



UNIVERSIDAD DE CONCEPCIÓN
FACULTAD DE INGENIERÍA

Un algoritmo para resolver el problema del vendedor viajero con ventanas de tiempo para servicios postales

Por: José Tomás Garcés Guzmán

Tesis presentada a la Facultad de Ingeniería de la Universidad de Concepción
para optar al Título de Ingeniero Civil Industrial

Agosto 2022
Concepción, Chile

Profesor Guía: Carlos Contreras Bolton

Agradecimientos

Quiero agradecer primero a mis padres que siempre me apoyaron en los momentos difíciles y celebraron conmigo en los momentos felices de mi vida universitaria.

En segundo lugar quiero agradecer a mi hermano y a mi polola, que me dieron su ayuda y consejos en momentos muy importantes y que seguramente si no fuese por ellos mi viaje por la universidad sería un poco más largo.

También a mis amigos, con quienes en varias ocasiones compartimos las angustias y las alegrías propias de la universidad.

Finalmente me gustaría agradecer a los profesores del departamento por demostrar siempre su interés en nuestro aprendizaje, y agradecer también al profesor Carlos que contribuyó en gran parte a que este trabajo sea un gran recuerdo de mi paso por la universidad.

Resumen

Esta memoria de título tiene como objetivo estudiar el problema del vendedor viajero con ventanas de tiempo aplicado al caso de los servicios postales, y elaborar una metaheurística para resolver el problema. Este problema es una variante del problema clásico del vendedor viajero que consiste en minimizar el costo de una ruta hamiltoniana con una restricción adicional, que consiste en la existencia de ventanas de tiempo para cada uno de los destinos en la ruta, generándose un aumento en la dificultad en la resolución. Esta dificultad se debe a que al llegar a un destino de forma prematura, se genera un aumento en el costo de la ruta y además, el arribo posterior al cierre de la ventana de tiempo produce una ruta no factible. En los servicios postales por lo general existen dos tipos de entregas, la paquetería y las cartas, para ambos casos la industria busca optimizar las rutas de entrega para reducir los costos operativos y de mano de obra. Además, existen muchos casos en los que el cliente tiene un tiempo acotado para recibir su entrega. Para resolver este problema se propone un algoritmo basado en una búsqueda local iterada. Este algoritmo trabaja con penalizaciones, perturbaciones y búsquedas locales. Los resultados de la metaheurística muestran buenos rendimientos hasta las instancias de 60 nodos.

Abstract

This objective work is to study the traveling salesman problem with time windows applied to postal services and develop a metaheuristic to solve the problem. This problem is a variant of the classic traveling salesman problem that tries to minimize the cost of a Hamiltonian route with an additional constraint, which consist of the existence of time windows for each point on the route, which makes it more difficult to solve the problem. This difficulty is because arriving at some point before the given time makes the route more expensive, and furthermore, arriving at some point after the given time generates an infeasible route. In the postal services generally are two types of deliveries, the parcels, and the letters deliveries, in both cases the industry tries to reduce de delivery routes to lower operating and human costs. Furthermore, there are many cases that the costumers have limited time to receive the package. To solve this problem, an algorithm based on an iterated local search is proposed. This algorithm works with penalties, perturbations and local searches. The results of the metaheuristic show a good performance up to the sixty cities instances.

© 2022, José Tomás Garcés Guzmán

Ninguna parte de esta tesis puede reproducirse o transmitirse bajo ninguna forma o por ningún medio o procedimiento, sin permiso por escrito del autor.

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

Índice de Contenidos

Agradecimientos	2
Resumen	3
Abstract.....	4
1. Introducción.....	11
1.1. Antecedentes generales	11
1.2. Objetivo general	12
1.3. Objetivos específicos	12
1.4. Estructura del documento	12
2. Problema del vendedor viajero con ventanas de tiempo	14
2.1. Descripción del problema.....	14
2.2. Modelo de programación lineal.....	18
2.3. Revisión de la literatura.....	20
3. Algoritmo de búsqueda local iterada	26
3.1. Estructura general.....	26
3.2. Solución inicial.....	27
3.3. Penalizaciones y representación de la solución	28
3.4. Perturbaciones	29
3.4.1. Intercambio Aleatorio de dos Ciudades	30
3.4.2. Inversión del Orden entre Ciudades	30
3.5. Búsqueda local.....	30
3.6. Variaciones ILS	32
3.6.1. ILS 1 e ILS 2	33
3.6.2. ILS 3 e ILS 4	34
4. Experimentos y resultados.....	36

4.1.	Detalles computacionales	36
4.2.	Descripción de las instancias	36
4.3.	Resultados.....	37
4.3.1.	Instancias de 20 ciudades	39
4.3.2.	Instancias de 40 ciudades	40
4.3.3.	Instancias de 60 ciudades	41
4.3.4.	Instancias de 80 ciudades	42
4.3.5.	Instancias de 100 ciudades	43
5.	Conclusiones.....	45
6.	Referencias	47
	Anexos.....	50

Índice de Tablas

TABLA 1: PARÁMETROS BÚSQUEDAS LOCALES	34
TABLA 2: SOLUCIONES ENCONTRADAS POR MÉTODO.....	38
TABLA 3: RESULTADOS MODELOS 20 CIUDADES.....	39
TABLA 4: RESULTADOS BÚSQUEDAS LOCALES 20 CIUDADES.....	39
TABLA 5: RESULTADOS MODELOS 40 CIUDADES.....	40
TABLA 6: RESULTADOS BÚSQUEDAS LOCALES 40 CIUDADES.....	41
TABLA 7: RESULTADOS MODELOS 60 CIUDADES.....	41
TABLA 8: RESULTADOS BÚSQUEDAS LOCALES 60 CIUDADES.....	42
TABLA 9: RESULTADOS MODELOS 80 CIUDADES.....	42
TABLA 10: RESULTADOS BÚSQUEDAS LOCALES 80 CIUDADES.....	43
TABLA 11: RESULTADOS MODELOS 100 CIUDADES.....	44
TABLA 12: RESULTADOS BÚSQUEDAS LOCALES 100 CIUDADES.....	44
TABLA A 1: RESULTADOS 20 CIUDADES MODELO MTZ.....	50
TABLA A 2: RESULTADO 20 CIUDADES MODELO GG.....	51
TABLA A 3: RESULTADOS 20 CIUDADES ILS 1	52
TABLA A 4: RESULTADOS 20 CIUDADES ILS 2	53
TABLA A 5: RESULTADOS 20 CIUDADES ILS 3	54
TABLA A 6: RESULTADOS 20 CIUDADES ILS 4	55
TABLA A 7: RESULTADOS 40 CIUDADES MODELO MTZ.....	56
TABLA A 8: RESULTADOS 40 CIUDADES MODELO GG	57
TABLA A 9: RESULTADOS 40 CIUDADES ILS 1	58
TABLA A 10: RESULTADOS 40 CIUDADES ILS 2	59
TABLA A 11: RESULTADOS 40 CIUDADES ILS 3	60
TABLA A 12: RESULTADOS 40 CIUDADES ILS 4	61
TABLA A 13: RESULTADOS 60 CIUDADES MODELO MTZ.....	62
TABLA A 14: RESULTADOS 60 CIUDADES MODELO GG	63
TABLA A 15: RESULTADOS 60 CIUDADES ILS 1	64
TABLA A 16: RESULTADOS 60 CIUDADES ILS 2	65

TABLA A 17: RESULTADOS 60 CIUDADES ILS 3	66
TABLA A 18: RESULTADOS 60 CIUDADES ILS 4	67
TABLA A 19: RESULTADOS 80 CIUDADES MODELO MTZ.....	68
TABLA A 20: RESULTADO 80 CIUDADES MODELO GG.....	69
TABLA A 21: RESULTADOS 80 CIUDADES ILS 1	70
TABLA A 22: RESULTADOS 80 CIUDADES ILS 2	71
TABLA A 23: RESULTADOS 80 CIUDADES ILS 3	72
TABLA A 24: RESULTADOS 80 CIUDADES ILS 4	73
TABLA A 25: RESULTADOS 100 CIUDADES MODELO MTZ.....	74
TABLA A 26: RESULTADOS 100 CIUDADES MODELO GG	75
TABLA A 27: RESULTADOS 100 CIUDADES ILS 1	76
TABLA A 28: RESULTADOS 100 CIUDADES ILS 2	77
TABLA A 29: RESULTADOS 100 CIUDADES ILS 3	78
TABLA A 30: RESULTADOS 100 CIUDADES ILS 4	79

Índice de Figuras

FIGURA 1: MINIMIZACIÓN TT CASO 1.....	15
FIGURA 2: MINIMIZACIÓN RD CASO 1.....	15
FIGURA 3: MINIMIZACIÓN TT CASO 2.....	17
FIGURA 4: MINIMIZACIÓN RD CASO 2.....	17
FIGURA 5: REPRESENTACIÓN DE LA SOLUCIÓN	29
FIGURA 6: INTERCAMBIO ALEATORIO DE DOS CIUDADES	27
FIGURA 7: INVERSIÓN DEL ORDEN ENTRE CIUDADES	30

1. Introducción

En este capítulo se presentan los antecedentes generales de este trabajo, sus objetivos, y finalmente, como se organiza el documento de la memoria de título.

1.1. Antecedentes generales

Uno de los factores más importantes para las empresas de servicios postales que entregan paquetería, es el momento de la entrega debido a que la entrega de paquetes antes o después genera costos no deseados para las empresas. Por ejemplo, si se llega antes al punto de entrega, es probable que se deba pagar estacionamiento y esperar a que el cliente llegue, generando nuevos costos tanto monetarios con el uso de los vehículos y humanos por el tiempo que el repartidor deba esperar. Por otro lado, llegar cuando el cliente no se encuentra implica costos relacionados con volver al sitio de forma posterior. Estos espacios de tiempo llamados ventanas de tiempo (TW, por sus siglas en inglés: *time windows*), bien administrados pueden evitar estas situaciones y gracias a la investigación de operaciones existen métodos para optimizar estas ventanas de tiempo mediante variaciones de problemas tradicionales.

El problema del vendedor viajero (TSP, por sus siglas en inglés: *traveling salesman problem*) es un problema clásico de la investigación de operaciones que consiste en encontrar un circuito óptimo en términos de longitud o costo, que debe seguir el vendedor para posteriormente volver al punto de partida (Applegate et al., 2007). A pesar de su complejidad NP – completo (Karp, 1972), el TSP tiene múltiples aplicaciones aún en sus formulaciones más simples, tales como, planificación y logística. Sin embargo, el número de soluciones o rutas posibles aumenta factorialmente debido a su complejidad a medida que la cantidad de ciudades aumenta.

El TSP tiene múltiples variantes que son mucho más difíciles, como por ejemplo, el problema del vendedor viajero con ventanas de tiempo (TSPTW, por sus siglas en inglés: *traveling salesman problema with time windows*) que además de conservar la dificultad del original, este agrega más elementos a considerar para resolver el problema de manera eficiente, eficaz y con aplicabilidad en la realidad. En función de lo anterior, se han investigado diversos enfoques para resolver problemas como el TSPTW, tales como,

algoritmos que tratan de encontrar soluciones exactas. Sin embargo, pueden resolver de manera óptima dimensiones de tamaño pequeño. Además, se han considerado enfoques heurísticos que generan soluciones iterativamente, intentando mejorar una solución candidata con respecto a una determinada medida de calidad. Las heurísticas hacen pocas o ninguna suposición sobre el problema que se está optimizando y pueden buscar grandes espacios de soluciones candidatas para encontrar soluciones óptimas o casi óptimas con buenos resultados (Wang & Chen, 2013).

En función de lo expresado anteriormente la principal diferencia que entre un TSPTW tradicional con la aplicación a los servicios postales trabajada en este trabajo es que la ruta optimizada considera todo el trayecto, desde la salida hasta la vuelta al depósito (*makespan*). Lo anterior quiere decir que las rutas que se desean generar consideran los tiempos de viajes entre nodos además de los posibles tiempo de espera, a diferencia de un TSPTW tradicional que por lo general busca optimizar la suma de los tiempos de viajes entre destinos.

1.2. Objetivo general

Implementar una metaheurística para resolver el problema del vendedor viajero con ventana de tiempo en servicios postales.

1.3. Objetivos específicos

- Realizar la revisión de literatura de los métodos usados para resolver el TSPTW.
- Diseñar e implementar una metaheurística para resolver el TSPTW.
- Resolver las instancias del TSPTW con la metaheurística
- Analizar los resultados obtenidos y realizar una comparación con la literatura aplicado al caso de los servicios postales.

1.4. Estructura del documento

El resto del documento se estructura como sigue. En el capítulo 2, se presenta el problema formalmente, incluyendo modelos matemáticos que permitan resolverlo. Continúa el capítulo 3 con el desarrollo y la explicación de la metaheurística propuesta, presentando pseudocódigos y descripción de su funcionamiento. Luego, el capítulo 4 aborda los

experimentos computacionales y resultados obtenidos. Finalmente, los resultados se discuten y analizan en el capítulo 5.

2. Problema del vendedor viajero con ventanas de tiempo

En este capítulo se muestra el problema de forma extensa y los modelos matemáticos asociados al mismo. También, se revisa la literatura más importante y reciente relacionada al problema.

2.1. Descripción del problema

Como se menciona anteriormente, el TSPTW tiene como principal característica la restricción generada por las TW, esta restricción genera un aumento de los costos o tiempo con llegadas prematuras y tardías, además, las llegadas tardías generan rutas no factibles. También, debido a las TW es necesario diferenciar y considerar a la vez dos tipos de costos diferentes.

El primer costo es conocido como tiempo de viaje o *time travel* (TT) y contempla el costo o tiempo total de las distancias entre destinos solamente. El segundo costo es el tiempo total de la ruta, conocido como *makespan* o *route duration* (RD) y contempla el TT más los posibles tiempos de espera. Por lo general, en los servicios postales se asocia el TT con los costos operacionales, como por ejemplo, el costo por hora de usar un vehículo para realizar los repartos. Mientras, la RD se asocia con el costo de mano de obra por hora, que por lo general es considerablemente mayor que los costos operacionales. Este trabajo se centra en la minimización de la ruta considerando tanto la RD como el TT, esto permite visitar más puntos en la ruta, evitar llegadas tardías y minimizar las llegadas prematuras.

El problema se puede formular a partir de un grafo dirigido $G = (V, A)$, donde V contiene a los vértices incluyendo al depósito y A contiene los arcos. Cada vértice $i \in V$ posee una ventana de tiempo $TW[r_i, d_i]$, donde r_i corresponde al tiempo inicial de la TW, y d_i es el tiempo de término de la TW que define el periodo de tiempo donde la visita debe ocurrir. Para cada arco $(i, j) \in A$ existe un costo $c_{ij}, \forall i, j \in V$ con $i \neq j$. Para este trabajo no existen tiempos de servicio y se permiten los tiempos de espera cuando se llega prematuramente, dónde el conductor debe esperar hasta r_i . A continuación, se ilustran unos ejemplos con la finalidad de entender por qué es importante que la función objetivo contemple el TT y la RD.

