problema que se puede abordar para las nuevas tecnologías en la Industria 4.0, por ejemplo, en aplicaciones de vehículo de guiado automático.

Incluso puede ser utilizado en la localización de instalaciones en pandemias, como la vacunación a gran escala del COVID-19. Finalmente, se puede aplicar a la ubicación de centros de pruebas en zonas urbanas, donde el objetivo es cubrir a una población que puede acudir a un centro cercano o de someterse a las pruebas en su domicilio mediante un vehículo sanitario móvil.

Por otra parte, el LoRP es la combinación de tres tipos de problemas de optimización combinatoria, la localización, ruteo de vehículos y la cobertura de conjuntos. Para la localización y el ruteo de vehículos existen problemas como el *Location Routing Problem* (LRP) y para la cobertura de conjuntos existe el *Set Covering Problem* (SCP) (Haghi *et al.*, 2023). Donde, el LRP consiste en abrir un conjunto de depósitos y designar un conjunto de rutas por cada depósito, considerando las cargas totales de las rutas en operación que no deben exceder la capacidad de los vehículos (Baldacci *et al.*, 2011). Mientras que, en el LRP, cada cliente debe ser parte de una ruta de un vehículo. En el LoRP, cada cliente puede ser cubierto por una ruta o un depósito (Haghi *et al.*, 2023). Además, tanto el LRP como el LoRP tienen el mismo objetivo de minimizar el costo total de la apertura de las instalaciones y de las rutas de los vehículos. En tanto, en el SCP cada instalación abierta tiene un rango de cobertura. El objetivo es abrir un conjunto de instalaciones de manera que cada usuario se encuentre dentro del rango de cobertura de una de las instalaciones abiertas (Haghi *et al.*, 2023).

Dado que el LoRP es una generalización del LRP y considerando que, tanto el SCP como el LRP son *NP-hard*, entonces, el LoRP también lo es. Esto implica que es computacionalmente muy complejo de resolver. Por tanto, contribuir con un nuevo y eficiente modelo para el LoRP, permite abordar mayores tamaños del problema. Por ello, en este trabajo se presenta un modelo compacto de programación lineal entera mixta y un conjunto de restricciones de reforzamiento para la resolución del LoRP. Además, se presenta un B&C que considera desigualdades válidas junto al algoritmo de separación. Se presentan experimentos computacionales en instancias de 30 a 40 clientes con cinco depósitos candidatos.

## 1.2. Objetivos

### 1.2.1. Objetivo General

Proponer un modelo compacto de programación lineal entera mixta para el LoRP.

### 1.2.2. Objetivos específicos

- Revisar el estado del arte del LoRP y problemas similares.
- Modelar e implementar de forma compacta del LoRP.
- Agregar restricciones de reforzamiento al modelo compacto.
- Implementar un B&C.
- Realizar experimentos computacionales.
- Analizar y comparar los resultados del modelo con el estado del arte.

## 1.3. Estructura del documento

El resto del documento se estructura como sigue. El Capítulo 2 presenta formalmente el problema abordado, incluyendo modelos matemáticos relacionados y una revisión de la literatura enfocada en métodos que permitan resolverlo. El Capítulo 3 presenta el modelo compacto de programación lineal entera mixta junto a las restricciones de reforzamiento y, además, la descripción del B&C utilizado. Luego, el Capítulo 4 aborda los experimentos computacionales y resultados obtenidos. Finalmente, los resultados se discuten y analizan en el Capítulo 5, donde también se presentan las conclusiones.

## Capítulo 2

# Location-or-Routing Problem

Este capítulo define y describe el LoRP. Además, se revisa la literatura más relevante y reciente asociada al LoRP, enfocándose en los algoritmos del estado del arte que resuelven el LoRP.

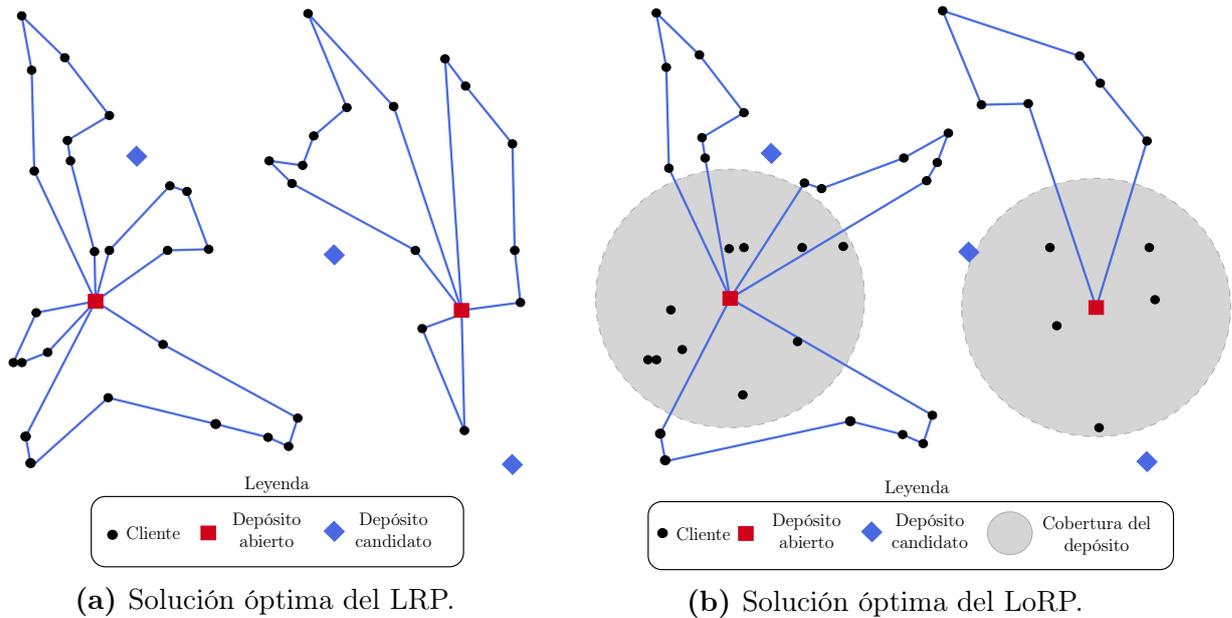
### 2.1. Descripción del problema

Dado un conjunto de depósitos candidatos ( $W$ ), un conjunto de clientes por cubrir ( $N$ ) y un grafo  $G = (V, E)$ , donde  $V = W \cup N$  es el conjunto de los nodos y  $E = \{(i, j) : i, j \in V, i \neq j\}$  es el conjunto de aristas. Se define  $t_{ij}$  como el costo del viaje en la arista  $(i, j) \in E$ . El costo de apertura de un depósito  $w \in W$  es  $f_w$  y  $r_w$  es el rango de cobertura. Adicionalmente,  $q_i$  es la demanda solicitada por el cliente  $i \in N$ , mientras que  $Q$  es la capacidad de los vehículos, considerando una flota de vehículos homogéneos y  $T$  es la distancia máxima de cada ruta que realiza un vehículo. Finalmente, un depósito  $w$  puede cubrir un cliente  $i \in N$  si  $t_{wi} \leq r_w$ .

El LoRP se define como la selección de un subconjunto de depósitos ( $W' \subseteq W$ ) que se deciden abrir y la determinación de un conjunto de rutas de vehículos. Cada ruta comienza en un depósito abierto  $w \in W'$  y considera un subconjunto de clientes ( $N' \subseteq N$ ) que regresa al mismo depósito  $w$ , respetando las restricciones de capacidad y distancia máxima del vehículo. Por tanto, el objetivo es cubrir a cada cliente  $i \in N$  ya sea mediante un depósito o mediante una ruta, minimizando el costo total de abrir depósitos, los costos de las rutas de los vehículos y los costos de los clientes cubiertos por el rango de cobertura. Sin embargo, los dos primeros costos se consideran costos “hard”. No así el costo de cubrir a los clientes por el rango de cobertura de un depósito. Este problema, específicamente se define como *Hard-cost minimizing Location-or-Routing Problem* (HMLoRP), siendo el símil al LoRP en esta descripción.

En la Figura 1a se presenta un ejemplo con 30 clientes y cinco depósitos candidatos.

El ejemplo es la solución óptima del LRP y tiene un costo de función objetivo de 753,03, mientras que en la Figura 1b es presentada la misma instancia. Sin embargo, con un rango de cobertura distinto de cero, siendo la solución óptima del LoRP con un costo de función objetivo de 618,16. Esto implica que el costo disminuye al considerar un rango de cobertura. Por tanto, cuando los depósitos no cubren a los clientes, se puede observar que se realizan más rutas de vehículos que en la Figura 1b. Esto implica que la distancia recorrida por las rutas es mayor que cuando se considera un rango de cobertura.



**Figura 1:** Ejemplos de una instancia con 30 clientes y cinco depósitos.

En este sentido, una de las observaciones más relevantes en Arslan (2021) es que el rango de cobertura de los depósitos es un determinante importante del número y localización de los depósitos por abrir. Por último, se puede observar en la Figura 1a y 1b que, cuando el rango de cobertura  $R$  es igual a cero, el LoRP se transforma en el LRP.

### 2.1.1. Modelo de programación lineal entera mixta existente

Antes de presentar el modelo propuesto por Arslan (2021), es necesario definir notación adicional. A modo de resumir y facilitar la comprensión del modelo, en la Tabla 1 se presentan los conjuntos, parámetros y variables de decisión del modelo. Así, se define  $p$  como una ruta que corresponde a un conjunto ordenado de arcos dirigidos que comienza y termina en el mismo nodo que representa el depósito candidato  $w \in W$ . Finalmente, una ruta  $p$  es factible si cumple con la distancia máxima de la ruta y la capacidad de los vehículos. Esto significa que se debe cumplir que  $\sum_{(i,j) \in E_p} d_{ij} \leq T$  y  $\sum_{(i,j) \in E_p: j \neq p_0} q_j \leq Q$ .

**Tabla 1:** Nomenclatura usada en la definición del modelo existente.

Conjuntos	
$I_j$	Conjunto de depósitos que cubren al cliente $j \in N$ .
$J_w$	Conjunto de los clientes cubiertos por el depósito $w \in W$ .
$J_p$	Conjunto de los clientes visitados en el camino $p$ .
$E_p$	Conjunto de arcos en $p$ .
$\tilde{\mathcal{P}}_j$	Conjunto de los caminos que visitan al cliente $j \in N$ .
$\tilde{\mathcal{P}}_w$	Conjunto de los caminos que comienzan en el depósito $w \in W$ .
$\hat{\mathcal{P}}_{wj}$	Conjunto intersección entre $\tilde{\mathcal{P}}_j$ y $\tilde{\mathcal{P}}_w$ , es decir, $\hat{\mathcal{P}}_{wj} = \tilde{\mathcal{P}}_j \cap \tilde{\mathcal{P}}_w$ .
$\mathcal{P}$	Familia de conjuntos de $\tilde{\mathcal{P}}_w$ , es decir, $\mathcal{P} = \bigcup_{w \in W} \tilde{\mathcal{P}}_w$ .
Parámetros	
$d_p$	Costo de la ruta $p$ .
$q_p$	Demanda que se cubre en la ruta $p$ .
$p_0$	Nodo de inicio y el final de la ruta $p$ .
Variables de decisión	
$x_w$	Toma valor 1, si el depósito $w$ es seleccionado, 0 caso contrario, $\forall w \in W$ .
$y_p$	Toma valor 1, si la ruta $p$ es seleccionada, 0 caso contrario, $\forall p \in \mathcal{P}$ .
$z_{wj}$	Toma valor 1, si el cliente $j$ es asignado al depósito $w$ , 0 caso contrario, $\forall j \in N, w \in W$ .

