

UNIVERSIDAD DE CONCEPCIÓN-CHILE
FACULTAD DE INGENIERÍA
DEPARTAMENTO DE INGENIERÍA INDUSTRIAL

**Un modelo y método metaheurístico, considerando
elementos estocásticos para la recolección de
residuos electrónicos en ciudades**

por

Raúl Alberto Raggio Monte

Profesor Guía:

Lorena Pradenas Rojas

Concepción, octubre de 2021

Tesis presentada a la

DIRECCIÓN DE POSTGRADO
UNIVERSIDAD DE CONCEPCIÓN



Para optar al grado de
MAGÍSTER EN INGENIERÍA INDUSTRIAL

© 2021 Raúl Alberto Raggio Monte

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



RESUMEN

Un modelo y método metaheurístico, considerando elementos estocásticos para la recolección de residuos electrónicos en ciudades

Raúl Raggio Monte
Octubre 2021

Profesor Guía: Lorena Pradenas Rojas
Programa: Magíster en Ingeniería Industrial

En el presente estudio se propone un modelo de optimización y métodos metaheurísticos para la recolección de residuos electrónicos en ciudades. El objetivo es minimizar el costo total de recolección y el costo de emisiones producidas por los vehículos. Se considera que la cantidad de residuos electrónicos en cada punto de recolección es aleatoria. Además, el tiempo de visita a cada nodo es restringido a través de ventanas de tiempo. El modelo de optimización estocástico es formulado a través de la metodología *chance-constrained programming*. Por otro lado, se propone un *Iterated Local Search* y *Simulated Annealing*, tratando la aleatoriedad mediante dos enfoques. En el primero se verifica la capacidad de los vehículos mediante el conjunto de restricciones de capacidad derivadas de la metodología *chance-constrained programming*. En el segundo enfoque, se verifica la capacidad de los vehículos mediante muestreo aleatorio. Para comparar los métodos, se utilizan instancias de la literatura de distinto tamaño adaptadas a la realidad de Chile. Con una significancia del 5%, el modelo exacto y las metaheurísticas encontraron soluciones estadísticamente iguales, aunque las soluciones de las metaheurísticas son en promedio mejor que las de CPLEX. Por otro lado, los costos totales aumentan al considerar mayores probabilidades de recolección, debido a que el problema se vuelve más complejo de resolver.

Palabras clave del estudio: Residuos electrónicos, optimización estocástica, ruteo de vehículos capacitados con ventanas de tiempo, metaheurísticas, *Iterated Local Search*, *Simulated Annealing*, *chance constrained programming*.

ABSTRACT

A model and metaheuristic method, considering Stochastic elements for e-waste collection in cities.

**Raúl Raggio Monte
October 2021**

Thesis supervisor: Lorena Pradenas Rojas
Program: Master in Industrial Engineering

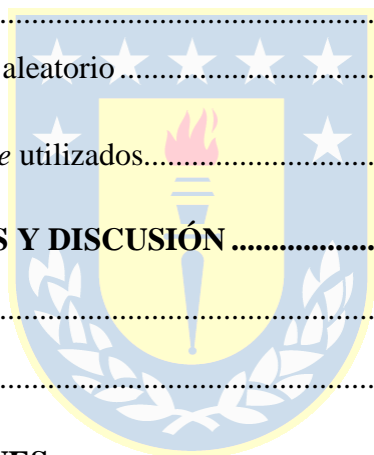
In this study, it is proposed an optimization model and metaheuristic methods for e-waste collection in cities. The aim is to minimize the total collection cost and the emissions cost produced for collection vehicles. It is considered that the e-waste quantity in each collection point is random. Besides, the visit time to each node is restricted through time windows. The optimization model is formulated through chance-constrained programming. On the other hand, it is proposed an Iterated Local Search and Simulated Annealing, treating the randomness through two approaches. In the first one, it is verified vehicles capacity using the set of capacity constraints derived from the chance-constrained programming methodology. In the second approach, it is verified vehicle capacity using random samples. For compare the methods, it is used literature instances adapted to the reality of Chile. With a significance of 5%, the exact model and the metaheuristics found statistically equal solutions, although the metaheuristics solutions are on average better than CPLEX solutions. On the other hand, total costs increase when greater collection probabilities are considered, because the problem becomes more complex to solve.

Study Keywords: e-waste, waste collection, stochastic optimization, vehicle routing problem with time windows, metaheuristics, Iterated Local Search, Simulated Annealing, chance constrained programming.

ÍNDICE DE CONTENIDOS

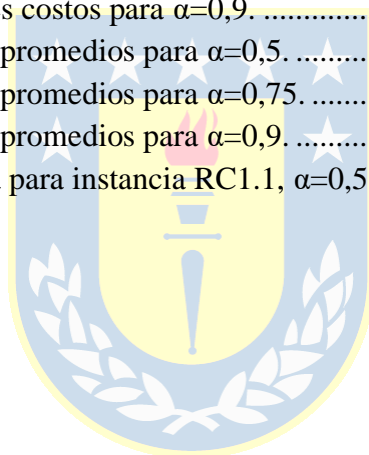
CAPÍTULO 1. INTRODUCCIÓN	1
1.1 Pregunta de investigación	2
1.2 Objetivo general.....	2
1.3 Objetivos específicos	2
1.4 Estructura del informe.....	2
CAPÍTULO 2. DESCRIPCIÓN DEL PROBLEMA	3
2.1 Situación mundial de la generación de residuos	3
2.2 Generación de residuos en Chile.....	4
2.3 Impacto Ambiental y económico de los residuos	6
CAPÍTULO 3. REVISIÓN DE LA LITERATURA.....	7
3.1 Manejo de residuos, tratamiento final y tipos de recolección.....	7
3.2 Principales variantes utilizadas para modelar la recolección de residuos.....	8
3.3 Estudios sobre el manejo de residuos en Chile	11
3.4 Comparación entre estudios	12
3.5 Brecha de conocimiento	13
CAPÍTULO 4. METODOLOGÍA.....	14
4.1 Presentación del problema	14
4.2 Modelo de programación matemática determinístico	15
4.3 <i>Chance-Constrained Programming</i> para el problema propuesto	18
4.4 Metaheurísticas propuestas	19
4.4.1 Enfoque A	22
4.4.2 Enfoque B.....	22

4.5 Elementos de las metaheurísticas.....	24
4.5.1 Representación de la solución.....	24
4.5.2 Función de evaluación.....	25
4.5.3 Solución inicial.....	25
4.5.4 Estrategia de búsqueda local.....	26
4.5.5 Estrategia de perturbación.....	28
4.5.6 Probabilidad de aceptación en SA.....	30
CAPÍTULO 5. INSTANCIAS DE PRUEBA Y RESULTADOS.....	31
5.1 Instancias de prueba.....	31
5.2 Parámetro de las metaheurísticas.....	33
5.2.1 Parámetros ILS.....	33
5.2.2 Parámetros SA.....	34
5.2.3 Parámetros muestreo aleatorio.....	34
5.3 Configuración y <i>hardware</i> utilizados.....	35
CAPÍTULO 6. RESULTADOS Y DISCUSIÓN.....	36
6.1 Resultados.....	36
6.2 Discusión de resultados.....	40
CAPÍTULO 7. CONCLUSIONES.....	49
CAPÍTULO 8. REFERENCIAS.....	51
ANEXOS.....	60
Anexo 1: Demostración <i>Chance-Constrained Programming</i>	60
Anexo 2: Análisis solución.....	63



ÍNDICE DE FIGURAS

Figura 2.1: Generación estimada de residuos electrónicos a nivel mundial.	3
Figura 2.2: Generación de residuos electrónicos de los principales países Latinoamericanos en 2016.	4
Figura 2.3: Generación de residuos electrónicos por habitante en Chile durante 2012-2027. ...	5
Figura 2.4: Porcentaje de residuos valorizados a nivel nacional entre 2015-2018.	6
Figura 3.1 Sistema de manejo de residuos en Chile.	7
Figura 4.1: Ejemplo solución encontrada.	24
Figura 4.2: Ejemplo búsqueda local <i>one-point move</i>	27
Figura 4.3: Ejemplo búsqueda local <i>two-point move</i>	27
Figura 6.1: Tiempo promedio de resolución CPLEX en cada instancia.	42
Figura 6.2: GAP soluciones CPLEX por instancia.	43
Figura 6.3: Comparación mejores costos para $\alpha=0,5$	46
Figura 6.4: Comparación mejores costos para $\alpha=0,75$	46
Figura 6.5: Comparación mejores costos para $\alpha=0,9$	47
Figura 6.6: Comparación costos promedios para $\alpha=0,5$	47
Figura 6.7: Comparación costos promedios para $\alpha=0,75$	48
Figura 6.8: Comparación costos promedios para $\alpha=0,9$	48
Figura A.1: Solución encontrada para instancia RC1.1, $\alpha=0,5$	63



ÍNDICE DE TABLAS

Tabla 3.1: Comparación estudios de recolección de residuos.	12
Tabla 4.1: Conjuntos y parámetros considerados.	15
Tabla 4.2: Parámetros agregados al modelo de optimización.	18
Tabla 4.3: Ejemplo verificación de capacidad en el enfoque B.	23
Tabla 5.1: Instancias consideradas.	31
Tabla 5.2: Valor de Q según la instancia y α	32
Tabla 5.3: Parámetro considerados en el problema.	33
Tabla 5.4: Parámetros ILS.	34
Tabla 5.5: Parámetros SA testeados experimentalmente.	34
Tabla 6.1: Resultados encontrados para $\alpha=0,5$	37
Tabla 6.2: Resultados obtenidos para $\alpha=0,75$	38
Tabla 6.3: Resultados obtenidos para $\alpha=0,9$	39
Tabla 6.4: Instancias donde el mejor costo de las metaheurísticas es peor que CPLEX.	40
Tabla 6.5: Instancias donde el enfoque B encontró mejores soluciones que CPLEX, 0% de GAP.	41
Tabla 6.6: Incremento de costos según α y método, promediando todas las instancias.	43
Tabla 6.7: GAP promedio costos respecto a CPLEX, promediando instancias y niveles de α	44
Tabla 6.8: GAP promedio enfoque B respecto al enfoque A.	44
Tabla 6.9: Wilcoxon <i>signed rank test</i> CPLEX vs metaheurísticas.	45
Tabla 6.10: Wilcoxon <i>signed rank test</i> entre metaheurísticas.	45
Tabla A.1: Rutas y cantidad de residuos recolectados para solución instancia RC1.1, $\alpha=0,5$	64
Tabla A.2: Tiempo de llegada a cada nodo para solución instancia RC1.1, $\alpha=0,5$	64

Glosario

- AGA: *Adaptive genetic algorithm*
- ALNS: *Adaptive large neighborhood search*
- BSA: *Backtracking Search Algorith*
- CCLVRP: *Chance-constrained low-carbon vehicle routing problem*
- CCCVRPTWSD: *Chance-constrained capacited vehicle routing problem with time windows and stochastic demand.*
- CCP: *Chance-Constrained Programming*
- CVRP: *Capacitated Vehicle Routing Problem*
- CVRPTW: *Capacitated Vehicle Routing Problem with time windows*
- Ee-VRP: *Eco-efficient Vehicle Routing Problem*
- E-waste: Residuos electrónicos
- Enfoque A: Enfoque propuesto para las metaheurísticas, el cual considera el conjunto de restricciones derivado de la metodología CCP para comprobar la factibilidad de una ruta.
- Enfoque B: Enfoque propuesto para las metaheurísticas, el cual considera muestreo aleatorio para comprobar la factibilidad de una ruta.
- GRASP: *Gready Randomized Adaptive Search Procedure*
- G-WCVRPTW: *Green waste collection problem with time windows*
- IG: *Iterated greedy*
- ILS: *Iterated Local Search*
- LC-2EHVRP-TW: *low-carbon two-echelon heterogeneous-fleet vehicle routing problem with time windows.*
- LNS: *Large neighborhood search*
- LSHA: *Local search hybrid algorithm*
- LRP: *Location routing problem*
- MDGVRP: *Multi-Depot Green Vehicle Routing Problem*
- MILP: *Mixed-integer linear programming*
- MOHCGA: *Multi objective hybrid cultural and genetic algorithm*
- MOPSO: *Multi-objective particle swarm optimization*

- NSGA-II: *Non-dominated sorting genetic algorithm II*
- PCARP: *Periodic capacitated arc routing problem*
- PCGVRP: *Priority considered green vehicle routing problem*
- PSO: *Particle swarm optimization*
- RL: *Reverse logistics*
- SA: *Simulated annealing*
- SAA-GP: *Sample average approximation based goal programming*
- SIRP: *Stochastic inventory routing problem*
- TSACS: *Two-stage Ant Colony System*
- VNS: *Variable neighborhood search*
- VRLP: *Vehicle routing loading problem*
- VRP: *Vehicle routing problem*
- VRPTW: *Vehicle routing problem with time windows*
- WCP: *waste collection problem*



CAPÍTULO 1. INTRODUCCIÓN

Los continuos avances tecnológicos han cambiado significativamente la vida de las personas, permitiéndoles realizar tareas que hace unos años eran impensadas, tales como comunicarse con personas de otros países, buscar información en segundos o trabajar desde casa. Además, los costos de estas tecnologías son cada vez más bajos, situación que, si bien ha sido ventajosa para las personas y les ha permitido tener más fácil acceso a estas, también ha generado que el ciclo de vida de los aparatos electrónicos sea cada vez menor, produciendo que aumente la cantidad de residuos electrónicos en el mundo.

El correcto manejo de estas altas cantidades de residuos electrónicos y su oportuna valorización, a través de reciclaje, reutilización o valoración energética, es necesaria para el desarrollo sustentable de un país, ayudando a disminuir la contaminación ambiental y proteger a la salud de las personas.

Es así que diversos países en todo el mundo han intentado incrementar las tasas de valorización de estos residuos. Un ejemplo es Chile, con la creación de diversas leyes que fomenten el reciclaje y de organización que velen por el cumplimiento de estas. A pesar de estos esfuerzos, cada año aumenta la cantidad de residuos electrónicos en Chile, mientras que las tasas de valorización se mantienen bajas.

En el presente estudio, se propone encontrar un conjunto de rutas que minimicen el costo total de recolección y el costo de las emisiones producidas por los vehículos de recolección. Se plantea un modelo de programación lineal entera mixta (MILP, por sus siglas en inglés), considerando que la cantidad de residuos electrónico en cada punto limpio es aleatoria. Adicionalmente, para instancias donde el método exacto es inviable o toma un tiempo excesivo en entregar una solución, se considera el uso de metaheurísticas, con el objetivo de encontrar soluciones factibles en tiempos computacionales más bajos.

1.1 Pregunta de investigación

¿Es posible encontrar rutas de recolección para los residuos electrónicos en ciudades como el Gran Concepción, que minimicen el costo total de recolección y el costo de las emisiones producidas, mediante el uso de optimización estocástica y métodos metaheurísticos?

1.2 Objetivo general

Encontrar un conjunto de rutas de recolección de residuos electrónicos en una ciudad como el Gran Concepción, que minimicen los costos de recolección y las emisiones generadas por cada vehículo, considerando que la cantidad de residuos electrónicos en cada punto limpios es aleatoria.

1.3 Objetivos específicos

- Representar el problema planteado a través de un modelo de programación matemática que considere elementos estocásticos.
- Implementar computacionalmente el modelo.
- Plantear algoritmos metaheurísticos adecuados para resolver el conjunto de instancias consideradas.
- Generar y recopilar instancias de prueba para los algoritmos planteados.
- Analizar resultados.

1.4 Estructura del informe

El informe se estructura de la siguiente forma: el Capítulo 2 proporciona la descripción y motivación del problema. En el Capítulo 3 se revisan antecedentes sobre métodos de optimización en la recolección de residuos, y se compara la propuesta del presente estudio, con lo disponible en la literatura. El Capítulo 4 presenta la metodología propuesta para resolver el problema. En el Capítulo 5 se presentan las instancias de prueba. En el Capítulo 6 se encuentran los resultados encontrados y su análisis. Finalmente, en el Capítulo 7 se presentan las conclusiones del estudio y en el Capítulo 8 las referencias utilizadas.

CAPÍTULO 2. DESCRIPCIÓN DEL PROBLEMA

En el presente capítulo, se revisa el estado actual sobre los residuos electrónicos, en Chile y en el mundo, las principales problemáticas asociadas con la alta generación de estos y los esfuerzos de gobernantes por mejorar esta situación.

2.1 Situación mundial de la generación de residuos

Cada año los aparatos tecnológicos son más baratos y eficientes, facilitando el acceso de estos a la mayor parte de la población mundial. Si bien esta situación presenta numerosos beneficios, permitiendo realizar tareas que antes eran imposibles, también ha generado que la cantidad de residuos electrónicos producidos por año sea cada vez mayor (Baldé et al., 2017). Los residuos electrónicos corresponden a todos los aparatos eléctricos y electrónicos, que han sido descartados por su dueño, sin la intención de reutilizarlos de ninguna manera (Raggio, 2020). Según Forti et al. (2020), los residuos electrónicos generados en 2019 alcanzaron 53,6 millones de toneladas, equivalentes a 7,3 kilogramos por habitante. Si la situación se mantiene, se espera que para 2030, incremente esta cantidad a 74,7 millones de toneladas o 9 kg por habitante (Forti et al., 2020). Esta situación se muestra en la Figura 2.1.

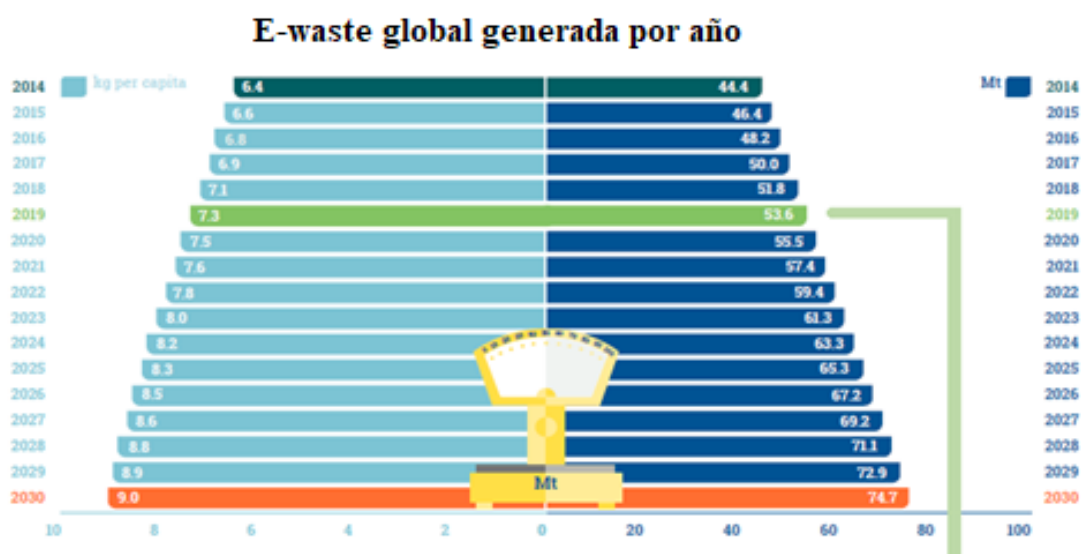


Figura 2.1: Generación estimada de residuos electrónicos a nivel mundial.
Fuente: Forti et al. (2020).

2.2 Generación de residuos en Chile

En relación a Chile, los generadores y destinatarios de residuos informaron por primera vez en 2015 al Sistema Nacional de Declaración de Residuos, por lo tanto, no se posee información oficial anterior a ese año (Raggio, 2020).

En la Figura 2.2, se observa que Chile fue el país de Latinoamérica que más residuos electrónicos generó por habitantes en 2016, alcanzando un promedio de 8,7 kg por habitante.

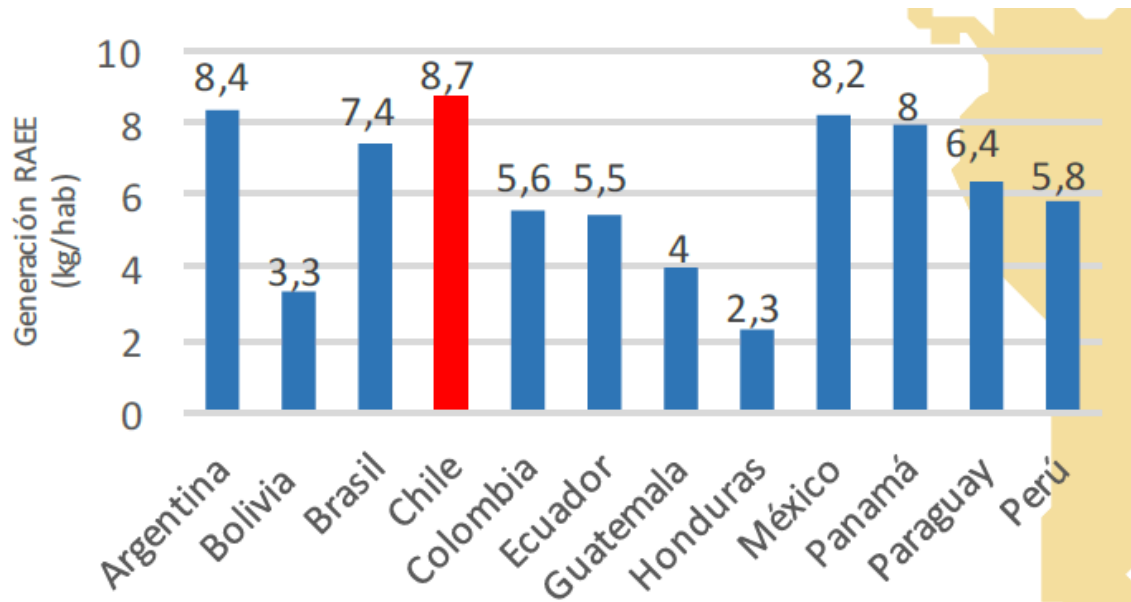


Figura 2.2: Generación de residuos electrónicos de los principales países Latinoamericanos en 2016.
Fuente: Fundación Chile & Ministerio del Medio Ambiente (2019).