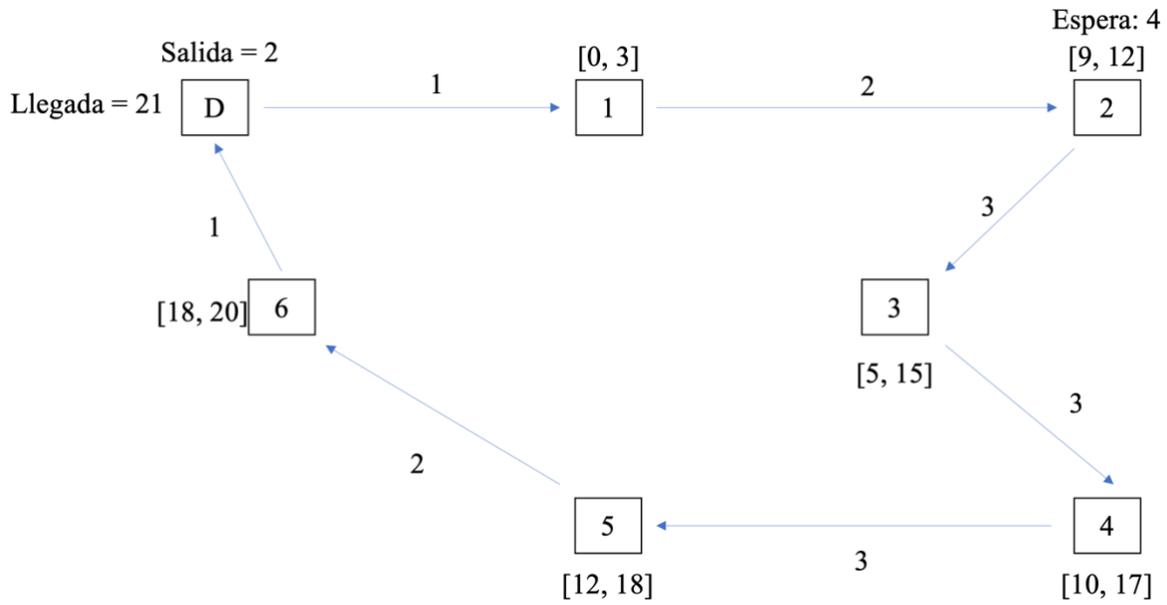


Figura 1: Minimización TT Caso 1.

Fuente: Bretin et al. (2021).

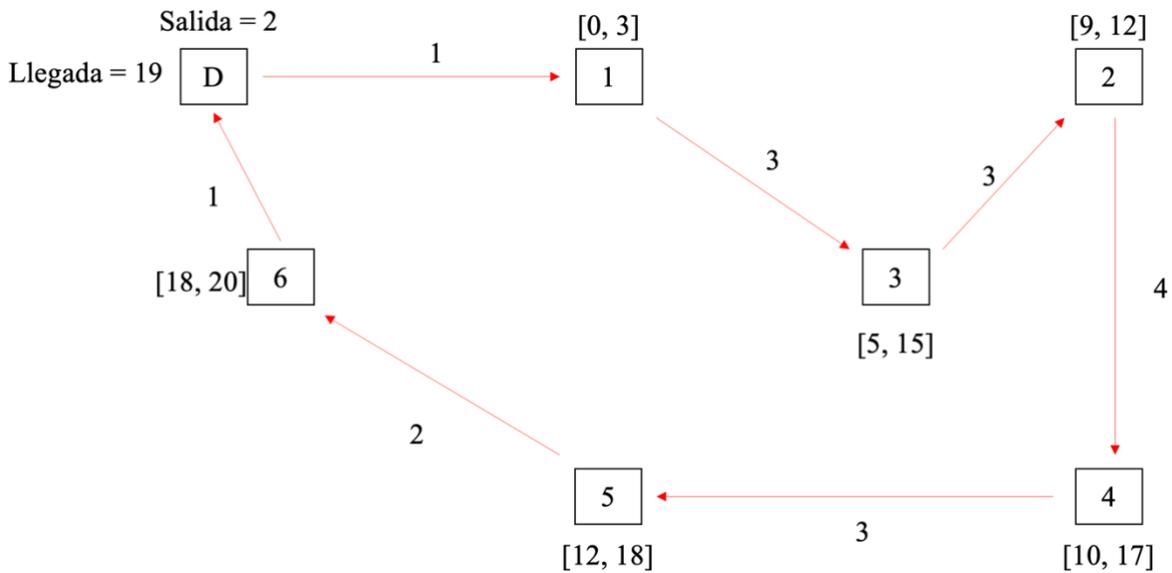


Figura 2: Minimización RD Caso 1.

Fuente: Bretin et al. (2021).

Considerando que el sueldo promedio de un repartidor es de 20 \$/h y el costo de utilizar un vehículo de entregas promedio se puede estimar en 8 \$/h se puede realizar el siguiente análisis. En la **Figura 1**, el objetivo es minimizar el TT, por lo que se requiere en el vértice 2 una espera de 4 horas. Para así obtener un TT de 15 horas (1 hora + 2 horas + 3 horas + 3 horas + 3 horas + 2 horas + 1 hora) y una RD de 19 horas (TT + 4 horas de espera). Luego, el objetivo es minimizar la RD en la **Figura 2**. Entonces, en vez de ser la ruta 1-2-3 como en la **Figura 1**, la ruta es 1-3-2, de esta forma se evita la espera de 4 horas y el TT es de 17 horas, (1 hora + 3 horas + 3 horas + 4 horas + 3 horas + 2 horas + 1 hora) igual que la RD, ya que en este caso no hay tiempos de espera. Con este análisis el costo total de la ruta minimizando el TT es de \$500 ($15 \text{ h} \times 8 \text{ \$/h} + 19 \text{ h} \times 20 \text{ \$/h}$) y el de la ruta que minimiza la RD es de \$476 ($17 \text{ h} \times 8 \text{ \$/h} + 17 \text{ h} \times 20 \text{ \$/h}$). Sin embargo, como se menciona anteriormente, concentrarse solo en la RD tampoco es recomendable. Como se observa en la **Figura 3** y **Figura 4**, si en el vértice 6 la TW pasa de ser [18, 20] a [20, 22], para la minimización del TT no se genera ninguna variación. Sin embargo, para la minimización de la RD, este valor aumenta debido a la aparición de un tiempo de espera de 2 horas en el vértice 6 generando un nuevo costo total de \$516 ($17 \text{ h} \times 8 \text{ \$/h} + 19 \text{ h} \times 20 \text{ \$/h}$). Esto ejemplifica porque ambos criterios deben ser considerados en la función objetivo.

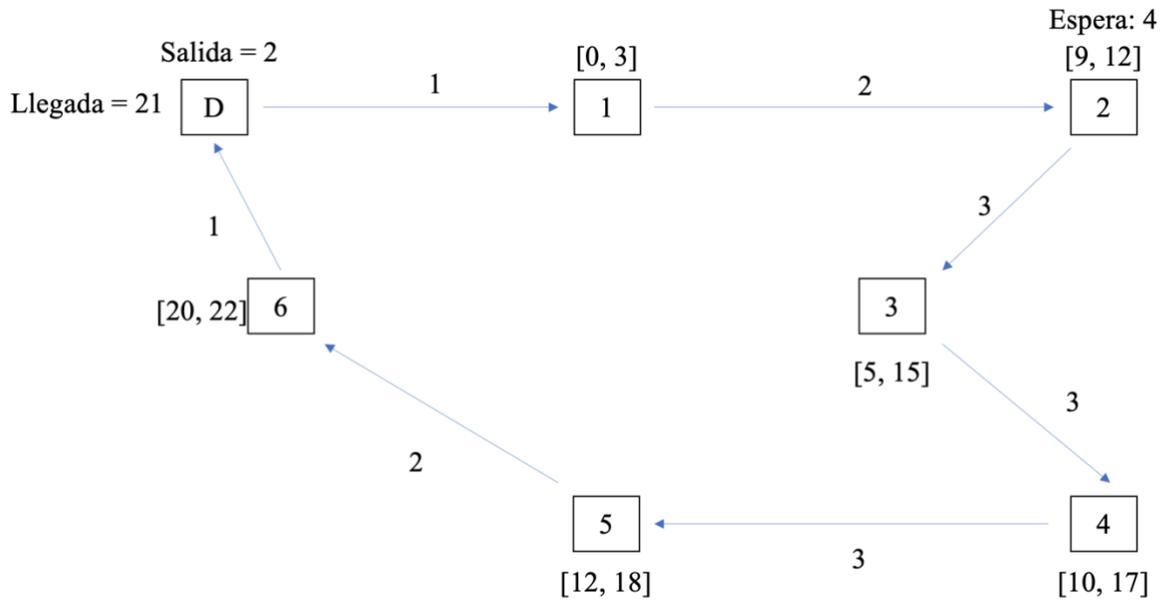


Figura 3: Minimización TT Caso 2.

Fuente: Bretin et al. (2021).

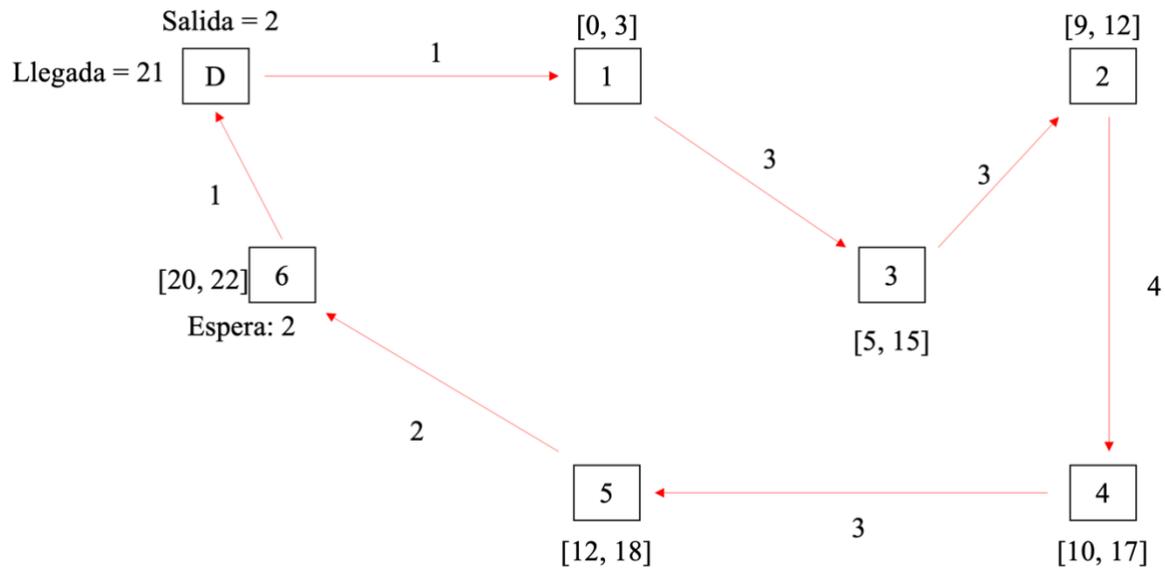


Figura 4: Minimización RD Caso 2.

Fuente: Bretin et al. (2021).

2.2. Modelo de programación lineal

A continuación, se presenta el modelo de programación lineal entera para resolver el TSPTW considerando el TT y la RD. Estos modelos se basan principalmente en los trabajos de Bretin et al. (2021) y Maffioli & Sciomachen (1997).

Variables de decisión:

$$x_{ij} = \begin{cases} 1, & \text{si se va del nodo } i \text{ al } j. \\ 0, & \text{en otro caso.} \end{cases}$$

y_{ij} = Tiempo de llegada y salida al nodo i si el nodo j es visitado después de i .

z = Cota superior que agrupa el TT y la RD.

Parámetros:

c_{ij} = Costo o distancia para ir del nodo i al nodo j .

r_i = Tiempo de inicio de la TW para el nodo i .

d_i = Tiempo de cierre de la TW para el nodo i .

$$\min z \tag{1}$$

$$\sum_{j \in V \setminus \{i\}} x_{ij} = 1, \quad \forall i \in V \setminus \{0\} \tag{2}$$

$$\sum_{i \in V \setminus \{j\}} x_{ij} = 1, \quad \forall j \in V \setminus \{0\} \tag{3}$$

$$\sum_{i \in V \setminus \{0\}} \sum_{j \in V \setminus \{0\}} x_{ij} c_{ij} \leq z \tag{4}$$

$$\sum_{i \in V \setminus \{0\}} y_{i0} + \sum_{i \in V \setminus \{0\}} x_{i0} c_{i0} \leq z \tag{5}$$

$$r_i x_{ij} \leq y_{ij} \leq d_i x_{ij}, \quad i \in V \setminus \{0\}, j \in V \setminus \{0\}, i \neq j \tag{6}$$

$$\sum_{i \in V \setminus \{0\}, i \neq j} y_{ij} + \sum_{i \in V \setminus \{0\}, i \neq j} x_{ij} c_{ij} \leq \sum_{k \in V \setminus \{0\}, k \neq j} y_{jk}, \quad \forall j \in V \setminus \{0\} \tag{7}$$

$$\sum_{i \in S} \sum_{j \in S} x_{ij} \leq |S| - 1, \quad S \subset V, \quad 2 \leq |S| \leq |V| - 2 \tag{8}$$

$$x_{ij} \in \{0,1\}, \quad i \in V, \quad j \in V \quad (9)$$

$$y_{ij} \geq 0, \quad i \in V \setminus \{0\}, \quad j \in V \setminus \{0\}, \quad i \neq j \quad (10)$$

La función objetivo (1) busca minimizar el costo total de la ruta a través de la minimización de una cota superior que engloba tanto el TT como la RD. Las restricciones (2) y (3) tienen como objetivo restringir que cada nodo sea alcanzado y abandonado una sola vez. Luego, la restricción (4) busca que la cota superior minimizada en la función objetivo contemple el tiempo de viaje. La restricción (5) tiene el mismo objetivo orientado a la minimización de la RD. Las restricciones (6) aseguran que el tiempo de llegada al nodo i sea mayor o igual al inicio de la TW y menor o igual al cierre de la TW. Las restricciones (7) están encargadas de generar que el tiempo de salida del nodo i más la distancia recorrida entre nodo i y el nodo j sea menor o igual que el tiempo de salida del nodo j al nodo k . Las restricciones (8) son la eliminación de sub-rutas propuesta por Dantzig et al. (1954), este conjunto de restricciones asegura la no formación de ciclos en la ruta, obligando a que solo se forme un único circuito hamiltoniano. Finalmente, las restricciones (9) y (10) son únicamente restricciones de integridad de las variables de decisión.