A continuación, se presenta el modelo de programación lineal entera mixta para el LoRP planteado por [Arslan \(2021\)](#).

$$\min \sum_{w \in W} f_w x_w + \sum_{p \in \mathcal{P}} d_p y_p \quad (1)$$

sujeto a:

$$\sum_{w \in I_j} z_{wj} + \sum_{p \in \tilde{\mathcal{P}}_j} y_p = 1, \quad j \in N \quad (2)$$

$$\sum_{j \in J_w} q_j z_{wj} + \sum_{p \in \tilde{\mathcal{P}}_w} q_p y_p \leq C_w x_w, \quad w \in W \quad (3)$$

$$z_{wj} \leq x_w, \quad w \in W, j \in J_w \quad (4)$$

$$\sum_{p \in \hat{\mathcal{P}}_{wj}} y_p \leq x_w, \quad w \in W, j \in J_w \quad (5)$$

$$\sum_{p \in \tilde{\mathcal{P}}_w} y_p = v_w, \quad w \in W \quad (6)$$

$$\sum_{w \in W} v_w \geq \left\lceil \frac{\sum_{j \in J \setminus \cup_{w \in W} J_w} q_j}{Q} \right\rceil, \quad w \in W, j \in J_w \quad (7)$$

$$x_w \in \{0, 1\}, \quad \forall w \in W \quad (8)$$

$$y_p \in \{0, 1\}, \quad \forall p \in \mathcal{P} \quad (9)$$

$$z_{wj} \in \{0, 1\}, \quad \forall j \in J_w, \forall w \in W \quad (10)$$

La función objetivo (1) minimiza: la suma del costo de abrir los depósitos, el costo de todas las rutas y de los clientes que son cubiertos por el rango de cobertura de un depósito. Las restricciones (2) aseguran que todos los clientes estén cubiertos solo una vez en la solución, aunque varios depósitos abiertos pueden estar dentro del rango de cobertura de un cliente. Las restricciones (3) respetan las capacidades de los depósitos y también aseguran que una ruta comience desde un depósito abierto. Además, las restricciones (3) aseguran abrir el depósito  $w \in W$  cuando existe un camino en la solución óptima que comienza en ese depósito. Las restricciones (4) vinculan al depósito y las variables de asignación que cubre dicho depósito a un nodo  $j$  y obligan a respetar las capacidades del depósito. Las restricciones (5), al igual que las restricciones (3), aseguran abrir el depósito  $w \in W$  cuando existe un camino en la solución óptima que comienza en el depósito  $w$ , pero son un conjunto de restricciones más fuertes. Las restricciones (6) asignan el número de vehículos utilizados en el depósito  $w \in W$  a la variable  $v_w$ . Las restricciones (7) establecen un límite inferior para el número de vehículos. Finalmente, las restricciones (8)-(10) indican el dominio de las variables.

## 2.2. Revisión de la literatura

El LoRP se plantea y se define por primera vez en Arslan (2021), quien formula un modelo de programación lineal entera mixta, desde la perspectiva del problema de cobertura de conjuntos junto a desigualdades válidas para esa formulación. El LoRP es planteado como una clase de problemas que se pueden relacionar con el problema de localización de  $p$  medianas y centradas, y también contemplan la localización considerando la máxima cobertura posible. Además, se propone un *Branch & Price* (B&P) como un método exacto para resolver el LoRP.

En Haggi *et al.* (2023) se extiende el LoRP mediante el *location-or-routing problem with partial and decaying coverage* (LoRPPDC). El LoRPPDC considera la cobertura decreciente para los depósitos, con una posibilidad de cubrir a los usuarios parcialmente y de tener visitas de múltiples vehículos a las ubicaciones de los usuarios. En este problema se consideran dos objetivos jerárquicos. El primer objetivo es maximizar el número de usuarios cubiertos, es decir, minimizar el número de usuarios no cubiertos. Mientras que el segundo objetivo es minimizar el porcentaje del presupuesto a gastar. Para su resolución se formula un modelo matemático no compacto y se resuelve utilizando un algoritmo de B&C. Además, para las restricciones de eliminación de subrutas, plantean una formulación de Miller-Tucker-Zemlin (MTZ) (Miller *et al.*, 1960). Finalmente, proponen una metaheurística *adaptive large neighborhood search* (ALNS) que es validada en un conjunto de instancias. Los resultados de ALNS muestran que en 36 segundos promedio

obtiene un gap promedio de 0,47 %. Además, pudo encontrar valores óptimos para más del 60 % de las instancias, confirmando la eficiencia del ALNS.

Recientemente, [De la Fuente et al. \(2023\)](#) formulan una nueva variante para el LoRP que llaman *location-or-routing problem with fixed destination multi-depot and distance constrained drones* (LoRP-FDM-DCD). Este problema es formulado para una aplicación de monitoreo de incendios forestales en una base de datos georeferenciados al sur de Chile. Los autores proponen depósitos como torres de vigilancia y las rutas de los vehículos son realizados con drones, considerando la capacidad de sus baterías. Para abordar el nuevo problema, proponen un modelo compacto de programación lineal entera mixta que se basa en la formulación de dos índices propuesto por [Burger et al. \(2018\)](#). Esta formulación ayuda a modelar el LoRP-FDM-DCD con menos variables binarias y restricciones sobre imposición de ciclos. Además, las restricciones de localización y de ruteo, se establecen de manera explícita, mediante restricciones de subrutas provenientes de Dantzig-Fulkerson-Johnson (DFJ) ([Dantzig et al., 1954](#)). Dado que las restricciones DFJ son intratables computacionalmente, utilizan las formulaciones de Gavish & Graves (GG) ([Gavish y Graves, 1978](#)) y MTZ. Los resultados muestran que los mejores resultados son obtenidos por la formulación GG. Sin embargo, el modelo no es capaz de obtener buenos resultados en instancias mayores. Por esto, proponen una matheurística que encuentra soluciones de buena calidad para todas las instancias requeridas en bajos tiempos computacionales promedio.

## Capítulo 3

# Modelo compacto de programación lineal entera mixta propuesto

Este capítulo presenta el modelo compacto de programación lineal entera mixta propuesto y restricciones adicionales de reforzamiento. Además, se presentan las desigualdades válidas utilizadas en el B&C, junto al algoritmo de separación utilizado.

### 3.1. Modelo propuesto para el LoRP

Con el fin de hacer más fácil la presentación del modelo propuesto, la notación del modelo es resumida en Tabla 2.

**Tabla 2:** Nomenclatura usada en la definición del modelo.

Conjuntos	
$N$	Conjunto de clientes por cubrir (indexado por $i$ ).
$W$	Conjunto de depósitos candidatos (indexado por $w$ ).
$V$	Conjunto de clientes de las rutas, $V = \{W \cup N\}$ (indexado por $j$ ).
Parámetros	
$\beta_{wi}$	Parámetro binario igual a 1, si un depósito $w$ cubre al cliente $i$ , $\forall w \in W, \forall i \in N$ .
$t_{jk}$	Distancia entre el cliente $j$ y $k$ , $\forall j, k \in V$ .
$q_i$	Demanda del cliente $i$ , $\forall i \in N$ .
$f_w$	Costo de apertura del depósito $w$ , $\forall w \in W$ .
$T$	Distancia máxima de una ruta.
$R$	Rango de cobertura de los depósitos.
$M$	Valor muy grande para el método <i>Big-M</i> .
Variables de decisión	
$L_{ij}$	Toma valor 1, si el nodo $i$ es cubierto por un nodo $j$ , 0 caso contrario, $\forall i, j \in V$ .
$P_i$	Toma valor 1, si el cliente $i$ es cubierto por una ruta, 0 caso contrario, $\forall i \in N$ .
$y_w$	Toma valor 1, si el depósito candidato $w$ es abierto, 0 caso contrario, $\forall w \in W$ .
$x_{ij}$	Toma valor 1, si un vehículo viaja desde el nodo $i$ al $j$ , 0 caso contrario, $i, j \in V$ .
$d_j$	Indica la etiqueta del nodo $j$ , $\forall j \in V$ .

El modelo compacto de programación lineal entera mixta propuesto en este estudio consiste en:

$$\min \sum_{w \in W} f_w y_w + \sum_{j \in V} \sum_{k \in V: j \neq k} t_{jk} x_{jk} \quad (11)$$

sujeto a:

$$P_i + \sum_{w \in W} L_{iw} \leq 1, \quad \forall i \in N \quad (12)$$

$$\sum_{i \in N} \left( \sum_{w \in W} L_{iw} + P_i \right) = |N| \quad (13)$$

$$\sum_{w \in W} L_{iw} \leq 1, \quad \forall i \in N \quad (14)$$

$$L_{iw} \leq \beta_{wi} y_w, \quad \forall i \in N, \quad \forall w \in W \quad (15)$$

$$\sum_{w \in W} \beta_{wi} y_w \geq \sum_{w \in W} L_{iw}, \quad \forall i \in N \quad (16)$$

$$\sum_{j \in V \setminus \{w\}} x_{wj} \leq M y_w, \quad \forall w \in W \quad (17)$$

$$\sum_{j \in V \setminus \{w\}} x_{jw} - \sum_{j \in V \setminus \{w\}} x_{wj} = 0, \quad \forall w \in W \quad (18)$$

$$\sum_{j \in V \setminus \{i\}} x_{ij} = P_i, \quad \forall i \in N \quad (19)$$

$$\sum_{j \in V \setminus \{i\}} x_{ji} = P_i, \quad \forall i \in N \quad (20)$$

$$d_w = w, \quad \forall w \in W \quad (21)$$

$$d_j - d_k \leq (|W| - 1)(1 - x_{jk}), \quad \forall j, k \in V : j \neq k \quad (22)$$

$$d_k - d_j \leq (|W| - 1)(1 - x_{jk}), \quad \forall j, k \in V : j \neq k \quad (23)$$

$$x_{wk} = 0, \quad \forall w, k \in W : w \neq k \quad (24)$$

$$\sum_{j \in S} \sum_{k \in S: j \neq k} x_{jk} \leq |S| - 1, \quad \forall S \subset V : 2 \leq |S| \leq |N| - 2 \quad (25)$$

$$y_w \in \{0, 1\}, \quad \forall w \in W \quad (26)$$

$$x_{jk} \in \{0, 1\}, \quad \forall j, k \in V \quad (27)$$

$$L_{iw} \in \{0, 1\}, \quad \forall i \in N, \quad \forall w \in W \quad (28)$$

$$P_i \in \{0, 1\}, \quad \forall i \in N \quad (29)$$

$$d_j \geq 0, \quad \forall j \in V \quad (30)$$

La función objetivo (11) minimiza: la suma del costo de abrir los depósitos, el costo de todas las rutas y de los clientes que son cubiertos por el rango de cobertura de un depósito. Las restricciones (12) y (13) aseguran que cada cliente  $i$  sea cubierto por un

depósito o una ruta. Las restricciones (14) señalan que un cliente  $i$  es cubierto sólo por un depósito. Mientras que las restricciones (15) y (16) indican que un cliente  $i$  está cubierto cuando se encuentra en el rango de cobertura de un depósito abierto. Las restricciones (17) permiten que si el depósito  $w$  es utilizado, entonces, pueden salir vehículos desde  $w$ . Luego, las restricciones (18) son utilizadas para controlar las aristas que entran en el depósito  $w$  sean las mismas que salgan. Además, las restricciones (19) y (20) controlan el flujo de entrada y de salida para cada cliente  $i$ , siempre y cuando dicho cliente es cubierto por una ruta. Por otra parte, las restricciones (21)-(23) son para controlar los subrutas de cada depósito, están basadas en la formulación de [Burger et al. \(2018\)](#), en la sección 3.2 se explican con mayor detalle. Las restricciones (24) permiten evitar las aristas entre depósitos. Finalmente, las restricciones (26)-(30) corresponden a las restricciones de dominio de las variables de decisión.