Por otro lado, según E2BIZ Consultores & Fundación Chile (2019), las organizaciones internacionales subestiman la cantidad de residuos electrónicos en Chile, siendo la generación real durante 2016 de 9,7 kg por habitante. Como se observa en la Figura 2.3, se espera que esta cantidad aumente con el pasar de los años, de 7,1 kg por habitantes en 2012, a 14,0 kg por habitante en 2027.

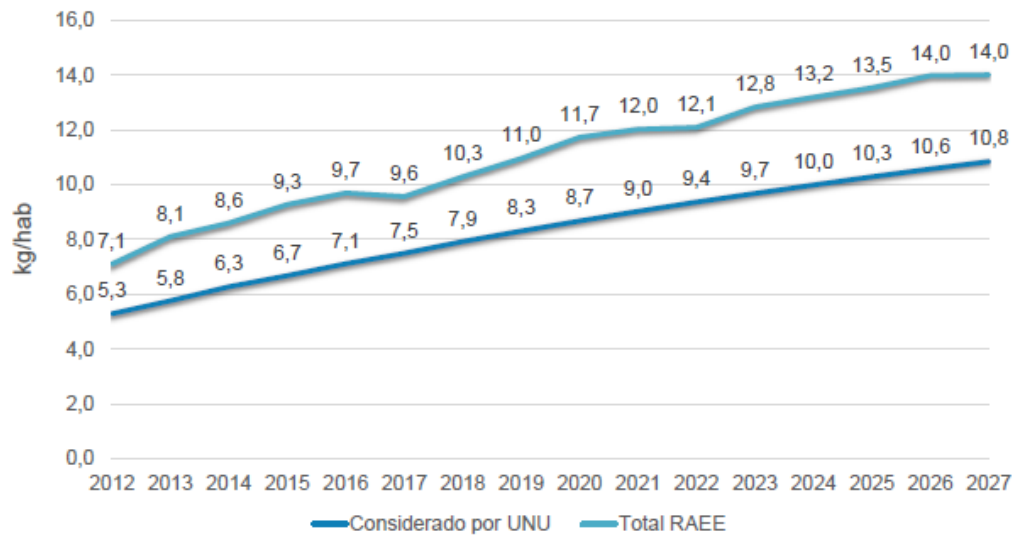


Figura 2.3: Generación de residuos electrónicos por habitante en Chile durante 2012-2027.
Fuente: E2BIZ Consultores & Fundación Chile (2019).

Por lo anterior, se han generado diversas leyes con el fin de mejorar la situación del medio ambiente del país y gestionar de la mejor manera posible estas grandes cantidades de residuos generados anualmente. Entre estas, se puede mencionar la Ley 19.300 que establece las bases generales el medio ambiente, la Ley 20.417 que crea el ministerio del medio ambiente y la Ley 20.920 que establece el marco legal para la gestión de residuos y fomento al reciclaje (Raggio, 2020).

A pesar de estos esfuerzos, el principal problema del país sigue siendo la falta de valorización de estos residuos (Ministerio del Medio Ambiente, 2016a). En la Figura 2.4 se muestra el porcentaje de residuos valorizados a nivel nacional entre 2015 y 2018, alcanzando 21,89% la tasa de valorización de residuos en 2018.

Tasa nacional de valorización final, 2015- 2018

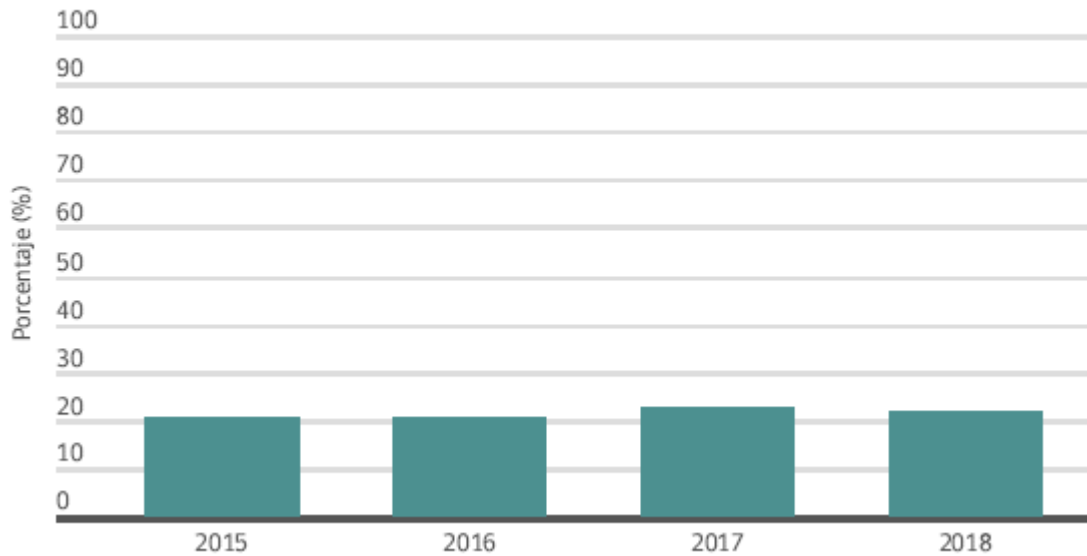


Figura 2.4: Porcentaje de residuos valorizados a nivel nacional entre 2015-2018.
Fuente: Ministerio del Medio Ambiente (2020).

2.3 Impacto Ambiental y económico de los residuos

La valorización de los residuos electrónicos es necesaria para el desarrollo sustentable de un país, relacionándose con: la conservación de la energía, la prevención de emisiones de gases de efecto invernaderos, la disminución de la contaminación en los océanos, el aumento del turismo y la protección de la salud pública (Kaza et al., 2018). Por otro lado, debido a la gran cantidad de materiales valiosos usados para la construcción de estos aparatos, como el oro, la plata o el platino, existe un gran incentivo económico para valorizarlos, estimándose que se desaprovechan 55 billones de euros anuales en estos materiales (Baldé et al., 2017).

En función de estos problemas, sumado a que los costos operativos de la recolección de residuos electrónicos pueden representar hasta el 70% del presupuesto total necesario (Kaza et al., 2018), cobra relevancia encontrar un conjunto de rutas de recolección para los residuos electrónicos, que generen los menores costos y emisiones, de modo de motivar a más empresas transportistas y al gobierno a realizar este servicio y eventualmente, incrementar las tasas de valorización.

CAPÍTULO 3. REVISIÓN DE LA LITERATURA

En el presente capítulo, se revisan distintos términos sobre el manejo de residuos electrónicos. Además, se presentan las principales variantes que se utilizan para modelar el problema de recolección de residuos.

3.1 Manejo de residuos, tratamiento final y tipos de recolección

El manejo de residuos se puede definir como: “todas las acciones operativas a las que se somete un residuo, incluyendo, entre otras, recolección, almacenamiento, transporte, pretratamiento y tratamiento” (Ministerio del Medio Ambiente, 2016b). De estos elementos, el más importante en términos de costos y peligro para la salud pública, es la recolección y transporte (Raggio, 2020). En la Figura 3.1, se observa el sistema de manejo de residuos en Chile.

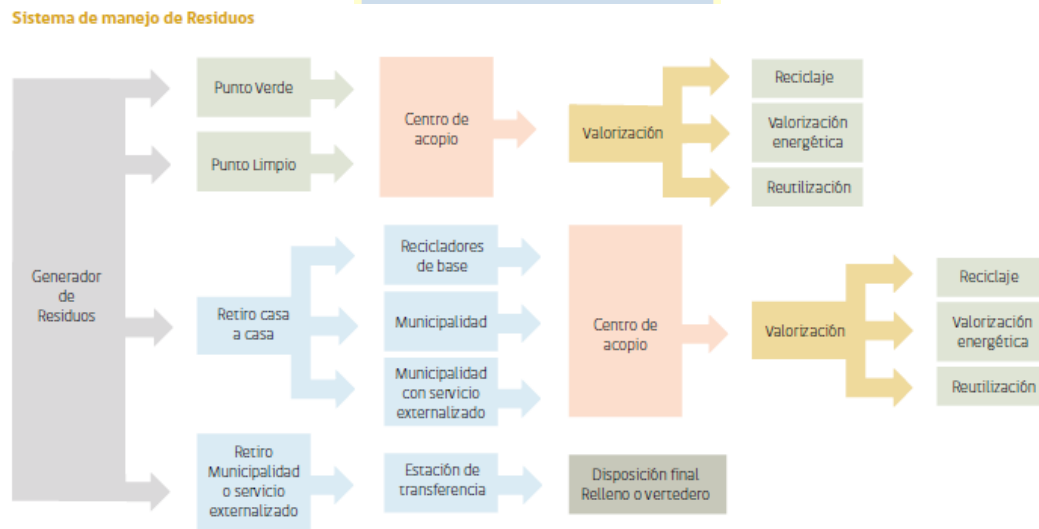


Figura 3.1 Sistema de manejo de residuos en Chile.
Fuente: Ministerio del Medio Ambiente (2016b).

En relación al tratamiento de residuos, existen dos principales formas de realizarlo. La primera es la valorización, en la cual se entrega un “valor” al medio ambiente, en forma de reutilización, reciclaje o valorización energética. La segunda es la eliminación, en la cual los residuos son incinerados o transportados a sitios de disposición final, como basurales, vertederos o rellenos sanitarios (Raggio, 2020).

Por otro lado, existen distintas formas de realizar el proceso de recolección y transporte de residuos, las cuales son adoptadas dependiendo de las características de la ciudad en cuestión. Una de las principales, es a través de la recolección en lugar designado, donde los residuos son colocados en distintos puntos fijos en la ciudad, evitando que los vehículos recolectores visiten muchos vecindarios y disminuir así la congestión vehicular en ciudades con alta población (Raggio, 2020).

3.2 Principales variantes utilizadas para modelar la recolección de residuos

Usualmente la recolección de residuos se modela a través de problemas de ruteo, con el objetivo de encontrar un conjunto de rutas óptimas de recolección. Entre los distintos tipos de problemas de ruteo, se utilizan los ruteos de vehículos (Kim et al., 2006; Akhtar et al., 2017; Hannana et al., 2018; Wu et al., 2020a), los ruteos en arcos (Cortinhal et al., 2016; Tirkolaee et al., 2018) y los ruteos de inventarios (Mes et al., 2014; Markov et al., 2020). De estos, el problema de ruteo de vehículos es uno de los más utilizados por los autores, el cual se define como: dado una flota de vehículos y un conjunto de solicitudes de transporte en los nodos de la red, se busca encontrar un conjunto de rutas para estos vehículos que realicen algunas o todas las solicitudes de transporte (Toth & Vigo, 2014).

Por otro lado, diversos autores optan por modelar la recolección y las demás fases de la cadena de suministro del manejo de residuos, a través del diseño de una red de reciclaje inversa. En esta estrategia, los residuos son asignados a distintos esquemas de recolección, determinando los sitios óptimos y las capacidades de diversos elementos, tales como los centros de recolección y plantas de reciclajes (Alumur et al., 2012). Entre estos autores, se encuentran Alumur et al. (2012), Ayvaz et al. (2015), Trochu et al. (2019), Yu et al. (2020).

Por otra parte, existen tres objetivos que son los más comunes a optimizar en este tipo de problemas. El primero consiste en minimizar los costos y/o distancias del manejo de residuos. Por ejemplo, Buhrkal et al. (2012) minimizan el costo total de recolección de residuos a través de un problema de ruteo de vehículos capacitado con ventanas de tiempo, aplicado a una empresa norte americana y otra danesa. Otro objetivo es maximizar el beneficio del manejo de residuos. Por ejemplo, Alumur et al. (2012) maximizan el beneficio de la red de logística inversa de residuos, considerando los ingresos de los centros de reciclaje, de las plantas de reacondicionamiento y de mercados secundarios, y los costos de establecer las instalaciones, de

transporte y de mantención. El tercero objetivo consiste en la minimización de emisiones producidas, especialmente en el proceso de recolección. Por ejemplo, Zhang et al. (2020) resuelven un *Green vehicle routing problem* multi-depósito y minimizan el total de emisiones de carbono producidos en cada ruta, utilizando instancias generadas aleatoriamente y de la literatura. Otros objetivos considerados son: minimizar el riesgo de exposición de las personas a los residuos peligrosos (Farrokhi-Asl et al., 2020; Yu et al., 2020), minimizar la tardanza de la recolección (Tirkolae et al., 2019; Ganji et al., 2020), maximizar la cantidad de residuos recolectados durante un horizonte de tiempo (Expósito-Márquez et al., 2019), entre otros.

Diversos autores consideran más de un objetivo a optimizar, ya sea en modelos con una función objetivo o multiobjetivo. Un ejemplo del primer caso lo abordan Wu et al. (2020a), que consideran el factor económico y ambiental en la recolección de residuos húmedos, a través de la minimización de los costos de recolección y los costos de emisiones de carbono en una misma función objetivo. Otros estudios similares, lo plantean Wang et al. (2020) y Wu et al. (2020b).

En relación a los problemas multiobjetivo, Farrokhi-Asl et al. (2020) abordan un problema multiobjetivo para la recolección de residuos, minimizando los costos operacionales del proceso y los costos sociales, tales como: minimizar la cantidad de personas entre las rutas de recolección y la minimización de las personas alrededor de los sitios de depósito de residuos. Otros ejemplos de problemas multiobjetivo en la recolección de residuos son planteados en Rabbani et al. (2018), Mostafayi et al. (2020) y Yu et al. (2020).

En relación a los conjuntos de restricciones considerados, las restricciones de capacidad son las más comunes, debido a que los vehículos recolectores tienen una cierta capacidad que no debe excederse. De la misma forma, se suelen considerar ventanas de tiempo, ya que muchos lugares son inaccesibles para los vehículos de recolección dependiente de la hora (Kim et al., 2006; Buhrkal et al., 2012; Król et al., 2016; Markov et al., 2020).

Existen otras variaciones en el problema que afectan a las restricciones planteadas en el modelo de optimización matemático, como considerar vehículos heterogéneos (Markov et al., 2016; Asefi et al., 2019a; Ganji et al., 2020), múltiples depósitos (Kim et al., 2006; Gruler et al., 2017), horizonte de recolección de varios días o meses (Alumur et al., 2012; Markov et al., 2020), múltiples viajes de cada vehículos (Kim et al., 2006; Tirkolae et al., 2019), equilibrio de trabajo y/o tiempo de descanso de los conductores (Kim et al., 2006; Buhrkal et al., 2012).

Otra variación importante ocurre cuando uno o más parámetros son aleatorios, pasando el problema a ser de optimización estocástica (Hvattum et al., 2006). El parámetro estocástico más común en estos problemas es la cantidad de residuos en cada punto de recolección, debido a la aleatoriedad de este parámetro en la vida real. Esta aleatoriedad se suele representar mediante restricciones probabilísticas (Xu et al., 2013; Edalatpour et al., 2018; Shah et al., 2018; Wu et al., 2020a) o modelo estocástico de dos etapas (Asefi et al., 2019b; Trochu et al., 2019). Otros posibles parámetros estocásticos son los clientes, los cuales pueden aparecer bajo una cierta probabilidad, y los tiempos de viaje y/o de servicios de cada cliente (Raggio, 2020).

En cuanto a los métodos para encontrar una solución, los autores optan por utilizar algoritmos metaheurísticos, debido a la naturaleza NP-difícil de estos tipos de problemas y al tamaño de las instancias que se utilizan. Por ejemplo, Kim et al. (2006) consideran el caso de una empresa real norteamericana, cuya cantidad de nodos varía entre 100 y 2.100.

Entre las metaheurísticas más utilizadas, se encuentra *Simulated Annealing* (Tirkolaei et al., 2018; Tirkolaei et al., 2019; Wu et al., 2020a), *Local Search* (Gruler et al., 2017; Markov et al., 2016; Blazquez et al., 2020), algoritmos genéticos (Król et al., 2016; Assaf & Saleh, 2017), *modified particle swarm optimization* (Hannan et al., 2018; Wu et al., 2020a) y *Tabu Search* (Cortinhal et al., 2016).

En relación a metaheurísticas aplicadas a problemas estocásticos, Grasas et al. (2014) plantean una “sim-metaheurística” la cual combina métodos de simulación para obtener muestras aleatorias y así estimar costos esperados, o verificar el cumplimiento de un conjunto de restricciones. Trabajos como los de Gruler et al. (2017) han utilizado este enfoque en un problema de recolección de residuos con demanda estocástica, múltiples depósitos y áreas urbanas agrupadas.

En cuanto a las metaheurísticas utilizadas para problemas multiobjetivo, destacan principalmente los algoritmos genéticos, tales como *Multi-objective Hybrid Cultural and Genetic Algorithm* (Farrokhi-Asl et al., 2018) y *non-dominated sorting genetic algorithm* (NSGA) (Rabbani et al., 2018; Rabbani et al., 2019; Farrokhi-Asl et al., 2020). Otras metaheurísticas utilizadas son *Particle Swarm Optimization* (Rabbani et al., 2018) y *Variable Neighborhood Search* (Delgado-Antequera et al., 2020).

Entre los estudios que optan el uso de métodos exactos, se pueden mencionar los siguientes *solvers*: CPLEX (Grunow et al., 2009; Alumur et al., 2012; Ayvaz et al., 2015) y *solver* de Excel (Shah et al., 2018).

3.3 Estudios sobre el manejo de residuos en Chile

Este problema ya ha sido abordado en Chile, debido a la importancia del correcto manejo de residuos. Entre los más importantes, se encuentra el trabajo de Eiselt et al. (2014), que presentan un MILP bi-objetivo para determinar la ubicación y tamaño óptimo de los rellenos sanitarios y estaciones de transferencia en la región de Bio-Bio, minimizando costos y contaminación. Por otro lado, Blazquez et al. (2020) diseñan un sistema de recolección de residuos domiciliarios para la comuna de Renca, mediante dos fases: primero resuelven un problema de localización y asignación de contenedores. En segundo lugar, resuelven un problema de ruteo de vehículos homogéneos, restringido por la duración del turno de trabajo y la capacidad del vehículo. Se utiliza una heurística de búsqueda de vecindario grande (LNS) para resolver el problema.

En cuanto a los estudios de recolección de residuos en la comuna de Concepción, Cárdenas (2020) resuelve un problema de gestión de residuos electrónicos para esta ciudad, mediante un problema de ruteo de vehículos capacitados y un problema de logística inversa. Para la resolución, consideran instancias con una cantidad entre 10-20 nodos. Por otro lado, Carreño (2020) resuelve un problema de recolección de residuos sólidos urbanos, a través de un *green waste collection problem with time windows* (G-WCVRPTW) multiobjetivo, que considera aspectos económicos y ambientales. Se utilizan las instancias de Solomon (1987) para un VRPTW, considerando hasta 40 nodos. Finalmente, Raggio (2020) plantea un modelo de optimización estocástico con restricciones probabilísticas para la recolección de residuos electrónicos, considerando demanda aleatoria. El objetivo es minimizar los costos de recorrer cada nodo y el de utilizar cada vehículo. Los resultados encuentran soluciones exactas para instancia de hasta 20 nodos.

3.4 Comparación entre estudios

En la Tabla 3.1 se muestra una comparación de diferentes estudios de la literatura relacionados al manejo de residuos.

Tabla 3.1: Comparación estudios de recolección de residuos.