Es importante mencionar que para este trabajo, las restricciones (8) de sub-rutas DFJ son poco prácticas por su naturaleza exponencial. Por ello, se utilizan las restricciones MTZ propuesta por Miller et al. (1960) y GG propuesta por Gavish & Graves (1978). Así, la formulación MTZ utiliza una variable continua u_i que representa el número de ciudades visitadas hasta el nodo i . Las restricciones (11), (12) y (13) son las restricciones MTZ que tienen orden cuadrático.

$$u_0 = 0 \quad (11)$$

$$u_i - u_j + 1 \leq (n - 1)(1 - x_{ij}), \quad \forall (i, j) \in V, \quad i \neq j \quad (12)$$

$$1 \leq u_i \leq m, \quad \forall i \in V \setminus \{p\} \quad (13)$$

Donde la restricción (11) indica que el depósito siempre es el punto de partida de la ruta con un valor de la variable auxiliar de 0. Luego, las restricciones (12) se encargan de que los únicos sub-tour que se pueden generar sean aquellos con únicamente 0 elementos en la ruta. Finalmente, las restricciones (13) indican que la variable u toma valores de 1 hasta m para todos los vértices menos el depósito en el origen.

Por otro lado, también, la formulación GG incorpora una variable continua g_{ij} que representa la existencia de un flujo entre el nodo i y j . Las restricciones (14), (15) y (16) son las restricciones GG y tienen un orden cuadrático.

$$\sum_{j \in V} g_{0j} = 0 \quad (14)$$

$$\sum_{j \in V} g_{ij} - \sum_{j \in V} g_{ji} = 1, \quad \forall i \in V \setminus \{p\} \quad (15)$$

$$0 \leq g_{ij} \leq nx_{ij}, \quad \forall i, j \in V \quad (16)$$

Donde la restricción (14) indica que el flujo entre el depósito y la primera ciudad visitada será siempre 0 y a la vez que el depósito es el punto de partida de la ruta. Luego, las restricciones (15) aseguran que sólo existe un flujo único entrando al vértice i y un único flujo saliendo del vértice i . Finalmente, las restricciones (13) indican que la variable g_{ij} toma valores de uno hasta n para todos los arcos.

2.3. Revisión de la literatura

Como se menciona anteriormente, este problema es relevante y ha sido estudiado en su área dando como resultado un gran número de publicaciones con diversos métodos para resolverlo. Dentro de estas es posible destacar.

En Gendreau et al. (1998) se trabajó una heurística de inserción para el TSPTW con el objetivo de minimizar el TT. Esta heurística se basaba en construir una ruta, insertando un nuevo vértice en cada iteración en el vecindario de la ruta que se está trabajando. Luego, se realiza una re-optimización de la ruta obtenida al momento. Posteriormente, se realiza una optimización posterior basada en la adición y sustracción de todos los vértices. Los resultados mostraron que el procedimiento se comparaba muy bien con otros métodos actuales y muchas veces produce resultados óptimos o cercanos al óptimo. Las pruebas se llevaron a cabo en 375 instancias de tamaño hasta 100 ciudades con ventanas de tiempo de máximo 200 unidades de tiempo.

Corne et al. (1999) proponen un algoritmo múltiple de colonia de hormigas para abordar el VRPTW, llamado MACS – VRPTW. Se organiza con una jerarquía artificial de colonia de

hormigas para sucesivamente ir optimizando la función objetivo. La primera colonia minimiza el número de vehículos y la segunda colonia minimiza las distancias viajadas. Un elemento fundamental de esta propuesta es un procedimiento que permite a las colonias compartir su información, actualizándose constantemente. Los resultados muestran que el método es competitivo con los métodos más avanzados de la época, tanto en calidad de la respuesta como en los tiempos computacionales.

Ohlmann & Thomas (2007) describen una variante del algoritmo *simulated annealing* (SA), que principalmente incorpora un método de penalización variable. El SA incorpora el concepto de presión que es análogo al multiplicador de penalidad, esta variante relaja las restricciones de TW integrando el método de penalización en conjunto con un procedimiento de búsqueda estocástico. Los resultados computacionales validan el concepto de penalización variable contra la penalización estática, ya que, esta variante obtuvo numerosos resultados óptimos conocidos en la literatura.

En Kim (2007) se propone un algoritmo de inserción iterativo como una heurística de búsqueda constructiva. Esto junto a una metaheurística híbrida que se combina con un algoritmo SA y una búsqueda tabú con una selección aleatoria de dos intercambios. Además, un operador de movimiento simple como heurística de mejora. Las pruebas realizadas en este trabajo generan soluciones óptimas o cercanas al óptimo en gran parte de las 400 instancias evaluadas. Las instancias consideran tamaños de 20 a 200 ciudades, con TW de 20 a 200 unidades de tiempo.

Luego, Cheng & Mao (2007) propusieron un algoritmo de colonia de hormigas modificado, llamado ACS – TSPTW. Dos heurísticas de búsqueda son insertadas para manejar las restricciones de TW. Los resultados obtenidos de las pruebas mostraron que el desempeño de la ACO modificada es similar a la ACS – Time propuesta por Corne et al. (1999) y la ACO ya existente para resolver el TSPTW.

Dash et al. (2009) trabajaron en una formulación extendida del TSPTW, basado en una división de las TW en sub ventanas con una fuerte relajación de la programación lineal. De manera que cuando se obtienen los valores óptimos del problema, se generan límites inferiores fuertes y se usaron heurísticas eficientes y el método de *Branch and Bound*. Los

resultados de este trabajo presentaron resultados eficaces para resolver problemas no resueltos anteriormente.

Más tarde, da Silva & Urrutia (2010) proponen una búsqueda variable generalizada en el vecindario (GVNS) que está compuesta por fases tanto constructivas como de optimización. En la primera etapa, la heurística construye una solución factible usando búsqueda variable en el vecindario (VNS). Luego, en la etapa de optimización, la heurística mejora la solución a través de la GVNS. Tanto la etapa de construcción y optimización utilizan pruebas de eliminación, evaluaciones parciales de vecindarios y técnicas de partición de vecindarios. Los resultados mostraron ser eficientes ya que los costos computacionales se redujeron considerablemente y se mejoraron algunos resultados obtenidos a la fecha.

Luego, en Baldacci et al. (2012) se propone resolver el TSPTW utilizando una relajación del recorrido para calcular un límite inferior válido para instancias más difíciles. Este límite inferior es obtenido mediante un solución dual cuasi óptima que busca una combinación convexa de peso mínimo de recorridos no necesariamente elementales. Luego, la solución óptima se obtiene mediante un procedimiento iterativo que utiliza un algoritmo de programación dinámica, usando los límites obtenidos en la relajación. Los resultados de este trabajo muestran que es capaz de resolver instancias de hasta 233 vértices. Además de resolver todos los casos propuestos menos uno y superar a todos los métodos exactos publicados hasta el momento.

Karabulut & Tasgetiren (2012) proponen un algoritmo discreto de colonia de abejas artificial (DABC) con el objetivo de minimizar el TT de las rutas. La DABC se basa en un algoritmo iterativo *greedy* con fases constructivas y destructivas, para generar fuentes de alimentación vecinas (soluciones) en el marco de un algoritmo de abejas artificial. Adicionalmente, utilizan una búsqueda local 1-opt para mejorar las soluciones. Los indicadores mostraron que el desempeño resulta competitivo e incluso mejor que los mejores algoritmos generalmente usados en la literatura.

López-Ibáñez et al. (2013) proponen un enfoque que utiliza dos algoritmos, *Compressed Annealing* y *Beam* – ACO, para resolver el TSPTW, considerando la minimización del TT y el RD del recorrido. Los resultados obtenidos de este trabajo mostraron que entre los dos

algoritmos utilizados, *Beam – ACO* obtuvo resultados mejores que el *algoritmo Compressed Annealing*. Sin embargo, ambos algoritmos obtuvieron mejores o similares resultados que los utilizados propiamente para resolver el TSPTW con enfoque en la RD.

Mladenović et al. (2013) proponen una GVNS que usa un conjunto de distintos vecindarios, un nuevo procedimiento de verificación de ruta factibles y una estructura de datos más eficiente que las GVNS más recientes, que puede ser considerada como el estado del arte en este método. Los resultados muestran que este GVNS es más rápida y efectiva que las anteriores y fue capaz de mejorar 14 de 25 mejores resultados conocidos hasta la fecha para grandes instancias. Las instancias utilizadas para probar este método están compuestas desde tamaños de 200 a 400 ciudades, con ventanas de tiempo desde 100 unidades de tiempo hasta 500 unidades de tiempo.

Posteriormente, Yurtkuran & Emel (2014) proponen un algoritmo del tipo electromagnético (EMA). EMA utiliza mecanismos de atracción y repulsión para minimizar los costos del viaje, estos mecanismos se utilizan en partículas cargadas (soluciones) en un espacio multidimensional de optimización global. Este trabajo investiga la capacidad dentro del marco de un EMA para manejar problemas con restricciones específicas usando una nueva estrategia de límites variables. Dentro del marco, las partículas (soluciones) con restricciones de límites asociados a las ventanas de tiempo de los consumidores son introducidas en conjunto con un sistema de penalizaciones para eliminar soluciones no factibles asociadas a la violación de las ventanas de tiempo. Los resultados de este trabajo muestran que el algoritmo propuesto genera resultados óptimos o cercanos al óptimo con tiempos computacionales más cortos con respecto a otros algoritmos utilizados como indicadores.

Kara & Derya (2015) plantean una situación en la que el costo por unidad de viaje y el costo por unidad de espera son aparentemente iguales, por tanto, el problema puede resolverse minimizando la duración de viaje. Sin embargo, no existe una formulación para instancias simétricas ni asimétricas del TSPTW, por lo que los autores proponen una formulación lineal entera de tamaño polinomial para instancias simétricas y asimétricas por separado. Las conclusiones de este trabajo indican que para los casos simétricos el modelo propuesto es capaz de encontrar soluciones óptimas en tiempos muy cortos para problemas con más de

400 vértices. Por otro lado, para instancias asimétricas encuentra soluciones óptimas para problemas con hasta 232 vértices en pocos segundos.

Kóczy et al. (2018) proponen un algoritmo memético bacteriano discreto (DMBEA) basado en la combinación de un algoritmo evolutivo de bacterias y técnicas de búsqueda local. Este método fue probado con varias instancias tanto para el TSP como para el TSPTW. Centrándonos en el TSPTW, gran parte de las instancias utilizadas como indicadores mostraron que el DMBEA es capaz de encontrar soluciones óptimas y quedó en segundo lugar con respecto a las técnicas de vanguardia. Si bien este método resultó ser eficaz, otros algoritmos especializados tienen mejor desempeño.

En Amghar et al. (2019) se propone una GVNS para resolver el TSPTW cuando el objetivo es minimizar el tiempo de llegada al depósito. La primera fase ayuda a encontrar una solución factible utilizando la VNS propuesta por da Silva & Urrutia (2010). Mientras en la segunda fase, se adapta y mejora una GVNS existente propuesta por Mladenović et al. (2013) que entrega buenos resultados cuando se quiere minimizar la distancia. Los autores usaron métodos básicos de pre-procesado y propiedades de la matriz de tiempos de viaje, lo que permitió eliminar el espacio de búsqueda sin eliminar ninguna de las soluciones factibles. Los resultados indican que esta metodología es tan buena al menos como el estado del arte y en particular genera mejores soluciones para seis instancias cuando el objetivo es minimizar el tiempo de llegada al depósito.

En Bretin et al. (2021) consideran que existen consumidores comerciales con TW restringidas y clientes privados sin esta restricción. Se propone una aproximación multi-objetivo para resolver el TSPTW basada en una programación con restricciones que permite balancear la optimización de recursos humanos y materiales. Para resolver problemas reales a gran escala con escasa restricción de tiempo y reducir la escala de problemas reales, se introduce un enfoque que primero agrupa a los consumidores en *clústers* y luego resuelve el problema agrupado para finalmente resolver la secuencia dentro de cada clúster. Los resultados muestran que este método produce soluciones de mejor calidad que los algoritmos comúnmente utilizados en tiempos computacionales más cortos en instancias industriales de hasta 475 clientes.

Gillard & Schaus (2022) proponen una búsqueda local para resolver problemas de optimización combinatoria de forma eficiente. A través de una búsqueda local en un vecindario extensa (LNS) y un algoritmo genético de exploración en el vecindario basado en diagramas de decisión restringidos (DD) construidos con base en la mejor solución actual. En este trabajo se experimentó con una DD-LNS en dos problemas secuenciales: el TSPTW y un problema de planeación de producción. Los resultados relacionados con el TSPTW arrojaron que la metodología utilizada mejora algunas soluciones actualmente conocidas para instancias estándar e incluso logra generar soluciones óptimas para algunas instancias.

3. Algoritmo de búsqueda local iterada

A continuación, se presenta la estructura general del algoritmo, la representación de la solución, la heurística del vecino más cercano, las perturbaciones y finalmente, cuatro versiones del algoritmo propuesto.

3.1. Estructura general

La búsqueda local iterada (ILS) es un metaheurística que se basa en la aplicación de un algoritmo de búsqueda local a una solución inicial, la cual se obtiene por una perturbación de un óptimo local encontrado en forma previa y esto se realiza iterativamente (Lourenço et al., 2003).

La ILS está compuesta por una solución inicial, S , construida mediante una heurística aleatoria; una perturbación, que aplica un cambio en la solución para investigar en otro posible conjunto de soluciones; una búsqueda local, que consiste en una heurística de mejora y finalmente, un criterio de aceptación, que determina si se acepta a la nueva mejor solución encontrada, S^* , como solución actual S . En el **Algoritmo 1** se presenta el algoritmo propuesto basado en el ILS.

Como muestra el **Algoritmo 1**, se construye una solución actual, S , mediante la heurística del vecino más cercano (línea 1) que es momentáneamente almacenada en S^* como la mejor solución (línea 2). Una vez dentro del ciclo, la solución actual es sometida a una perturbación y posteriormente a una búsqueda local (líneas 4 y 5) que la modifica para compararla con S^* . Si S resulta ser mejor que la mejor solución encontrada ocurre una actualización de la mejor solución S^* (líneas 6 y 7). Posteriormente, el algoritmo pregunta si la solución actual es como máximo un 5% peor que la mejor solución encontrada (línea 8 y 9), si es así, ocurre una actualización de la solución actual. Caso contrario, se sigue con la solución actual (S) de la iteración anterior y así sucesivamente hasta la condición de término. Para el desarrollo de este trabajo se realizaron modificaciones al **Algoritmo 1** que son explicadas más adelante. Además, se explica el mecanismo de penalizaciones y otras variantes del algoritmo utilizado, las perturbaciones utilizadas y la representación de la solución.