## 3.2. Imposición de subrutas y formulación de la corriente en nodos

Las restricciones de eliminación de subrutas se utilizan para asegurar que no existan subrutas dentro del conjunto de nodos. Sin embargo, las restricciones de imposición de subrutas aseguran un número mínimo de subrutas en un conjunto de nodos.

Para comprender la formulación de [Burger et al. \(2018\)](#), se utiliza el ejemplo de la Figura 2. Las soluciones de destino fijo se obtienen con  $|V|$  variables continuas  $k_i \geq 0$  que satisfacen las restricciones de imposición de subrutas:

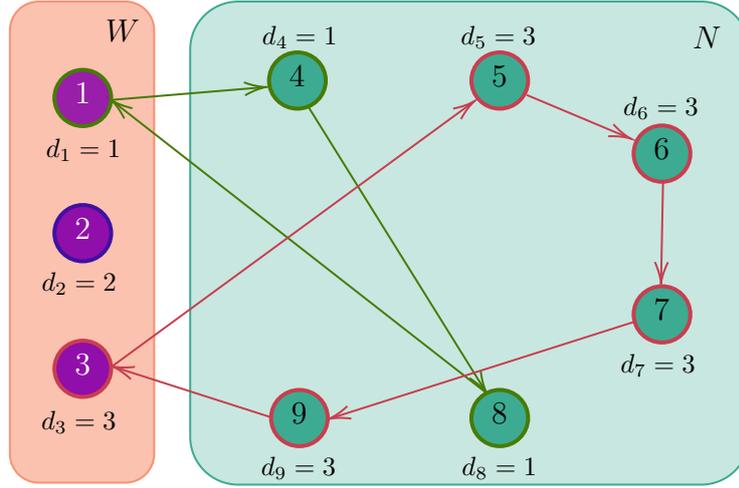
$$d_w = w \quad \forall w \in W \quad (31)$$

$$d_i - d_j \leq (|W| - 1)(1 - x_{ij}) \quad \forall i, j \in V : i \neq j \quad (32)$$

$$d_j - d_i \leq (|W| - 1)(1 - x_{ij}) \quad \forall i, j \in V : i \neq j \quad (33)$$

Luego, de las restricciones (31) se concluye que  $1 \leq d_w \leq |W|, \forall w \in W$ . Además, para la solución óptima se cumple que  $1 \leq d_i \leq |W|, \forall i \in V$ .

Por otra parte, mediante inducción se demostró que, cualquier camino que empieza en  $w$  también debe terminar en el mismo depósito  $w$ . Adicionalmente, en [Burger et al. \(2018\)](#) demuestran que todas las restricciones de eliminación de subrutas y de imposición de subrutas resultan en un grafo con, exactamente,  $\sum_{w \in W} v_w$  subrutas. Donde cada depósito  $w \in W$  contiene exactamente  $v_w$  subrutas, siendo  $v_w$  el número de vehículos asociados al depósito  $w$ . En el modelo propuesto para el LoRP,  $v_w$  no es definido, es el modelo el que decide cuántos es la cantidad óptima de vehículos. Esto se observa en las restricciones (17), donde usan la formulación *Big-M*.



**Figura 2:** Ejemplo de tres depósitos y seis clientes con la variable  $d_i$  respectiva (Burger *et al.*, 2018).

Analizando el ejemplo de la Figura 2, se observa que hay dos subrutas, uno que sale desde el depósito 1 y otro del depósito 3. Considerando algunas de las restricciones activas, se tiene lo siguiente:

$$d_1 = 1 \quad (34)$$

$$d_1 - d_4 \leq (|W| - 1)(1 - x_{14}) \quad (35)$$

$$d_4 - d_1 \leq (|W| - 1)(1 - x_{14}) \quad (36)$$

Como  $x_{14} = 1$ , entonces de la Ecuación (35) tiene que  $d_1 - d_4 \leq 0$ , o sea  $d_4 \leq d_1$ . Luego, si se considera la Ecuación (36), como  $x_{14} = 1$ , entonces  $d_1 - d_4 \geq 0$ . Siendo  $d_4 = 1$ , la única posibilidad para dicha variable. Lo mismo ocurre con  $d_8$ , donde se considera que  $d_4 = 1$  y  $x_{48} = 1$ .

La número de variables a utilizar son  $|W| + |N|$  en una formulación con dos índices como las de las Ecuaciones (31)-(33). Esto implica una disminución de la cantidad de variables en relación a una formulación de tres índices con  $\mathcal{O}(|V|^2|W|)$  variables binarias, donde el número de variables binarias aumenta cúbicamente con el número de depósitos (a diferencia del aumento cuadrático para formulaciones de dos índices). El número de restricciones a utilizar en esta formulación es de  $(|W| + |N|)^2 - |N|$  (Burger *et al.*, 2018).

### 3.3. Formulación de subrutas

Notar que las restricciones (25) corresponden a la formulación DFJ de eliminación de subrutas (Dantzig *et al.*, 1954). Esta formulación no es práctica debido a su complejidad computacional  $\mathcal{O}(2^{|V|})$ . Por tanto, las restricciones (25) son reemplazadas por la formulación

de Gavish y Graves (1978). Así, se definen  $g_{ij}$  como las variables que representan la demanda cubierta desde un depósito a una arista  $(i, j), \forall i, j \in V$  y se necesitan  $|V|(|V| - 1)$  variables continuas. Las restricciones son:

$$\sum_{j \in V \setminus \{i\}} g_{ij} - \sum_{j \in V \setminus \{i\}} g_{ji} = P_i q_i \quad \forall i \in N \quad (37)$$

$$q_i x_{ij} \leq g_{ij} \leq (Q - q_j) x_{ij} \quad \forall i, j \in N \quad (38)$$

$$g_{wi} = 0, \quad \forall w \in W, \forall i \in N \quad (39)$$

$$g_{iw} \leq Q x_{iw} \quad \forall w \in W, \forall i \in N \quad (40)$$

$$g_{jk} \geq 0 \quad \forall i, j \in V : i \neq j \quad (41)$$

Las restricciones (37)-(41) corresponden a la capacidad máxima a cargar un vehículo en una ruta. Las restricciones (37) aseguran que la diferencia del flujo al pasar por el cliente  $i$  sea exactamente su demanda  $q_i$  ( $i \in N$ ), siempre y cuando sea cubierto por una ruta. Las restricciones (38) son una cota inferior del flujo antes de llegar al cliente  $i$  y una cota superior del flujo del vehículo antes de llegar al cliente  $j$ . Adicionalmente, las restricciones (39) y (40) indican que el flujo es cero al salir de un depósito, mientras las segundas indican que la cota superior del flujo es  $Q$  al volver al depósito. Finalmente, las restricciones (41) son el dominio de la variables de flujo  $g$ .

De manera similar a la formulación GG de subrutas considerando demanda, se presenta la formulación GG que considera la distancia máxima de la ruta. Así, se definen  $h_{ij}$  como las variables que representan la distancia recorrida desde un depósito a una arista  $(i, j), \forall i, j \in V$  y se necesitan  $|V|(|V| - 1)$  variables continuas. Las restricciones son:

$$\sum_{j \in V \setminus \{i\}} h_{ij} - \sum_{j \in V \setminus \{i\}} h_{ji} = \sum_{j \in V \setminus \{i\}} t_{ij} x_{ij} \quad \forall i \in N \quad (42)$$

$$t_{ij} x_{ij} \leq h_{ij} \leq T x_{ij} \quad \forall i, j \in N \quad (43)$$

$$h_{wi} = t_{wi} x_{wi} \quad \forall w \in W, \forall i \in N \quad (44)$$

$$h_{iw} \leq T x_{iw} \quad \forall w \in W, \forall i \in N \quad (45)$$

$$h_{ij} \geq 0 \quad \forall i, j \in V : i \neq j \quad (46)$$

Las restricciones (42)-(46) aseguran la distancia máxima que puede recorrer un vehículo en una ruta. Las restricciones (42) aseguran que la diferencia del flujo al pasar por la arista  $(i, j)$  sea exactamente la distancia recorrida hasta esa arista, si es visitado. Las restricciones (43) son la cota inferior para el flujo, siendo como mínimo  $t_{ij}$  y la cota superior del flujo que es  $T$ , si se utiliza la arista  $(i, j)$ . Las restricciones (44) indican que al salir de un depósito, el flujo es igual a la distancia en esa arista  $(i, j)$ . Las restricciones (45) son la

cota superior  $T$  que puede tener el flujo de distancia al volver al depósito. Finalmente, las restricciones (46) son el dominio de la variables de flujo  $h$ .

### 3.4. Restricciones de reforzamiento

Con el fin de mejorar el rendimiento computacional del modelo, se proponen restricciones adicionales. Estas restricciones adicionales consideran  $C_w = \{i : \beta_{wi} = 1, \forall i \in N\}$  como el conjunto de clientes que son cubiertos por el depósito  $w \in W$ ,  $\overline{C_w}$  como el conjunto de clientes que no cubre el depósito  $w \in W$  y  $N_w = V \setminus \{\bigcup_{w \in W} C_w\}$  como el conjunto de los clientes que no pueden ser cubiertos por ningún depósito. Así, el primer conjunto de restricciones de reforzamiento son las siguientes:

$$\sum_{i,j \in C_w: i \neq j} x_{ij} \leq M(1 - y_w) \quad \forall w \in W : C_w \neq \emptyset \quad (47)$$

Las restricciones (47) indican que si un depósito se abre, todas las aristas que existen entre los clientes que están dentro de su rango de cobertura deben ser cero, debido a que no se permiten rutas dentro de ese rango.