Autores	Estrategia utilizada	Naturaleza de parámetros	Método de solución	Objetivo
Król et al. (2016)	CVRPTW	Determinístico	Metaheurística (GA) y lógica difusa	Minimizar vehículos utilizados y longitud de las rutas de recolección.
Akhtar et al. (2017)	CVRP	Determinístico	Metaheurística (BSA)	Minimizar la suma de la distancia de recolección.
Nowakowski (2017)	VRLP	Determinístico	Metaheurística (Optaplanner)	Maximizar la cantidad de residuos en contenedor y minimizar los costos de recolección.
Farrokhi-Asl et al. (2018)	WCP multiobjetivo	Determinístico	Metaheurística (MOHCGA)	Minimizar costos operacionales, de emisiones y sociales.
Hannan et al. (2018)	CVRP	Determinístico	Metaheurística (PSO)	Minimizar la suma de la distancia de recolección.
Rabbani et al. (2018)	hazard waste-location routing problem	Determinístico	Metaheurística (NSGA-II y MOPSO)	Minimizar costos y riesgo de transporte de los residuos peligrosos, y riesgo del sitio.
Tirkolaee et al. (2018)	PCARP	Estocástico (demanda)	Metaheurística (SA híbrido)	Minimizar distancia total recorrida y el costo de uso de los vehículos.
Expósito-Márquez et al. (2019)	Ee-VRP	Determinístico	Metaheurística (GRASP)	Maximizar la cantidad de residuos recolectados durante un horizonte de tiempo.
Tirkolaee et al. (2019)	VRPTW con viajes múltiples	Determinístico	Metaheurística (SA)	Minimizar el costo de viaje, de vehículos utilizados y por tardanza de entrega.
Blazquez et al. (2020)	Diseño de recolección de residuos (dos etapas)	Determinístico	Metaheurística (LNS)	Minimizar el costo de localización de contenedores y la distancia total recorrida.
Cárdenas (2020)	CVRP-RL	Determinístico	Exacto (CPLEX)	Minimizar costos de recolección y costos asociados a la ubicación de centros de almacenamiento e inspección.
Carreño (2020)	GWCVRPTW multiobjetivo	Determinístico	Exacto (CPLEX)	Minimizar costos asociados al proceso de recolección y cantidad de emisiones producidas.
Delgado-Antequera et al. (2020)	Problema de ruteo multiobjetivo	Determinístico	Metaheurística (IG-VNS)	Minimizar el costo de viaje y número de rutas, y balancear las longitudes y el tiempo de cada ruta.
Farrokhi-Asl et al. (2020)	LRP multiobjetivo	Determinístico	ϵ -constraint y metaheurístico (NSGA-II)	Minimizar costos operacionales, daño a la salud de las personas, ocurrencia de accidentes y daño al medio ambiente.
Markov et al. (2020)	SIRP no lineal con modelo de estimación de demanda	Estocástico (demanda)	Metaheurística (ALNS)	Minimizar costo esperado de desbordamiento, de recolección de emergencia, de ruteo y esperado de falla de ruta.
Raggio (2020)	CCVRPTWSD	Estocástico (demanda)	Exacto (CPLEX)	Minimizar costos de recolección y de vehículos utilizados.
Wang et al. (2020)	LC-2EHVRP-TW	Determinístico	Metaheurística (AGA)	Minimizar costos de recolección total, emisiones de carbono y maximizar la satisfacción de los clientes
Wu et al. (2020a)	CCLVRP	Estocástico (demanda)	Metaheurística (PSO-SA)	Minimizar costos de recolección y emisiones producidas.
Wu et al. (2020b)	PCGVRP	Determinístico	Metaheurística (LSHA)	Minimizar distancia total de recolección, costos de emisiones totales y costos de recolección
Yu et al. (2020)	MILP estocástico biobjetivo	Estocástico (costo, demanda y población afectada)	SAA-GP	Minimizar el riesgo de exposición de la población y maximizar la rentabilidad de la recolección.
Zhang et al. (2020)	MDGVRP	Determinístico	Metaheurística (TSACS)	Minimizar las emisiones de carbono totales.
Este estudio	CCCVPTWSD	Estocástico (demanda)	Exacto y Metaheurístico (ILS y SA)	Minimización costos de recolección y costos de emisiones.

Fuente: Elaboración propia.

3.5 Brecha de conocimiento

A diferencia de otros estudios, en este se presenta un modelo de optimización estocástico y métodos metaheurísticos para la recolección de residuos electrónicos en ciudades como el Gran Concepción, considerando costos totales de recolección y costos de emisiones generadas. Si bien, los estudios de Cárdenas (2020) y Carreño (2020) plantean un modelo para la recolección de residuos en esta ciudad, lo realizan considerando que todos los parámetros son determinísticos. Por otro lado, Raggio (2020) no considera aspectos ambientales ni métodos metaheurísticos para resolver este problema. Es más, con el método exacto planteado en este último estudio, el tiempo de resolución aumenta considerablemente en instancias con más de 20 nodos, lo que hace que su eficiencia no sea la mejor para resolver instancias reales.



CAPÍTULO 4. METODOLOGÍA

En el presente capítulo se describe la metodología utilizada, presentando el problema, el modelo de optimización matemático y las metaheurísticas propuestas.

4.1 Presentación del problema

En el problema de recolección de residuos electrónicos del presente estudio, se dispone: un único depósito en el cual los vehículos recolectores inician y terminan su ruta, y un conjunto de puntos limpios, cada uno con una cantidad aleatoria de residuos electrónicos. Los vehículos deben recolectar la totalidad de los residuos en los puntos limpios (recolección en lugar designado) sin superar su capacidad máxima y respetando las ventanas de tiempo de cada locación. El principal objetivo es encontrar un conjunto de rutas factibles para cada vehículo que disminuyan el costo de recolección y el costo de emisiones generadas en cada ruta. Las principales consideraciones de este problema son:

- Los costos de recolección consideran: los costos de recorrer cada nodo y el costo fijo de utilizar un vehículo extra.
- Los arcos son simétricos: el costo/distancia de ir desde punto limpio i al j es el mismo que el costo/distancia de ir desde el punto limpio j a i .
- Se asume que el costo de emisiones depende solo de la distancia total recorrida.
- Vehículos homogéneos: Tienen la misma capacidad máxima, rendimiento y costo de utilización.
- El lugar de almacenamiento de residuos coincide con el depósito de los vehículos.
- Cada vehículo debe empezar y terminar su ruta en el depósito.
- Se permite solo una ruta por vehículo.
- Cada punto limpio debe ser visitado por exactamente un vehículo de recolección.
- La demanda de cada punto limpio debe ser atendida completamente.
- La demanda de cada punto limpio es aleatoria.
- Cada punto limpio debe ser visitado exactamente una vez y en su ventana de tiempo.
- Existe un tiempo de viaje entre locaciones (puntos limpios y depósito).
- Existe un tiempo de servicio en cada punto limpio.

4.2 Modelo de programación matemática determinístico

Bajo las consideraciones propuestas, el problema es modelado como un problema de ruteo de vehículos capacitado con pickups y ventanas de tiempos. En la Tabla 4.1 se presentan los conjuntos y parámetros considerados

Tabla 4.1: Conjuntos y parámetros considerados.

Parámetro	Descripción
n	Número de puntos limpios.
K	Número de vehículos.
C	Conjunto de puntos limpios. $C = \{1,2,3, \dots, n\}$.
N	Conjunto de todas las localizaciones: Depósito y puntos limpios. $\{0\} \cup C$, $N = \{0,1,2,3, \dots, n\}$.
V	Conjunto de vehículos. $V = \{1,2,3, \dots, K\}$.
c_{ij}	Costo de viaje, en pesos chilenos, entre la locación i y j . Con $i, j \in N$.
f	Costo de utilizar un vehículo en pesos chilenos.
Q	Capacidad máxima de un vehículo en kg.
d_i	Cantidad de residuos electrónicos, en kg, en el punto limpio i . $d_i \leq Q$. Con $i \in C$.
a_i	Tiempo inicial, en minutos, permitido para atender al punto i . $i \in N$.
b_i	Tiempo final, en minutos, permitido para atender al punto i . $i \in N$.
t_{ij}	Tiempo de viaje, en minutos, entre la locación i y j . Con $i, j \in N$.
p_i	Tiempo de atención, en minutos, al punto limpio i . Con $i \in C$.
M_{ij}	Constante de penalización. Con $i, j \in N$.
$CosCO2_{ij}$	Costo de emisiones, en pesos chilenos, entre la locación i y j . Con $i, j \in N$.
dis_{ij}	Distancia, en kilómetros, entre la locación i y j . Con $i, j \in N$.
Fac_{co2}	Factor de emisiones de carbono (kg/L).
Im_{co2}	Impuesto de emisiones de CO2 en Chile (<i>pesos chilenos/kg</i>).
pc	Precio de combustible diésel por litro (<i>pesos chilenos/L</i>).
r	Rendimiento del vehículo (<i>kilómetros/L</i>).

Fuente: Elaboración propia.

Las variables de decisión consideradas son:

Variables de decisión:

$$x_{ijk} = \begin{cases} 1; & \text{Si el vehículo } k \text{ viaja directamente de la locación } i \text{ hasta la } j. \forall i, j \in N, \forall k \in V \\ 0; & \text{e. o. c} \end{cases}$$

s_{ik} = Tiempo en el que el vehículo k inicia el servicio de la locación i . $\forall i \in N, \forall k \in V$.

*Si el vehículo k no visita el nodo i , la variable s_{ik} pierde su significado.

Así, la función objetivo y restricciones consideradas se muestran a continuación.

Función objetivo

$$\text{Min} \sum_{i \in N} \sum_{j \in N} \sum_{k \in V} c_{ij} x_{ijk} + f \sum_{k \in V} (1 - x_{00k}) + \sum_{i \in N} \sum_{j \in N} \sum_{k \in V} \text{CosCO2}_{ij} x_{ijk} \quad (1)$$

Sujeto a:

$$\sum_{\substack{i \in N \\ i \neq j}} \sum_{k \in V} x_{ijk} = 1, \quad \forall j \in C \quad (2)$$

$$\sum_{j \in N} x_{0jk} = 1, \quad \forall k \in V \quad (3)$$

$$\sum_{\substack{i \in N \\ i \neq j}} x_{ijk} = \sum_{\substack{i \in N \\ i \neq j}} x_{jik}, \quad \forall j \in C, \forall k \in V \quad (4)$$

$$\sum_{i \in N} x_{i0k} = 1, \quad \forall k \in V \quad (5)$$

$$\sum_{\substack{i \in C \\ i \neq j}} \sum_{j \in N} d_i x_{ijk} \leq Q, \quad \forall k \in V \quad (6)$$

$$s_{ik} + p_i + t_{ij} - M_{ij}(1 - x_{ijk}) \leq s_{jk}, \quad \forall i \in C, \forall j \in N, \forall k \in V \quad (7)$$

$$a_i \leq s_{ik} \leq b_i, \quad \forall i \in N, \forall k \in V \quad (8)$$

$$x_{ijk} \in \{0,1\}, \quad \forall i, j \in N, \forall k \in V \quad (9)$$

$$s_{ik} \geq 0, \quad \forall i \in N, \forall k \in V \quad (10)$$

La ecuación (1) corresponde a la función objetivo, la cual minimiza el costo total de recolección y el costo de emisiones producidas de CO_2 . El costo total de recolección es la suma entre el costo de viaje entre cada nodo y el costo de utilizar cada vehículo. El costo de viaje entre cada nodo, se asume que es dependiente de: la distancia recorrida (dis_{ij}), el precio del combustible (pc) y el rendimiento del vehículo (r) (ecuación (11)). En relación al costo de emisiones de CO_2 , según Wu et al. (2019a), está directamente relacionado con el consumo de combustible del vehículo, el cual, para motivos de este estudio, se asume que solo depende de la distancia recorrida (dis_{ij}). Así, junto al factor de kg de CO_2 producidos por litro (Fac_{co2}), el costo por kg de CO_2 producido (a través del impuesto al CO_2 Im_{co2}) y el rendimiento del vehículo (r), se obtiene el costo de emisiones producidas de CO_2 (ecuación (12)).

$$c_{ij} = \frac{dis_{ij} * pc}{r}, \quad \forall i, j \in N \quad (11)$$

$$CosCO2_{ij} = \frac{dis_{ij} * Im_{co2} * Fac_{co2}}{r}, \quad \forall i, j \in N \quad (12)$$

Las restricciones de tipo (2) aseguran que un punto limpio es visitado exactamente una vez por un vehículo. Las restricciones de tipo (3) aseguran que cada vehículo sale del depósito hacia exactamente una locación, ya sea un punto limpio o el mismo depósito. Las restricciones de tipo (4) restringen que, si un vehículo llega a un punto limpio, este debe salir a otra locación, ya sea, un punto limpio o el depósito. Las restricciones de tipo (5) indican que cada vehículo debe llegar al depósito. El conjunto de restricciones de tipo (6) aseguran que la capacidad máxima del vehículo no sea superada. Las restricciones de tipo (7) aseguran que el vehículo k usa el tiempo correcto en viajar desde i a j . Esto es, si el vehículo k viaja directamente desde i a j ($x_{ijk} = 1$), entonces el tiempo en que comienza el servicio en el nodo j (s_{jk}), es mayor o igual a la suma entre: el tiempo en que comienza el servicio a la locación i (s_{ik}); el tiempo de servicio del nodo i (p_i) y el tiempo de viaje entre las locaciones (t_{ij}). Además, este conjunto restringe las subrutas. Las restricciones de tipo (8) establecen las ventanas de tiempo para el nodo i . Finalmente, las restricciones de tipo (9) y (10) restringen los valores que pueden tomar las variables de decisión. El valor de M_{ij} es definido de forma de ser lo suficientemente grande, esto es:

$$M_{ij} = b_i + t_{ij} + p_i - a_j \quad (13)$$

4.3 *Chance-Constrained Programming* para el problema propuesto

Como se menciona, el problema considera que la cantidad de residuos en cada punto limpio es aleatoria, esto con el fin de representar incertidumbres propias de la vida real.

Una manera eficiente de agregar esta aleatoriedad al problema de optimización planteado y que permite resolver un problema probabilístico como uno lineal determinístico, es a través de *Chance-Constrained Programming* (CCP). Esta garantiza que la viabilidad de una ruta se logra por sobre un nivel dado (limitando la región factible), alcanzando un alto nivel de confiabilidad de la solución (Toth & Vigo, 2014).

Para aplicar esta metodología, se agregan los parámetros al modelo de optimización matemático mostrados en la Tabla 4.2.

Tabla 4.2: Parámetros agregados al modelo de optimización.

Parámetro	Descripción
α	Mínima probabilidad que la ruta funcione (no falle).
μ_i	Media de la distribución de probabilidad de d_i . Con $i \in C$.
σ_i	Desviación estándar de la distribución de probabilidad de d_i . Con $i \in C$.
z_α	Función inversa de la distribución Normal estándar, evaluada en la probabilidad α .

Fuente: Elaboración propia.

Para el problema estocástico, el conjunto de restricciones (6) es transformado en las restricciones (14), las cuales aseguran que la probabilidad de que la cantidad de residuos electrónicos recolectados en la ruta sea menor a la capacidad del vehículo, es mayor a α .

$$P\left(\sum_{i \in C} \sum_{j \in N} d_i x_{ijk} \leq Q\right) \geq \alpha, \quad \forall k \in V \quad (14)$$

Asumiendo que la cantidad de residuos en cada punto d_i se distribuye normal con media μ_i y varianza σ_i^2 , que son independientes e idénticamente distribuidas entre todos los puntos limpios (esto es, $\mu_i = \mu$ y $\sigma_i = \sigma \forall i \in C$) y como las variables x_{ijk} son binarias, se cumple que $x_{ijk}^2 = x_{ijk}$, el conjunto de restricciones (14) puede ser expresado según la ecuación (15), donde $\pi = \frac{\sigma^2}{\mu}$ (Stewart & Golden, 1983). Para más detalle de esta transformación, consultar la sección Anexo 1.

$$\sum_{i \in C} \sum_{j \in N} \mu x_{ijk} \leq \frac{2Q + z_\alpha^2 \pi - \sqrt{z_\alpha^4 \pi^2 + 4Q z_\alpha^2 \pi}}{2} = \bar{Q}, \quad \forall k \in V \quad (15)$$

Así, considerando las restricciones (15) y las demás restricciones presentadas en el apartado anterior, el problema con demanda estocástica puede ser resuelto como uno lineal determinístico. Sin embargo, si bien este modelo es capaz de encontrar soluciones exactas en instancias pequeñas (hasta 25 nodos) como se muestra en (Raggio, 2020), su eficiencia disminuye en instancias más grandes. Por lo tanto, es necesario el uso de otros métodos para encontrar buenas soluciones factibles.

4.4 Metaheurísticas propuestas

Una alternativa para encontrar soluciones factibles en tiempos computacionales bajos en problemas NP-difícil (como los problemas de ruteo de vehículos), es mediante métodos metaheurísticos (Elshaer & Awad, 2020). Es así que se proponen dos algoritmos metaheurísticos de búsqueda local: *Iterated local search* (ILS) y *simulated annealing* (SA).

En el Algoritmo 1 se encuentra el pseudocódigo de la *Iterated local search* basado en lo presentado en Grasas et al. (2014). En esta metaheurística, a partir de una solución inicial s_0 , se aplican búsquedas locales con el fin de encontrar una mejor solución s (líneas 1-3). Luego, se entra a un ciclo en el cual se perturba aleatoriamente a la solución s bajo una probabilidad φ , donde el procedimiento de perturbación escogido depende de una probabilidad κ (líneas 5-11). Luego, se aplican procedimientos de búsquedas locales, obteniendo una solución s^* (líneas 12-13). Si el costo de s^* es menor al mejor costo hasta ahora s_m , entonces esta solución pasa a ser la mejor hasta ahora. Además, si el costo de esta solución s^* es menor a la suma entre el mejor

costo s_m y el costo fijo multiplicado por γ (es decir, se permite que la solución empeore en un vehículo), se escoge como solución s y se vuelve a repetir el ciclo. De otro modo, se selecciona como solución s a la mejor solución hasta ahora s_m (líneas 14-19). El procedimiento se repite hasta cumplir un tiempo máximo de ejecución.

Algoritmo 1: Seudocódigo *iterated local search*.

```

1:   $s_0 = \text{generar\_solución\_inicial}$ 
2:   $s = \text{busqueda\_local1}(s_0)$ 
3:   $s = \text{busqueda\_local2}(s)$ 
4:  While Tiempo ejecución < Tiempo máximo:
5:      If  $\text{rand}(0,1) < \varphi$ :
6:          If  $\text{rand}(0,1) < \kappa$ :
7:               $s' = \text{Perturbación1}(s)$ 
8:          Else:
9:               $s' = \text{Perturbación2}(s)$ 
10:         Else:
11:              $s' = s$ 
12:          $s^* = \text{Busqueda\_local1}(s')$ 
13:          $s^* = \text{Busqueda\_local2}(s^*)$ 
14:         If  $\text{costo}(s^*) < \text{costo}(s_m)$ :
15:              $s_m = s^*$ 
16:         If  $\text{costo}(s^*) < \text{costo}(s_m) + \gamma * f$ :
17:              $s = s^*$ 
18:         Else:
19:              $s = s_m$ 
20:  End

```

Fuente: Elaboración propia basado en Grasas et al. (2014).

En el Algoritmo 2 se muestra el pseudocódigo para *simulated annealing*, basado en Wei et al. (2018) y Normasari et al. (2019). En esta metaheurística, a partir de una solución inicial s_0 , se aplican búsquedas locales con el fin de encontrar una mejor solución s , y se determina una temperatura inicial T_0 (líneas 1-4). Luego se ingresa a un ciclo, en el cual se perturba aleatoriamente a la solución s , donde el procedimiento de perturbación depende de una probabilidad χ (líneas 7-10). Con el fin de encontrar mejores soluciones sin empeorar el tiempo computacional, se aplica un procedimiento de búsqueda local con una probabilidad ψ , obteniendo una solución s^* (líneas 11-12). Si el costo de s^* es menor al mejor costo hasta ahora s_m , entonces esta solución pasa a ser la mejor hasta ahora (s_m). Luego si el costo de s^* es menor

al costo de s , se escoge como nueva solución s . Por otro lado, si el costo de s^* es mayor al costo de s , se escoge como nueva solución s con una probabilidad p , la cual depende de la temperatura T (líneas 13-18). El proceso se vuelve a repetir por una cierta cantidad de iteraciones. Una vez se cumplan estas, se disminuye la temperatura multiplicándola por un parámetro β y se vuelven a aplicar dos procedimientos de búsquedas locales a la solución s (líneas 20-22). La solución resultante se selecciona como nueva solución s y se vuelve a entrar al ciclo de iteraciones. El procedimiento se repite hasta que cumplir un tiempo máximo de ejecución.

Algoritmo 2: Seudocódigo *simulated annealing*.

```

1:   $s_0 = \text{generar\_solución\_inicial}$ 
2:   $s = \text{busqueda\_local1}(s_0)$ 
3:   $s = \text{busqueda\_local2}(s)$ 
4:   $T = T_0$ 
5:  While Tiempo ejecución < Tiempo máximo:
6:      For  $i$  in iteraciones:
7:          If  $\text{rand}(0,1) < \chi$ :
8:               $s^* = \text{Perturbación1}(s)$ 
9:          Else:
10:              $s^* = \text{Perturbación2}(s)$ 
11:             If  $\text{rand}(0,1) < \psi$ :
12:                  $s^* = \text{busqueda\_local2}(s^*)$ 
13:             If  $\text{costo}(s^*) < \text{costo}(s_m)$ :
14:                  $s_m = s^*$ 
15:             If  $\text{costo}(s^*) < \text{costo}(s)$ :
16:                  $s = s^*$ 
17:             Else:
18:                  $s = s^*$  con probabilidad  $p, p = \exp\left(-\frac{\text{costo}(s^*) - \text{costo}(s)}{T}\right)$ 
19:         End For
20:          $T = T * \beta$ 
21:          $s = \text{busqueda\_local1}(s)$ 
22:          $s = \text{busqueda\_local2}(s)$ 
23:  End

```

Fuente: Elaboración propia basado en Wei et al. (2018) y Normasari et al. (2019).