Algoritmo 1: Búsqueda Local Iterada

Entradas: $V, c_{ij}, TW[r_i, d_i]$

Salidas: S^* , costo de la ruta

- 1) $S \leftarrow$ Vecino más cercano()
- 2) $S^* \leftarrow S$
- 3) **Repetir**
- 4) $S \leftarrow$ Perturbación(S)
- 5) $S \leftarrow$ Búsqueda local(S)
- 6) **Si** $f(S) < f(S^*)$
- 7) $S^* \leftarrow S$
- 8) **Si** Criterio de aceptación($f(S) \leq f(S^*) \times 1.05$)
- 9) $S_{i+1} \leftarrow S$
- 10) **Hasta** Condición de término

3.2. Solución inicial

El vecino más cercano (Flood, 1956) es una heurística constructiva clásica para el TSP. Esta funciona como solución inicial del **Algoritmo 1** y se presenta en el **Algoritmo 2**, esta asigna como punto de partida el depósito (línea 1). Luego, el depósito es removido del conjunto de nodos por visitar (línea 2). Posteriormente, mientras existan clientes por visitar (línea 3), se selecciona el que quede a menor distancia del destino actual (línea 4) y lo agrega a la ruta, mientras, es retirado del conjunto de cliente por visitar. Este algoritmo posee la cualidad de ser un buen punto de partida para la ILS ya que el tiempo computacional requerido para obtener la ruta es muy corto. Por último, al ser sólo una solución inicial no es necesario que la solución sea factible

Algoritmo 2: Vecino más cercano

Entradas: V, c_{ij}

Salida: S

1) $S \leftarrow 0$

2) $V \leftarrow V \setminus \{0\}$

3) **Mientras** $V \neq \emptyset$

4) $k \leftarrow \min \{c_{ij} : j \in V\}$

5) $S \leftarrow S \cup \{k\}$

6) $V \leftarrow V \setminus \{k\}$

7) **Finalizar**

3.3. Penalizaciones y representación de la solución

Debido a la gran cantidad de posibles rutas que pueden ser creadas a partir de diferentes destinos, es necesarios idear mecanismos que permitan reducir el espacios de búsqueda de posibles soluciones. Para esto, se utiliza como base el trabajo de Yurkuran & Emel (2014), que proponen un sistema de penalizaciones dinámico calculado que permite penalizar de mayor forma una ruta a medida que los tiempo de llegada a un nodo exceden el cierre de la ventana de tiempo. Es decir, si la llegada a un cliente excede el cierre de su TW por 5 unidades de tiempo, su penalización es menor que si la excede por 10 unidades de tiempo. El cálculo de las penalizaciones está incluido en la función de costos del algoritmo y las penalizaciones se van agregando a medida que la función de costo evalúa la ruta nodo a nodo. A continuación, se describe de mejor forma el sistema de penalizaciones propuesto, donde RD_i corresponde a la duración de la ruta hasta el nodo i y d_i corresponde al cierre de la ventana de tiempo en el nodo i .

En caso que la llegada al siguiente destino exceda d_i :

$$\text{Penalización} = RD_i - d_i$$

Esta penalización es evaluada desde el depósito en la partida y hasta el depósito en la finalización de la ruta, indicándole que la penalización tiene la misma magnitud que el retraso en el cliente i .

Por otro lado, durante el proceso de experimentación se observaron algunos casos en los que la ILS propuesta genera rutas con inicios distintos al depósito, a pesar de que el depósito es fijado para que siempre sea considerado como el primer nodo, lo que no es deseado, por lo que para esos casos se implementa la penalización:

$$\text{Penalización} = d_0$$

Esta penalización indica que en caso de que el punto de partida sea distinto al depósito, la penalización es equivalente al cierre de la TW del depósito, que siempre es el $\max \{d_i \forall i \in V\}$.

La solución del ILS está representada por una única lista que corresponde a la mejor ruta encontrada. La lista siempre comienza desde el depósito para posteriormente recorrer desde la primera hasta la última ciudad. En la **Figura 5** se presenta una solución de 5 ciudades.

0	C1	C2	C3	C4	C5
---	----	----	----	----	----

Figura 5: Representación de la solución.

Fuente: Elaboración propia.

3.4. Perturbaciones

Durante la búsqueda de mejores soluciones en el algoritmo, es posible que la solución quede atrapada en óptimos locales. Para evitar lo anterior, se aplican perturbaciones a la solución, y de esta manera poder escapar de los óptimos locales con la esperanza de acercarnos cada vez más al resultado óptimo. En el algoritmo propuesto se usan dos perturbaciones, con el depósito como nodo fijo en todo momento, el intercambio aleatorio de dos ciudades y la inversión del orden de las ciudades entre dos puntos seleccionados.

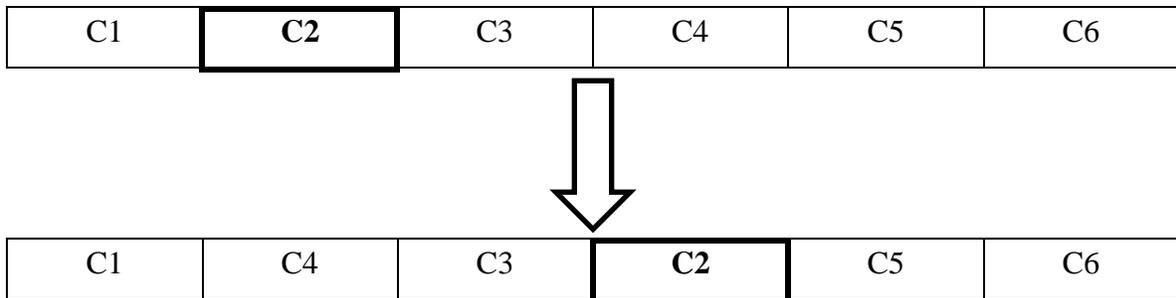


Figura 6: Intercambio aleatorio de dos ciudades.

Fuente: Elaboración propia.

3.4.1. Intercambio Aleatorio de dos Ciudades

Esta perturbación selecciona de forma aleatoria una ciudad en la ruta y la coloca en otro lugar, generando una nueva ruta, como se muestra en la **Figura 6**.

3.4.2. Inversión del Orden entre Ciudades

Esta perturbación selecciona aleatoriamente dos ciudades en la ruta e invierte el orden de todas las ciudades en la ruta entre las ciudades elegidas en la ruta. En la **Figura 7** se muestra un ejemplo en el que se invierten las ciudades que se encuentran entre las ciudades 2 y 6.

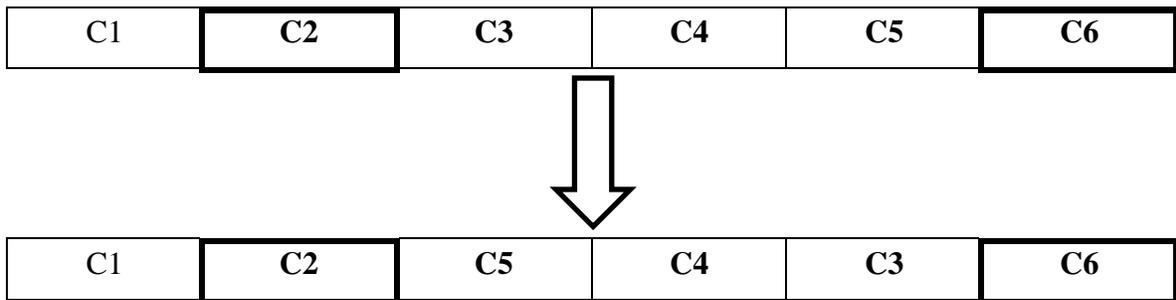


Figura 7: Inversión del orden entre ciudades.

Fuente: Elaboración propia.

3.5. Búsqueda local

Las búsquedas locales se utilizan con el fin de mejorar una solución mediante múltiples iteraciones en un mismo vecindario. La búsqueda utilizada en este trabajo consiste en generar un intercambio aleatorio en un ruta existente. Primero, evalúa si el costo de esta nueva ruta es menor al de la ruta existente y ambas no están penalizadas. Segundo, si la situación anterior

no ocurre, la búsqueda analiza si la ruta existente tiene una penalización mayor que cero y la nueva ruta no está penalizada. Por último, la búsqueda analiza si ambas están penalizadas y la nueva ruta tiene un menor costo.

En el **Algoritmo 3** se describe la búsqueda local. Esta comienza guardando el costo y la penalización de la solución actual (S) (líneas 1 y 2). Luego, comienza un ciclo iterativo aplicando un intercambio aleatorio a la ruta (línea 4) actual (S), generando una nueva solución candidata (S') para posteriormente almacenar el costo y la penalización de la solución candidata (líneas 5 y 6). El algoritmo analiza si la solución candidata (S') y la solución actual no tienen penalizaciones, si es así, la búsqueda analiza si la solución actual es mejor que la existente, en cuyo caso termina el ciclo (líneas 7 - 9) y entrega como nueva ruta a la solución candidata. De no ser así, se analiza si la solución actual es una ruta penalizada con respecto a la solución candidata, si la ruta candidata no está penalizada y la solución actual si, el ciclo iterativo termina entregando la solución candidata (líneas 10 y 11). Como último caso, se analiza cuál de las dos rutas tiene un costo menor considerando que ambas están penalizadas, si la solución candidata tiene un menor costo el ciclo iterativo termina retornando la solución candidata (líneas 12, 13 y 14).

Esta búsqueda procura analizar solamente los casos en que la aplicación de la perturbación genera una mejoría, en casos donde la perturbación no genera ninguna mejoría, luego de 10 iteraciones la ruta ya actual (S) continúa por la ILS inalterada.

Algoritmo 3: Búsqueda Local

Entradas: $S, c_{ij}, TW[r_i, d_i]$

Salida: S'

- 1) $C_0 \leftarrow f(S)$
- 2) $P_0 \leftarrow \text{penalización}(S)$
- 3) **Repetir**
- 4) $S' \leftarrow \text{Intercambio Aleatorio}(S)$
- 5) $C_1 \leftarrow f(S')$
- 6) $P_1 \leftarrow \text{penalización}(S')$
- 7) **Si** $P_0 = 0$ **y** $P_1 = 0$
- 8) **Si** $C_1 < C_0$
- 9) **Retornar** S'
- 10) **Si** $P_0 > 0$ **y** $P_1 = 0$
- 11) **Retornar** S'
- 12) **Si** $P_0 > 0$ **y** $P_1 > 0$
- 13) **Si** $C_1 < C_0$
- 14) **Retornar** S'
- 15) **Hasta** 10 iteraciones

3.6. Variaciones ILS

En esta sección se presenta de forma más extensa las cuatro variantes del ILS utilizadas que se basan en el **Algoritmo 1**. Así, las variaciones ILS 1 e ILS 2 construyen la nueva mejor solución sobre la anterior, sin importar si la solución actual está penalizada. Por otro lado, las variaciones ILS 3 e ILS 4 diferencian las penalizaciones, permitiendo generar nuevas soluciones factibles sobre la anterior solución factible y lo mismo ocurre con las infactibles.

3.6.1. ILS 1 e ILS 2

En el **Algoritmo 4** se presentan las variables ILS 1 e ILS 2.

Algoritmo 4: Búsqueda Local Iterada V1
Entradas: $V, c_{ij}, TW[r_i, d_i]$
Salidas: S^* , Costo de la ruta
1) $S \leftarrow \text{Vecino más cercano}()$
2) $S^* \leftarrow S$
3) Repetir
4) Si Aleatorio(0, 1) < α
5) Intercambio Aleatorio entre 2 Ciudades(S)
6) En otro caso Aleatorio(0, 1) < β
7) Búsqueda Local(S)
8) En otro caso
9) Inversión del Orden entre 2 Ciudades(S)
10) Si $f(S) < f(S^*)$
11) $S^* \leftarrow S$
12) Si Criterio de aceptación($f(S) \leq f(S^*) \times 1.05$)
13) $S_{i+1} \leftarrow S$
14) Hasta 100.000 iteraciones

Cómo se observa en el **Algoritmo 4**, la principal diferencia con el **Algoritmo 1** es el inicio del ciclo, en vez de realizar en forma sucesiva una perturbación seguida de una búsqueda local, se implementa de forma aleatoria el intercambio de ciudades, la búsqueda local y la inversión de la ruta entre dos ciudades (líneas 3 a la 9). Los valores de α y β son de 0.6 y 0 para la ILS 1 e ILS 3 y de 0.6 y 0.3 para la ILS 2 e ILS 4.

3.6.2. ILS 3 e ILS 4

En el **Algoritmo 5** se presentan las variantes ILS 3 e ILS 4. ILS 3 e ILS 4 poseen los mismos parámetros descritos para ILS 1 e ILS 2, respectivamente. En estas versiones inicialmente se utiliza el vecino más cercano para obtener la solución inicial (línea 1). Posteriormente, el algoritmo analiza si la solución actual (S) tiene o no penalización para realizar la actualización de las variables, S_f (mejor solución factible) si no hay penalización y S_{in} (mejor solución infactible) si la hay (líneas 2 a la 6). Desde la línea 6 a la 12, es igual a lo explicado en las líneas 3 a la 9 en el **Algoritmo 4**. Posteriormente, el algoritmo analiza si las ruta obtenidas están o no penalizadas (líneas 13 y 18) para realizar la actualización de variables si corresponde (líneas 14 - 15 y líneas 19 - 20). Finalmente, dependiendo si la última ruta obtenida tiene penalización, esta pasa a la siguiente iteración como solución actual, si es hasta un 5% peor que la mejor solución factible o infactible actual, según corresponda (líneas 16 - 17 y líneas 21 - 22).

Tabla 1: Parámetros búsquedas locales

ILS	α	β	Algoritmo 3
1 y 3	0,6	0,0	No
2 y 4	0,6	0,3	Sí

Fuente: Elaboración Propia

Algoritmo 5: Búsqueda Local Iterada V2

Entradas: $V, c_{ij}, TW[r_i, d_i]$

Salidas: S_f , Costo de la ruta

- 1) $S \leftarrow \text{Vecino más cercano}()$
- 2) **Si** penalización(S) = 0
- 3) $S_f \leftarrow S$
- 4) **En otro caso**
- 5) $S_{in} \leftarrow S$
- 6) **Repetir**
- 7) **Si** Aleatorio(0, 1) < α
- 8) Intercambio Aleatorio entre 2 Ciudades(S)
- 9) **En otro caso** Aleatorio(0, 1) < β
- 10) Búsqueda Local(S)
- 11) **En otro caso**
- 12) Inversión del Orden entre 2 Ciudades(S)
- 13) **Si** penalización(S) = 0
- 14) **Si** $f(S) < f(S_f)$
- 15) $S_f \leftarrow S$
- 16) **Si** Criterio de aceptación($f(S) \leq f(S_f) \times 1.05$)
- 17) $S_{i+1} \leftarrow S$
- 18) **En otro caso**
- 19) **Si** $f(S) < f(S_{in})$
- 20) $S_{in} \leftarrow S$
- 21) **Si** Criterio de aceptación($f(S) \leq f(S_{in}) \times 1.05$)
- 22) $S_{i+1} \leftarrow S$
- 23) **Hasta** 100.000 iteraciones

4. Experimentos y resultados

En este capítulo se presentan los detalles computacionales, la descripción de las instancias y los resultados.