Además, existen más conjuntos de restricciones de reforzamiento asociadas a la variable  $x$  que, se presentan, a continuación:

$$x_{wi} \leq (1 - \beta_{wi})y_w \quad \forall w \in W, \forall i \in N \quad (48)$$

$$x_{iw} \leq (1 - \beta_{wi})y_w \quad \forall w \in W, \forall i \in N \quad (49)$$

$$\sum_{w \in W} wx_{wi} \leq d_i \quad \forall i \in N \quad (50)$$

$$\sum_{w \in W} wx_{iw} \leq d_i \quad \forall i \in N \quad (51)$$

Las restricciones (48) y (49) señalan que si un depósito no se abre, entonces, la arista entre el cliente  $i \in N$  y el depósito  $w \in W$  no es parte de la solución. Sin embargo, si abre ese depósito y cubre al cliente, de todas formas esa arista no pertenece a una solución factible. Adicionalmente, las restricciones (50) y (51) aseguran que el primer y el último cliente  $i \in N$  de la ruta sea al menos  $w$ .

Con respecto a la variable  $P$ , existen dos tipos de restricciones:

$$(1 - y_w) + (1 - \beta_{wi}) \geq P_i \quad \forall i \in N, \forall w \in W \quad (52)$$

$$P_i = 1 \quad \forall i \in N_w : N_w \neq \emptyset \quad (53)$$

Donde las restricciones (52) señalan que si se abre un depósito  $w$  y cubre al cliente  $i$ ,

entonces,  $P_i$  toma el valor cero. En un ámbito similar, las restricciones (53) fuerzan a los clientes que no se puedan cubrir por depósitos, deban ser cubiertos por una ruta.

Para complementar a la cobertura de conjuntos, a continuación, se proponen las siguientes restricciones de refuerzo asociadas a la variable  $L$ :

$$L_{ij} = 0 \quad \forall i, j \in N : i \neq j \quad (54)$$

$$L_{wk} = 0 \quad \forall w, k \in W : w \neq k \quad (55)$$

$$L_{iw} + L_{wi} \leq 1 \quad \forall i \in N, \forall w \in W \quad (56)$$

$$L_{iw} \leq \beta_{wi} y_w, \quad \forall i \in C_w, \forall w \in W \quad (57)$$

$$L_{wi} \leq \beta_{wi} y_w, \quad \forall i \in C_w, \forall w \in W \quad (58)$$

$$L_{iw} + L_{wi} = 0 \quad \forall i \in \overline{C_W}, \forall w \in W \quad (59)$$

$$\sum_{i \in N_w} (L_{iw} + L_{wi}) = 0 \quad \forall w \in W \quad (60)$$

Las restricciones (54) y (55) aseguran que no se puedan cubrir entre clientes o entre depósitos. Las restricciones (56) buscan controlar que exista sólo una variable  $L$  asociada a la cobertura de un cliente  $i \in N$  desde un depósito  $w \in W$ . Las restricciones (57) y (58) reemplazan a las restricciones (15), generando una menor cantidad de restricciones necesarias. Sin embargo, al reemplazar (15), es necesario añadir las restricciones (59) y (60). Además, las restricciones (59) y (60) refuerzan que los clientes que no son cubiertos por el rango de cobertura del depósito  $w \in W$ , no pueden tomar el valor uno de la variable  $L_{iw}$  o  $L_{wi}$ . Finalmente, a continuación, se proponen restricciones adicionales asociadas a las variables de flujo  $g$  y  $h$ .

$$g_{wk} = 0 \quad \forall w, k \in W : w \neq k \quad (61)$$

$$h_{wk} = 0 \quad \forall w, k \in W : w \neq k \quad (62)$$

$$\sum_{i \in N} \sum_{w \in W} g_{iw} \leq \sum_{i \in N} q_i \quad (63)$$

$$(t_{ij} + \min_{w \in W} \{d_{wi}\}) x_{ij} \leq h_{ij} \leq (T - \min_{w \in W} \{d_{jw}\}) x_{ij} \quad \forall i, j \in N \quad (64)$$

Las restricciones (61) y (62) refuerzan que el flujo entre depósitos sea cero. La restricción (63) asegura que la suma de todas las demandas de cada ruta sea menor o igual a la demanda de todos los clientes en  $N$ . Notar que no es igual a la demanda de todos los clientes, ya que puede existir demanda cubierta por depósitos. Por otra parte, las restricciones (64) reemplazan a las restricciones (43), añadiendo una mejor cota inferior y superior de la variable  $h$  asociada. El flujo  $h_{ij}$  no debe ser menor que la distancia  $t_{ij}$  más la distancia entre el cliente  $i$  y el depósito  $w$  más cercano. Por otra parte, la última arista de una ruta,

antes de volver a un depósito, no puede ser mayor que  $T$  menos la distancia entre el cliente  $j$  y el depósito  $w$  más cercano.

### 3.5. Complejidad computacional del modelo propuesto

Para analizar la complejidad computacional de los modelos propuestos, primero se deben definir de qué conjunto de restricciones están compuestos. Así, el modelo básico (MB) propuesto está compuesto por (11)-(24), (26)-(30) y (37)-(46). Luego, el modelo con restricciones de refuerzo (MR) consta de (11)-(24), (26)-(30), (37)-(42) y (44)-(64).

Para la formulación de subrutras de distancia y capacidad se deben generar  $2|V|(|V| - 1)$  variables y  $|V|(|V| + 2)$  restricciones, entonces, la complejidad computacional para las variables y restricciones es  $\mathcal{O}(|V|^2)$ . Adicionalmente, para el MB y el MR, la cantidad de variables a utilizar son  $|V|$  y de restricciones son  $|V|^2 - |W|$ . Por tanto, la complejidad es  $\mathcal{O}(|V|)$  y  $\mathcal{O}(|V|^2 - |W|)$ , respectivamente. Sin embargo, la cantidad de variables está dominada por la variable  $x$ , ya que son  $|V|^2$ . Por tanto, la complejidad computacional es  $\mathcal{O}(|V|^2)$ .

En la Tabla 3 se presenta un resumen de la complejidad computacional y de las restricciones que consta cada modelo de programación lineal entera mixta. La cantidad de variables enteras está sujeto a la formulación de subrutras y la imposición de ciclos, mientras que para las variables binarias está sujeto a la combinación posible de aristas del conjunto  $|V|$ . Por último, la cantidad de variables continuas está determinada por la formulación GG, es decir, dos veces  $2|V|(|V| - 1)$  de variables continuas considerando la formulación GG de demanda como de distancia máxima de la ruta.

**Tabla 3:** Complejidad computacional de los modelos MB y MR.

Modelo	Conjunto de restricciones	# Variables enteras	# Variables binarias	# Variables continuas	# Restricciones
MB	(11)-(24), (26)-(30), (37)-(46)	$\mathcal{O}( V ^2)$	$\mathcal{O}( V ^2)$	$\mathcal{O}( V ^2)$	$\mathcal{O}( V ^2)$
MR	(11)-(24), (26)-(30), (37)-(42), (44)-(64)	$\mathcal{O}( V ^2)$	$\mathcal{O}( V ^2)$	$\mathcal{O}( V ^2)$	$\mathcal{O}( V ^2)$

### 3.6. Desigualdades válidas

Esta sección introduce la familia de restricciones para fortalecer la relajación del modelo de programación lineal entera mixta propuesto. Las desigualdades válidas desempeñan un rol fundamental en el B&C, al ayudar a mejorar la formulación del problema, reducir el espacio de búsqueda, acelerar la convergencia y mejorar la calidad de las soluciones

obtenidas (Nemhauser y Wolsey, 1988). En general, son esenciales para abordar problemas de optimización combinatoria de manera eficiente y efectiva.

Las desigualdades válidas propuestas están basadas en versiones adaptadas para el LoRP que provienen de desigualdades de otro tipos de problemas como el LRP y *Distance Constrained Vehicle Routing Problem* (DCVRP).

Por tanto, primero se deben definir ciertas estructuras y conjuntos utilizados para definir dichas desigualdades válidas. Se define el conjunto  $X = \{x_{ij} > 0, \forall i, j \in V\}$  que contiene las aristas con valor mayor que cero de la solución factible entera actual de la relajación del modelo compacto de programación lineal entera mixta. Así, sea  $\bar{G}(X)$  el grafo encogido o “*shrunk graph*” con un sólo depósito que es un subgrafo de  $G$ . Esto permite utilizar desigualdades válidas para el *Capacitated Vehicle Routing Problem* (CVRP). Considerando la componente conexa  $S$ , se define el conjunto que contiene todas las aristas de la subruta, excepto la arista que entra y sale del depósito  $w$ , como  $D = \{(i, j) : x_{ij} = 1, \forall i, j \in S, i \neq j\}$ . Además,  $\hat{E}$  es el conjunto de aristas de corte definido por toda arista  $(i, j)$  y  $(j, i)$  tal que  $i \in S$  y  $j \in V \setminus S$ , donde  $S$  es una componente conexa del grafo  $G'$ .

### 3.6.1. *Rounded capacity inequality (RCI)*

La desigualdad utilizada para eliminar las subrutas con demanda mayor que la capacidad del vehículo, son establecidas por Lysgaard *et al.* (2004) en la inecuación (65).

$$\sum_{e \in \hat{E}} x_e \leq |S| - \left\lceil \frac{\sum_{i \in S} q_i}{Q} \right\rceil \quad (65)$$

La desigualdad (65) permite que la subruta que existe en  $S$ , se divida en las  $\left\lceil \frac{\sum_{i \in S} q_i}{Q} \right\rceil$  rutas necesarias para cubrir la demanda de  $S$ . Por tanto, la idea es que ciertos clientes de  $S$  se pueden unir a otros clientes que no estén en  $S$ .

### 3.6.2. *Rounded distance inequality (RDI)*

Para eliminar las subrutas que exceden la distancia máxima  $T$  de una ruta, se necesitan todas las aristas que contiene la subruta excepto por la arista que entra y sale del depósito  $w \in W$ . Así, se tiene lo siguiente:

$$\sum_{e \in \hat{E}} x_e \leq |S| - \left\lceil \frac{\sum_{(i,j) \in D} d_{ij}}{T} \right\rceil \quad (66)$$

### 3.6.3. Laporte's inequality (LI)

LI es utilizada para reforzar las inecuaciones (65) y (66). Esto implica que no se considera la distancia entre el depósito  $w \in W$  y algún cliente  $i \in S$ . En esencia esta es la diferencia con la formulación de Laporte *et al.* (1985). Así, la desigualdad válida para el LoRP está dada por lo siguiente:

$$\sum_{e \in \hat{E}} x_e \leq |S| - \max \left\{ \left\lceil \frac{\sum_{(i,j) \in D} d_{ij}}{T} \right\rceil, \left\lceil \frac{\sum_{i \in S} q_i}{Q} \right\rceil \right\} \quad (67)$$

Donde el segundo término del lado derecho de la desigualdad (67), implica que la subruta se debe eliminar, reduciendo el número de aristas en una posible subruta en  $S$ . Siendo ese número de aristas la cantidad de vehículos necesarios para satisfacer a esa subruta, ya sea por distancia o por demanda. En consecuencia, el objetivo es forzar algunos clientes de  $S$  a que dejen de pertenecer a esta componente y deba formar otras subrutas con otros clientes que no están en  $S$ .