Además, se proponen dos enfoques para representar la aleatoriedad de la demanda en estas metaheurísticas:

- El primer enfoque (desde ahora, enfoque A) considera el problema lineal determinístico derivado de la metodología CCP, verificando la capacidad de un vehículo mediante las ecuaciones (15). Así, este enfoque puede ser comparado con un método exacto que

resuelva el modelo de optimización matemático formulado, debido a que consideran la misma metodología.

- El segundo enfoque (desde ahora, enfoque B) considera el conjunto de ecuaciones (16) y (17) para verificar una ruta, utilizando muestras aleatorias de la distribución de probabilidad. Este enfoque es planteado con el fin de representar la demanda de una manera más aleatoria. Además, comprobar como varían los resultados al considerar una metodología distinta al modelo de optimización matemático.

$$\sum_{i \in C} \sum_{j \in N} m_{ih} x_{ijk} \leq Q, \quad \forall k \in V, \forall h \in \text{cantidad de experimentos} \quad (16)$$

Donde m_{ih} corresponde a la demanda del nodo i en el experimento h .

$$\frac{\text{Cantidad de experimentos factibles}}{\text{Cantidad de experimentos totales}} \geq \alpha \quad \forall k \in V \quad (17)$$

A continuación, se detalla cada uno de estos enfoques.

4.4.1 Enfoque A

En el enfoque A se aplican las metaheurísticas al problema de optimización determinístico con demanda estocástica, resultante de aplicar la metodología CCP. Por lo tanto, se considera el conjunto de restricciones (15) para verificar si se cumple la restricción de capacidad de los vehículos. De este modo, la demanda de los puntos limpios pasa a ser la media de la distribución de probabilidad (μ) y la capacidad de los vehículos el parámetro \bar{Q} , el cual es calculado tal cual se muestra en el apartado anterior.

4.4.2 Enfoque B

El enfoque B se basa en lo presentado en Grasas et al. (2014), los cuales presentan una *sim iterated local Search* para problemas de optimización estocásticos. El propósito es verificar si una solución satisface las restricciones de capacidad de los vehículos con una cierta probabilidad. Esta verificación se realiza mediante muestreo aleatorio a una solución candidata, la cual es calculada considerando μ como demanda de los puntos limpios y Q como capacidad de los vehículos. Luego, se reemplaza la demanda de los puntos limpios por m_{ih} y se verifica la factibilidad de la ruta en el experimento h según el conjunto de ecuaciones (16). Finalmente se calcula la proporción de experimentos en los que se cumple la restricción de capacidad, según la ecuación (17). Si esta proporción es mayor o igual a α , la ruta es factible.

En la Tabla 4.3 se muestra un ejemplo para un vehículo que visita cuatro puntos limpios. Si $\alpha = 0,5$, entonces el vehículo debe recolectar los residuos electrónicos un 50% de las veces. Además, se considera: $\mu = 5 \text{ kg}$, $Q = 20 \text{ kg}$ y *cantidad de experimentos* = 4. Así, la solución candidata es factible, debido a que la cantidad de residuos recolectados considerando μ como demanda son 20 kg , igualando la capacidad máxima del vehículo. Por lo tanto, esta solución pasa al muestreo aleatorio. En la Tabla 4.3 se observa que en dos de las cuatro muestras se mantiene la factibilidad, verificándose que en el 50% de las veces el vehículo puede recolectar los residuos. Por lo tanto, esta solución es factible según el enfoque B.

Tabla 4.3: Ejemplo verificación de capacidad en el enfoque B.

	Puntos limpios				Residuos recolectados	Capacidad del vehículo	Factibilidad
	1	2	3	4			
Candidata	5 kg	5 kg	5 kg	5 kg	20 kg	20 kg	Si
Muestra 1	6 kg	4 kg	7 kg	4 kg	21 kg	20 kg	No
Muestra 2	8 kg	5 kg	5 kg	4 kg	22 kg	20 kg	No
Muestra 3	2 kg	5 kg	6 kg	3 kg	16 kg	20 kg	Si
Muestra 4	5 kg	6 kg	2 kg	1 kg	14 kg	20 kg	Si

Fuente: Elaboración propia.

Si bien, en el trabajo de Grasas et al. (2014) el muestreo aleatorio se realiza después de encontrar una solución en la búsqueda local y/o perturbación, en este estudio se incluye dentro de estos algoritmos, con el fin de evitar que estos se estancuen y no encuentren soluciones factibles. En el Algoritmo 3 se muestra cómo se realiza el muestreo aleatorio.

Algoritmo 3: Testeo muestreo aleatorio.

Entrada: Solución factible utilizando μ como demanda, cantidad de experimentos, α

Salida: Si la solución es factible según el muestreo aleatorio

- 1: **While** Todas las rutas sean verificadas:
- 2: seleccionar ruta en el vector solución, $count \leftarrow 0$
- 3: **For** cantidad de experimentos:
- 4: **For** cada ciudad en la ruta:
- 5: generar demanda para la ciudad según la distribución $N(\mu, \sigma)$
- 6: **End For**
- 7: **If** la ruta con las demandas generadas es factible:
- 8: $count \leftarrow count + 1$
- 9: **End For**
- 10: **If** $\frac{count}{\text{cantidad de experimentos}} \geq \alpha$:
- 11: Ruta factible, se pasa a la siguiente
- 12: **Else:**
- 13: Ruta infactible, termina el muestreo aleatorio
- 14: **End**

Fuente: Elaboración propia en base a lo expuesto en Grasas et al. (2014).

4.5 Elementos de las metaheurísticas

A continuación, se presentan los principales elementos de las metaheurísticas.

4.5.1 Representación de la solución

Una solución factible para el problema de ruteo de vehículos presentado, corresponde a un conjunto de rutas, una para cada vehículo, en las que se visiten exactamente una vez todos los puntos limpios y que se respeten las restricciones de capacidad y ventanas de tiempo. Esta se representa a través de un vector que contiene cada una de las rutas como se muestra a continuación:

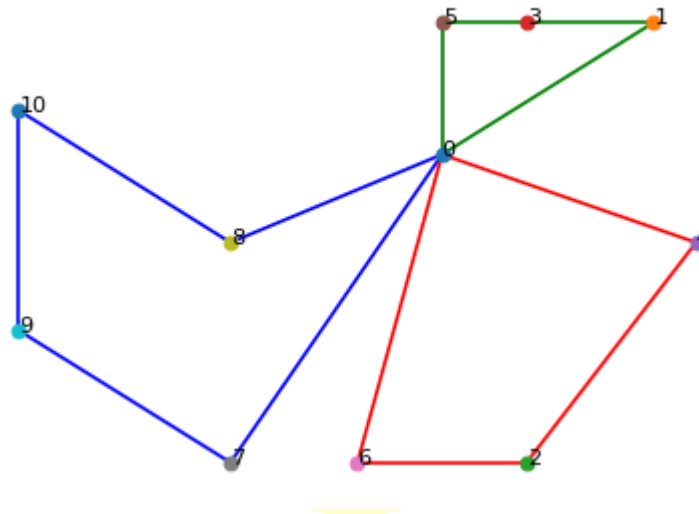


Figura 4.1: Ejemplo solución encontrada.
Fuente: Elaboración propia

$$S = [0,8,10,9,7,0,6,2,4,0,5,3,1,0]$$

$$Ruta\ 1: [0,8,10,9,7,0]$$

$$Ruta2: [0,6,2,4,0]$$

$$Ruta3: [0,5,3,1,0]$$

4.5.2 Función de evaluación

La función de evaluación utilizada se muestra en la ecuación (18), la cual para un vector solución, se calcula el costo de recorrer cada nodo, el costo de emisiones en cada ruta y el costo de los vehículos utilizados.

$$F(S) = \sum_{i=0}^{\text{Largo de } S-2} \{c_{(S[i],S[j+1])} + \text{CosCO2}_{(S[i],S[j+1])}\} + |V| * f \quad (18)$$

4.5.3 Solución inicial

Debido a la complejidad de las ventanas de tiempo y a la existencia de un costo fijo por la utilización de vehículos, se propone una heurística golosa que busque el siguiente nodo en el cual se produzca el menor aumento en el tiempo transcurrido, con el fin de que las rutas sean compactas, evitando la utilización de un vehículo extra. El pseudocódigo de este procedimiento se muestra en el Algoritmo 4. Esta heurística partiendo del nodo depósito (nodo 0), busca la ciudad j que genere el menor tiempo de finalización de servicio en este nodo. Luego, se repite este procedimiento hasta que el vehículo no pueda realizar el servicio a otro punto limpio, ya sea por su capacidad o porque el tiempo no es suficiente. Cuando ocurre esta situación, se escoge como siguiente nodo el depósito y se repite el procedimiento hasta que todos los puntos limpios sean atendidos. El proceso de verificación de factibilidad de la capacidad, mostrado en la línea 4, se realiza con la ecuación (15) para el enfoque A. Por otro lado, en el enfoque B se realiza en mediante muestreo aleatorio.

Algoritmo 4: Heurística para encontrar la solución inicial.

```
1:  $A \leftarrow \{0\}, B \leftarrow \{0\}$ 
2:  $C \leftarrow \{1,2,3, \dots, n\}$ 
3: While  $|A| \leq$  número de ciudades:
4:      $k \leftarrow$  Seleccionar ciudad  $j \in C$  que genere el menor tiempo de finalización
        de servicio y que la solución  $B \cup \{k, 0\}$  sea factible.
5:     If  $k = \emptyset$ 
6:          $B \leftarrow \{0\}, A \cup \{0\}$ 
7:         número de ciudades  $\leftarrow$  número de ciudades + 1
8:     Else
9:          $A \leftarrow A \cup \{k\}, B \leftarrow B \cup \{k\}$ 
10:         $C \leftarrow C \setminus \{k\}$ 
11: End While
12: Return  $A$ 
```

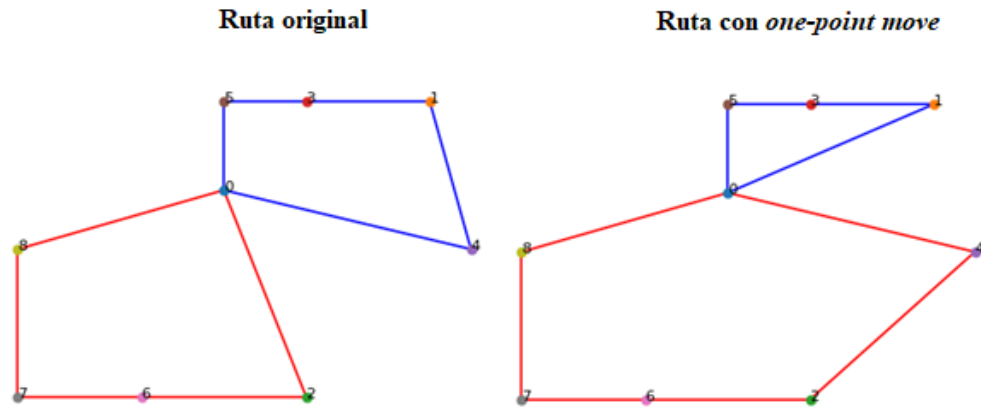
*El proceso de verificación de factibilidad depende del enfoque.

Fuente: Elaboración propia.

4.5.4 Estrategia de búsqueda local

Las estrategias de búsqueda local son utilizadas para mejorar la solución actual, explorando el vecindario de una solución, generando nuevas soluciones factibles mediante operaciones específicas tales como el intercambio de clientes a una nueva ruta (Groër et al., 2010). En los algoritmos propuestos se trabaja con dos algoritmos de búsqueda local:

- El primero corresponde a *one-point move*, en el cual se traslada un nodo específico a una nueva locación. En el caso de esta búsqueda local, se realizan todos los traslados posibles y se elige el que genere el menor costo. Este es mostrado en la Figura 4.2 y su pseudocódigo en el Algoritmo 5.
- El segundo corresponde a *two-point move*, en el que se intercambia la posición de dos nodos. A diferencia de la primera búsqueda local y de modo de que no aumente demasiado el tiempo computacional, esta búsqueda local se detiene al encontrar el primer intercambio que mejore el costo actual. Este es mostrado en la Figura 4.3 y su pseudocódigo en el Algoritmo 6.



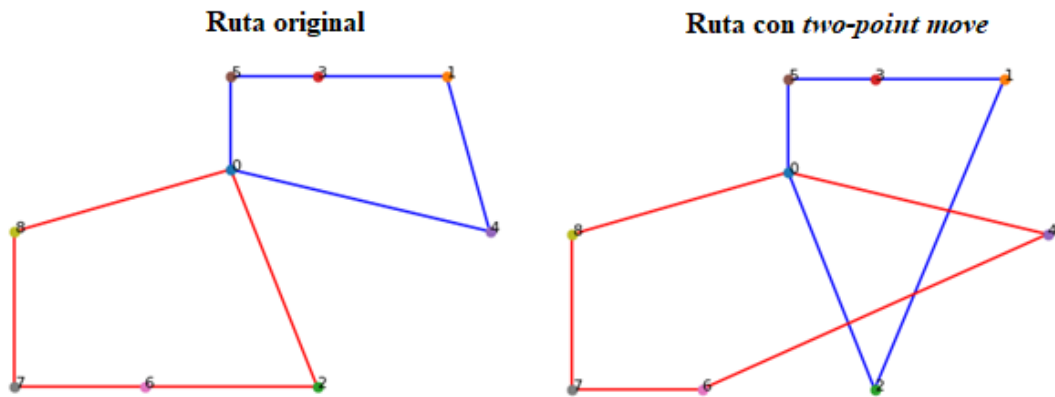
Inserción nodo 4 entre nodos 0 y 2.
 Figura 4.2: Ejemplo búsqueda local *one-point move*.
 Fuente: Elaboración propia.

Algoritmo 5: Búsqueda local *one-point move*.

```

Entrada:  $S$ 
1:  $N \leftarrow \{1,2,3,4..n\}$ 
2:  $S' \leftarrow S, S'' \leftarrow S$ 
3: For  $i$  en  $N$ 
4:   For  $j$  en  $(1, \text{largo de } S - 1)$ 
5:      $S \leftarrow \text{Insertar el nodo } i \text{ en la posición } j \text{ de } S, \text{ si y solo si } S[j] \neq i$ 
6:     If  $f(S) < f(S')$  y  $S$  es una solución factible:
7:        $S' \leftarrow S$ 
8:        $S \leftarrow S''$ 
9:   End For
10: End For
11: return  $S'$ 
  
```

*El proceso de verificación de factibilidad depende del enfoque.
 Fuente: Elaboración propia.



Intercambio nodos 4 y 2.
 Figura 4.3: Ejemplo búsqueda local *two-point move*.
 Fuente: Elaboración propia.

Algoritmo 6: Búsqueda local *two-point move*.

Entrada: S
1: $N \leftarrow \{1,2,3,4..n\}$
2: $S' \leftarrow S, S'' \leftarrow S$
3: **For** i en N
4: **For** j en N
5: $S \leftarrow$ Se intercambia el nodo i y j , si $i \neq j$
6: **If** $f(S) < f(S')$ y S es una solución factible:
7: $S' \leftarrow S$
8: **Return** S'
9: $S \leftarrow S''$
10: **End For**
11: **End For**
12: **Return** S'

*El proceso de verificación de factibilidad depende del enfoque.

Fuente: Elaboración propia.

4.5.5 Estrategia de perturbación

Las estrategias de perturbación son utilizadas para escapar de los óptimos locales. A diferencia de los procedimientos de búsqueda local que aceptan movimientos que mejoren la solución actual, las perturbaciones alteran la solución de manera aleatoria, sin importar si esta empeora. Se proponen dos estrategias de perturbación:

- En la primera perturbación se realizan ν intercambios aleatorios de pares de puntos limpios. Estos intercambios se realizan siempre y cuando se mantenga la factibilidad de la solución. Además, cada par de nodos seleccionado aleatoriamente, ya sea intercambiado o no, es guardado en una lista para que no sean seleccionados nuevamente. Este procedimiento es mostrado con más detalle en el Algoritmo 7.
- En la segunda perturbación se realizan ν traslados aleatorios de puntos limpios a una nueva posición en el vector solución. Al igual que la primera perturbación, se debe mantener la factibilidad con los traslados. Además, cada vez que se traslada un nodo, este es guardado en una lista para que no sea trasladado nuevamente. Este procedimiento es mostrado con más detalle en el Algoritmo 8.

Algoritmo 7: Primera perturbación, intercambio aleatorio de nodos.

Entrada: S, v

- 1: $S' \leftarrow S, \text{Intercambios} \leftarrow \{\}$
- 2: **While** $v > 0$:
- 3: *Escoger aleatoriamente dos nodos i y j , tal que $i \neq j, i, j \neq 0$ y $(i, j) \notin \text{Intercambios}$*
- 4: $\text{Intercambios} \cup \{(i, j) \text{ y } (j, i)\}$
- 5: $S \leftarrow$ intercambiar los nodos i, j en S
- 6: **If** S es factible:
- 7: $S' \leftarrow S$
- 8: $v \leftarrow v - 1$
- 9: **Else:**
- 10: $S \leftarrow S'$
- 11: **End While**
- 12: **Return** S'

*El proceso de verificación de factibilidad depende del enfoque.
Fuente: Elaboración propia.

Algoritmo 8: Segunda perturbación, inserción aleatoria.

Entrada: S, v

- 1: $S' \leftarrow S, \text{nodos} \leftarrow \{\}$
- 2: **While** $v > 0$:
- 3: *Escoger aleatoriamente un nodo i y una posición j de S , tal que $i \neq 0$ y $i \notin \text{nodos}$:*
- 4: $S \leftarrow$ se inserta el nodo i en la posición j en S
- 5: **If** S es factible:
- 6: $S' \leftarrow S$
- 7: $\text{nodos} \cup \{i\}$
- 8: $v \leftarrow v - 1$
- 9: **Else:**
- 10: $S \leftarrow S'$
- 11: **End While**
- 12: **Return** S'

*El proceso de verificación de factibilidad depende del enfoque.
Fuente: Elaboración propia.

4.5.6 Probabilidad de aceptación en SA

Una característica importante de la metaheurística SA, es que puede aceptar movimientos que empeoren la solución actual según una cierta probabilidad (p), de forma de evitar quedar atrapado en óptimos locales (Wei et al., 2018). Esta probabilidad p es calculada según la ecuación (19), donde S^* es la solución candidata peor que S , y T corresponde a la temperatura, la cual disminuye después de una cierta cantidad de iteraciones según la ecuación (20). Así, a medida que la temperatura disminuye, también lo hace la probabilidad de aceptar peores soluciones.

$$p = \exp\left(-\frac{\text{costo}(S^*) - \text{costo}(S)}{T}\right) \quad (19)$$

$$T = T * \beta \quad (20)$$

Además, es necesario fijar una temperatura inicial T_0 al principio del procedimiento. Esta temperatura es determinada con la ecuación (19), bajo el criterio de que una solución peor que la solución inicial en w pesos chilenos, tenga un 50% de ser aceptada (Wei et al., 2018). Por lo tanto, en la ecuación (21) se muestra como es calculada esta temperatura.

$$T_0 = -\frac{w}{\ln(0,5)} \quad (21)$$

CAPÍTULO 5. INSTANCIAS DE PRUEBA Y RESULTADOS

En el siguiente capítulo se detallan las instancias, configuración y hardware utilizado.

5.1 Instancias de prueba

Para validar el modelo y algoritmos propuestos, se utilizan como base las instancias para un problema de ruteo de vehículos con ventanas de tiempo de Solomon (1987). En la Tabla 5.1 se muestran las instancias consideradas, su tamaño, capacidad máxima de los vehículos, tiempo de servicio en cada nodo, media y desviación estándar de la demanda en cada nodo. Con tal de comprobar al rendimiento del método exacto y los algoritmos, se consideran instancias con 26, 51 y 101 nodos, donde uno de estos corresponde al nodo depósito y el resto a los puntos limpios.

Tabla 5.1: Instancias consideradas.

Instancia	Abreviación utilizada	Cantidad de nodos (n)	Capacidad máxima (Q)	Tiempo de servicio de cada nodo	Media de demanda (μ)	Desviación estándar demanda (σ)
R101.26	R1.1	26	200 kg	10 minutos	13,3 kg	7,15 kg
R101.51	R1.2	51	200 kg	10 minutos	14,4 kg	8,26 kg
R101.101	R1.3	101	200 kg	10 minutos	14,6 kg	8,83 kg
C101.26	C1.1	26	200 kg	90 minutos	18,4 kg	10,1 kg
C101.51	C1.2	51	200 kg	90 minutos	17,2 kg	9,39 kg
C101.101	C1.3	101	200 kg	90 minutos	18,1 kg	10,4 kg
RC101.26	RC1.1	26	200 kg	10 minutos	21,6 kg	10,1 kg
RC101.51	RC1.2	51	200 kg	10 minutos	19,4 kg	9,88 kg
RC101.101	RC1.3	101	200 kg	10 minutos	17,2 kg	9,37 kg
R201.26	R2.1	26	1.000 kg	10 minutos	13,3 kg	7,16 kg
R201.51	R2.2	51	1.000 kg	10 minutos	14,4 kg	8,26 kg
R201.101	R2.3	101	1.000 kg	10 minutos	14,6 kg	8,82 kg
C201.26	C2.1	26	700 kg	90 minutos	18,4 kg	10,1 kg
C201.51	C2.2	51	700 kg	90 minutos	17,2 kg	9,39 kg
C201.101	C2.3	101	700 kg	90 minutos	18,1 kg	10,4 kg
RC201.26	RC2.1	26	1.000 kg	10 minutos	21,6 kg	10,1 kg
RC201.51	RC2.2	51	1.000 kg	10 minutos	19,4 kg	9,88 kg
RC201.101	RC2.3	101	1.000 kg	10 minutos	17,2 kg	9,37 kg

Fuente: Elaboración propia en base a instancias de Solomon (1987).