4.1. Detalles computacionales

Para analizar el rendimiento de las ILS propuestas se comparan los resultados obtenidos con respecto a la programación lineal propuesta, a partir de las instancias propuestas en Gendreau et al. (1998). En el modelo matemático los experimentos son ejecutados con un tiempo límite de 3600 segundos, entregando un costo óptimo, factible o en su defecto el mejor límite inferior. Por otro lado, los experimentos resueltos por los algoritmos entregaron los mejores resultados a partir de un proceso de 10 iteraciones con una semilla independiente en cada una.

Los algoritmos fueron codificados usando Python 3.10.1 y el modelo matemático fue resuelto utilizando Gurobi 9.5.1. Todos los experimentos se llevaron a cabo utilizando un computador personal con un procesador Apple M1, utilizando solo un hilo de procesamiento y 8 GB de RAM con el sistema operativo macOS Monterey 12.3.1.

4.2. Descripción de las instancias

Para este trabajo se utilizaron 130 instancias propuestas en Gendreau et al. (1998) Las instancias seleccionadas están descritas de la forma $nXX.wYYY.00d$. Donde XX corresponde a la cantidad de ciudades que deben ser visitadas en la ruta e YYY corresponde al ancho de las TW. La cantidad de ciudades que deben ser visitadas (XX) abarca un rango de 20, 40, 60, 80 y 100 ciudades, para las cuales, las rutas con hasta 60 ciudades tienen TW (YYY) de 120, 140, 160, 180 y 200 unidades de tiempo, las rutas con 80 ciudades tienen TW de 100, 120, 140, 160, 180 y 200 unidades de tiempo, y las rutas con 100 ciudades tienen TW de 80, 100, 120, 140 y 160 unidades de tiempo. Finalmente, corresponde a la instancia número d para una ruta con XX ciudades y una ventana de tiempo de YYY unidades de tiempo siendo d un valor de 1 a 5. Para cada una de las instancias, el costo del viaje es igual a la distancia euclidiana entre ciudades, además, no existen los tiempos de servicio.

4.3. Resultados

La **Tabla 2** muestra la cantidad de instancias resueltas con una solución al menos factible para cada método. Luego, las siguientes tablas muestran el promedio de los valores obtenidos de todas las instancias correspondientes a la ruta que se está evaluando, es decir, abarca la ruta con todas sus ventanas de tiempo y las cinco instancias para cada TW. El Gap que es calculado como el cociente, de la resta del valor obtenido y el valor óptimo en la literatura, con el valor obtenido para la instancia, es obtenido con respecto a los valores entregados por Gendreau et al. (1998). Además, los resultados comparados entre los modelos de programación lineal entera mixta y la literatura corresponden sólo a aquellos que presentan una solución factible en los dos modelos matemáticos. Mientras que para las ILS, los resultados comparados corresponden a las instancias en que las cuatro alternativas encontraron una solución factible.

En lo que respecta a los modelos matemáticos en términos generales, las rutas de hasta 60 ciudades no representan mayor dificultad para ninguno de los dos modelos, como se observa en la **Tabla 2**. Posteriormente, el modelo MTZ para la ruta de 80 ciudades comienza a mostrar problemas en relación con la capacidad para resolver todas las instancias en 3600 segundos. Por otro lado, el modelo GG mantiene un buen desempeño. Finalmente, para las rutas de 100 ciudades, el modelo MTZ presenta una leve mejoría, aunque no es capaz de resolver la mayoría de las instancias. Mientras, el modelo GG mantiene el buen desempeño no logrando resolver sólo tres instancias.

Por otro lado, en lo que respecta a las ILS, los Gap obtenidos con respecto a las instancias de referencia tiende a ser casi 0 y son capaces de generar más respuestas óptimas o factibles en un menor tiempo. Además, las ILS presentan un comportamiento similar al modelo matemático MTZ, ya que el desempeño disminuye de forma moderada en las rutas con 60 ciudades para posteriormente ser notorio en las rutas de 80 y 100 ciudades.

Con respecto a la convergencia de los resultados en las búsquedas locales existe una tendencia clara con respecto a la iteración en la que la metaheurística encuentra la mejor solución. En la ILS 1 el número de iteraciones necesarias para encontrar la mejor solución sube de forma sostenida, para en promedio, necesitar aproximadamente 13 mil iteraciones

para instancias con 20 ciudades, 19 mil iteraciones para instancias con 40 ciudades, 26 mil iteraciones para 60 ciudades, 57 y 58 mil iteraciones para 80 y 100 ciudades respectivamente. Esta tendencia se mantiene en las demás instancias y búsquedas locales, a excepción de las iteraciones en la ILS 2 para instancias de 40 ciudades donde las iteraciones bajan de 20 mil a 13 mil iteraciones aproximadamente y en ILS 3 para instancias de 60 ciudades donde las iteraciones para encontrar la mejor solución en promedio bajan de 32 mil a 23 mil iteraciones.

Considerando que las rutas de 20, 40 y 60 nodos tienen TW de 120, 140, 160, 180 y 200 unidades de tiempo, las rutas de 80 nodos agregan a las TW anteriores las de 100 unidades de tiempo y las rutas de 100 nodos tienen TW de 80, 100, 120, 140 y 160 unidades de tiempo.

Los resultados muestran que para la ILS 1, 13 de los 20 resultados no factibles están en instancias con TW de 120 unidades de tiempo o menos, 2 en las instancias con TW de 140 unidades de tiempo y el resto de las instancias están por sobre esta ventana de tiempo. También, para ILS 2 los resultados muestran que 16 de las 20 instancias con resultados no factibles están en las TW de 120 unidades de tiempo o menos, 1 instancia en la TW de 140 unidades de tiempo y 1 instancia en la TW de 180 unidades de tiempo.

Para ILS 3, 11 de las 15 instancias no factibles están en las TW de 120 unidades de tiempo o menos, 2 en las instancias con TW de 140 unidades de tiempo y los restantes 2 resultados no factibles en instancias con 160 y 180 unidades de tiempo. En ILS 4 los resultados muestran que 14 de las 15 instancias no factibles están en TW de 120 unidades de tiempo o menos y la restante instancia tiene una TW de 140 unidades de tiempo.

Tabla 2: Soluciones encontradas por método.

Método	Número de instancias resueltas
MTZ	94/130 (72%)
GG	123/130 (95%)
ILS 1	110/130 (84%)
ILS 2	112/130 (86%)
ILS 3	115/130 (88%)
ILS 4	115/130 (88%)

Fuente: Elaboración Propia.

4.3.1. Instancias de 20 ciudades

Cómo se puede observar en la **Tabla 3**, para las rutas de 20 ciudades ninguno de los dos modelos tiene mayores dificultades para resolver las instancias propuestas. Obteniendo solamente resultados óptimos y resolviendo todas la instancias con una holgura considerable, con respecto de los 3600 segundos entregados para resolver cada instancia.

Tabla 3: Resultados modelos 20 ciudades

BKS	MTZ	Best bound	Gap (%)	Tiempo promedio	Infactibles	GG	Best bound	Gap (%)	Tiempo promedio	Infactibles
302,04	302,04	302,04	0,00	99,17	0	302,04	302,04	0,00	88,38	0

Fuente: Elaboración propia

En lo que respecta a las ILS, la **Tabla 4** permite realizar un análisis bastante similar con resultados en su mayoría óptimos, con un Gap Promedio máximo del 2,7% y con todas las instancias resueltas de forma satisfactoria. En relación con cada una de las ILS, las que obtienen mejores resultados corresponden a la ILS 2 e ILS 4. Sin embargo, la diferencia de los resultados obtenidos con respecto a la ILS 1 e ILS 3 para instancias con 20 ciudades es despreciable y estas últimas requieren sólo la mitad del tiempo.

Tabla 4: Resultados búsquedas locales 20 ciudades.

	Solución (Promedio mejores resultados)	Gap solución (%)	Promedio (Factibles y óptimas)	Gap promedio soluciones (%)	Tiempo promedio (s)	Rutas infactibles	Iteración mejor solución
ILS 1	303,680	0,5	309,404	2,4	3,272	0	13204
ILS 2	302,560	0,2	308,248	2,0	7,397	0	20663
ILS 3	303,160	0,4	310,427	2,7	3,109	0	25242
ILS 4	302,840	0,3	309,297	2,3	7,046	0	13455

Fuente: Elaboración Propia.

4.3.2. Instancias de 40 ciudades

Para las instancias con tamaño de 40 ciudades, la **Tabla 5** muestra que ambos modelos son capaces de encontrar soluciones factibles u óptimas. Sin embargo, para este tamaño de instancias, el modelo GG es levemente mejor que el modelo MTZ debido a que los resultados del modelo GG son iguales a la literatura y lo del modelo MTZ son ligeramente mayores. También el modelo GG tiene un menor Gap, sin embargo, presenta un tiempo promedio para resolver una instancia 58 segundos mayor que el modelo MTZ y unos límites inferiores también levemente mayores.

Tabla 5: Resultados modelos 40 ciudades

BKS	MTZ	Best bound	Gap (%)	Tiempo promedio	Infactibles	GG	Best bound	Gap (%)	Tiempo promedio	Infactibles
439,00	439,60	417,96	5,13	1916,63	0	439,00	420,16	4,51	1974,96	0

Fuente: Elaboración propia

En lo que respecta a las ILS, en la **Tabla 6**, se observa que aparecen las primera rutas sin resultados factibles en los algoritmos ILS 1 e ILS 2 para instancias de 40 ciudades. Hay una diferencia marcada entre los algoritmos que no diferencian entre rutas factibles o no para construir una mejor solución (ILS 1 e ILS 2) y los que si lo hacen (ILS 3 e ILS 4) a favor de estos últimos. También, entre los algoritmos con mejores resultados es el ILS 3 que, no utiliza la búsqueda local, siendo este también el que menor tiempo para encontrar un solución.

Tabla 6: Resultados búsquedas locales 40 ciudades.

	Solución (Promedio mejores resultados)	Gap solución (%)	Promedio (Factibles y óptimas)	Gap promedio soluciones (%)	Tiempo promedio (s)	Rutas infactibles	Iteración mejor solución
ILS 1	436,455	0,5	468,650	6,7	6,096	1	19274
ILS 2	436,591	0,5	468,105	6,6	14,173	3	13066
ILS 3	437,182	0,6	450,159	3,5	5,777	0	32251
ILS 4	439,864	1,2	453,375	4,2	13,333	0	22146

Fuente: Elaboración propia.

4.3.3. Instancias de 60 ciudades

La **Tabla 7** muestra que para instancias de tamaño de 60 ciudades, ambos modelos presentan sus primeras rutas sin soluciones factibles en el tiempo disponible. Sin embargo, aunque el modelo MTZ tiene 2 instancias sin solución y el GG una sola instancia sin solución. El modelo GG es ampliamente superior en el tiempo necesario para resolver las instancias con un tiempos 25,4% más rápidos. Además, el Gap del modelo GG es mejor en un 18,22%, a pesar de que las diferencias son pequeñas entre el BKS, el modelo GG sigue teniendo un mejor desempeño.

Tabla 7: Resultados modelos 60 ciudades

BKS	MTZ	Best bound	Gap (%)	Tiempo promedio	Infactibles	GG	Best bound	Gap (%)	Tiempo promedio	Infactibles
588,55	590,13	565,35	4,50	1641,80	2	589,83	569,57	3,68	1223,42	1

Fuente: Elaboración propia

La **Tabla 8** muestra que las variantes de la ILS para instancias con 60 ciudades, también tuvieron casos en los que no fueron capaces de encontrar soluciones como mínimo factible. Las ILS que tuvieron incapacidad de tener todas las instancias una solución factible, corresponden a aquellas que no utilizan la búsqueda local (ILS 1 e ILS 3). En este caso, ocurre por primera vez que las ILS que demandan más tiempo son las que obtienen los

mejores resultados. Esto muestra de forma evidente que la dificultad va incrementando, necesitando de procesos más complejos que perturbaciones.

Tabla 8: Resultados búsquedas locales 60 ciudades.

	Solución (Promedio mejores resultados)	Gap solución (%)	Promedio (Factibles y óptimas)	Gap promedio soluciones (%)	Tiempo promedio (s)	Rutas infactibles	Iteración mejor solución
ILS 1	589,609	0,2	730,513	15,6	8,870	2	25829
ILS 2	591,217	0,4	670,352	10,7	20,350	0	26556
ILS 3	589,652	0,2	602,943	2,3	8,448	2	23985
ILS 4	591,261	0,4	601,259	2,1	19,573	0	38240

Fuente: Elaboración propia.

4.3.4. Instancias de 80 ciudades

La **Tabla 9** muestra un rendimiento muy diferente en los dos modelos, siendo el MTZ el modelo de peor desempeño y el GG con un rendimiento bastante similar al de las instancias de 60 ciudades. El modelo MTZ tiene una disminución en su rendimiento drástica, pasando de solo dos instancias no factibles a veinte. Por otro lado, si bien el GG también tiene una disminución en su rendimiento, este modelo solo pasa de una instancias no factible a tres. Además, los tiempos promedios para encontrar una solución son muy distintos, con una diferencia del 33% entre ambos modelos para encontrar una solución, siendo el modelo MTZ el con peor desempeño. También, las diferencias entre las soluciones y el Gap son bastante grandes. Finalmente, el *best bound* entre ambos es similar, pero el modelo MTZ es el peor.

Tabla 9: Resultados modelos 80 ciudades

BKS	MTZ	Best bound	Gap (%)	Tiempo promedio	Infactibles	GG	Best bound	Gap (%)	Tiempo promedio	Infactibles
679,70	692,90	663,40	4,24	2351,47	20	679,70	661,3	2,87	1572,77	3

Fuente: Elaboración propia

El rendimiento observado en la **Tabla 10**, para las ILS es similar que el obtenido para instancias de 60 ciudades en términos generales. Las ILS que obtuvieron mejores resultados son aquellas que diferencian entre las soluciones según su factibilidad para generar una nueva mejor solución. Sin embargo, a diferencia de las instancias de 60 ciudades, donde ILS 4 fue el algoritmo con mejor rendimiento, para instancias de 80 ciudades fue ILS 3. Los resultados obtenidos por ILS 3 muestran resultados levemente mejores que ILS 4. Sin embargo, los tiempos computacionales requeridos son menos de la mitad y centrándonos en la cantidad de instancias resultas no hay dudas sobre el procedimiento con mejores resultados.

Tabla 10: Resultados búsquedas locales 80 ciudades.

	Solución (Promedio mejores resultados)	Gap solución (%)	Promedio (Factibles y óptimas)	Gap promedio soluciones (%)	Tiempo promedio (s)	Rutas infactibles	Iteración mejor solución
ILS 1	671,700	0,4	968,685	24,9	11,729	7	57506
ILS 2	674,950	0,9	1004,780	26,0	27,149	5	50513
ILS 3	674,400	0,8	682,620	2,0	11,398	4	52062
ILS 4	675,700	1,0	686,296	2,5	25,960	5	49251

Fuente: Elaboración propia.