### 3.6.4. Rounded distance subtour inequality (RDSI)

Si  $S$  es una componente conexa y si solo si existe una ruta en  $S$  que parte de la arista  $x_{w\hat{i}}$  y termina en  $x_{\hat{j}w}$ . Entonces, basta considerar la distancia del camino  $\{\hat{i} + 1, \dots, \hat{j} - 1\}$  y la distancia de la arista  $(j, i)$  para obtener una desigualdad válida debido a la desigualdad triangular en distancias euclidianas. Así, se tiene lo siguiente:

$$\sum_{e \in \hat{E}} x_e \leq |S| - \left\lceil \frac{d_{\hat{j}\hat{i}} + \sum_{(i,j) \in D} d_{ij}}{T} \right\rceil \quad (68)$$

### 3.6.5. Subtour elimination inequality (SEI)

Esta desigualdad es un caso especial de la restricción de eliminación de subrutas DFJ, ya que impide que se forme una subruta específica, es decir, la desigualdad va asociada a un conjunto específico de aristas que están activas en la componente conexa  $S$ . La subruta específica que se impide mediante esta desigualdad puede ser considerada en ambos sentidos, es decir, si parte de la arista  $x_{wi}$  y termina en  $x_{jw}$ , también se considera la ruta que termina en  $x_{iw}$  y parte en  $x_{wj}$ . Así, el conjunto  $\hat{D}$  es el que contiene las aristas de esa ruta, es decir, para un sentido es  $\hat{D}^+ = \{(i, j) : x_{ij} = 1, \forall i, j \in S \cup W, i \neq j\}$

$$\sum_{e \in \hat{D}^+} x_e \leq |S| - 1 \quad (69)$$

Luego, el otro sentido es  $\hat{D}^- = \{(j, i) : x_{ij} = 1, \forall i, j \in S \cup W, i \neq j\}$ . Por tanto, la desigualdad está dada por:

$$\sum_{e \in \hat{D}^-} x_e \leq |S| - 1 \quad (70)$$

### 3.6.6. Projection inequality (PI)

En [Letchford y Salazar-González \(2006\)](#) plantean una desigualdad válida para el DCVRP como un caso particular del *Vehicle Routing Problem with Time Windows* (VRPTW). Para el LoRP es similar a la desigualdad del DCVRP, pero debe ser válida para múltiples depósitos. Modificando la desigualdad, se tiene lo siguiente:

$$T \sum_{i \in S} \sum_{j \in V \setminus S} x_{ij} \geq \sum_{i \in S} \sum_{j \in V \setminus S} (\min_{w \in W} \{d_{wj}\} + d_{ij})(x_{ij} + x_{ji}) + \sum_{i \in S} \sum_{j \in S: i \neq j} d_{ij} x_{ij} \quad (71)$$

La desigualdad (71) establece que para cualquier  $S$ , la suma de la distancia máxima para los vehículos que salen de  $S$  debe ser mayor o igual a la suma de las distancias para los vehículos que entran a  $S$ , más la distancia que los vehículos pasan viajando dentro de  $S$ .

### 3.6.7. Depot degree inequality (DDI)

Dado que cualquier par de clientes  $i, j \in S$  pueden hacer una ruta, siempre es posible combinar dos viajes de ida y vuelta en sólo una ruta factible, si la suma de las demandas es menor a la capacidad del vehículo. La misma idea se puede aplicar a las rutas con más de un cliente, lo que conduce a la siguiente desigualdad válida planteada por [Belenguer et al. \(2011\)](#):

$$\sum_{j \in S \setminus \{w\}} x_{wj} + \sum_{j \in S \setminus \{w\}} x_{jw} + \sum_{i \in S} \sum_{j \in S: i \neq j} x_{ij} \leq 2y_w + |S| - 1 \quad \forall w \in W \quad (72)$$

Notar que si  $y_w = 0$  se cumple siempre, ya que la desigualdad (72) se convierte en una restricción DFJ (25). Por otra parte, si  $y_w = 1$ , el lado derecho de la desigualdad es  $|S| + 1$ , ya que a lo más se puede contener una ruta completa con los nodos  $S \cup \{w\}$ .

## 3.7. Branch and Cut

El modelo propuesto del LoRP en la Sección 3.1 y las desigualdades válidas descritas en la Sección 3.6 se han incorporado mediante planos de corte para producir un límite inferior válido o *Lower Bound* (LB) para el problema. El algoritmo de planos de corte se

implementa en el *Branch and Bound* (B&B) de Gurobi, mediante el uso de *callbacks* y *lazy constraints*. El proceso de planos de corte utilizado se basa en el propuesto por [Belenguer et al. \(2011\)](#). Inicialmente, se relaja el MR, removiendo las restricciones (37)-(42) y las restricciones de refuerzo (61)-(64). Luego, en cada iteración, se resuelve el MR actual, se buscan un conjunto de desigualdades válidas que sean violadas por la solución óptima del MR, se añaden al MR y se continúa el proceso como antes. Los planos de corte se ingresan en soluciones que son enteras. Por último, el algoritmo se detiene cuando se cumplen las condiciones del LoRP, es decir, que no existan subrutas entre clientes y las rutas asociadas a los depósitos abiertos no excedan la distancia máxima y/o la demanda del vehículo.

En la Tabla 4 se definen los algoritmos B&C en base a las desigualdades válidas utilizadas en cada uno. Cada columna corresponde a la abreviación de la desigualdad válida junto a las dos últimas columnas que tratan de la formulación GG para demanda y distancia, respectivamente. En el B&C-QT se consideran las restricciones del modelo con restricciones de reforzamiento (MR) sin la formulación GG para demanda y distancia de las rutas. En el B&C-Q considera el MR junto a la formulación GG de distancia de las rutas y no considera la formulación GG de la demanda. Por último, el B&C-T se define con el MR y las restricciones GG sobre la demanda sin considerar la formulación GG de las distancias de las rutas.

**Tabla 4:** Desigualdades válidas y formulación GG para cada B&C.

Algoritmo	RCI	RDI	LI	RDSI	SEI	PI	DDI	GG-Q	GG-T
B&C-QT	✓	✓	✓	✓	✓	✓	✓	×	×
B&C-Q	✓	×	×	×	✓	×	✓	×	✓
B&C-T	×	✓	×	✓	✓	✓	✓	✓	×

### 3.7.1. Solución inicial

Con el fin de obtener una mejor solución incumbente (es la solución que se ha encontrado hasta el momento y es utilizada como punto de referencia) y evitar explorar nodos del árbol de búsqueda del B&C con mayor valor de esa función objetivo de la mejor solución incumbente. Entonces, se propone una heurística constructiva para obtener una solución inicial del LoRP. Esta solución inicial es dada en el nodo raíz del B&C como solución entera y factible, donde, por un tiempo determinado para cada tipo de instancia según el rango de cobertura, se aplica la heurística por defecto de Gurobi que mejora la solución inicial propuesta buscando soluciones factibles de alta calidad antes de resolver el nodo raíz ([Gurobi, 2023](#)). Para los rangos de cobertura ( $R$ ) menor o igual que 10 el tiempo de ejecución de la heurística de Gurobi es de 5 segundos. Mientras que para los rangos de cobertura mayor o igual que 40 el tiempo es de 0,1 segundos. Finalmente, para  $R = 20$  el tiempo es de 3 segundos y con  $R = 30$ , se ejecuta por 1 segundo.

### 3.7.1.1. Heurística constructiva propuesta

La heurística propuesta en este estudio consiste en dos fases: determinar los depósitos que se abren y generar las rutas. La primera fase consiste en aplicar una heurística de cobertura de conjuntos, para obtener los depósitos abiertos y los conjuntos de nodos para realizar rutas de vehículos. Mientras, la segunda fase es para obtener las rutas asociadas a cada depósito abierto. Esta solución siempre es entera y factible para el MB y MR. Así, se presenta el procedimiento del algoritmo para obtener una solución inicial del LoRP:

1. Aplicar *clustering* mediante el algoritmo de K-Means (Jain *et al.*, 1999). El número de *clusters*,  $k$ , está definido por el número de vehículos necesarios para cubrir la demanda de los clientes. Esto implica que  $k = \left\lceil \frac{\sum_{i \in N} q_i}{Q} \right\rceil$ .
2. Asociar un depósito  $w$  a cada *cluster*. Para esto, se eliminan todos los depósitos asociados en cada *cluster*, excepto el más cercano al centroide del *cluster*. Si el *cluster* no tiene asociado a ningún depósito, se selecciona el depósito más cercano al centroide de ese *cluster*.
3. Obtener los conjuntos de clientes candidatos para formar rutas. Esto implica que si se abre el depósito  $w$ , los clientes de los *clusters* asociados a este depósito que son cubiertos por el rango de cobertura se eliminan de esos conjuntos.
4. Calcular  $\delta_w, \forall w \in W$ . Este coeficiente contempla el porcentaje de cobertura de clientes de  $w$ , el porcentaje de *clusters* asociados a  $w$  y la cercanía al centroide de los centroides de los *clusters*.
5. Ordenar  $\delta_w, \forall w \in W$  y seleccionar el primero. Así, este depósito es el primer depósito que se abre. Luego, se abren más depósitos si es que la distancia entre los depósitos asociados a un *cluster* es mayor que  $\frac{T}{2}$ . Con el fin de comparar si es beneficioso abrir otro depósito o no. Como mínimo se abren dos depósitos.
6. Como en el paso 3 hay *clusters* con depósitos que no se abrirán según el paso 6, entonces, los depósitos que se abren se asocian con los demás *clusters* que no tienen un depósito asociado que pertenezca al conjunto solución. Se eliminan los depósitos de los *clusters* que no serán abiertos. El criterio para asociar un depósito es la cercanía al centroide del *cluster*.
7. Este último paso de la fase 1 es para verificar factibilidad. Para realizar rutas de vehículos no se pueden considerar los clientes que no son cubiertos por el depósito asociado al *cluster*, pero sí por otro depósito abierto. Por tanto, se eliminan los clientes que son cubiertos por algún depósito abierto y no es considerado para una ruta.

8. Utilizar la heurística modificada de [Clarke y Wright \(1964\)](#) que considera la capacidad del vehículo y la distancia máxima de la ruta. Esto se aplica para cada conjunto de nodos por *cluster*, obteniendo las rutas asociadas a cada depósito abierto.

### 3.7.2. Algoritmo de separación

Los algoritmos de separación son una parte fundamental de las técnicas de B&C utilizadas en la resolución de modelos de programación lineal entera mixta. Estos algoritmos se utilizan para identificar desigualdades válidas que pueden agregar restricciones al modelo y ayudar a reducir el espacio de búsqueda, mejorando así la eficiencia de la resolución ([Mitchell, 2009](#)). En el Algoritmo 1, se presenta el pseudocódigo del algoritmo de separación junto con la descripción de las líneas principales del algoritmo.