Además, se consideran tres niveles de α : 0,5, 0,75 y 0,9. En la Tabla 5.2 se muestra el valor de \bar{Q} dependiendo de la instancia y el valor de α , calculado según la ecuación (15).

Tabla 5.2: Valor de \bar{Q} según la instancia y α .

Instancia	Valor de α		
	0,5	0,75	0,90
R1.1	200 kg	182 kg	167 kg
R1.2	200 kg	180 kg	164 kg
R1.3	200 kg	179 kg	162 kg
C1.1	200 kg	179 kg	162 kg
C1.2	200 kg	180 kg	163 kg
C1.3	200 kg	178 kg	160 kg
RC1.1	200 kg	180 kg	164 kg
RC1.2	200 kg	180 kg	163 kg
RC1.3	200 kg	180 kg	163 kg
R2.1	1000 kg	959 kg	923 kg
R2.2	1000 kg	955 kg	916 kg
R2.3	1000 kg	952 kg	911 kg
C2.1	700 kg	659 kg	625 kg
C2.2	700 kg	661 kg	627 kg
C2.3	700 kg	658 kg	622 kg
RC2.1	1000 kg	955 kg	916 kg
RC2.2	1000 kg	953 kg	913 kg
RC2.3	1000 kg	953 kg	913 kg

Fuente: Elaboración propia.

La distancia entre nodos (dis_{ij}), inicio y final de la ventana de tiempo de cada nodo (a_i y b_i) depende de la instancia de estudio, disponibles en Solomon (1987). Además, en estas instancias se asume que el tiempo de viaje (t_{ij}) es igual a la distancia entre cada nodo (dis_{ij}). El resto de los parámetros considerados en el modelo de optimización se encuentran en la Tabla 5.3, los cuales fueron definidos con tal de representar la realidad del Gran Concepción y basados en Wu et al. (2020a).

Tabla 5.3: Parámetro considerados en el problema.

Parámetro	Valor
α	0,5, 0,75 y 0,90
f	\$500.000
Fac_{co2}	$2,63 \frac{Kg}{L}$
Im_{co2}	$3,639 \frac{\$}{kg}$
pc	$655 \frac{\$}{L}$
r	$2 \frac{km}{L}$

Fuente: Elaboración propia.

5.2 Parámetro de las metaheurísticas

En relación a los parámetros utilizados en cada metaheurística, estos son extraídos de estudios similares en la literatura. Sin embargo, existen otros parámetros que resulta difícil extraerlos de algún estudio o que necesitan ser ajustados al problema actual.

Por lo cual, se realiza un ajuste de manera experimental al igual que en Li et al. (2015), testeando el valor de un parámetro específico, mientras los demás son dejado fijos. Para esto, se utilizan las siguientes instancias: R1.2, R1.3, RC1.2, RC1.3, R2.2 y R2.3, con $\alpha = 0,75$. Cada experimento es ejecutado 10 veces en cada instancia. El tiempo máximo de cada algoritmo se limita a 45 segundos. Finalmente, se seleccionan los valores que mejor costo promedio generen.

A continuación, se revisan los parámetros según cada metaheurística y/o enfoque considerado.

5.2.1 Parámetros ILS

Se necesitan determinar los siguientes parámetros en la metaheurística ILS: probabilidad de perturbación φ , probabilidad de elección de cada perturbación κ , cantidad de nodos perturbados ν y el parámetro que multiplica al costo fijo en el criterio de aceptación γ . Estos son determinados de manera experimental, mostrando su resultado en la Tabla 5.4.

Tabla 5.4: Parámetros ILS.

Parámetro	Rango probado	Valor
φ	[0,05-1]	0,2
κ	[0-1]	0,6
ν	[1-10]	5
γ	[0,1-2]	1,5

Fuente: Elaboración propia.

5.2.2 Parámetros SA

Con respecto a la metaheurística SA, se necesita determinar el parámetro con el cual decae la temperatura β , el parámetro w necesario para determinar la solución inicial, cantidad de iteraciones, probabilidad de elección de cada metaheurística χ y probabilidad de búsqueda local ψ .

El parámetro β es fijado en 0,9, según el estudio de Wei et al. (2018). Los parámetros w , χ , ψ y cantidad de iteraciones, son determinados experimentalmente, cuyos valores se muestran en la Tabla 5.5.

Tabla 5.5: Parámetros SA testeados experimentalmente.

Parámetro	Rango probado	Valor
w	[1.000, 20.000]	3.000
χ	[0-1]	0,7
ψ	[0-1]	0,2
ν	[1-10]	1
Cantidad de iteraciones	[500-10.000]	1.000

Fuente: Elaboración propia.

5.2.3 Parámetros muestreo aleatorio

En el muestreo aleatorio, es necesario determinar la cantidad de experimentos que se realizan para verificar el cumplimiento de las restricciones de capacidad. En este caso, para la solución inicial, se define que se realicen 1000 experimentos. Por otro lado, para cada elemento de la metaheurística, se determina que se realicen 75 experimentos, debido a que el empeoramiento de las soluciones aumenta considerablemente al aumentar el número de experimentos.

5.3 Configuración y *hardware* utilizados

El método exacto es resuelto mediante el *solver* CPLEX 12.9.0. Los algoritmos metaheurísticos son implementados en Python 3.7.1. Este *solver* y metaheurísticas son implementadas en un computador con sistema operativo Windows 10 home 64 bits, procesador Intel Core i5-7200U @ 2,5GHz y 12GB de memoria RAM.

Con respecto al tiempo máximo para encontrar una solución, se definen 3.200 segundos en CPLEX y 200 segundos para los algoritmos metaheurísticos. Además, se realizan 10 experimentos por instancia en cada metaheurística, obteniendo el mejor resultado y el promedio de estos.



CAPÍTULO 6. RESULTADOS Y DISCUSIÓN

En el presente capítulo se muestran los resultados obtenidos y se realiza la discusión sobre estos.

6.1 Resultados

Los resultados encontrados son presentados en las Tablas 6.1, 6.2 y 6.3 según nivel de α . Cada tabla se divide según el método considerado. Además, la última fila muestra el promedio de cada columna. En los resultados del modelo exacto se muestran los costos totales, tiempo de resolución y GAP encontrados con CPLEX. Los promedios de los costos totales y GAP de CPLEX se realizaron con 16 de las 18 instancias, debido a que CPLEX no encontró soluciones en todas las instancias. Un GAP mayor a cero, indica que podría existir una solución factible mejor a la encontrada. Por ejemplo, en la instancia RC1.3 para $\alpha = 0,5$, se encontró un costo total de 10.707.826 con un GAP de 75,6%. Esto significa que podría existir una solución mejor a esta, pero de ser así, esta no será mejor que 2.612.709. Esta cota es calculada según la ecuación (22).

$$0,756 = 1 - \frac{\text{cota}}{10.707.826} \quad (22)$$

En los resultados de los métodos metaheurísticos, se presenta el mejor costo y el costo promedio encontrados en los diez experimentos realizados. Además, se muestra un GAP de estos costos respecto al costo encontrado con CPLEX. Este GAP es calculado según la ecuación (23). Un GAP negativo, indica que la metaheurística encuentra una solución con menor costo que CPLEX.

$$GAP = \frac{(\text{Costo metaheurística} - \text{Costo de CPLEX})}{\text{Costo de Cplex}} \quad (23)$$

En el Anexo 2 se muestra una demostración que las soluciones obtenidas cumplen con las restricciones establecidas, por lo tanto, la implementación es correcta.

Tabla 6.1: Resultados encontrados para $\alpha=0,5$.

Instancia	Modelo exacto (CPLEX)			ILS-A				SA-A				ILS-B				SA-B			
	Costo total	Tiempo	GAP CPLEX (%)	Mejor costo	GAP mejor costo (%)	Costo promedio	GAP costo promedio (%)	Mejor costo	GAP mejor costo (%)	Costo promedio	GAP costo promedio (%)	Mejor costo	GAP mejor costo (%)	Costo promedio	GAP costo promedio (%)	Mejor costo	GAP mejor costo (%)	Costo promedio	GAP costo promedio (%)
R1.1	4.205.462	3,69	0	4.205.462	0	4.205.462	0	4.205.462	0	4.205.462	0	4.205.462	0	4.205.462	0	4.205.462	0	4.205.933	0
R1.2	5.865.753	21,14	0	6.347.803	8,2	6.349.801	8,3	6.347.803	8,2	6.348.977	8,2	6.349.563	8,2	6.351.338	8,3	6.350.594	8,3	6.439.686	9,8
R1.3	10.047.597	10,46	0	10.557.735	5,1	10.731.539	6,8	10.556.484	5,1	10.718.015	6,7	10.579.398	5,3	10.931.320	8,8	10.583.900	5,3	10.903.078	8,5
RC1.1	2.153.568	77	0	2.153.568	0	2.153.568	0	2.153.568	0	2.153.568	0	2.153.568	0	2.153.568	0	2.153.568	0	2.153.568	0
RC1.2	4.819.615	3.600	40,8	4.817.899	0	4.818.818	0	4.314.201	-10,5	4.720.787	-2,1	4.818.526	0	4.821.963	0	4.820.224	0	4.821.498	0
RC1.3	10.707.826	3.600	75,6	8.573.438	-19,9	8.850.454	-17,3	8.585.306	-19,8	9.111.762	-14,9	8.582.925	-19,8	8.992.994	-16	8.569.310	-20	9.246.714	-13,6
C1.1	1.564.210	3,44	0	1.564.210	0	1.564.210	0	1.564.210	0	1.564.210	0	1.564.210	0	1.564.210	0	1.564.210	0	1.564.210	0
C1.2	2.620.701	14,63	0	2.620.701	0	2.620.701	0	2.620.701	0	2.620.749	0	2.620.701	0	2.620.701	0	2.620.701	0	2.620.939	0
C1.3	5.277.333	139	0	5.277.333	0	5.278.911	0	5.277.333	0	5.278.521	0	5.278.094	0	5.291.544	0,3	5.278.590	0	5.282.044	0,1
R2.1	1.174.004	319	0	1.174.004	0	1.174.488	0	1.174.004	0	1.174.584	0	1.174.004	0	1.174.488	0	1.174.004	0	1.178.585	0,4
R2.2	2.784.502	3.600	72,9	1.801.903	-35,3	1.808.870	-35	1.791.305	-35,7	1.801.543	-35,3	1.791.007	-35,7	1.803.448	-35,2	1.791.908	-35,6	1.795.587	-35,5
R2.3	-	3.600	-	2.951.973	-100	2.991.755	-100	2.926.408	-100	2.953.080	-100	2.943.782	-100	2.986.417	-100	2.949.631	-100	2.985.493	-100
RC2.1	1.143.646	619	0	1.143.646	0	1.143.646	0	1.143.646	0	1.146.248	0,2	1.143.646	0	1.143.646	0	1.143.646	0	1.157.245	1,2
RC2.2	3.345.871	3.600	94,8	1.778.706	-46,8	1.803.788	-46,1	1.780.916	-46,8	1.785.141	-46,6	1.779.319	-46,8	1.790.382	-46,5	1.778.636	-46,8	1.782.338	-46,7
RC2.3	-	3.600	-	3.012.443	-100	3.066.421	-100	3.023.634	-100	3.098.303	-100	3.041.134	-100	3.070.082	-100	3.010.660	-100	3.037.076	-100
C2.1	1.071.622	3,88	0	1.071.622	0	1.071.622	0	1.071.622	0	1.071.622	0	1.071.622	0	1.071.622	0	1.071.622	0	1.074.763	0,3
C2.2	1.147.854	9,44	0	1.147.854	0	1.147.854	0	1.147.854	0	1.147.854	0	1.147.854	0	1.147.854	0	1.147.854	0	1.147.854	0
C2.3	1.696.565	3.600	5,1	1.696.565	0	1.696.565	0	1.696.565	0	1.696.565	0	1.696.565	0	1.696.565	0	1.696.566	0	1.696.566	0
Promedio	3.726.633	1.468	18,1	3.438.715	-16,0	3.471.026	-15,7	3.410.057	-16,6	3.477.611	-15,8	3.441.188	-16,0	3.489.867	-15,6	3.439.505	-16,0	3.505.177	-15,3

Fuente: Elaboración propia.

Tabla 6.2: Resultados obtenidos para $\alpha=0,75$.

Instancia	Modelo exacto (CPLEX)			ILS A				SA A				ILS B				SA B			
	Costo total	Tiempo	GAP CPLEX (%)	Mejor costo	GAP mejor costo (%)	Costo promedio	GAP costo promedio (%)	Mejor costo	GAP mejor costo (%)	Costo promedio	GAP costo promedio (%)	Mejor costo	GAP mejor costo (%)	Costo promedio	GAP costo promedio (%)	Mejor costo	GAP mejor costo (%)	Costo promedio	GAP costo promedio (%)
R1.1	4.205.462	2,33	0%	4.205.462	0	4.205.462	0	4.205.462	0	4.205.490	0	4.205.462	0	4.205.462	0	4.205.462	0	4.205.933	0
R1.2	5.865.753	14,51	0%	6.349.025	8,2	6.352.282	8,3	6.347.803	8,2	6.352.762	8,3	6.349.558	8,2	6.350.848	8,3	6.347.803	8,2	6.348.750	8,2
R1.3	10.047.597	796	0%	10.551.586	5	10.886.180	8,3	10.571.823	5,2	10.952.806	9	10.563.389	5,1	10.772.705	7,2	10.570.281	5,2	10.827.125	7,8
RC1.1	2.153.568	128	0%	2.153.568	0	2.153.568	0	2.153.568	0	2.153.568	0	2.153.568	0	2.153.568	0	2.153.568	0	2.153.568	0
RC1.2	4.821.198	3.600	43,4%	4.818.526	-0,1	4.820.723	0	4.314.718	-10,5	4.670.698	-3,1	4.817.606	-0,1	4.869.255	1,0	4.819.525	0	4.820.667	0
RC1.3	10.711.623	3.600	69,9%	8.573.918	-20	8.988.909	-16,1	8.579.997	-19,9	9.080.251	-15,2	8.587.629	-19,8	8.945.488	-16,5	8.582.474	-19,9	9.256.564	-13,6
C1.1	1.571.661	154	0%	1.571.661	0	1.571.661	0	1.571.661	0	1.571.661	0	1.571.661	0	1.571.661	0	1.571.661	0	1.571.716	0
C1.2	2.628.031	174	0%	2.628.031	0	2.628.031	0	2.628.031	0	2.628.031	0	2.624.025	-0,2	2.628.819	0	2.628.031	0	2.628.031	0
C1.3	6.318.690	3.600	7,6%	6.313.558	-0,1	6.328.667	0,2	6.824.098	8	6.851.573	8,4	6.326.623	0,1	6.674.447	5,6	6.829.016	8,1	6.886.000	9
R2.1	1.174.314	3.600	42,6%	1.174.004	0	1.174.141	0	1.174.004	0	1.174.141	0	1.174.004	0	1.174.004	0	1.175.060	0,1	1.177.100	0,2
R2.2	3.304.443	3.600	77,2%	1.791.889	-45,8	1.802.165	-45,5	1.791.107	-45,8	1.795.885	-45,7	1.795.588	-45,7	1.803.234	-45,4	1.790.836	-45,8	1.795.826	-45,7
R2.3	-	3.600	-	2.944.551	-100	3.012.932	-100	2.935.391	-100	2.972.048	-100	2.960.629	-100	3.001.622	-100	2.974.812	-100	2.990.111	-100
RC2.1	1.143.645	900	0%	1.143.645	0	1.143.645	0	1.143.645	0	1.143.645	0	1.143.645	0	1.143.645	0	1.143.645	0	1.154.025	0,9
RC2.2	2.807.588	3.600	91,4%	1.781.116	-36,6	1.795.168	-36,1	1.778.636	-36,6	1.782.466	-36,5	1.785.781	-36,4	1.798.106	-36,0	1.779.172	-36,6	1.783.664	-36,5
RC2.3	-	3.600	-	3.036.296	-100	3.067.526	-100	3.001.363	-100	3.022.789	-100	3.053.062	-100	3.137.136	-100	3.024.927	-100	3.053.792	-100
C2.1	1.071.622	5,52	0%	1.071.622	0	1.071.622	0	1.071.622	0	1.071.622	0	1.071.622	0	1.071.622	0	1.071.622	0	1.071.622	0
C2.2	1.147.854	21,41	0%	1.147.854	0	1.147.854	0	1.147.854	0	1.147.854	0	1.147.854	0	1.147.854	0	1.147.854	0	1.147.854	0
C2.3	1.696.565	3.600	5,59%	1.696.565	0	1.696.565	0	1.696.565	0	1.696.565	0	1.696.565	0	1.696.565	0	1.696.565	0	1.696.565	0
Promedio	3.791.851	1.922	21,1%	3.497.382	-16,1	3.547.061	-15,6	3.496.519	-16,2	3.570.770	-15,3	3.501.571	-16	3.563.669	-15,3	3.528.462	-15,6	3.587.162	-15

Fuente: Elaboración propia.

Tabla 6.3: Resultados obtenidos para $\alpha=0,9$.

Instancia	Modelo exacto (CPLEX)			ILS A				SA A				ILS B				SA B			
	Costo total	Tiempo	GAP CPLEX (%)	Mejor costo	GAP mejor costo (%)	Costo promedio	GAP costo promedio (%)	Mejor costo	GAP mejor costo (%)	Costo promedio	GAP costo promedio (%)	Mejor costo	GAP mejor costo (%)	Costo promedio	GAP costo promedio (%)	Mejor costo	GAP mejor costo (%)	Costo promedio	GAP costo promedio (%)
R1.1	4.205.462	2,15	0%	4.205.462	0	4.205.462	0	4.205.462	0	4.205.462	0	4.205.462	0	4.205.462	0	4.205.462	0	4.205.960	0
R1.2	5.865.753	11,29	0%	6.348.834	8,2	6.350.437	8,3	6.347.803	8,2	6.349.260	8,2	6.349.025	8,2	6.350.816	8,3	6.347.803	8,2	6.350.235	8,3
R1.3	10.047.597	3.600	4,86%	10.558.344	5,1	10.768.855	7,2	10.567.428	5,2	10.813.838	7,6	10.553.771	5	10.770.504	7,2	10.584.666	5,3	10.947.476	9
RC1.1	2.669.102	655	0%	2.669.102	0	2.669.102	0	2.669.102	0	2.669.102	0	2.153.568	-19,3	2.204.155	-17,4	2.154.048	-19,3	2.558.572	-4,1
RC1.2	4.819.553	3.600	43,5%	4.818.139	0	5.121.959	6,3	4.315.454	-10,5	4.924.372	2,2	4.820.922	0	5.272.914	9,4	4.821.045	0	5.226.895	8,5
RC1.3	10.787.825	3.600	77,8%	8.571.701	-20,5	8.989.468	-16,7	9.082.245	-15,8	9.141.634	-15,3	8.584.047	-20,4	9.047.306	-16,1	9.116.236	-15,5	9.406.807	-12,8
C1.1	2.074.126	27,42	0%	2.074.126	0	2.074.126	0	2.074.126	0	2.074.126	0	1.571.232	-24,2	1.572.534	-24,2	1.574.419	-24,1	2.031.840	-2
C1.2	3.142.193	3.600	0,48%	3.142.193	0	3.144.626	0,1	3.647.416	16,1	3.647.611	16,1	3.137.522	-0,1	3.247.762	3,4	3.142.193	0	3.395.263	8,1
C1.3	6.875.569	3.600	4,86%	6.882.631	0,1	7.552.345	9,8	7.370.404	7,2	7.486.503	8,9	7.350.217	6,9	7.568.181	10,1	7.857.568	14,3	8.082.430	17,6
R2.1	1.174.314	3.600	42,5%	1.174.004	0	1.174.035	0	1.174.004	0	1.174.035	0	1.174.004	0	1.174.699	0	1.174.004	0	1.174.140	0
R2.2	2.803.033	3.600	73,0%	1.792.436	-36,1	1.803.941	-35,6	1.792.054	-36,1	1.802.868	-35,7	1.798.259	-35,8	1.810.117	-35,4	1.793.506	-36	1.797.992	-35,9
R2.3	-	3.600	-	2.965.101	-100	2.995.387	-100	2.974.040	-100	2.985.377	-100	2.958.422	-100	2.986.970	-100	3.044.134	-100	3.076.427	-100
RC2.1	1.143.646	968	0%	1.143.646	0	1.143.646	0	1.143.646	0	1.143.646	0	1.143.646	0	1.143.710	0	1.143.646	0	1.143.645	0
RC2.2	2.918.532	3.600	92,0%	1.781.116	-39	1.803.680	-38,2	1.781.046	-39	1.789.449	-38,7	1.782.430	-38,9	1.797.066	-38,4	1.778.765	-39,1	1.794.030	-38,5
RC2.3	-	3.600	-	3.019.030	-100	3.070.119	-100	2.988.581	-100	3.013.146	-100	3.025.906	-100	3.089.812	-100	3.064.131	-100	3.137.513	-100
C2.1	1.071.622	3,42	0%	1.071.622	0	1.071.622	0	1.071.622	0	1.071.622	0	1.071.622	0	1.071.622	0	1.071.622	0	1.071.622	0
C2.2	1.147.854	24,19	0%	1.147.854	0	1.147.854	0	1.147.854	0	1.147.854	0	1.147.854	0	1.147.854	0	1.147.854	0	1.147.854	0
C2.3	1.708.122	3.600	5,62%	1.706.519	-0,1	1.706.724	-0,1	1.706.519	-0,1	1.706.519	-0,1	1.696.566	-0,7	1.700.821	-0,4	1.706.519	-0,1	1.807.369	5,8
Promedio	3.903.394	2.294	21,5%	3.615.103	-15,7	3.710.744	-14,4	3.669.934	-14,7	3.730.357	-13,7	3.584.693	-17,7	3.675.684	-16,3	3.651.535	-17	3.797.559	-13,1

Fuente: Elaboración propia.