4.3.5. Instancias de 100 ciudades

Finalmente, para las instancias con 100 ciudades, los resultados obtenidos para el modelo MTZ siguen siendo los peores como muestra la **Tabla 11**. Las instancias sin una solución son catorce para el modelo MTZ y el modelo GG mantuvo el rendimiento con sólo tres instancias no factibles. Luego, el tiempo promedio es menor para el modelo MTZ, pasando de un 33% más de tiempo que el modelo GG, a necesitar un 22% menos de tiempo que el modelo GG. Además, el Gap para este tamaño de instancia es mucho mejor para el modelo MTZ, siendo de 1,75% en contraste con el GG que es de 7,75%. Por otro lado, las mejores cotas encontradas son las del modelo GG que es mejor con diferencia y las soluciones encontradas por este modelo se aproximan de mejor forma a las de la literatura.

Tabla 11: Resultados modelos 100 ciudades

BKS	MTZ	Best bound	Gap (%)	Tiempo promedio	Infactibles	GG	Best bound	Gap (%)	Tiempo promedio	Infactibles
878,36	884,30	864,36	1,75	1868,83	14	878,64	808,91	7,75	1911,17	3

Fuente: Elaboración propia

Posteriormente, como se muestra en la **Tabla 12**, las ILS 3 e ILS 4 tienen un mejor desempeño para las instancias de 100 ciudades en términos generales. Mientras que en ILS 3 y 4 se repiten el mismo desempeño con la tercera opción como la alternativa con mejor desempeño. Sin embargo, centrándose únicamente en la cantidad de soluciones factibles encontradas, es la primera vez que tres de las cuatro ILS propuestas no son capaces de encontrar la misma cantidad de soluciones factibles, con 10 instancias con soluciones infactibles para las alternativas 1, 2 y 4.

Tabla 12: Resultados búsquedas locales 100 ciudades.

	Solución (Promedio mejores resultados)	Gap solución (%)	Promedio (Factibles y óptimas)	Gap promedio soluciones (%)	Tiempo promedio (s)	Rutas infactibles	Iteración mejor solución
ILS 1	889,200	0,0	1231,207	17,7	15,447	10	58150
ILS 2	889,200	0,0	1344,140	24,1	36,355	10	60697
ILS 3	889,200	0,0	893,303	0,5	15,933	9	64345
ILS 4	889,200	0,0	893,263	0,5	32,506	10	64154

Fuente: Elaboración Propia.

5. Conclusiones

En este trabajo se utilizaron distintos métodos para resolver el TSPTW con la función objetivo que considera los términos de *time travel* y *route duration*, que se adecúa mejor para el caso de los servicios postales. El uso de metaheurísticas resulta positivo en lo que respecta al tiempo necesario para obtener buenos resultados con tiempos considerablemente más pequeños que el modelo de programación lineal entera mixta. Sin embargo, es claro que un factor muy importante en el desempeño de los modelos es la restricción de sub – rutas, ya que, este método requiere tiempos mucho mayores. Pero, fue el modelo GG que obtuvo la mejor calidad y cantidad de soluciones.

El desempeño tanto de los modelos propuestos como de los distintos algoritmos, muestra de forma clara que la dificultad para generar nuevas soluciones aumenta en conjunto con la cantidad de ciudades en la instancia. Sin embargo, el desempeño de los dos modelos muestra diferencias evidentes, ya que el modelo MTZ obtiene 36 instancias sin solución y el GG sólo siete instancias sin solución. Lo anterior, permite suponer que las restricciones basadas en flujos a través de los arcos resultan ser más eficientes y eficaces que las restricciones centradas en los vértices.

Los resultados muestran que si bien las perturbaciones son útiles para encontrar un buen número de soluciones. ILS 1, que sólo utiliza perturbaciones, tiene en promedio cuatro instancias por ruta sin solución. Posteriormente, el uso de la búsqueda local mejora este rendimiento con 3,6 instancias infactibles por ruta para la ILS 2, mostrando que procedimientos más elaborados pueden ser útiles para encontrar más soluciones. Sin embargo, las ILS 3 y 4, ambas obtienen tres instancias no factibles, muestran que puede ser más importante utilizar métodos más inteligentes para generar soluciones que métodos basados en fuerza bruta. Lo anterior, se desprende de que ILS 3 utiliza búsqueda local y la ILS 4 no, sin embargo, ambas opciones discriminan si una nueva solución se debe generar en base a la mejor solución infactible o factible.

El estudio de las ventanas de tiempo muestra que mientras más acotadas son le es más difícil a la metaheurística encontrar soluciones a las rutas, no siendo generalmente el tamaño de la

ruta un factor importante. Si bien rutas más grandes son más difíciles de resolver, la mayoría de las instancias sin solución se concentran en las TW más acotadas.

También se observa una tendencia que muestra que a mientras las rutas van aumentando la cantidad de nodos en el ciclo se necesitan más iteraciones para encontrar una mejor solución.

Además, existen distintos mecanismos en etapas más tempranas del procedimientos que pueden ayudar a disminuir los tiempos computacionales y mejorar los resultados obtenidos. Ejemplo de esos pueden ser sistemas de penalizaciones variables que discriminen de mejor forma la magnitud de la penalización o métodos de procesamiento que descarten sub – rutas antes de iniciar la búsqueda. Finalmente, implementar este trabajo en un lenguaje de programación compilado, como C++, pues Python es un lenguaje interpretado y suelen ser más lentos que los compilados.

Finalmente, en relación con los servicios postales este trabajo se puede considerar como una buena base para seguir trabajando la optimización de los servicios de postales debido a que presenta resultados aceptables en cuanto a servicios postales simples como puede ser la entrega de cartas . La modelación matemática muestra que el modelo GG se puede seguir refinando agregando más parámetros de costo y refinando las cotas. Por otro lado también en las metaheurísticas se pueden agregar nuevos parámetros como costos y tiempo de servicio de la mano con otros algoritmos para generar soluciones mejores y en menor tiempo.

6. Referencias

- Amghar, K., Cordeau, J.-F., & Gendron, B. (2019). A General Variable Neighborhood Search Heuristic for the Traveling Salesman Problem with Time Windows under Completion Time Minimization. In *CIRRELT* (Vol. 29).
- Baldacci, R., Mingozzi, A., & Roberti, R. (2012). New state-space relaxations for solving the traveling salesman problem with time windows. *INFORMS Journal on Computing*, 24(3), 356–371.
- Bretin, A., Desaulniers, G., & Rousseau, L. M. (2021). The traveling salesman problem with time windows in postal services. *Journal of the Operational Research Society*, 72(2), 383–397.
- Cheng, C. bin, & Mao, C. P. (2007). A modified ant colony system for solving the travelling salesman problem with time windows. *Mathematical and Computer Modelling*, 46(9–10), 1225–1235.
- Corne, I. D., Dorigo, M., & Glover, F. (1999). MACS-VRPYW: A multiple ant colony system for vehicle routing problems with time windows. In *McGraw-Hill* (pp. 63–76).
- da Silva, R. F., & Urrutia, S. (2010). A General VNS heuristic for the traveling salesman problem with time windows. *Discrete Optimization*, 7(4), 203–211.
- Dantzig, G., Fulkerson, R., & Johnson, S. (1954). Solution of a Large-Scale Traveling-Salesman Problem. *Journal of the Operations Research Society of America*, 2(4), 393–410.
- Dash, S., Günlük, O., Lodi, A., & Tramontani, A. (2009). A Time Bucket Formulation for the TSP with Time Windows. *Infoms Journal of Computing*, 24(1), 132–147.
- David L. Applegate, Robert E. Bixby, Vašek Chvátal, & William J. Cook. (2007). *The Traveling Salesman Problem: A Computational Study*. Princeton University Press.
- Flood, M. M. (1956). The Traveling-Salesman Problem. *Operations Research*, 4(1), 61–75.
- Gavish, B., & Graves, S. C. (1978). *The Traveling Salesman Problem and Related Problems* (OR 078-78; Operation Research Center Working Paper).
- Gendreau, M., Hertz, A., Laporte, G., & Stan, M. (1998). A generalized insertion heuristic for the traveling salesman problem with time windows. *Operations Research*, 46(3).

- Gillard, X., & Schaus, P. (2022, July 23). Large Neighborhood Search with Decision Diagrams. *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence*.
- Kara, I., & Derya, T. (2015). Formulations for Minimizing Tour Duration of the Traveling Salesman Problem with Time Windows. *Procedia Economics and Finance*, 26, 1026–1034.
- Karabulut, K., & Tasgetiren, M. F. (2012). A discrete artificial bee colony algorithm for the traveling salesman problem with time windows. *2012 IEEE Congress on Evolutionary Computation, CEC 2012*.
- Karp, R. M. (1972). Reducibility Among Combinatorial Problems. *The IBM Research Symposia Series*. Springer, Boston, MA., 85–113.
- Kim, B.-I. (2007). An Iterative Insertion Algorithm and a Hybrid Meta Heuristic for the Traveling Salesman Problem with Time Windows. In *Journal of the Korean Institute of Industrial Engineers* (Vol. 33, Issue 1).
- Kóczy, L. T., Földesi, P., & Tű-Szabó, B. (2018). Enhanced discrete bacterial memetic evolutionary algorithm - An efficacious metaheuristic for the traveling salesman optimization. *Information Sciences*, 460–461, 389–400.
- López-Ibáñez, M., Blum, C., Ohlmann, J. W., & Thomas, B. W. (2013). The travelling salesman problem with time windows: Adapting algorithms from travel-time to makespan optimization. *Applied Soft Computing Journal*, 13(9), 3806–3815.
- Lourenço, H. R., Martin, O. C., & Stützle, T. (2003). Iterated Local Search. In F. Glover & G. A. Kochenberger (Eds.), *Handbook of Metaheuristics* (Vol. 57, pp. 320–353). Springer.
- Maffioli, F., & Sciomachen, A. (1997). A mixed-integer model for solving ordering problems with side constraints. *Annals of Operations Research*, 69, 277–297.
- Miller, C. E., Tucker, A. W., & Zemlin, R. A. (1960). Integer Programming Formulation of Traveling Salesman Problems. *Journal of the ACM*, 7(4), 326–329.
- Mladenović, N., Todosijević, R., & Urošević, D. (2013). An efficient general variable neighborhood search for large travelling salesman problem with time windows. *Yugoslav Journal of Operations Research*, 23(1), 19–30.

- Ohlmann, J. W., & Thomas, B. W. (2007). A Compressed-Annealing Heuristic for the Traveling Salesman Problem with Time Windows. *INFORMS Journal on Computing*, 19(1), 80–90.
- Wang, F.-S., & Chen, L.-H. (2013). Encyclopedia of Systems Biology. In *Encyclopedia of Systems Biology*. Springer New York.
- Yurtkuran, A., & Emel, E. (2014). Efficient constraint handling in electromagnetism-like algorithm for traveling salesman problem with time windows. *The Scientific World Journal*, 2014.

Anexos

Tabla A 1: Resultados 20 ciudades modelo MTZ.

Instancia	BKS	MTZ	Best bound	Gap	Tiempo ejecución
n20w120.001	337	337	337	0,00	6,71
n20w120.002	246	246	246	0,00	4,95
n20w120.003	347	347	347	0,00	2,02
n20w120.004	353	353	353	0,00	19,85
n20w120.005	315	315	315	0,00	7,38
n20w140.001	230	230	230	0,00	313,29
n20w140.002	307	307	307	0,00	5,61
n20w140.003	301	301	301	0,00	409,53
n20w140.004	318	318	318	0,00	5,54
n20w140.005	275	275	275	0,00	4,23
n20w160.001	347	347	347	0,00	3,08
n20w160.002	250	250	250	0,00	25,86
n20w160.003	331	331	331	0,00	8,74
n20w160.004	287	287	287	0,00	4,96
n20w160.005	342	342	342	0,00	163,94
n20w180.001	353	353	353	0,00	15,05
n20w180.002	347	347	347	0,00	6,16
n20w180.003	315	315	315	0,00	56,41
n20w180.004	284	284	284	0,00	824,79
n20w180.005	257	257	257	0,00	56,06
n20w200.001	259	259	259	0,00	51,29
n20w200.002	242	242	242	0,00	104,39
n20w200.003	305	305	305	0,00	40,52
n20w200.004	326	326	326	0,00	274,83
n20w200.005	277	277	277	0,00	64,01

Fuente: Elaboración Propia.

Tabla A 2: Resultado 20 ciudades modelo GG.

Instancia	BKS	GG	Best bound	Gap	Tiempo ejecución
n20w120.001	337	337	337	0,00	3,91
n20w120.002	246	246	246	0,00	3,81
n20w120.003	347	347	347	0,00	1,49
n20w120.004	353	353	353	0,00	22,96
n20w120.005	315	315	315	0,00	4,81
n20w140.001	230	230	230	0,00	181,61
n20w140.002	307	307	307	0,00	13,54
n20w140.003	301	301	301	0,00	369,32
n20w140.004	318	318	318	0,00	5,14
n20w140.005	275	275	275	0,00	4,52
n20w160.001	347	347	347	0,00	6,89
n20w160.002	250	250	250	0,00	18,14
n20w160.003	331	331	331	0,00	13,20
n20w160.004	287	287	287	0,00	2,62
n20w160.005	342	342	342	0,00	133,48
n20w180.001	353	353	353	0,00	54,25
n20w180.002	347	347	347	0,00	6,54
n20w180.003	315	315	315	0,00	32,12
n20w180.004	284	284	284	0,00	477,00
n20w180.005	257	257	257	0,00	36,29
n20w200.001	259	259	259	0,00	23,64
n20w200.002	242	242	242	0,00	395,62
n20w200.003	305	305	305	0,00	34,57
n20w200.004	326	326	326	0,00	116,09
n20w200.005	277	277	277	0,00	47,92

Fuente: Elaboración Propia.

Tabla A 3: Resultados 20 Ciudades ILS 1.

Instancia	Mínimo	Gap Min ILS	Promedio	Gap prom ILS	Tiempo ejecución
n20w120.001	337	0,00%	339,80	0,82%	3,32
n20w120.002	246	0,00%	251,20	2,07%	3,28
n20w120.003	347	0,00%	350,90	1,11%	3,30
n20w120.004	353	0,00%	355,10	0,59%	3,28
n20w120.005	315	0,00%	321,20	1,93%	3,25
n20w140.001	232	0,86%	241,80	4,88%	3,25
n20w140.002	308	0,32%	315,00	2,54%	3,26
n20w140.003	301	0,00%	302,00	0,33%	3,24
n20w140.004	318	0,00%	320,50	0,78%	3,26
n20w140.005	275	0,00%	275,00	0,00%	3,25
n20w160.001	347	0,00%	347,00	0,00%	3,25
n20w160.002	254	1,57%	255,20	2,04%	3,30
n20w160.003	331	0,00%	331,70	0,21%	3,29
n20w160.004	287	0,00%	287,00	0,00%	3,25
n20w160.005	342	0,00%	342,00	0,00%	3,26
n20w180.001	353	0,00%	365,80	3,50%	3,25
n20w180.002	347	0,00%	354,00	1,98%	3,30
n20w180.003	315	0,00%	329,60	4,43%	3,33
n20w180.004	288	1,39%	290,40	2,20%	3,26
n20w180.005	257	0,00%	257,50	0,19%	3,25
n20w200.001	259	0,00%	259,60	0,23%	3,26
n20w200.002	242	0,00%	266,90	9,33%	3,26
n20w200.003	328	7,01%	341,80	10,77%	3,27
n20w200.004	333	2,10%	353,40	7,75%	3,27
n20w200.005	277	0,00%	280,70	1,32%	3,27

Fuente: Elaboración Propia.