Para añadir las desigualdades válidas como *lazy constraints* mediante *callbacks* dentro del B&C, se necesita una solución entera del modelo relajado en el nodo actual. En la línea 4, dicha solución se transforma en el *shrunk graph*  $\bar{G}(X) = (\bar{E}, \bar{V})$ , donde  $\bar{V} = N \cup \{0\}$ . Posteriormente, en la línea 6, comienza un ciclo para toda componente conexa  $S$  que exista en el  $\bar{G}(X)$ . Entre las líneas 7-8 se calculan los valores necesarios para realizar cortes. En las líneas 9-13, se agrega la desigualdad DDI siempre y cuando  $S$  no esté conectada a un depósito y que cumpla con las condiciones de toda ruta. Esto permite que se enlace esa ruta válida al depósito más cercano. Luego, si el depósito no está en  $S$ , en las líneas 14-16 se agrega una de las desigualdades RCI, RDI o LI, dependiendo de la versión de B&C escogido.

Cuando el depósito está en  $S$ , en la línea 17, si la demanda cubierta en  $S$  es mayor que  $Q$ , entonces, se agrega la desigualdad RCI antes que cualquier desigualdad de distancia en la línea 19. Caso contrario, se calculan valores necesarios para realizar cortes asociados a la distancia de la componente conexa (líneas 21-23). En consecuencia, si la distancia entre el primer cliente que se visita y el último es mayor que la distancia máxima, entonces, se agrega la desigualdad RDI. Caso contrario, es menor y además, si la distancia en  $S^+$  versus la distancia máxima  $T$  es mayor (línea 26). Entonces, en la línea 27, se obtienen los clientes que se visitan primero y último para obtener la distancia de toda la ruta. Luego, se verifica que esa componente conexa  $S$  no haga esa ruta con ningún depósito, agregando la desigualdad RDSI. Caso contrario, se agregan las desigualdades SEI y PI (línea 32).

El orden de inserción de los cortes acelera la poda de nodos del árbol de búsqueda del B&C, ya que si la demanda de  $S$  excede a  $Q$ , entonces, esa ruta ya es infactible para el problema sin importar la distancia recorrida en  $S$ . Del mismo modo, si la distancia del camino entre el primer cliente y el último es mayor que la distancia máxima, entonces, ese camino no es parte de la solución óptima. Por tanto, no es necesario realizar cortes que contemplen a  $S^+$ .

**Algoritmo 1** Algoritmo de separación**Entrada:** alg (algoritmo B&C),  $X$  (Solución entera)**Salida:**  $\alpha$ 


---

```

1:  $\bar{E} \leftarrow \text{obtener-aristas}(X)$ 
2:  $\hat{E} \leftarrow \text{obtener-aristas-deps}(X, \bar{E})$ 
3:  $\bar{w} \leftarrow 0, \alpha \leftarrow \emptyset$ 
4:  $\bar{G}(X) \leftarrow \text{obtener-grafo-encogido}(X, \bar{w})$ 
5:  $G(X) \leftarrow \text{obtener-grafo}(X)$ 
6: for  $S$  en Componentes( $G(X)$ ) do
7:    $T(S) \leftarrow \sum_{i \in S} \sum_{j \in S: i \neq j} c_{ij} x_{ij}$  ;  $Q(S) \leftarrow \sum_{i \in S} q_i$ 
8:    $w(S) \leftarrow \left\lceil \frac{Q(S)}{Q} \right\rceil$  ;  $l(S) \leftarrow \left\lceil \frac{T(S)}{T} \right\rceil$  ;  $k(S) \leftarrow \max\{l(S), w(S)\}$ 
9:   if  $|S| \geq 2$  and  $\bar{w} \notin S$  and  $Q(S) < Q$  and  $T(S) < T$  then
10:     for  $w \in W$  do
11:        $\alpha \leftarrow \alpha \cup \text{DDI}(S, w)$ 
12:     end for
13:   end if
14:   if  $|S| \geq 2$  and  $\bar{w} \notin S$  then
15:     if alg  $\in \{\text{B\&C-Q}, \text{B\&C-QT}\}$  then  $\alpha \leftarrow \alpha \cup \text{RCI}(S, w(S))$ 
16:     if alg  $\in \{\text{B\&C-T}, \text{B\&C-QT}\}$  then  $\alpha \leftarrow \alpha \cup \text{RDI}(S, l(S))$ 
17:     if alg  $\in \{\text{B\&C-QT}\}$  then  $\alpha \leftarrow \alpha \cup \text{LI}(S, k(S))$ 
18:   end if
19:   if  $|S| \geq 2$  and  $\bar{w} \in S$  then
20:     if  $Q(S) > Q$  then
21:        $\alpha \leftarrow \alpha \cup \text{RCI}(S, w(S))$ 
22:     else
23:        $w^* \leftarrow \text{obtener-deposito}(S, G(X))$ 
24:        $S^+ \leftarrow S \cup \{w^*\}$ 
25:        $T(S^+) \leftarrow \sum_{i \in S^+} \sum_{j \in S^+: i \neq j} c_{ij} x_{ij}$  ;  $l(S^+) \leftarrow \left\lceil \frac{T(S^+)}{T} \right\rceil$ 
26:       if  $T(S) > T$  then
27:          $\alpha \leftarrow \alpha \cup \text{RDI}(S, l(S))$ 
28:       else if  $T(S^+) > T$  then
29:          $\hat{i}, \hat{j} \leftarrow \text{obtener-arista-cierre}(S, \hat{E})$ 
30:         if  $T(S) + d_{\hat{j}, \hat{i}} > T$  then
31:            $\hat{l}(S) \leftarrow \left\lceil \frac{T(S) + d_{\hat{j}, \hat{i}}}{T} \right\rceil$ 
32:            $\alpha \leftarrow \alpha \cup \text{RDSI}(S, \hat{l}(S))$ 
33:         else
34:            $\alpha \leftarrow \alpha \cup \text{SEI}(S) \cup \text{PI}(S)$ 
35:         end if
36:       end if
37:     end if
38:   end if
39: end for

```

---

# Capítulo 4

## Resultados y Análisis

Este capítulo presenta los resultados computacionales de los modelos y de los diferentes B&C propuestos. Además, se presentan las instancias utilizadas en los experimentos.

### 4.1. Configuración de la experimentación

Los modelos y algoritmos de B&C son implementados mediante Python 3.9.17, donde los experimentos son realizados en un procesador Intel(R) Core(TM) i7-3770 CPU @ 3.40GHz de 8 núcleos y con 8 GB de RAM, bajo el sistema operativo Ubuntu 22.04.2 LTS. Además, los experimentos son ejecutados en un hilo y de manera secuencial con la condición de término de 3600 segundos para todas las instancias utilizadas.

### 4.2. Instancias de prueba

Los experimentos computacionales son ejecutados en 12 instancias introducidas por [Akca et al. \(2009\)](#). Estas constan de seis instancias con 30 clientes y otras seis con 40 clientes, donde todas tienen cinco depósitos candidatos. Las instancias son modificadas por [Arslan \(2021\)](#), añadiéndoles el rango de cobertura  $R \in \{0, 10, 20, 30, 40, 50\}$  y la distancia máxima  $T \in \{100, 125, 150\}$ , ya que fueron definidos para el LRP. Así, las instancias son definidas según el número de clientes ( $|N|$ ), número de depósitos ( $|W|$ ), rango de cobertura  $R$  y la distancia máxima  $T$ . Por ejemplo, la instancia denominada ‘r30x5a-2-0-100’, hace referencia a una instancia de 30 clientes, con cinco depósitos, con rango de cobertura 10 y distancia máxima 100. Dada la combinación de  $R$  y  $T$ , en total se tienen 216 instancias del LoRP, donde 192 son factibles para el problema. En la Tabla 5 se presenta un desglose sobre las instancias factibles con respecto a cada  $R$  y  $T$  versus las instancias totales. Específicamente, las instancias infactibles para el problema son ‘r30x5b-3’ y ‘r40x5b-2’ con  $T = 100$ , junto a ‘r40x5b-1’ con  $T \in \{100, 125\}$ . Las instancias consideran un escalamiento

de las distancias de las aristas, de modo que la arista entre los dos nodos más lejanos sea igual a 100. Este procedimiento se realiza con una regla de tres clásica.

**Tabla 5:** Desglose sobre las instancias de *Akca et al. (2009)*.

T	R						Total
	0	10	20	30	40	50	
100	9/12	9/12	9/12	9/12	9/12	9/12	54/72
125	11/12	11/12	11/12	11/12	11/12	11/12	66/72
150	12/12	12/12	12/12	12/12	12/12	12/12	72/72
<b>Total</b>	32/36	32/36	32/36	32/36	32/36	32/36	192/216

### 4.3. Resultados computacionales

Los modelos y algoritmos propuestos en el Capítulo 3 son validados en las 192 instancias, considerando un tiempo máximo de ejecución de 3600 segundos como condición de término. En la Tabla 6 y 7 se presentan los resultados promedios y en la Sección 4.3.4 se presentan los resultados promedios de los algoritmos y modelos propuestos con respecto al rango de cobertura. Además, en la Sección 4.3.5 se presentan las instancias específicas en las que se obtienen una nueva mejor solución.

#### 4.3.1. MB vs MR

En la Tabla 6 se presentan los resultados de MB y MR, mostrando los promedios de la cota superior (UB: upper bound), cota inferior (LB: lower bound), gap (es la brecha porcentual entre el LB y UB encontrado al final del proceso), el tiempo computacional (expresado en segundos) que tomó el modelo respectivo en encontrar la mejor solución y la cantidad de soluciones óptimas encontradas. Adicionalmente, la última columna señala la cantidad de nodos explorados

**Tabla 6:** Comparación resultados promedios de MB y MR.

Modelo	UB	LB	Gap (%)	Tiempo [s]	# Opt.	Nodos expl.
MB	574,80	556,39	2,26	1430,24	123/192	49809,05
MR	574,32	564,10	1,28	1108,99	141/192	47408,18

A pesar de que el UB del MB es cercano al del MR, con una diferencia menor a 0,5 unidades, el LB toma mayor distancia con aproximadamente ocho unidades de costo promedio. Esto conlleva a que el MR tenga un punto porcentual menos de gap promedio, siendo un 1,28% versus el 2,26% del MB. Adicionalmente, en tiempos de ejecución, el MR sigue siendo superior al MB en un 22,5%. Debido a esto, el MR también recorre menos nodos del árbol durante el proceso de B&C. El modelo MR explora, en promedio,

47408,18 nodos, mientras que el MB explora 49809,05 nodos. Finalmente, el MR supera al MB en cantidad de soluciones óptimas encontradas por 18 unidades. Así, las restricciones adicionales de refuerzo cumplen el objetivo de disminuir los resultados promedios.

### 4.3.2. MR vs B&C's

En la Tabla 7 se presentan los resultados de MR comparados con el B&C-Q, B&C-QT y B&C-T. Esta tabla muestra los promedios de la cota superior (UB), cota inferior (LB), gap, el tiempo computacional (expresados en segundos) al encontrar la mejor solución, la cantidad de nodos explorados por algoritmo respectivo. Además, la última columna muestra la cantidad de soluciones óptimas encontradas.