6.2 Discusión de resultados

En la mayoría de instancias las metaheurísticas obtuvieron buenos resultados, igualando o superando las soluciones encontradas en CPLEX. Por ejemplo, en la instancia RC1.3, CPLEX encontró un costo de 10.707.826 para $\alpha=0,5$. Todas las metaheurísticas encontraron un costo mejor a este, donde ILS A obtuvo un mejor costo de 8.573.438, SA-A de 8.585.306, ILS-B de 8.582.925 y SA-B de 8.569.310.

El modelo implementado en CPLEX no encontró soluciones factibles en las instancias R2.3 y RC2.3 en ningún nivel de α , a diferencia de las metaheurísticas. El mejor costo y mejor costo promedio en la instancia R2.3, fueron encontrados por la metaheurística SA-A para $\alpha=0,5$ y $0,75$. Para $\alpha=0,9$, la metaheurística ILS-B encontró el mejor costo y SA-A el mejor costo promedio. Por otro lado, el mejor costo y mejor costo promedio en la instancia RC2.3, fueron encontrados por la metaheurística SA-B para $\alpha=0,5$. Para $\alpha=0,75$ y $0,9$, la metaheurística SA-A encontró el mejor costo y mejor costo promedio.

Los métodos metaheurísticos encontraron soluciones considerablemente peores a CPLEX en las instancias R1.2 y R1.3 en todos los niveles de α . Por ejemplo, CPLEX encontró una solución de 5.865.753 pesos chilenos en la instancia R1.2 con $\alpha=0,5$, la cual es mejor en 482.050 pesos chilenos que el mejor costo encontrado en ILS-A y SA-A en esta misma instancia. Esta situación se repite para $\alpha=0,75$ y $\alpha=0,9$.

La Tabla 6.4 resume las instancias en las que CPLEX encontró una mejor solución que las metaheurísticas. El mayor número de instancias en las que ocurre esta situación, son 4 de 18 en SA-A con $\alpha = 0,9$ y SA-B con $\alpha = 0,75$.

Tabla 6.4 Instancias donde el mejor costo de las metaheurísticas es peor que CPLEX.

Valor α	Enfoque A		Enfoque B	
	ILS	SA	ILS	SA
0,5	R1.2 y R1.3	R1.2 y R1.3	R1.2 y R1.3	R1.2 y R1.3
0,75	R1.2 y R1.3	R1.2, R1.3 y C1.3	R1.2, R1.3 y C1.3	R1.2, R1.3, C1.3 y R2.1
0,9	R1.2, R1.3 y C1.3	R1.2, R1.3, C1.2 y C1.3	R1.2, R1.3 y C1.3	R1.2, R1.3 y C1.3

Fuente: Elaboración propia.

Con respecto a la diferencia entre el mejor costo y costo promedio de las metaheurísticas, se observa igualdad entre ambos costos. Entre las instancias con mayor diferencia, se encuentra C1.3 con $\alpha=0,9$, donde la metaheurística ILS A obtuvo un mejor costo de 6.882.631 y el costo promedio fue de 7.552.345. Otro ejemplo es RC1.3 con $\alpha=0,75$, donde la metaheurística SA-B obtuvo un mejor costo de 8.582.474 y el costo promedio fue de 9.256.564. A pesar de estas instancias, en la mayoría se encuentran mejores costos y costos promedios parecidos, por lo tanto, las metaheurísticas encuentran soluciones similares en los diez experimentos realizados.

Se observa en las Tablas 6.2 y 6.3 que las metaheurísticas del enfoque B encuentra mejores soluciones que CPLEX en las instancias C1.2 para $\alpha=0.75$, RC1.1 y C1.1 para $\alpha=0.9$, a pesar que el GAP de CPLEX es cero. Estas instancias son resumidas en la Tabla 6.5. Esto se debe a que el enfoque B considera una metodología distinta al enfoque A y al método exacto para tratar la aleatoriedad de la demanda, haciendo uso de muestreo aleatorio. Estas muestras podrían ser considerablemente menores a la media de la distribución (ocupada como demanda en el método exacto), encontrándose soluciones más compactas y con menor costo. Por lo tanto, distintas formas de tratar la aleatoriedad pueden entregar distintas soluciones, mejorando incluso las soluciones óptimas encontradas con un enfoque específico. Sin embargo, destacar que esta mejora solo ocurrió en 3 de las 54 instancias consideradas.

Tabla 6.5: Instancias donde el enfoque B encontró mejores soluciones que CPLEX, 0% de GAP.

Nivel de α	Instancia	Costo CPLEX	GAP	Mejor costo ILS B	Costo promedio ILS B	Mejor costo SA B	Costo promedio SA B
0,75	C1.2	2.628.031	0%	2.624.025	2.628.819	2.628.031	2.628.031
0,9	RC1.1	2.669.102	0%	2.153.568	2.204.155	2.154.048	2.558.572
	C1.1	2.074.126	0%	1.571.232	1.572.533	1.574.419	2.031.839

Fuente: Elaboración propia.

Por otro lado, se puede observar que tanto el tiempo computacional, como el GAP de las soluciones encontradas con CPLEX, aumentan al incrementar la cantidad de nodos. En 27 instancias se alcanzó el límite de 3.600 segundos. De estas, 15 instancias consideran 101 nodos, 10 instancias 51 nodos y solo 2 instancias 26 nodos. Un ejemplo de este empeoramiento ocurre en la instancia C2 con $\alpha = 0,9$, donde el tiempo de cálculo aumentó de 3,42 segundos con 26 nodos a 24,19 segundos con 51 y a 3600 segundos con 101 nodos. Con respecto al GAP de CPLEX, notar la instancia RC1 con $\alpha = 0,5$, donde el GAP aumentó de 0% con 26 nodos, a

40,8% con 51 nodos y a 75,6% con 101 nodos. En las instancias en que menos aumentó este GAP fue en las R1, C1 y C2. Para complementar esto, en la Figura 6.1 se muestra el tiempo de resolución de CPLEX y en la Figura 6.2 se presenta el GAP de las soluciones en cada instancia, promediando los resultados de los tres niveles de α , lográndose ver gráficamente este empeoramiento en las soluciones. Estos comportamientos son esperables, debido a que el problema de ruteo de vehículos es un problema NP-difícil. Así, la complejidad del problema aumenta al incrementar la cantidad de nodos, empeorando las soluciones considerablemente al usar métodos exactos.

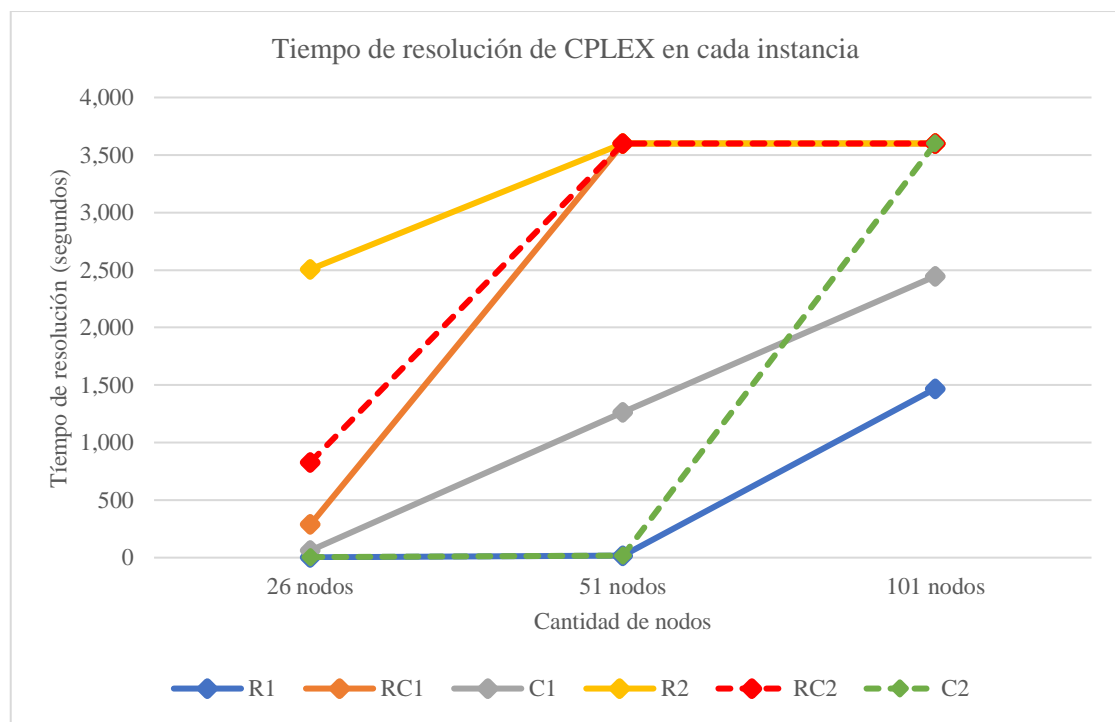


Figura 6.1: Tiempo promedio de resolución CPLEX en cada instancia.
Fuente: Elaboración propia.

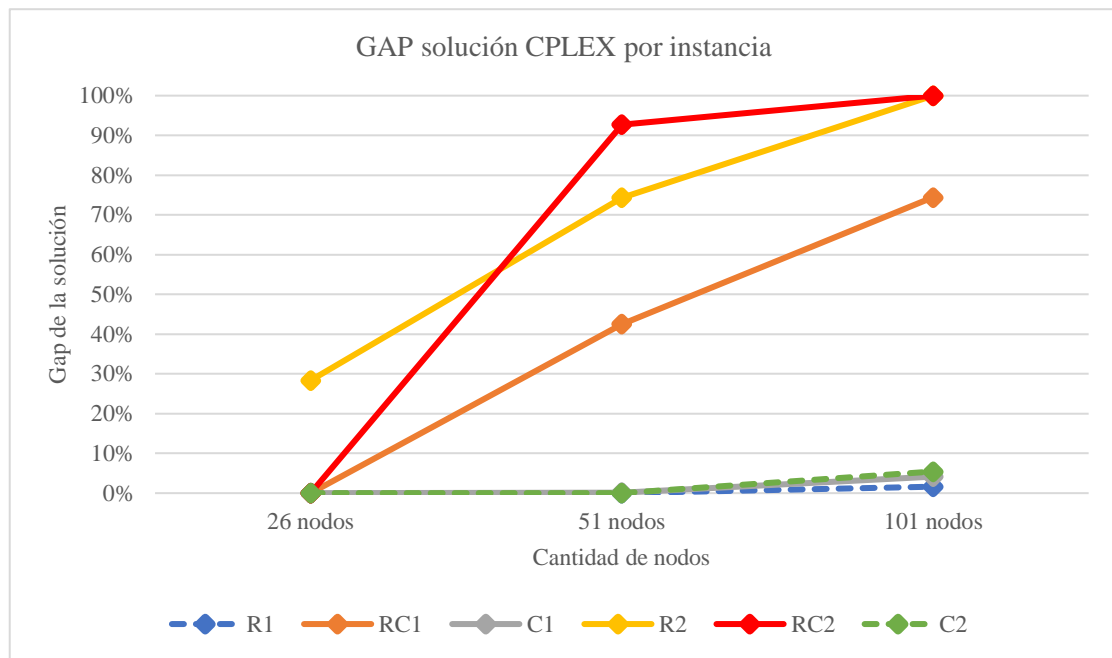


Figura 6.2: GAP soluciones CPLEX por instancia.
Fuente: Elaboración propia.

De igual forma, se observa que en varias instancias el costo total de recolección aumenta al considerar mayores valores de α . Por ejemplo, en la instancia C1.2 el mejor costo encontrado, aumenta de 2.620.701 pesos chilenos en $\alpha = 0,5$ (todos los métodos), a 2.624.025 en $\alpha = 0,75$ (ILS-B) y a 3.137.522 en $\alpha = 0,9$ (ILS-B). Se resume el incremento de costos según método y nivel de α en la Tabla 6.6, promediando todas las instancias. Este comportamiento es esperable, debido a que se quiere asegurar con mayor probabilidad que los vehículos son capaces de cumplir cada ruta. Además, el valor de \bar{Q} disminuye a mayores valores de α , por lo que el conjunto de ecuaciones (14) es más restrictivo, aumentando la cantidad de vehículos y el costo total.

Tabla 6.6: Incremento de costos según α y método, promediando todas las instancias.

Instancia	CPLEX		ILS-A		SA-A		ILS-B		SA-B	
	Costo	Mejor costo	Costo promedio	Mejor costo	Costo promedio	Mejor costo	Costo promedio	Mejor costo	Costo promedio	
0.5	3.726.633	3.438.715	3.471.026	3.410.057	3.477.611	3.441.188	3.489.867	3.439.505	3.505.177	
0.75	3.791.851	3.497.382	3.547.061	3.496.519	3.570.770	3.501.571	3.563.669	3.528.462	3.587.162	
0.9	3.903.394	3.615.103	3.710.744	3.669.934	3.730.357	3.584.693	3.675.684	3.651.535	3.797.559	

Fuente: Elaboración propia.

Comparando el rendimiento de las metaheurísticas mediante los promedios de cada costo, SA A tuvo el menor promedio en el mejor costo y ILS A el menor promedio en los costos promedios en $\alpha=0,5$ y $\alpha=0,75$. Para $\alpha=0,9$, la metaheurística ILS B tuvo el menor promedio en el mejor costo y costo promedio. En la Tabla 6.7 se resume el GAP promedio de estos costos respecto a la solución de CPLEX (promediando los tres niveles de α). Se aprecia que las metaheurísticas encuentran en promedio tanto mejores costos, como costos promedios mejores que CPLEX. Además, ambas metaheurísticas encuentran en promedio soluciones similares, siendo la metaheurística ILS B la que supera ligeramente a las otras.

Tabla 6.7: GAP promedio costos respecto a CPLEX, promediando instancias y niveles de α .

Costos	Enfoque A		Enfoque B	
	GAP ILS A (%)	GAP SA A (%)	GAP ILS B (%)	GAP SA B (%)
Mejores costos	-15,9	-15,8	-16,6	-16,2
Costos promedios	-15,2	-14,9	-15,7	-14,5

Fuente: Elaboración propia.

Comparando ambos enfoques, en la Tabla 6.8 se presenta el GAP promedio de los mejores costos y costos promedios del enfoque B respecto al A, según metaheurística y nivel de α (calculado con la ecuación (23)). Se observa que no existe gran diferencia entre los mejores costos y costos promedios encontrados con ambos enfoques. La mayor diferencia ocurre cuando $\alpha = 0,9$ y la menor cuando $\alpha = 0,5$.

Tabla 6.8: GAP promedio enfoque B respecto al enfoque A

GAP promedio	$\alpha = 0,5$		$\alpha = 0,75$		$\alpha = 0,9$		Promedio	
	GAP ILS (%)	GAP SA (%)	GAP ILS (%)	GAP SA (%)	GAP ILS (%)	GAP SA (%)	GAP ILS (%)	GAP SA (%)
Mejores costos	0,0	0,7	0,1	0,8	-2,0	-1,9	-0,6	-0,1
Costos promedio	0,2	0,4	0,4	0,4	-1,9	1,0	-0,5	0,6

Fuente: Elaboración propia.

Para comprobar la igualdad entre los métodos, se realiza el Wilcoxon *signed rank test*. Este test no paramétrico compara muestras de los resultados de dos métodos de resolución para comprobar si existe diferencia estadísticamente significativa entre ambos (Kumar & Kumar Singh, 2021). El test considera la hipótesis nula que los resultados de ambos métodos son estadísticamente iguales. Por lo otro lado, la hipótesis alternativa considera que los métodos son estadísticamente diferentes. En la Tabla 6.9 se presenta los resultados de este test comparando CPLEX con las metaheurística, y la Tabla 6.10 muestra los resultados de este test entre las metaheurísticas. Este test se realiza con un tamaño de muestras de 40 soluciones.

Considerando una significancia del 5%, se observa en la Tabla 6.9 que no se rechaza la hipótesis nula en ninguna comparación pareada. Por lo tanto, no existiría diferencia significativa entre CPLEX y los métodos metaheurísticos, a pesar de que las metaheurísticas encontraron en promedio mejores resultados. Misma situación se observa en la Tabla 6.10, donde no existe evidencia para rechazar la hipótesis nula en ninguna comparación pareada entre metaheurísticas. Por lo tanto, los métodos metaheurísticos entregarían soluciones estadísticamente iguales, verificando la igualdad de resultados entre los métodos y enfoques.

Tabla 6.9: Wilcoxon *signed rank test* CPLEX vs metaheurísticas.

Comparación pareada	Valor p
CPLEX- ILS A	0,0850
CPLEX-SA A	0,1866
CPLEX-ILS B	0,1123
CPLEX- SA B	0,2799

Fuente: Elaboración propia.

Tabla 6.10: Wilcoxon *signed rank test* entre metaheurísticas.

Comparación pareada	Valor p
ILS A-SA A	0,7412
ILS B-SA B	0,1838
ILS A- ILS B	0,7532
SA A- SA B	0,3722
ILS A-SA B	0,9453
ILS B- SA A	0,8288

Fuente: Elaboración propia.

En la Figura 6.3 ,6.4 y 6.5 se aprecia de manera gráfica los mejores costos encontrados en cada enfoque por nivel de α , observándose que, en líneas generales, en CPLEX y las metaheurísticas se encuentran similares mejores costos.

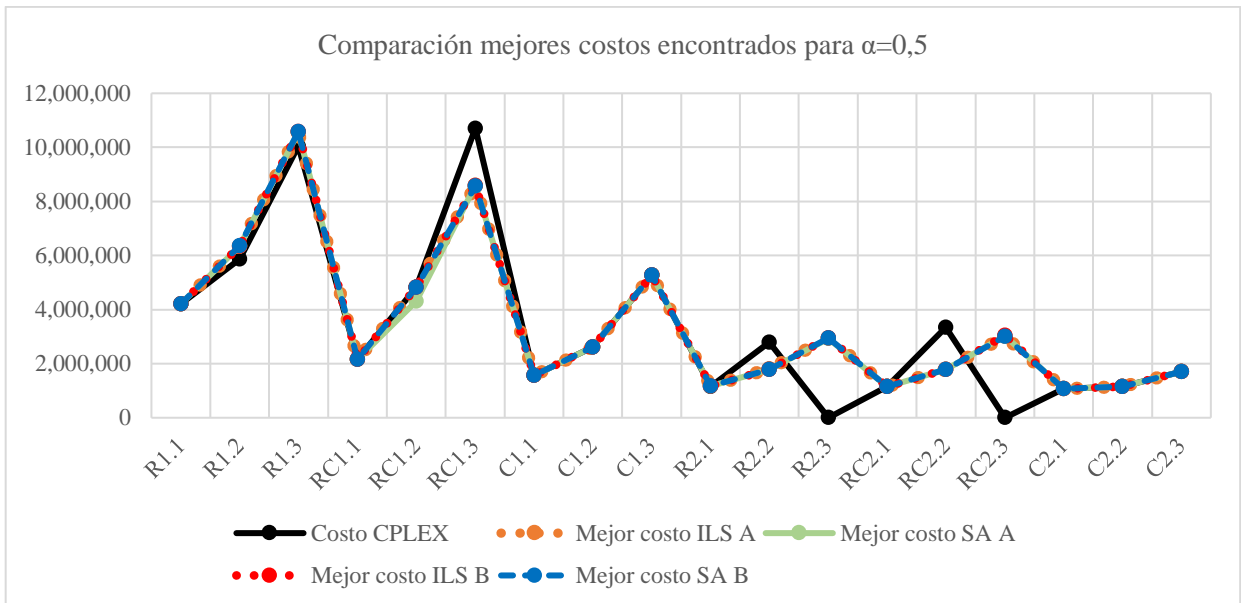


Figura 6.3: Comparación mejores costos para $\alpha=0,5$.
Fuente: Elaboración propia.