Tabla A 4: Resultados 20 Ciudades ILS 2.

Instancia	Mínimo	Gap Min ILS	Promedio	Gap prom ILS	Tiempo ejecución
n20w120.001	337	0,00%	343,50	1,89%	7,45
n20w120.002	246	0,00%	253,80	3,07%	7,38
n20w120.003	347	0,00%	351,40	1,25%	7,44
n20w120.004	353	0,00%	360,10	1,97%	7,42
n20w120.005	315	0,00%	318,90	1,22%	7,39
n20w140.001	232	0,86%	234,90	2,09%	7,38
n20w140.002	307	0,00%	311,20	1,35%	7,41
n20w140.003	301	0,00%	301,60	0,20%	7,38
n20w140.004	318	0,00%	320,50	0,78%	7,38
n20w140.005	275	0,00%	275,00	0,00%	7,41
n20w160.001	347	0,00%	347,00	0,00%	7,39
n20w160.002	250	0,00%	256,00	2,34%	7,41
n20w160.003	331	0,00%	332,70	0,51%	7,44
n20w160.004	287	0,00%	287,00	0,00%	7,38
n20w160.005	342	0,00%	342,00	0,00%	7,39
n20w180.001	353	0,00%	361,20	2,27%	7,39
n20w180.002	347	0,00%	360,50	3,74%	7,43
n20w180.003	315	0,00%	326,90	3,64%	7,41
n20w180.004	288	1,39%	291,10	2,44%	7,40
n20w180.005	257	0,00%	257,00	0,00%	7,37
n20w200.001	259	0,00%	268,30	3,47%	7,40
n20w200.002	242	0,00%	258,90	6,53%	7,37
n20w200.003	305	0,00%	328,00	7,01%	7,37
n20w200.004	333	2,10%	341,50	4,54%	7,36
n20w200.005	277	0,00%	277,20	0,07%	7,37

Fuente: Elaboración Propia.

Tabla A 5: Resultados 20 Ciudades ILS 3.

Instancia	Mínimo	Gap Min ILS	Promedio	Gap prom ILS	Tiempo ejecución
n20w120.001	337	0,00%	343,00	1,75%	3,08
n20w120.002	246	0,00%	251,20	2,07%	3,07
n20w120.003	347	0,00%	351,50	1,28%	3,10
n20w120.004	353	0,00%	362,00	2,49%	3,11
n20w120.005	315	0,00%	328,50	4,11%	3,13
n20w140.001	232	0,86%	240,89	4,52%	3,11
n20w140.002	310	0,97%	318,33	3,56%	3,11
n20w140.003	301	0,00%	301,80	0,27%	3,10
n20w140.004	318	0,00%	318,20	0,06%	3,11
n20w140.005	275	0,00%	278,00	1,08%	3,10
n20w160.001	347	0,00%	347,00	0,00%	3,11
n20w160.002	254	1,57%	257,40	2,87%	3,11
n20w160.003	331	0,00%	332,10	0,33%	3,13
n20w160.004	287	0,00%	287,00	0,00%	3,11
n20w160.005	342	0,00%	342,00	0,00%	3,11
n20w180.001	353	0,00%	367,00	3,81%	3,10
n20w180.002	347	0,00%	365,00	4,93%	3,12
n20w180.003	323	2,48%	330,70	4,75%	3,11
n20w180.004	288	1,39%	292,50	2,91%	3,11
n20w180.005	257	0,00%	257,00	0,00%	3,11
n20w200.001	259	0,00%	259,50	0,19%	3,12
n20w200.002	242	0,00%	263,40	8,12%	3,12
n20w200.003	305	0,00%	322,25	5,35%	3,12
n20w200.004	333	2,10%	360,11	9,47%	3,10
n20w200.005	277	0,00%	284,30	2,57%	3,11

Fuente: Elaboración Propia.

Tabla A 6: Resultados 20 Ciudades ILS 4.

Instancia	Mínimo	Gap Min ILS	Promedio	Gap prom ILS	Tiempo ejecución
n20w120.001	337	0,00%	343,50	1,89%	7,11
n20w120.002	246	0,00%	251,20	2,07%	7,05
n20w120.003	347	0,00%	356,40	2,64%	7,09
n20w120.004	353	0,00%	358,00	1,40%	7,09
n20w120.005	319	1,25%	324,60	2,96%	6,98
n20w140.001	232	0,86%	235,90	2,50%	7,00
n20w140.002	310	0,97%	312,60	1,79%	7,10
n20w140.003	301	0,00%	302,50	0,50%	6,99
n20w140.004	318	0,00%	318,30	0,09%	7,03
n20w140.005	275	0,00%	275,00	0,00%	7,08
n20w160.001	347	0,00%	347,00	0,00%	7,00
n20w160.002	250	0,00%	258,40	3,25%	7,05
n20w160.003	331	0,00%	332,20	0,36%	7,06
n20w160.004	287	0,00%	287,00	0,00%	6,99
n20w160.005	342	0,00%	342,00	0,00%	7,03
n20w180.001	353	0,00%	365,50	3,42%	7,02
n20w180.002	347	0,00%	369,60	6,11%	7,06
n20w180.003	315	0,00%	327,30	3,76%	7,10
n20w180.004	288	1,39%	292,90	3,04%	6,97
n20w180.005	257	0,00%	257,30	0,12%	7,03
n20w200.001	259	0,00%	269,10	3,75%	7,06
n20w200.002	242	0,00%	255,80	5,39%	7,07
n20w200.003	305	0,00%	325,22	6,22%	7,07
n20w200.004	333	2,10%	343,60	5,12%	7,09
n20w200.005	277	0,00%	281,50	1,60%	7,04

Fuente: Elaboración Propia.

Tabla A 7: Resultados 40 ciudades modelo MTZ.

Instancia	BKS	MTZ	Best bound	Gap	Tiempo ejecución
n40w120.001	470	470	470	0,00	24,21
n40w120.002	557	557	557	0,00	53,72
n40w120.003	464	464	464	0,00	328,98
n40w120.004	392	398	340	14,65%	3600,00
n40w120.005	470	470	470	0,00	319,53
n40w140.001	460	460	460	0,00	2140,95
n40w140.002	459	459	459	0,00	41,830
n40w140.003	476	476	476	0,00	1576,29
n40w140.004	458	458	458	0,00	942,90
n40w140.005	438	438	438	0,00	381,90
n40w160.001	470	470	470	0,00	604,24
n40w160.002	451	451	448	0,67%	3600,00
n40w160.003	415	415	414	0,24%	3600,00
n40w160.004	425	425	425	0,00	804,60
n40w160.005	373	373	373	0,00	2019,68
n40w180.001	444	453	364	19,59%	3600,00
n40w180.002	448	448	447	0,22%	3600,00
n40w180.003	398	398	388	2,51%	3600,00
n40w180.004	409	409	409	0,00	604,17
n40w180.005	438	438	361	17,55%	3600,00
n40w200.001	416	416	416	0,00	595,38
n40w200.002	402	402	333	17,10%	3600,00
n40w200.003	408	408	310	23,99%	3600,00
n40w200.004	426	426	291	31,65%	3600,00
n40w200.005	408	408	408	0,00	1477,35

Fuente: Elaboración Propia.

Tabla A 8: Resultados 40 ciudades modelo GG.

Instancia	BKS	GG	Best bound	Gap	Tiempo ejecución
n40w120.001	470	470	470	0,00	33,39
n40w120.002	557	557	557	0,00	108,54
n40w120.003	464	464	464	0,00	336,20
n40w120.004	392	392	340	13,18%	3600,01
n40w120.005	470	470	470	0,00	383,42
n40w140.001	460	460	460	0,00	2304,16
n40w140.002	459	459	459	0,00	50,46
n40w140.003	476	476	476	0,00	569,57
n40w140.004	458	458	458	0,00	739,85
n40w140.005	438	438	438	0,00	594,28
n40w160.001	470	470	470	0,00	288,75
n40w160.002	451	451	439	2,66%	3600,01
n40w160.003	415	414	414	0,00	1477,88
n40w160.004	425	425	425	0,00	137,70
n40w160.005	373	373	373	0,00	1965,05
n40w180.001	444	444	429	3,38%	3600,01
n40w180.002	448	448	447	0,22%	3600,01
n40w180.003	398	398	388	2,51%	3600,01
n40w180.004	409	409	409	0,00	3556,92
n40w180.005	438	438	368	16,00%	3600,01
n40w200.001	416	416	416	0,00	827,71
n40w200.002	402	402	338	15,94%	3600,01
n40w200.003	408	408	312	23,43%	3600,01
n40w200.004	426	426	290	31,90%	3600,01
n40w200.005	408	408	394	3,43%	3600,01

Fuente: Elaboración Propia.

Tabla A 9: Resultados 40 Ciudades ILS 1.

Instancia	Mínimo	Gap Min ILS	Promedio	Gap prom ILS	Tiempo ejecución
n40w120.001	470	0%	580,50	19%	6,13
n40w120.002	----	----	----	----	----
n40w120.003	465	0%	477,90	3%	6,15
n40w120.004	399	2%	407,80	4%	6,15
n40w120.005	470	0%	478,30	2%	6,14
n40w140.001	460	0%	472,70	3%	6,09
n40w140.002	459	0%	491,10	7%	6,10
n40w140.003	477	0%	508,60	6%	6,13
n40w140.004	465	2%	473,50	3%	6,09
n40w140.005	438	0%	526,20	17%	6,10
n40w160.001	470	0%	475,30	1%	6,08
n40w160.002	453	0%	463,20	3%	6,08
n40w160.003	415	0%	445,10	7%	6,10
n40w160.004	425	0%	437,50	3%	6,08
n40w160.005	373	0%	398,00	6%	6,07
n40w180.001	453	2%	484,80	8%	6,08
n40w180.002	448	0%	466,10	4%	6,09
n40w180.003	398	0%	407,40	2%	6,09
n40w180.004	421	3%	614,50	33%	6,11
n40w180.005	438	0%	440,10	0%	6,06
n40w200.001	416	0%	431,60	4%	6,11
n40w200.002	402	0%	433,40	7%	6,08
n40w200.003	416	2%	433,20	6%	6,09
n40w200.004	426	0%	426,90	0%	6,07
n40w200.005	408	0%	421,90	3%	6,07

Fuente: Elaboración Propia.

Tabla A 10: Resultados 40 Ciudades ILS 2.

Instancia	Mínimo	Gap Min ILS	Promedio	Gap prom ILS	Tiempo ejecución
n40w120.001	470	0%	608,20	23%	13,98
n40w120.002	----	----	----	----	----
n40w120.003	----	----	----	----	----
n40w120.004	399	2%	410,70	5%	13,93
n40w120.005	470	0%	475,50	1%	13,97
n40w140.001	460	0%	487,70	6%	13,93
n40w140.002	459	0%	525,70	13%	13,92
n40w140.003	476	0%	498,30	4%	13,97
n40w140.004	458	0%	471,40	3%	13,92
n40w140.005	438	0%	484,00	10%	13,94
n40w160.001	470	0%	475,20	1%	13,94
n40w160.002	455	1%	464,30	3%	13,93
n40w160.003	418	1%	439,20	6%	13,95
n40w160.004	426	0%	446,10	5%	13,93
n40w160.005	373	0%	395,60	6%	13,93
n40w180.001	453	2%	474,00	6%	13,92
n40w180.002	448	0%	469,40	5%	13,91
n40w180.003	----	----	----	----	----
n40w180.004	434	6%	579,90	29%	13,91
n40w180.005	438	0%	439,00	0%	13,87
n40w200.001	416	0%	431,50	4%	13,97
n40w200.002	402	0%	416,70	4%	13,88
n40w200.003	408	0%	436,20	6%	13,90
n40w200.004	426	0%	436,30	2%	18,84
n40w200.005	408	0%	433,40	6%	14,36

Fuente: Elaboración Propia.

Tabla A 11: Resultados 40 Ciudades ILS 3.

Instancia	Mínimo	Gap Min ILS	Promedio	Gap prom ILS	Tiempo ejecución
n40w120.001	470	0%	502,00	6%	5,77
n40w120.002	565	1%	565,56	2%	5,77
n40w120.003	476	3%	496,60	7%	5,76
n40w120.004	399	2%	404,00	3%	5,78
n40w120.005	470	0%	476,33	1%	5,79
n40w140.001	460	0%	474,60	3%	5,78
n40w140.002	459	0%	459,00	0%	5,78
n40w140.003	477	0%	479,57	1%	5,79
n40w140.004	467	2%	473,00	3%	5,78
n40w140.005	438	0%	475,67	8%	5,77
n40w160.001	470	0%	479,00	2%	5,78
n40w160.002	453	0%	464,30	3%	5,78
n40w160.003	415	0%	433,22	4%	5,79
n40w160.004	425	0%	440,70	4%	5,80
n40w160.005	373	0%	403,11	7%	5,80
n40w180.001	453	2%	460,00	3%	5,76
n40w180.002	448	0%	460,29	3%	5,77
n40w180.003	398	0%	413,70	4%	5,78
n40w180.004	435	6%	458,60	11%	5,77
n40w180.005	438	0%	441,90	1%	5,75
n40w200.001	416	0%	422,89	2%	5,79
n40w200.002	402	0%	412,33	3%	5,77
n40w200.003	416	2%	428,29	5%	5,78
n40w200.004	426	0%	426,20	0%	5,76
n40w200.005	408	0%	428,50	5%	5,76

Fuente: Elaboración Propia.

Tabla A 12: Resultados 40 Ciudades ILS 4.