**Tabla 7:** Comparación promedios entre MR y B&C's.

Modelo	UB	LB	Gap (%)	Tiempo [s]	# Opt.	Nodos expl.
B&C-Q	579,14	557,51	2,59	1172,56	135/192	103179,74
B&C-QT	600,55	551,89	4,80	1444,76	119/192	100372,71
B&C-T	576,22	551,50	3,02	1535,28	117/192	124692,63
MR	574,32	564,10	1,28	1108,99	141/192	47408,18

El B&C-T obtiene la mejor cota superior de los tres algoritmos propuestos, es decir, encuentra las mejores soluciones enteras. Sin embargo, cuando se trata de demostrar la optimalidad, su cota inferior es superada por los otros dos B&C. Además, es el algoritmo con menor cantidad de soluciones óptimas encontradas y tiene una mayor cantidad de nodos explorados en promedio. A pesar de esto, su gap es más bajo que el B&C-QT, pero mayor que el B&C-Q. Este último supera a los otros dos B&C en cota inferior, gap, tiempo de ejecución y cantidad de soluciones óptimas. Sin embargo, está a tres unidades de la mejor cota superior de los tres algoritmos y su cantidad de nodos explorados es levemente superior al B&C-QT. Por tanto, en general, el B&C-Q es el algoritmo de mejor rendimiento. Finalmente, el B&C-Q es superado por el MR en todos los resultados obtenidos, siendo la mayor diferencia en la cota superior e inferior, gap y cantidad de nodos explorados. Cabe destacar que el B&C-QT es el algoritmo que en promedio recorre una menor cantidad de nodos, que es la idea principal en un algoritmo B&C. Sin embargo, pareciera que esta menor medida no se compara con las restricciones adicionales de cortes, que lo lleva a resolver más relajaciones en cada nodo del B&C.

### 4.3.3. MR vs B&P

En la Tabla 8 se presentan los resultados de MR comparado con el B&P de [Arslan \(2021\)](#). Esta tabla muestra los promedios de la cota superior (UB), cota inferior (LB), gap

y el tiempo computacional (en segundos) al encontrar la mejor solución. Además, la última columna muestra la cantidad de soluciones óptimas encontradas.

**Tabla 8:** Comparación promedios entre MR y B&P.

Modelo	UB	LB	Gap (%)	Tiempo [s]	# Opt.
B&P	573,84	573,74	0,01	344,36	189/192
MR	574,32	564,10	1,28	1108,99	141/192

El MR tiene una cota superior muy cercana al B&P, teniendo una diferencia de 1,5 unidades aproximadamente. Sin embargo, en la cota superior si existe una diferencia mayor de más de nueve unidades. Esto se ve reflejado en el gap promedio de cada uno, ya que el gap del MR es 1,28 %, mientras el del B&P es 0,01 %. Otro resultado donde el B&P domina por sobre el MR es en el tiempo computacional promedio que lo supera entre tres y cuatro veces su valor. Por último, producto del gap promedio del MR, la diferencia de la cantidad de óptimos asciende a 48 versus el B&P.

#### 4.3.4. Análisis respecto al rango de cobertura

Para realizar un análisis con respecto al rango de cobertura, se agregan los resultados en dos tablas. En la Tabla 9 se presentan los MR y MB junto a los algoritmos B&C-QT, B&C-Q y B&C-T. En las columnas están los promedios de la cota superior (UB), cota inferior (LB), gap y el tiempo computacional que tomó el modelo respectivo en encontrar la mejor solución, considerando los rangos de cobertura 0, 10 y 20.

**Tabla 9:** Comparación de los promedios con  $R \in \{0, 10, 20\}$ .

Algoritmo	UB	LB	Gap (%)	Tiempo [s]	# Opt.	Nodos expl.
B&C-QT	805,47	708,16	9,61	2852,37	23/96	190048,15
B&C-Q	762,59	719,31	5,17	2335,76	39/96	205848,27
B&C-T	756,81	708,20	5,91	2896,57	24/96	226423,14
MB	753,90	717,09	4,52	2832,48	27/96	98156,32
MR	752,94	732,51	2,55	2198,46	45/96	94164,05
B&P	751,97	751,79	0,02	685,47	93/96	-

En la Tabla 10, también se presentan los promedios de la cota superior (UB), cota inferior (LB), gap y el tiempo de ejecución en encontrar la mejor solución, pero considerando el resto de los rangos posibles, es decir, 30, 40 y 50. Esta comparación se debe a que existe una amplia diferencia entre estos dos intervalos de rangos de cobertura. Cabe destacar que los resultados desagregados por  $R$  están presentes en la Tabla A.12, A.13, A.14, A.15, A.16 y A.17.

En la Tabla 9, se observa que el número de soluciones óptimas de los algoritmos propuestos no supera el 50 %, siendo MR el de menor gap, tiempos de ejecución, nodos

explorados y cantidad de soluciones óptimas. Luego, el MB es segundo con menor gap promedio a pesar de no ser el segundo con más cantidad de soluciones óptimas, siendo superando por el B&C-Q. Sin embargo, el B&C-Q es el de mejor rendimiento en tiempo computacional promedio, gap promedio y cantidad de soluciones óptimas.

**Tabla 10:** Comparación de los promedios con  $R \in \{30, 40, 50\}$ .

Algoritmo	UB	LB	Gap (%)	Tiempo [s]	# Opt.	Nodos expl.
B&C-QT	395,70	395,70	0,00	37,15	96/96	10697,27
B&C-Q	395,70	395,70	0,00	9,36	96/96	511,21
B&C-T	395,62	394,80	0,13	174,00	93/96	22962,11
MB	395,70	395,70	0,00	28,00	96/96	1461,78
MR	395,70	395,70	0,00	19,52	96/96	652,30
B&P	395,70	395,70	0,00	3,26	96/96	-

En la Tabla 10, se observa que prácticamente todos los algoritmos propuestos, excepto el B&C-T, obtienen la solución óptima de las 96 instancias. Por tanto, existe un mejor rendimiento que con los rangos de cobertura 0, 10 y 20. Sin embargo, la mayor diferencia entre los algoritmos es el tiempo promedio de ejecución junto a los nodos explorados. El B&C-Q es el que más se acerca al B&P con un promedio de 9,36 segundos, siendo el algoritmo con menor número de nodos explorados. Luego, el MR es el segundo más cercano al B&P con 19,52 segundos promedio de ejecución. Tras esto, los peores rendimientos en tiempos de ejecución son el B&C-T, B&C-QT y el MB, respectivamente.

#### 4.3.5. Instancias con nueva mejor solución

Un hallazgo importante dentro de los resultados son presentados en la Tabla 11. La primera columna señala la instancia de Akca ejecutada, mientras que la segunda y tercera columna señalan la distancia máxima junto al rango de cobertura asociado a los depósitos. Luego, hay dos columnas principales que son el B&P y el MR. Donde, cabe destacar que todos los modelos restantes propuestos encontraron el mismo valor óptimo que MR en las instancias presentadas y sólo se eligió como punto de comparación de los nuevos valores óptimos. Cada versión presenta la cota superior (UB), cota inferior (LB), gap y el tiempo de ejecución.

**Tabla 11:** Instancias con óptimos distintos de Arslan (2021).

Instancia	T	R	B&P		MR/MB		Dif
			UB	LB	UB	LB	
r30x5a-2	150	20	692,07	692,07	691,99	691,99	<b>0,08</b>
r40x5a-1	100	20	703,99	703,99	703,24	703,24	<b>0,75</b>
r40x5a-2	150	20	619,11	619,11	618,16	618,16	<b>0,95</b>
r40x5a-3	100	30	625,39	625,39	625,36	625,36	<b>0,03</b>
r40x5b-3	150	20	720,04	720,04	717,11	717,11	<b>2,93</b>

En la Tabla 11 se presentan cinco instancias donde se encontró un nuevo óptimo menor que los presentados por Arslan (2021). Esto se debe a que el autor utiliza dos parámetros en el *pulse algorithm* que permite resolver el problema de la ruta elemental más corta o *elementary shortest path* en inglés (Lozano *et al.*, 2016). El primer parámetro es la cantidad de pasos y el segundo es el límite de tiempo inferior. Estos valores pueden conducir a no encontrar las rutas óptimas en el *pricing problem* (para identificar las variables de ruta con costos reducidos negativos). Si la cantidad de pasos y el límite de tiempo inferior se establecieran en un valor más bajo, entonces podría encontrar esas rutas, pero eso conlleva un tiempo de ejecución más alto. Por tanto, el autor utilizó los valores de los parámetros declarados en Lozano *et al.* (2016). Así, en Arslan (2021) los resultados provienen bajo esa configuración del *pulse algorithm*, es decir, el B&P obtiene el óptimo siempre y cuando el *pulse algorithm* proporcione la solución óptima del *elementary shortest path*.

# Capítulo 5

## Conclusiones

Esta memoria de título presenta un modelo compacto de programación lineal entera mixta, restricciones adicionales de reforzamiento para el modelo y diversos B&C diseñados para resolver el LoRP, empleando distintos planos de corte con desigualdades válidas. Además, se lleva a cabo un análisis exhaustivo de la literatura asociada al LoRP, poniendo especial atención en las técnicas más avanzadas que abordan esta cuestión.

Los modelos garantizan optimalidad pero de manera muy lenta en instancias con rango de cobertura menor o igual a 20. Adicionalmente, mientras mayor es el valor de  $T$ , el problema tiene menor dificultad, generando que los tiempos de ejecución disminuyan. Sin embargo, cuando el rango de cobertura es mayor o igual a 30, el modelo con las restricciones de refuerzo está cerca de los resultados del estado del arte, sobre todo en las instancias con  $R = 40$  y  $R = 50$ . Se exponen los hallazgos computacionales obtenidos al ejecutar 192 instancias previamente utilizados en el estado del arte.

En promedio, el modelo con restricciones de refuerzo fue el de mejor desempeño entre los algoritmos y modelos propuestos. Además, el B&C-Q fue el mejor B&C propuesto, siendo el segundo entre los algoritmos y modelos propuestos. Cabe destacar que los modelos tienen mejor rendimientos mientras más grande es el  $T$  y mientras mayor sea el rango de cobertura. A pesar de esto, los algoritmos presentan un potencial de mejora en la búsqueda de las rutas óptimas, ya que reducen el número de nodos explorados y posibles mejoras podrían llegar a competir con el algoritmo del estado del arte. Los modelos y algoritmos no compiten en tiempos de ejecución con respecto al B&P cuando el rango de cobertura es 0, 10 y 20, pero sí se acercan más en los rangos 30, 40 y 50. Por último, todos los objetivos de este trabajo, general y específicos, fueron logrados satisfactoriamente.

Existen diversas aristas que se pueden abordar. En primer lugar, implementar el B&C en un lenguaje de programación compilado como C++ con el fin de obtener mejor rendimiento al ser un lenguaje de programación de más bajo nivel que Python, que además es un lenguaje interpretado. Lo segundo que se puede abordar es mejorar las

desigualdades que están asociadas a la distancia de la ruta para disminuir la cantidad de nodos explorados. Además, se pueden implementar estrategias de separación y ramificación. Estas aristas pueden producir una mejora sustancial en el desempeño del B&C. Luego, otra mejora es utilizar cortes fraccionarios dentro del algoritmo de separación, siendo una gran alternativa utilizar la librería CVRPSEP implementada en C++ (Lysgaard, 2003). En esta librería están implementadas las estrategias de separación para las desigualdades de capacidad, entre las cuales se encuentran las desigualdades *homogeneous multistar*, *generalized large multistar*, *framed capacity*, *strengthened comb* e *hypotour* que son válidas para el CVRP. Adicionalmente, CVRPSEP contiene un procedimiento para generar conjuntos de candidatos para realizar la ramificación (Lysgaard, 2003). Finalmente, se pueden desarrollar otro tipo de *solvers* como metaheurísticas o matheurísticas que disminuyan los tiempos de ejecución, obteniendo una buena solución factible. Sin embargo, se comprometería la optimalidad de esas soluciones.