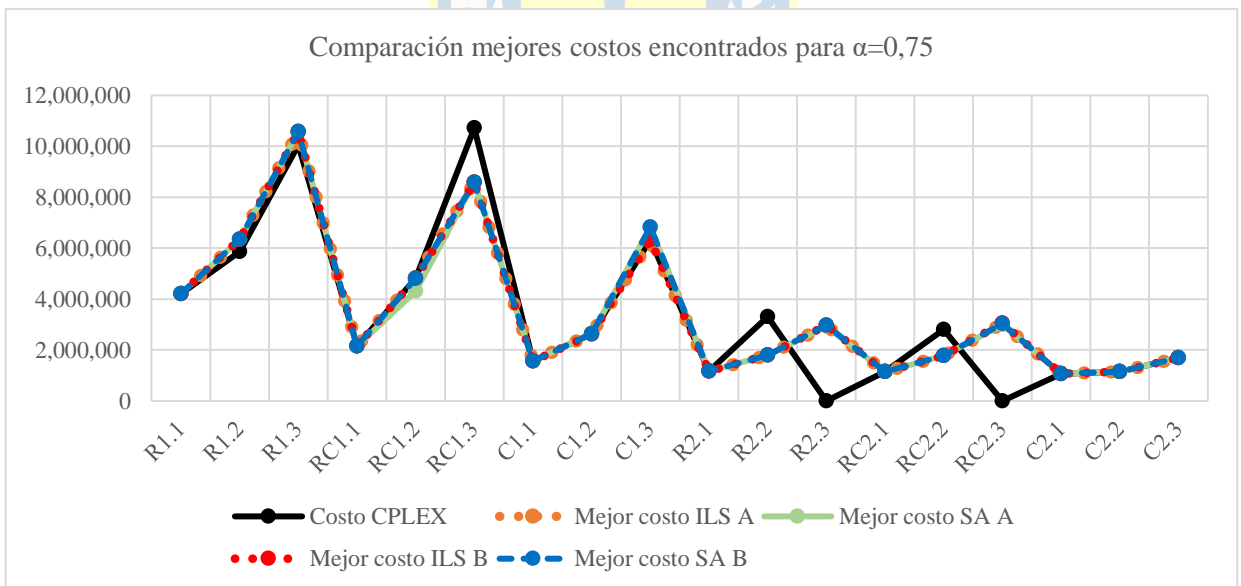


Figura 6.4: Comparación mejores costos para $\alpha=0,75$.
Fuente: Elaboración propia.

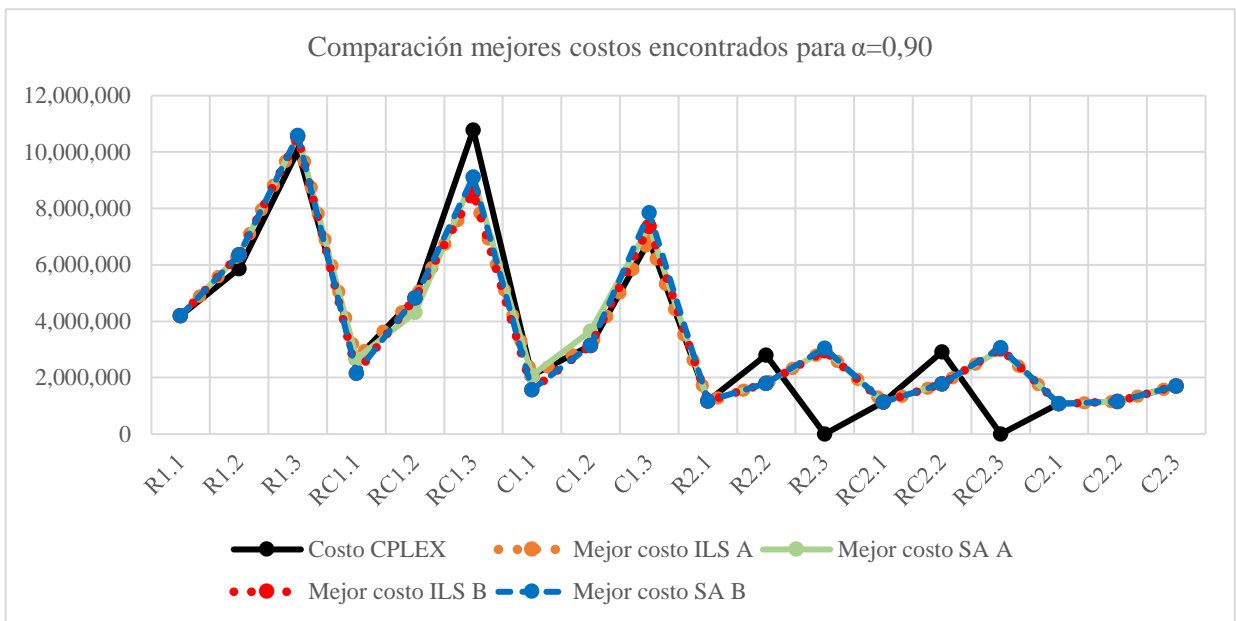
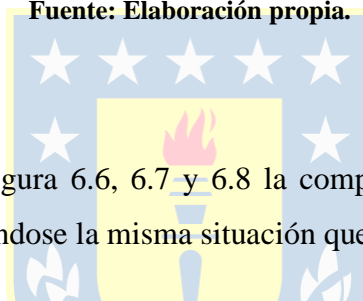


Figura 6.5: Comparación mejores costos para $\alpha=0,9$.
Fuente: Elaboración propia.



Por último, se muestra en la Figura 6.6, 6.7 y 6.8 la comparación de los costos promedios obtenidos en cada método, teniendo la misma situación que los mejores costos encontrados.

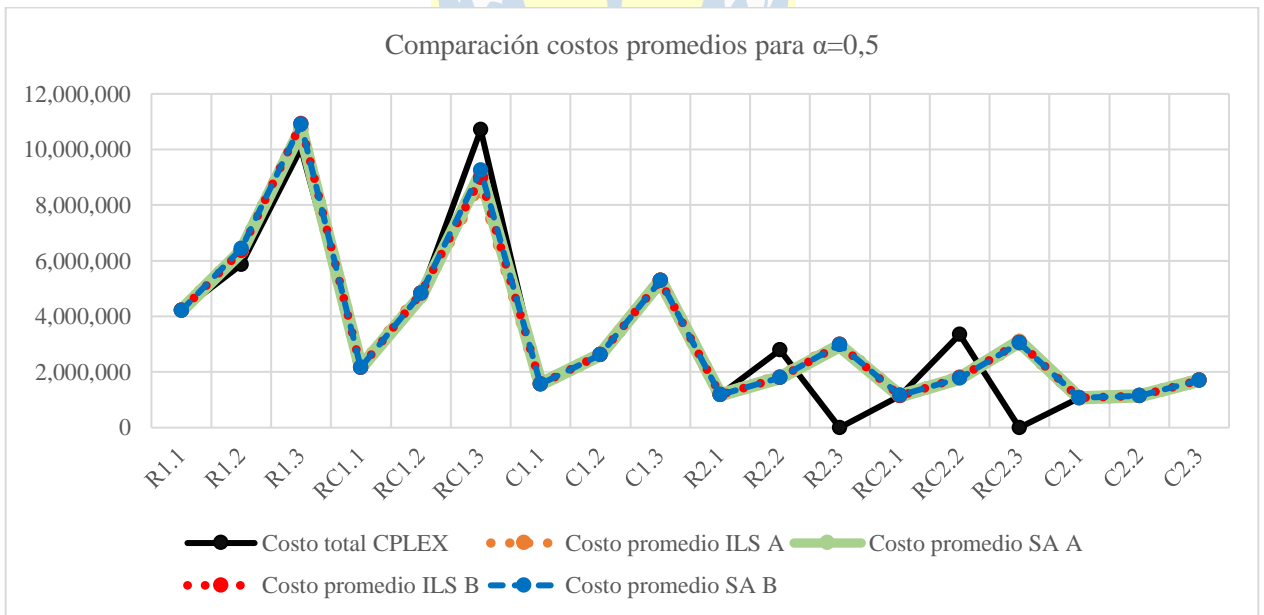


Figura 6.6: Comparación costos promedios para $\alpha=0,5$.
Fuente: Elaboración propia.

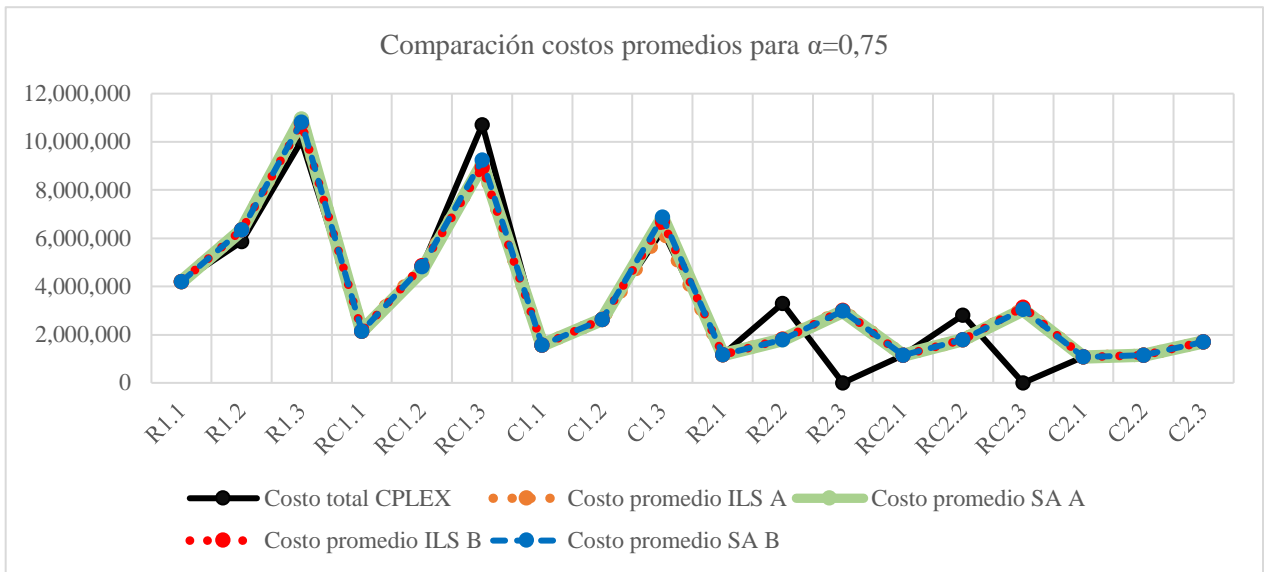


Figura 6.7: Comparación costos promedios para $\alpha=0,75$.
Fuente: Elaboración propia.

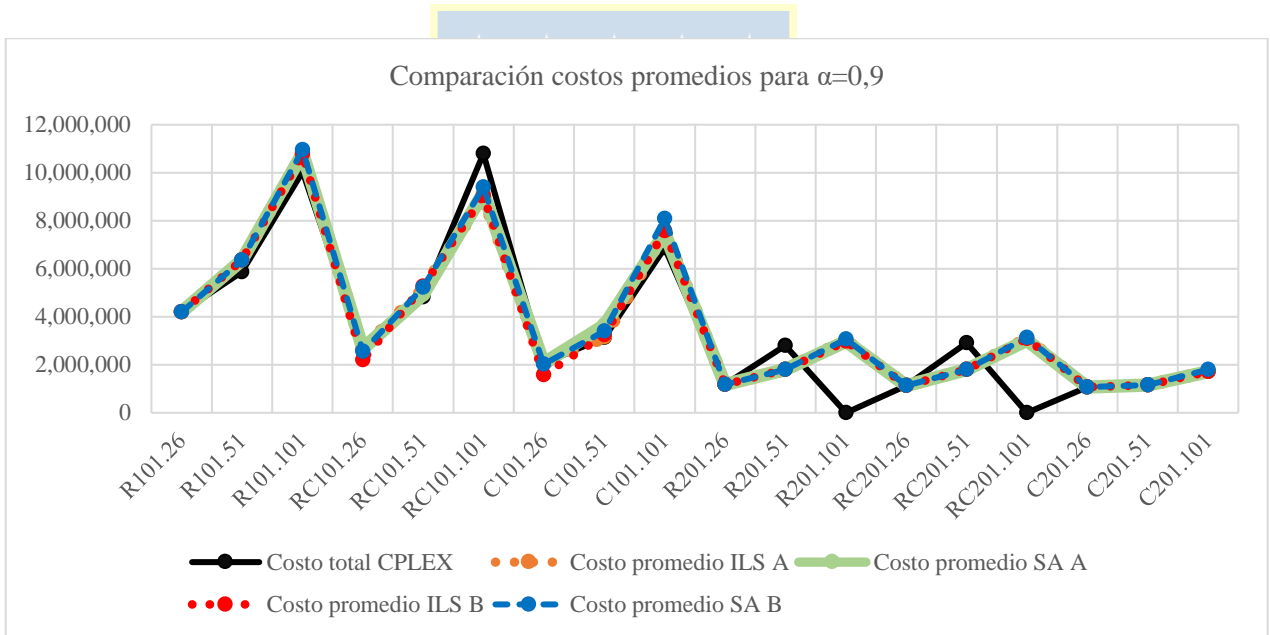


Figura 6.8: Comparación costos promedios para $\alpha=0,9$.
Fuente: Elaboración propia.

CAPÍTULO 7. CONCLUSIONES

Cada año crece la cantidad de residuos electrónicos generados en Chile, mientras que las tasas de valorización de estos residuos se mantienen bajas, lo cual puede provocar problemas al medio ambiente y a la salud de las personas. Esto ha conllevado que aumente la importancia de optimizar el proceso de manejo de residuos, con el fin de volverlo más rentable e incentivar la realización de este servicio. De todo el proceso de manejo de residuos, la fase de recolección representa aproximadamente el 70% de los costos totales, siendo el elemento más importante de este proceso.

El principal objetivo de este estudio fue plantar un modelo de programación matemático y un método metaheurístico que permitan minimizar el costo y las emisiones producidas en el proceso de recolección de residuos electrónicos en una ciudad, considerando que la cantidad de residuos generados en los puntos limpios es aleatoria. El problema fue modelado a través de un problema de ruteo de vehículos con ventanas de tiempo y demanda estocástica. La metodología CCP permitió transformar este problema a determinístico, asumiendo que la distribución de los residuos en cada punto limpio es normal, independiente e idénticamente distribuida entre estas. Con respecto a las metaheurísticas, se desarrolló un *Iterated local Search* y un *Simulated Annealing*. Además, se propusieron dos enfoques para representar la aleatoriedad de la demanda en las metaheurísticas, con el fin de comprobar como varían los resultados al considerar maneras distintas de tratar la aleatoriedad. El primer enfoque (denominado enfoque A) consideró el problema determinístico derivado de las suposiciones realizadas en la formulación CCP, verificando la capacidad de un vehículo mediante las ecuaciones (15). El segundo enfoque (denominado enfoque B) consideró el conjunto de ecuaciones (16), verificando la factibilidad de cada solución mediante muestras aleatorias de la distribución de probabilidad.

Para validar los métodos, se realizaron varios experimentos consideraron las instancias de Solomon (1987), las cuales fueron adaptadas para este problema.

Sobre los resultados, el método exacto permitió encontrar rutas de recolección factibles en las instancias pequeñas en tiempos computacionales aceptables. Sin embargo, se verificó que a medida que la cantidad de puntos limpios es mayor, este método se vuelve ineficiente, aumentando el GAP y el tiempo computacional. Por otro lado, tanto *Iterated local Search*, como *Simulated Annealing* encontraron buenas soluciones en la mayoría de las instancias, igualando o superando el resultado encontrado por CPLEX en mucho menor tiempo. Además, a pesar de que se propusieron distintas metodologías para tratar la aleatoriedad de la demanda, se encontraron soluciones bastante similares con ambos enfoques. Esta igualdad fue comprobada a través de métodos estadísticos, la cual arrojó que no existe diferencia significativa entre los métodos. Por lo tanto, los métodos metaheurísticos planteados resultan ser tan efectivos como un método exacto al momento de encontrar rutas de recolección de residuos electrónicos, aumentando su utilidad si se necesitan resultados en poco tiempo.

En relación a los costos totales, estos aumentan al asegurar con mayor probabilidad que la capacidad de un vehículo es suficientemente grande para servir a una ruta específica. Esto se debe a que el problema es más restrictivo, lo que genera que disminuya el largo de las rutas y aumente el costo total de recolección.

Por último, para futuros estudios, se propone: considerar enfoque multiobjetivo, metaheurísticas basadas en población, múltiples meses, vehículos heterogéneos, tiempos de descanso para los trabajadores, equilibrio de trabajo para cada vehículo, determinar la ubicación y número óptimo de puntos limpios, agregar otras fases de la cadena de suministro del proceso de recolección y eliminación/valorización de residuos, entre otras.

CAPÍTULO 8. REFERENCIAS

- Akhtar, M., Hannan, M., Begum, R., Basri, H., & Scavino, E. (2017). Backtracking search algorithm in CVRP models for efficient solid waste collection and route optimization. *Waste Management*, *61*, 117-128. doi:<https://doi.org/10.1016/j.wasman.2017.01.022>
- Alumur, S., Nickel, S., Saldanha-da-Gama, F., & Verter, V. (2012). Multi-period reverse logistics network design. *European Journal of Operational Research*, *220*, 67-78. doi:<https://doi.org/10.1016/j.ejor.2011.12.045>
- Arey, M., Baetz, B., Macdonald, P., & Byer, P. (1993). Use of Mixed Probability Distributions For The Analysis of Solid Waste Generation Data. *Waste Management and Research*, *11*, 387-402. doi:<https://doi.org/10.1006/wmre.1993.1041>
- Asefi, H., Shahparvari, S., & Chhetri, P. (2019b). Integrated Municipal Solid Waste Management under uncertainty: A tri-echelon city logistics and transportation context. *Sustainable Cities and Society*, *50*. doi:<https://doi.org/10.1016/j.scs.2019.101606>
- Asefi, H., Shahparvari, S., Chhetri, P., & Lim, S. (2019a). Variable fleet size and mix VRP with fleet heterogeneity in Integrated Solid Waste Management. *Journal of Cleaner Production*, *230*, 1376-1395. doi:<https://doi.org/10.1016/j.jclepro.2019.04.250>
- Assaf, R., & Saleh, Y. (2017). Vehicle Routing Optimization for Municipal Solid Waste Collection Using Genetic Algorithm: The Case of Southern Nablus City. *Civil and Environmental Engineering Reports*, *26*, 43-67. doi:<https://doi.org/10.1515/ceer-2017-0034>
- Ayvaz, B., Bolat, B., & Aydin, N. (2015). Stochastic reverse logistics network design for waste of electrical and electronic equipment. *Resources, Conservation and Recycling*, *104*, 391-404. doi:<https://doi.org/10.1016/j.resconrec.2015.07.006>

- Baldé, C., Forti, V., Gray, V., Kuehr, R., & Stegmann, P. (2017). *The Global E-waste Monitor*. United Nations University (UNU), International Telecommunication Union (ITU) & International Solid Waste Association (ISWA), Bonn/Geneva/Vienna.
- Blazquez, C., & Paredes-Belmar, G. (2020). Network design of a household waste collection system: A case study of the commune of Renca in Santiago, Chile. *Waste Management*, 116, 179-189. doi:<https://doi.org/10.1016/j.wasman.2020.07.027>
- Buhrkal, K., Larsen, A., & Ropke, S. (2012). The Waste Collection Vehicle Routing Problem with Time Windows in a City Logistics Context. *Procedia - Social and Behavioral Sciences*, 39, 241-254. doi:<https://doi.org/10.1016/j.sbspro.2012.03.105>
- Cardenas, P. (2020). *Optimizando la gestión de residuos electrónicos en la provincia de Concepción*. Concepción.
- Carreño, E. (2020). *Modelo y solución computacional para el problema de manejo de residuos utilizando herramientas multi-objetivo*. Concepción.
- Cortinhal, M. J., Mourão, M. C., & Nunes, A. C. (2016). Local search heuristics for sectoring routing in a household waste collection context. *European Journal of Operational Research*, 255, 68-79. doi:<https://doi.org/10.1016/j.ejor.2016.04.013>
- Delgado-Antequera, L., Caballero, R., Sánchez-Oro, J., & Colmenar, J. M. (26 de Noviembre de 2020). Iterated greedy with variable neighborhood search for a multiobjective waste collection problem. *Expert Systems With Applications*, 145. doi:<https://doi.org/10.1016/j.eswa.2019.113101>
- E2BIZ Consultores & Fundación Chile. (2019). *Antecedentes para la elaboración de análisis económicos de metas de recolección y valorización para "Aparatos Eléctricos y Electrónicos"*.