# Bibliografía

- Akca, Z., Berger, R., y Ralphs, T. (2009). A branch-and-price algorithm for combined location and routing problems under capacity restrictions. En *Operations research and cyber-infrastructure*, pp. 309–330. Springer.
- Arslan, O. (2021). The location-or-routing problem. *Transportation Research Part B: Methodological*, 147:1–21.
- Baldacci, R., Mingozzi, A., y Calvo, R. W. (2011). An exact method for the capacitated location-routing problem. *Operations Research*, 59(5):1284–1296.
- Belenguer, J.-M., Benavent, E., Prins, C., Prodhon, C., y Wolfler Calvo, R. (2011). A branch-and-cut method for the capacitated location-routing problem. *Computers Operations Research*, 38(6):931–941.
- Burger, M., Su, Z., y De Schutter, B. (2018). A node current-based 2-index formulation for the fixed-destination multi-depot travelling salesman problem. *European Journal of Operational Research*, 265(2):463–477.
- Clarke, G. y Wright, J. W. (1964). Scheduling of vehicles from a central depot to a number of delivery points. *Operations Research*, 12(4):568–581.
- Dantzig, G., Fulkerson, R., y Johnson, S. (1954). Solution of a large-scale traveling-salesman problem. *Journal of the Operations Research Society of America*, 2(4):393–410.
- De la Fuente, R., Aguayo, M. M., y Contreras-Bolton, C. (2023). An optimization-based approach for an integrated forest fire monitoring system with multiple technologies and surveillance drones. *European Journal of Operational Research*.
- Demir, E., Syntetos, A., y van Woensel, T. (2022). Last mile logistics: Research trends and needs. *IMA Journal of Management Mathematics*, 33(4):549–561.
- Gavish, B. y Graves, S. C. (1978). The travelling salesman problem and related problems. *Operations Research Center Working Paper; OR 078-78*.
- Gurobi (2023). Gurobi Optimizer Reference Manual.
- Haghi, M., Arslan, O., y Laporte, G. (2023). A location-or-routing problem with partial and decaying coverage. *Computers & Operations Research*, 149:106041.
- Jain, A. K., Murty, M. N., y Flynn, P. J. (1999). Data clustering: a review. *ACM computing surveys (CSUR)*, 31(3):264–323.
- Laporte, G., Nobert, Y., y Desrochers, M. (1985). Optimal routing under capacity and distance restrictions. *Operations Research*, 33(5):1050–1073.

- Letchford, A. N. y Salazar-González, J.-J. (2006). Projection results for vehicle routing. *Mathematical Programming*, 105:251–274.
- Lozano, L., Duque, D., y Medaglia, A. L. (2016). An exact algorithm for the elementary shortest path problem with resource constraints. *Transportation Science*, 50(1):348–357.
- Lysgaard, J. (2003). *CVRPSEP: A Package of Separation Routines for the Capacitated Vehicle Routing Problem*. Research reports from Department of Management Science. Aarhus School of Business.
- Lysgaard, J., Letchford, A. N., y Eglese, R. W. (2004). A new branch-and-cut algorithm for the capacitated vehicle routing problem. *Mathematical programming*, 100:423–445.
- Miller, C. E., Tucker, A. W., y Zemlin, R. A. (1960). Integer programming formulation of traveling salesman problems. *Journal of the Association for Computing Machinery*, 7(4):326–329.
- Mitchell, J. E. (2009). *Integer programming: branch and cut algorithms* *Integer Programming: Branch and Cut Algorithms*, pp. 1643–1650. Springer US, Boston, MA.
- Nemhauser, G. y Wolsey, L. (1988). *The Theory of Valid Inequalities*, capítulo II.1, pp. 203–258. John Wiley Sons, Ltd.

# Anexo

## Resultados promedios desagregados

En este anexo se presentan, de manera desagregada, todos los resultados promedios obtenidos durante la ejecución de los experimentos computacionales. La Tabla A.12 presenta los resultados promedios del B&P de Arslan (2021), considerando la cota superior (UB), cota inferior (LB), Gap y el tiempo computacional promedio. En la Tabla A.13, A.14, A.15, A.16 y A.17 se observan los resultados promedios del algoritmo respectivo propuesto con respecto a al rango de cobertura  $R$ . Estos consideran la cota superior (UB), cota inferior (LB), Gap, tiempo de ejecución y los nodos explorados.

**Tabla A.12:** Resultados promedios de B&P con respecto a  $R$ .

<b>R</b>	<b>UB</b>	<b>LB</b>	<b>Gap (%)</b>	<b>Tiempo [s]</b>
0	786,67	786,42	0,03	1200,01
10	776,99	776,69	0,04	824,52
20	692,26	692,26	0,00	31,87
30	547,88	547,88	0,00	8,58
40	398,68	398,68	0,00	0,72
50	240,53	240,53	0,00	0,47
<b>Prom.</b>	<b>573,84</b>	<b>573,74</b>	<b>0,01</b>	<b>344,36</b>

**Tabla A.13:** Resultados promedios de B&C-Q con respecto a  $R$ .

<b>R</b>	<b>UB</b>	<b>LB</b>	<b>Gap (%)</b>	<b>Tiempo [s]</b>	<b>Nodos expl.</b>
0	803,77	734,08	8,23	3194,27	265831,88
10	791,90	737,66	6,48	2802,09	246353,69
20	692,09	686,20	0,80	1010,92	105359,25
30	547,88	547,88	0,00	24,32	1427,41
40	398,68	398,68	0,00	2,76	105,22
50	240,53	240,53	0,00	0,99	1,00
<b>Prom.</b>	<b>579,14</b>	<b>557,51</b>	<b>2,59</b>	<b>1172,56</b>	<b>103179,74</b>

**Tabla A.14:** Resultados promedios de B&C-QT con respecto a  $R$ .

<b>R</b>	<b>UB</b>	<b>LB</b>	<b>Gap (%)</b>	<b>Tiempo [s]</b>	<b>Nodos expl.</b>
0	847,75	723,63	13,79	3500,50	163774,31
10	874,13	728,14	12,23	3014,51	177165,69
20	694,51	672,71	2,80	2042,11	229204,44
30	547,89	547,89	0,00	104,65	30545,63
40	398,46	398,46	0,00	6,16	1542,97
50	240,53	240,53	0,00	0,65	3,22
<b>Prom.</b>	<b>600,55</b>	<b>551,89</b>	<b>4,80</b>	<b>1444,76</b>	<b>100372,71</b>

**Tabla A.15:** Resultados promedios de B&C-T con respecto a  $R$ .

<b>R</b>	<b>UB</b>	<b>LB</b>	<b>Gap (%)</b>	<b>Tiempo [s]</b>	<b>Nodos expl.</b>
0	794,09	729,81	7,70	3387,59	239722,22
10	784,10	727,28	6,82	3332,79	248481,13
20	692,25	667,51	3,21	1969,33	191066,06
30	547,88	545,41	0,38	508,02	67076,91
40	398,46	398,46	0,00	12,94	1806,19
50	240,53	240,53	0,00	1,03	3,25
<b>Prom.</b>	<b>576,22</b>	<b>551,50</b>	<b>3,02</b>	<b>1535,28</b>	<b>124692,63</b>

**Tabla A.16:** Resultados promedios de MB con respecto a  $R$ .

<b>R</b>	<b>UB</b>	<b>LB</b>	<b>Gap (%)</b>	<b>Tiempo [s]</b>	<b>Nodos expl.</b>
0	790,53	739,68	6,16	3390,44	91550,28
10	778,94	732,94	5,63	3336,10	114495,69
20	692,24	678,66	1,76	1770,89	88423,00
30	547,88	547,88	0,00	75,08	4229,56
40	398,68	398,68	0,00	7,27	154,78
50	240,53	240,53	0,00	1,63	1,00
<b>Prom.</b>	<b>574,80</b>	<b>556,39</b>	<b>2,26</b>	<b>1430,24</b>	<b>49809,05</b>

**Tabla A.17:** Resultados promedios de MR con respecto a  $R$ .

<b>R</b>	<b>UB</b>	<b>LB</b>	<b>Gap (%)</b>	<b>Tiempo [s]</b>	<b>Nodos expl.</b>
0	788,22	758,19	3,71	2990,18	125394,44
10	778,46	752,27	3,25	2663,87	119061,94
20	692,14	687,07	0,70	941,32	38035,78
30	547,88	547,88	0,00	52,71	1882,22
40	398,68	398,68	0,00	4,61	73,75
50	240,53	240,53	0,00	1,25	0,94
<b>Prom.</b>	<b>574,32</b>	<b>564,10</b>	<b>1,28</b>	<b>1108,99</b>	<b>47408,18</b>

**UNIVERSIDAD DE CONCEPCION – FACULTAD DE INGENIERIA  
RESUMEN DE MEMORIA DE TITULO**

Departamento de Ingeniería		Industrial	
Título		Un modelo compacto de programación lineal entera mixta para el <i>Location-or-Routing Problem</i>	
Nombre Memorista		Matías Sebastián Barrientos Zagal	
Modalidad		Profesor(es) Patrocinante	
Concepto		Carlos Contreras-Bolton	
Calificación			
Fecha	25/09/2023	Ingeniero Supervisor	Institución
Comisión (Nombre y Firma)			
Resumen			
<p>El <i>Location-or-Routing Problem</i> (LoRP) es una generalización del problema de localización y ruteo de vehículos. El objetivo del LRP es minimizar el costo de abrir un subconjunto de depósitos candidatos y el costo de las rutas realizadas desde estos depósitos, con el fin de cubrir a todos los clientes mediante rutas. En el LoRP, la cobertura se puede realizar con un depósito o con una ruta. Así, la función objetivo es la suma del costo de abrir un subconjunto de depósitos candidatos, el costo de las rutas realizadas desde estos depósitos y el costo de cubrir a los clientes con cierto rango de cobertura. La investigación de este problema tiene una gran relevancia, ya que se sitúa en la última milla y es uno de los principales costos dentro de una empresa. Por tanto, este trabajo propone un modelo compacto de programación lineal entera mixta para el LoRP y también, restricciones de refuerzo para mejorar el modelo propuesto. Además, se presenta un <i>Branch &amp; Cut</i> (B&amp;C) que considera desigualdades válidas relacionadas con las subrutas, basadas en la capacidad y la distancia máxima de los vehículos. El rendimiento del modelo y algoritmo propuesto es validado sobre instancias de la literatura. Los resultados muestran que el modelo con restricciones de refuerzo obtiene el mejor rendimiento promedio entre los algoritmos y modelos propuestos. Finalmente, el B&amp;C que sólo considera las restricciones de distancia de subrutas y desigualdades válidas de demanda, obtiene los mejores resultados entre los algoritmos propuestos en promedio.</p>			