- Edalatpour, M., Al-e-hashem, S. M., Karimi, B., & Bahli, B. (2018). Investigation on a novel sustainable model for waste management in megacities: A case study in tehran municipality. *Sustainable Cities and Society*, 36, 286-301. doi:<https://doi.org/10.1016/j.scs.2017.09.019>
- Eiselt, H., & Marianov, V. (2014). A bi-objective model for the location of landfills for municipal solid waste. *European Journal of Operational Research*, 235, 187-194. doi:<https://doi.org/10.1016/j.ejor.2013.10.005>
- Elshaer, R., & Awad, H. (February de 2020). A taxonomic review of metaheuristic algorithms for solving the vehicle routing problem and its variants. *Computers & Industrial Engineering*, 140. doi:<https://doi.org/10.1016/j.cie.2019.106242>
- Expósito-Márquez, A., Expósito-Izquierdo, C., Brito-Santana, J., & Moreno-Pérez, J. A. (2019). Greedy randomized adaptive search procedure to design waste collection routes in La Palma. *Computers & Industrial Engineering*, 137. doi:<https://doi.org/10.1016/j.cie.2019.106047>
- Farrokhi-Asl, H., Makui, A., Ghousi, R., & Rabbani, M. (2020). Developing a hazardous waste management system with consideration of health, safety, and environment. *Computers and Electrical Engineering*, 82. doi:<https://doi.org/10.1016/j.compeleceng.2020.106553>
- Farrokhi-Asl, H., Makui, A., Jabbarzadeh, A., & Barzinpour, F. (2018). Solving a multi-objective sustainable waste collection problem considering a new collection network. *Operational Research*, 20, 1977–2015. doi:<https://doi.org/10.1007/s12351-018-0415-0>
- Forti, V., Baldé, C. P., Kuehr, R., & Bel, G. (2020). *The Global E-waste Monitor 2020, Quantities, flows, and the circular economy potential*. United Nations University (UNU), Bonn/Geneva/Rotterdam.

Fundación Chile & Ministerio del Medio Ambiente. (2019). *¿Qué es la Ley 20.920? Responsabilidad Extendida del Productor (REP) y Residuos de Aparatos Eléctricos y Electrónicos (RAEE)*.

Fundación Chile. (4 de Diciembre de 2019). *Residuos electrónicos*. Obtenido de <https://fch.cl/iniciativa/residuos-electronicos/>

Ganji, M., Kazemipoor, H., Hadji Molana, S. M., & Sajadi, S. M. (2020). A green multi-objective integrated scheduling of production and distribution with heterogeneous fleet vehicle routing and time windows. *Journal of Cleaner Production*, 259. doi:<https://doi.org/10.1016/j.jclepro.2020.120824>

Grasas, A., Juan, A. A., & Lourenço, H. (2014). SimILS: a simulation-based extension of the iterated local search metaheuristic for stochastic combinatorial optimization. *Journal of Simulation*, 1-9. doi:<https://doi.org/10.1057/jos.2014.25>

Groër, C., Golden, B., & Wasil, E. (Abril de 2010). A library of local search heuristics for the vehicle routing problem. *Mathematical Programming Computation*, 79-101. doi:<https://doi.org/10.1007/s12532-010-0013-5>

Gruher, A., Fikar, C., Juan, A., Hirsch, P., & Contreras, C. (2017). Supporting multi-depot and stochastic waste collection management in clustered urban areas via simulation-optimization. *Journal of Simulation*, 11, 11-19. doi:<https://doi.org/10.1057/s41273-016-0002-4>

Grunow, M., & Gobbi, C. (2009). Designing the reverse network for WEEE in Denmark. *CIRP Annals - Manufacturing Technology*, 58, 391-398. doi:<https://doi.org/10.1016/j.cirp.2009.03.036>

Hannan, M., Akhtar, M., Begum, R., Basri, H., Hussain, A., & Scavino, E. (2018). Capacitated vehicle-routing problem model for scheduled solid waste collection and route

- optimization using PSO algorithm. *Waste Management*, 71, 31-41.
doi:<https://doi.org/10.1016/j.wasman.2017.10.019>
- Hvattum, L., Løkketangen, A., & Laporte, G. (2006). Solving a Dynamic and Stochastic Vehicle Routing Problem with a Sample Scenario Hedging Heuristic. *Transportation Science*, 40, 421-438. doi:<https://doi.org/10.1287/trsc.1060.0166>
- Kaza, S., Yao, L., Bhada-Tata, P., & Van Woerden, F. (2018). *What a Waste 2.0: A Global Snapshot of Solid Waste Management to 2050*. World Bank, Washington DC.
- Kim, B. I., Kim, S., & Sahoo, S. (2006). Waste collection vehicle routing problem with time windows. *Computers & Operations Research*, 33, 3624-3642.
doi:<https://doi.org/10.1016/j.cor.2005.02.045>
- Król, A., Nowakowski, P., & Mrówczyńska, B. (2016). How to improve WEEE management? Novel approach in mobile collection with application of artificial intelligence. *Waste Management*, 50, 222-233. doi:<https://doi.org/10.1016/j.wasman.2016.02.033>
- Kumar, J., & Kumar Singh, A. (2021). Performance evaluation of metaheuristics algorithms for workload prediction in cloud environment. *Applied Soft Computing*, 113.
doi:<https://doi.org/10.1016/j.asoc.2021.107895>
- Li, J., Pardalos, P., Sun, J., & Zhang, Y. (2015). Iterated local search embedded adaptive neighborhood selection approach for the multi-depot vehicle routing problem with simultaneous deliveries and pickups. *Expert Systems with Applications*, 42, 3551-3561.
doi:<http://dx.doi.org/10.1016/j.eswa.2014.12.004>
- Markov, I., Bierlaire, M., Cordeau, J.-F., Maknoon, Y., & Varone, S. (2020). Waste collection inventory routing with non-stationary stochastic demands. *Computers & Operations Research*, 113, 107-120. doi:<https://doi.org/10.1016/j.cor.2019.104798>

- Markov, I., Varone, S., & Bierlaire, M. (2016). Integrating a heterogeneous fixed fleet and a flexible assignment of destination depots in the waste collection VRP with intermediate facilities. *Transportation Research Part B: Methodological*, 84, 256-273. doi:<https://doi.org/10.1016/j.trb.2015.12.004>
- Mes, M., Schutten, M., & Pérez Rivera, A. (2014). Inventory routing for dynamic waste collection. *Waste Management*, 34, 1564-1576. doi:<https://doi.org/10.1016/j.wasman.2014.05.011>
- Ministerio del Medio Ambiente. (2016a). *Informe del estado del Medio Ambiente*. Chile.
- Ministerio del Medio Ambiente. (2016b). *Guía de educación ambiental y residuos*. Chile.
- Ministerio del Medio Ambiente. (2019). *Quinto reporte del medio ambiente*. Chile.
- Ministerio del medio ambiente. (2020). *Informe del estado del medio ambiente 2020*. Chile.
- Mostafayi, S., Moazzeni, S., & Hvattum, L. (2020). Multi-objective sustainable location-districting for the collection of municipal solid waste: Two case studies. *Computers & Industrial Engineering*. doi:<https://doi.org/10.1016/j.cie.2020.106965>
- Normasari, N. E., Yu, V., Bachtiyar, C., & Sukoyo. (2019). A Simulated Annealing Heuristic for the Capacitated Green Vehicle Routing Problem. *Mathematical Problems in Engineering*, 18. doi:<https://doi.org/10.1155/2019/2358258>
- Nowakowski, P. (2017). A proposal to improve e-waste collection efficiency in urban mining: Container loading and vehicle routing problems – A case study of Poland. *Waste Management*, 60, 494-504. doi:<https://doi.org/10.1016/j.wasman.2016.10.016>
- Rabbani, M., Heidari, R., & Yazdanparast, R. (2019). A stochastic multi-period industrial hazardous waste location-routing problem: Integrating NSGA-II and Monte Carlo simulation. *European Journal of Operational Research*, 272, 945-961. doi:<https://doi.org/10.1016/j.ejor.2018.07.024>

- Rabbani, M., Heidari, R., Farrokhi-Asl, H., & Rahimi, N. (2018). Using metaheuristic algorithms to solve a multi-objective industrial hazardous waste location-routing problem considering incompatible waste types. *Journal of Cleaner Production*, 170, 227-241. doi:<https://doi.org/10.1016/j.jclepro.2017.09.029>
- Raggio, R. (2020). *Un Modelo Y Solución Preliminar, Considerando Elementos Estocásticos Para La Red De Reciclaje de E-waste en Ciudades*. Concepción.
- Shah, P., Anagnostopoulos, T., Zaslavsky, A., & Behdad, S. (2018). A stochastic optimization framework for planning of waste collection and value recovery operations in smart and sustainable cities. *Waste Management*, 78, 104-114. doi:<https://doi.org/10.1016/j.wasman.2018.05.019>
- Solomon, M. M. (1987). Algorithms for the Vehicle Routing and Scheduling Problems with Time Window Constraints. *Operations Research*, 35, 254-265. doi:<http://dx.doi.org/10.1287/opre.35.2.254>
- Stewart, W., & Golden, B. (1983). Stochastic vehicle routing: A Comprehensive Approach. *European Journal of Operational Research*, 14, 371-385. doi:[https://doi.org/10.1016/0377-2217\(83\)90237-0](https://doi.org/10.1016/0377-2217(83)90237-0)
- Tirkolaee, E. B., Abbasian, P., Soltani, M., & Ghaffarian, A. (2019). Developing an applied algorithm for multi-trip vehicle routing problem with time windows in urban waste collection: A case study. *Waste Management & Research*, 37, 4-13. doi:<https://doi.org/10.1177/0734242X18807001>
- Tirkolaee, E. B., Mahdavi, I., & Seyyed Esfahani, M. M. (2018). A robust periodic capacitated arc routing problem for urban waste collection considering drivers and crew's working time. *Waste Management*, 76, 138-146. doi:<https://doi.org/10.1016/j.wasman.2018.03.015>

- Toth, P., & Vigo, D. (2014). *Vehicle Routing Problems, Methods, and Applications* (Vol. II).
- Trochu, J., Chaabane, A., & Ouhimmou, M. (2019). A two-stage stochastic optimization model for reverse logistics network design under dynamic suppliers' locations. *Waste Management, 95*, 569-583. doi:<https://doi.org/10.1016/j.wasman.2019.06.012>
- Wang, Z., & Wen, P. (2020). Optimization of a Low-Carbon Two-Echelon Heterogeneous-Fleet Vehicle Routing for Cold Chain Logistics under Mixed Time Window. *Sustainability, 12*. doi:<https://doi.org/10.3390/su12051967>
- Wei, L., Zhang, Z., Zhang, D., & Heung, S. (2018). A simulated annealing algorithm for the capacitated vehicle routing problem with two-dimensional loading constraints. *European Journal of Operational Research, 243*, 843–859. doi:<http://dx.doi.org/10.1016/j.ejor.2017.08.035>
- Wu, H., Tao, F., & Yang, B. (2020b). Optimization of Vehicle Routing for Waste Collection and Transportation. *International Journal of Environmental Research and Public Health, 17*. doi:<https://doi.org/10.3390/ijerph17144963>
- Wu, H., Tao, F., Qiao, Q., & Zhang, M. (2020a). A Chance-Constrained Vehicle Routing Problem for Wet Waste Collection and Transportation Considering Carbon Emissions. *International Journal of Environmental Research and Public Health, 17*, 458-471. doi:<https://doi.org/10.3390/ijerph17020458>
- Xu, Y., Wu, S., Zang, H., & Hou, G. (2013). An interval joint-probabilistic programming method for solid waste management: a case study for the city of Tianjin, China. *Frontiers of Environmental Science & Engineering, 8*, 239-255. doi:<https://doi.org/10.1007/s11783-013-0536-x>

Yu, H., Sun, X., Solvang, D., Laporte, G., Lee, M., & Ka, C. (2020). A stochastic network design problem for hazardous waste management. *Journal of Cleaner Production*, 277. doi:<https://doi.org/10.1016/j.jclepro.2020.123566>

Zhang, W., Gajpal, Y., Appadoo, S. S., & Wei, Q. (Abril de 2020). Multi-Depot Green Vehicle Routing Problem to Minimize Carbon Emissions. *Sustainability*, 12. doi:<https://doi.org/10.3390/su12083500>



ANEXOS

Anexo 1: Demostración *Chance-Constrained Programming*

Las restricciones de capacidad son transformadas en las ecuaciones (24), las cuales aseguran que la probabilidad de que la cantidad de residuos electrónicos recolectados en la ruta sea menor a la capacidad del vehículo, es mayor a α .

$$P\left(\sum_{i \in C} \sum_{j \in N} d_i x_{ijk} \leq Q\right) \geq \alpha, \quad \forall k \in V \quad (24)$$

Si d_i son variables aleatorias con media μ_i y varianza σ_i^2 , entonces:

$$U_k = \sum_{i \in C} \sum_{j \in N} \mu_i x_{ijk}, \quad \forall k \in V \quad (25)$$

$$S_k = \sqrt{\sum_{i \in C} \sum_{j \in N} \sigma_i^2 x_{ijk}^2}, \quad \forall k \in V \quad (26)$$

Donde U_k corresponde a la media de los residuos recolectados en la ruta del vehículo k y S_k a su desviación estándar. Para aplicar esta formulación, es necesario que la distribución de $\frac{\sum_{i \in C} \sum_{j \in N} d_i x_{ijk} - U_k}{S_k}$ sea igual a la de $\frac{d_i - \mu_i}{\sigma_i}$. Esto se cumple si la cantidad de residuos en cada punto limpio se distribuyen Normal, Binomial, Poisson, entre otras (Stewart & Golden, 1983). Para este estudio, se asume que la cantidad de residuos generados en cada punto limpio se distribuye Normal. Usando las ecuaciones (25) y (26), el conjunto de restricciones (24) resulta:

$$P\left(\frac{\sum_{i \in C} \sum_{j \in N} d_i x_{ijk} - U_k}{S_k} \leq \frac{Q - U_k}{S_k}\right) = \alpha, \quad \forall k \in V \quad (27)$$

$$P\left(z \leq \frac{Q - U_k}{S_k}\right) = \alpha, \quad \forall k \in V \quad (28)$$

Las ecuaciones (28) son equivalente a:

$$z_\alpha \leq \frac{Q - U_k}{S_k}, \quad \forall k \in V \quad (29)$$

$$U_k + z_\alpha S_k \leq Q, \quad \forall k \in V \quad (30)$$

$$\sum_{i \in C} \sum_{j \in N} \mu_i x_{ijk} + z_\alpha \sqrt{\sum_{i \in C} \sum_{j \in N} \sigma_i^2 x_{ijk}^2} \leq Q, \quad \forall k \in V \quad (31)$$

Donde z_α , corresponde a la función inversa de la distribución Normal estándar, evaluada en la probabilidad α .

Sin embargo, las restricciones (31) son no lineales. Asumiendo que las demandas de los puntos limpios son independientes e idénticamente distribuidas y que, como las variables x_{ijk} son binarias, se cumple que $x_{ijk}^2 = x_{ijk}$. Además, estableciendo que σ^2 puede ser obtenido multiplicando μ por un parámetro π ($\sigma^2 = \pi\mu$), entonces según Stewart et al. (1983), existe un valor \bar{Q} tal que el conjunto de restricciones (31) es equivalente a:

$$\sum_{i \in C} \sum_{j \in N} \mu x_{ijk} \leq \bar{Q}, \quad \forall k \in V \quad (32)$$

Aplicando las condiciones establecidas, los conjuntos de ecuaciones (25) y (26) resultan:

$$U_k = \sum_{i \in C} \sum_{j \in N} \mu_i x_{ijk} = \sum_{i \in C} \sum_{j \in N} \mu x_{ijk}, \quad \forall k \in V \quad (33)$$

$$S_k = \sqrt{\sum_{i \in C} \sum_{j \in N} \sigma_i^2 x_{ijk}^2} = \sqrt{\sum_{i \in C} \sum_{j \in N} \sigma^2 x_{ijk}} = \sqrt{\sum_{i \in C} \sum_{j \in N} \pi \mu x_{ijk}}, \quad \forall k \in V \quad (34)$$

Así, las restricciones (31) pueden ser escritas como:

$$\sum_{i \in C} \sum_{j \in N} \mu x_{ijk} + z_\alpha \sqrt{\sum_{i \in C} \sum_{j \in N} \pi \mu x_{ijk}} \leq Q, \quad \forall k \in V \quad (35)$$

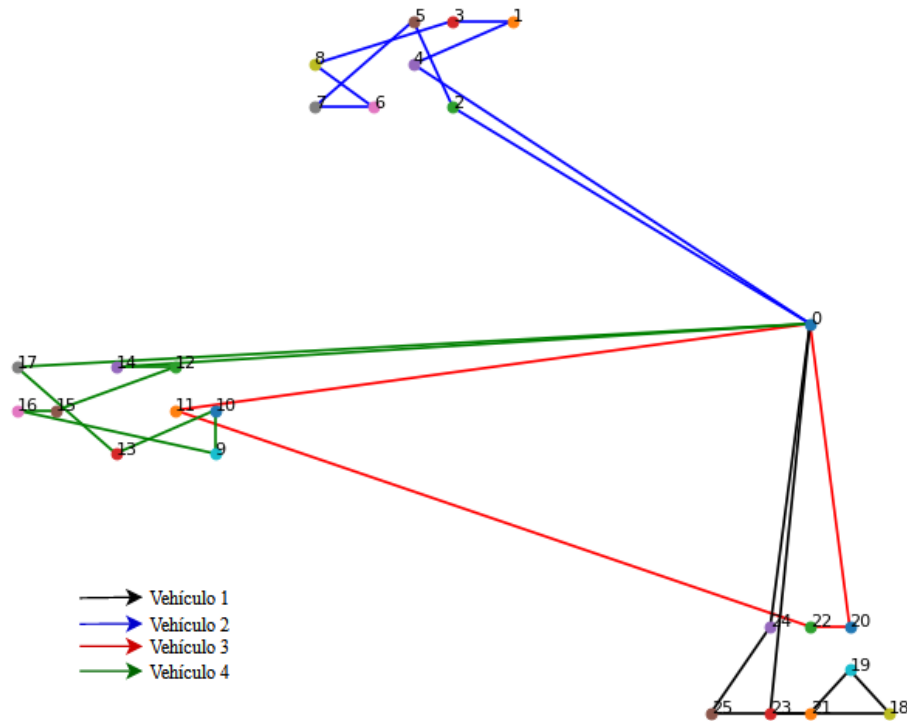
$$U_k + z_\alpha \sqrt{\pi U_k} \leq Q, \quad \forall k \in V \quad (36)$$

Agrupando los términos en el conjunto de ecuaciones (36), se obtiene el conjunto de restricciones lineales (32), tal como se muestra a continuación en la ecuación (37).

$$U_k = \sum_{i \in C} \sum_{j \in N} \mu x_{ijk} \leq \frac{2Q + z_\alpha^2 \pi - \sqrt{z_\alpha^4 \pi^2 + 4Q z_\alpha^2 \pi}}{2} = \bar{Q}, \quad \forall k \in V \quad (37)$$

Anexo 2: Análisis solución

En relación con los resultados obtenidos, se observa coherencia con las restricciones establecidas para el problema: las rutas de cada vehículo empiezan y terminan en el depósito, cada punto limpio es visitado por exactamente un vehículo y solo una vez, no se supera la capacidad máxima de los vehículos y se cumple las ventanas de tiempo de cada nodo. En la Figura A.1 se muestra la solución encontrada por todos los métodos para la instancia RC1.1, considerando $\alpha = 0,5$. Además, en la Tabla A.1 se muestran las rutas de recolección y la cantidad de residuos recolectado y en la Tabla A.2 el tiempo de llegada a cada nodo, verificándose todas las restricciones establecidas para el modelo de optimización y las metaheurísticas del enfoque A. Para el enfoque B, las restricciones de capacidad dependen de muestreo aleatorio, por lo tanto, no se incluye en la Tabla A.1.



**Figura A.1: Solución encontrada para instancia RC1.1, $\alpha=0,5$.
Fuente: Elaboración propia.**

Tabla A.1: Rutas y cantidad de residuos recolectados para solución instancia RC1.1, $\alpha=0,5$.

Ruta	Cantidad de residuos	Capacidad máxima (\bar{Q})
0-2-5-7-6-8-3-1-4-0	173 kg	200 kg
0-11-22-20-0	64 kg	
0-14-12-15-16-9-10-13-17-0	173 kg	
0-23-21-19-18-25-24-0	130 kg	

Fuente: Elaboración propia.**Tabla A.2: Tiempo de llegada a cada nodo para solución instancia RC1.1, $\alpha=0,5$.**

Ruta 1	Ventana de tiempo	Tiempo de llegada	Ruta 2	Ventana de tiempo	Tiempo de llegada	Ruta 3	Ventana de tiempo	Tiempo de llegada	Ruta 4	Ventana de tiempo	Tiempo de llegada
0	[0,240]	0	0	[0,240]	0	0	[0,240]	0	0	[0,240]	0
2	[50,80]	60	11	[59,89]	59	14	[35,65]	35	23	[65,95]	65
5	[41,71]	70	22	[92,122]	110	12	[64,94]	64	21	[67,97]	77
7	[79,109]	91	20	[122,152]	122	15	[58,88]	82	19	[72,102]	92
6	[95,125]	104	0	[0,240]	167	16	[72,102]	94	18	[87,117]	108
8	[91,121]	121				9	[91,121]	115	25	[154,184]	154
3	[109,139]	138				10	[119,149]	130	24	[148,178]	174
1	[145,175]	152				13	[142,172]	147	0	[0,240]	219
4	[141,171]	169				17	[149,179]	168			
0	[0,240]	215				0	[0,240]	218			

Fuente: Elaboración propia.