Instancia	Mínimo	Gap Min ILS	Promedio	Gap prom ILS	Tiempo ejecución
n40w120.001	509	8%	519,67	10%	13,35
n40w120.002	565	1%	565,00	1%	13,33
n40w120.003	476	3%	489,90	5%	13,29
n40w120.004	399	2%	404,63	3%	13,35
n40w120.005	470	0%	476,30	1%	13,32
n40w140.001	460	0%	482,67	5%	13,30
n40w140.002	459	0%	480,43	4%	13,33
n40w140.003	476	0%	483,88	2%	13,38
n40w140.004	466	2%	473,20	3%	13,33
n40w140.005	438	0%	460,86	5%	13,32
n40w160.001	470	0%	472,00	0%	13,33
n40w160.002	455	1%	471,60	4%	13,33
n40w160.003	427	3%	447,50	7%	13,36
n40w160.004	425	0%	445,80	5%	13,29
n40w160.005	373	0%	395,60	6%	13,37
n40w180.001	453	2%	461,38	4%	13,29
n40w180.002	448	0%	465,56	4%	13,35
n40w180.003	411	3%	412,40	3%	13,31
n40w180.004	442	7%	465,50	12%	13,40
n40w180.005	438	0%	439,00	0%	13,31
n40w200.001	416	0%	424,57	2%	13,40
n40w200.002	402	0%	416,50	3%	13,28
n40w200.003	417	2%	429,86	5%	13,31
n40w200.004	426	0%	430,00	1%	13,33
n40w200.005	408	0%	427,78	5%	13,31

Fuente: Elaboración Propia.

Tabla A 13: Resultados 60 ciudades modelo MTZ.

Instancia	BKS	MTZ	Best bound	Gap	Tiempo ejecución
n60w120.001	536	536	536	0,00	265,99
n60w120.002	606	607	606	0,16%	3600,01
n60w120.003	541	541	541	0,00	2499,41
n60w120.004	607	607	607	0,00	455,28
n60w120.005	579	579	579	0,00	867,32
n60w140.001	596	596	596	0,00	170,32
n60w140.002	647	647	647	0,00	83,52
n60w140.003	625	625	625	0,00	1577,46
n60w140.004	578	inf	578	None	3600,01
n60w140.005	554	inf	554	None	3600,01
n60w160.001	672	672	672	0,00	1619,81
n60w160.002	665	665	665	0,00	368,03
n60w160.003	569	569	569	0,00	76,44
n60w160.004	573	573	472	17,70%	3600,01
n60w160.005	619	619	619	0,00	1363,80
n60w180.001	556	556	556	0,00	794,66
n60w180.002	561	561	403	28,11%	3600,01
n60w180.003	606	606	606	0,00	1046,95
n60w180.004	629	629	629	0,00	885,00
n60w180.005	528	528	464	12,12%	3600,01
n60w200.001	526	530	394	25,71%	3600,01
n60w200.002	572	572	572	0,00	2343,85
n60w200.003	560	562	452	19,61%	3600,01
n60w200.004	575	575	575	0,00	1426,70
n60w200.005	618	618	618	0,00	316,81

Fuente: Elaboración Propia.

Tabla A 14: Resultados 60 ciudades modelo GG.

Instancia	BKS	GG	Best bound	Gap	Tiempo ejecución
n60w120.001	536	536	536	0,00	454,71
n60w120.002	606	606	606	0,00	405,10
n60w120.003	541	541	487	9,98%	3600,01
n60w120.004	607	607	607	0,00	52,67
n60w120.005	579	579	579	0,00	472,14
n60w140.001	596	596	596	0,00	370,33
n60w140.002	647	647	647	0,00	54,26
n60w140.003	625	625	625	0,00	175,92
n60w140.004	578	inf	578	None	3600,00
n60w140.005	554	554.0	554	0,00	711,29
n60w160.001	672	672	672	0,00	184,97
n60w160.002	665	665	665	0,00	53,93
n60w160.003	569	569	569	0,00	75,27
n60w160.004	573	573	550	4,01%	3600,01
n60w160.005	619	619	619	0,00	1292,12
n60w180.001	556	556	556	0,00	476,65
n60w180.002	561	561	405	27,89%	3600,01
n60w180.003	606	606	606	0,00	353,89
n60w180.004	629	629	629	0,00	295,11
n60w180.005	528	528	528	0,00	1635,11
n60w200.001	526	526	416	20,85%	3600,01
n60w200.002	572	572	572	0,00	2629,44
n60w200.003	560	560	437	21,98%	3600,01
n60w200.004	575	575	575	0,00	883,53
n60w200.005	618	618	618	0,00	273,56

Fuente: Elaboración Propia.

Tabla A 15: Resultados 60 Ciudades ILS 1.

Instancia	Mínimo	Gap Min ILS	Promedio	Gap prom ILS	Tiempo ejecución
n60w120.001	536	0%	718,70	25%	8,94
n60w120.002	606	0%	642,80	6%	8,96
n60w120.003	541	0%	548,00	1%	8,88
n60w120.004	607	0%	1118,20	46%	8,96
n60w120.005	----	----	----	----	----
n60w140.001	596	0%	710,90	16%	8,94
n60w140.002	647	0%	678,70	5%	8,93
n60w140.003	625	0%	640,40	2%	8,91
n60w140.004	----	----	----	----	----
n60w140.005	554	0%	1381,40	60%	8,95
n60w160.001	672	0%	850,50	21%	8,83
n60w160.002	665	0%	668,40	1%	8,84
n60w160.003	569	0%	978,10	42%	8,84
n60w160.004	573	0%	645,00	11%	8,83
n60w160.005	619	0%	666,80	7%	8,84
n60w180.001	556	0%	590,00	6%	8,81
n60w180.002	561	0%	780,10	28%	8,95
n60w180.003	606	0%	616,90	2%	8,80
n60w180.004	629	0%	635,20	1%	8,79
n60w180.005	528	0%	561,10	6%	8,83
n60w200.001	536	2%	610,30	14%	8,82
n60w200.002	572	0%	610,60	6%	8,79
n60w200.003	570	2%	689,00	19%	8,86
n60w200.004	575	0%	791,30	27%	8,82
n60w200.005	618	0%	669,40	8%	8,87

Fuente: Elaboración Propia.

Tabla A 16: Resultados 60 Ciudades ILS 2.

Instancia	Mínimo	Gap Min ILS	Promedio	Gap prom ILS	Tiempo ejecución
n60w120.001	536	0%	685,40	22%	20,46
n60w120.002	609	0%	802,70	25%	20,47
n60w120.003	543	0%	556,40	3%	20,39
n60w120.004	607	0%	719,00	16%	20,45
n60w120.005	611	5%	1140,80	49%	20,58
n60w140.001	596	0%	606,00	2%	20,59
n60w140.002	647	0%	666,80	3%	20,41
n60w140.003	625	0%	637,90	2%	20,40
n60w140.004	578	0%	1140,60	49%	20,47
n60w140.005	554	0%	1019,50	46%	20,39
n60w160.001	672	0%	808,20	17%	20,31
n60w160.002	665	0%	679,20	2%	20,27
n60w160.003	569	0%	576,70	1%	20,26
n60w160.004	586	2%	727,90	21%	20,32
n60w160.005	619	0%	662,30	7%	20,29
n60w180.001	556	0%	591,60	6%	20,26
n60w180.002	561	0%	697,20	20%	20,28
n60w180.003	606	0%	606,00	0%	20,21
n60w180.004	629	0%	639,50	2%	20,24
n60w180.005	528	0%	549,10	4%	20,28
n60w200.001	561	6%	604,40	13%	20,35
n60w200.002	572	0%	595,60	4%	20,35
n60w200.003	564	1%	619,20	10%	20,44
n60w200.004	575	0%	735,50	22%	20,33
n60w200.005	618	0%	632,00	2%	20,32

Fuente: Elaboración Propia.

Tabla A 17: Resultados 60 Ciudades ILS 3.

Instancia	Mínimo	Gap Min ILS	Promedio	Gap prom ILS	Tiempo ejecución
n60w120.001	536	0%	545,78	2%	8,43
n60w120.002	607	0%	608,78	0%	8,44
n60w120.003	545	1%	548,70	1%	8,41
n60w120.004	607	0%	607,00	0%	8,45
n60w120.005	0	----	0,00	----	0,00
n60w140.001	596	0%	612,86	3%	8,46
n60w140.002	647	0%	655,17	1%	8,46
n60w140.003	625	0%	641,50	3%	8,42
n60w140.004	0	----	0,00	----	0,00
n60w140.005	554	0%	557,00	1%	8,54
n60w160.001	673	0%	698,00	4%	8,45
n60w160.002	665	0%	668,40	1%	8,40
n60w160.003	569	0%	570,29	0%	8,46
n60w160.004	573	0%	606,00	5%	8,45
n60w160.005	619	0%	656,33	6%	8,44
n60w180.001	556	0%	569,67	2%	8,42
n60w180.002	561	0%	570,38	2%	8,44
n60w180.003	606	0%	607,56	0%	8,42
n60w180.004	629	0%	638,90	2%	8,46
n60w180.005	528	0%	544,67	3%	8,48
n60w200.001	536	2%	553,33	5%	8,45
n60w200.002	572	0%	598,33	4%	8,43
n60w200.003	565	1%	573,67	2%	8,51
n60w200.004	575	0%	614,83	6%	8,43
n60w200.005	618	0%	620,57	0%	8,45

Fuente: Elaboración Propia.

Tabla A 18: Resultados 60 Ciudades ILS 4.

Instancia	Mínimo	Gap Min ILS	Promedio	Gap prom ILS	Tiempo ejecución
n60w120.001	536	0%	548,57	2%	19,52
n60w120.002	609	0%	609,00	0%	19,54
n60w120.003	546	1%	548,89	1%	19,45
n60w120.004	607	0%	607,00	0%	19,59
n60w120.005	611	5%	625,00	7%	19,57
n60w140.001	596	0%	605,56	2%	19,59
n60w140.002	647	0%	656,33	1%	19,55
n60w140.003	625	0%	639,20	2%	19,44
n60w140.004	578	0%	578,00	0%	19,59
n60w140.005	554	0%	582,00	5%	19,57
n60w160.001	672	0%	689,17	2%	19,62
n60w160.002	665	0%	676,20	2%	19,52
n60w160.003	569	0%	570,00	0%	19,56
n60w160.004	587	2%	588,83	3%	19,59
n60w160.005	619	0%	655,00	5%	19,65
n60w180.001	556	0%	563,00	1%	19,55
n60w180.002	561	0%	563,57	0%	19,57
n60w180.003	606	0%	606,00	0%	19,54
n60w180.004	629	0%	631,70	0%	19,56
n60w180.005	528	0%	543,33	3%	19,65
n60w200.001	561	6%	563,67	7%	19,64
n60w200.002	572	0%	600,30	5%	19,53
n60w200.003	561	0%	575,00	3%	19,66
n60w200.004	575	0%	582,63	1%	19,54
n60w200.005	618	0%	624,00	1%	19,73

Fuente: Elaboración Propia.

Tabla A 19: Resultados 80 ciudades modelo MTZ.

Instancia	BKS	MTZ	Best bound	Gap	Tiempo ejecución
n80w100.001	676	inf	676	None	3600,006
n80w100.002	721	inf	721	None	3600,004
n80w100.003	729	inf	729	None	3600,005
n80w100.004	759	inf	710	None	3600,004
n80w100.005	671	inf	658	None	3600,004
n80w120.001	675	675	675	0,00	2492,475
n80w120.002	748	748	748	0,00	120,848
n80w120.003	677	677	677	0,00	1269,341
n80w120.004	644	644	644	0,00	3470,115
n80w120.005	743	743	743	0,00	2008,324
n80w140.001	689	689	689	0,00	304,6614
n80w140.002	651	736	651	11,5%	3600,006
n80w140.003	673	677	673	0,59%	3600,005
n80w140.004	612	inf	429	None	4877,285
n80w140.005	739	inf	670	None	3791,58
n80w160.001	624	inf	451	None	3626,867
n80w160.002	654	inf	510	None	3883,426
n80w160.003	690	inf	483	None	4055,874
n80w160.004	655	inf	452	None	3957,28
n80w160.005	645	inf	457	None	4014,587
n80w180.001	678	inf	514	None	3600,005
n80w180.002	633	inf	488	None	3600,004
n80w180.003	680	inf	671	None	3600,004
n80w180.004	659	659	659	0,00	3048,924
n80w180.005	632	inf	449	None	3600,005
n80w200.001	626	inf	453	None	3600,004
n80w200.002	638	681	475	30,2%	3600,007
n80w200.003	679	inf	457	None	3600,004
n80w200.004	667	inf	483	None	3600,004
n80w200.005	621	inf	424	None	3600,005

Fuente: Elaboración Propia.

Tabla A 20: Resultado 80 ciudades modelo GG.

Instancia	BKS	GG	Best bound	Gap	Tiempo ejecución
n80w100.001	676	676	676	0,00	1048,01
n80w100.002	721	721	721	0,00	3156,03
n80w100.003	729	729	729	0,00	685,36
n80w100.004	759	759	710	6,46%	3600,02
n80w100.005	671	inf	651	None	3600,01
n80w120.001	675	675	675	0,00	599,85
n80w120.002	748	748	748	0,00	115,92
n80w120.003	677	677	677	0,00	522,55
n80w120.004	644	644	644	0,00	770,92
n80w120.005	743	743	743	0,00	1840,12
n80w140.001	689	689	689	0,00	245,66
n80w140.002	651	651	651	0,00	649,63
n80w140.003	673	673	667	0,89%	3783,02
n80w140.004	612	inf	500	None	3600,01
n80w140.005	739	739	725	1,89%	3600,04
n80w160.001	624	624	507	18,80%	3600,04
n80w160.002	654	665	652	1,95%	3600,01
n80w160.003	690	699	687	1,72%	3600,01
n80w160.004	655	655	571	12,82%	3600,03
n80w160.005	645	646	478	26,00%	3691,80
n80w180.001	678	inf	532	None	3600,01
n80w180.002	633	633	494	22,02%	3600,01
n80w180.003	680	680	671	1,32%	3600,01
n80w180.004	659	659	654	0,76%	3600,01
n80w180.005	632	635	459	27,64%	3600,01
n80w200.001	626	626	451	28,01%	3600,01
n80w200.002	638	638	465	27,04%	3600,01
n80w200.003	679	679	679	0,00%	2612,01
n80w200.004	667	672	482	28,29%	3600,01
n80w200.005	621	621	433	30,35%	3600,01

Fuente: Elaboración Propia.

Tabla A 21: Resultados 80 Ciudades ILS 1.

Instancia	Mínimo	Gap Min ILS	Promedio	Gap prom ILS	Tiempo ejecución
n80w100.001	----	----	----	----	----
n80w100.002	721	0%	1897,90	62%	11,79
n80w100.003	729	0%	1779,50	59%	11,79
n80w100.004	----	----	----	----	----
n80w100.005	671	0%	937,80	28%	11,74
n80w120.001	675	0%	842,70	20%	11,69
n80w120.002	748	0%	1248,50	40%	11,77
n80w120.003	680	0%	829,80	18%	11,74
n80w120.004	645	0%	1211,70	47%	11,69
n80w120.005	743	0%	1028,60	28%	11,75
n80w140.001	689	0%	829,40	17%	11,69
n80w140.002	651	0%	968,50	33%	11,75
n80w140.003	696	3%	1288,70	48%	11,76
n80w140.004	613	0%	1217,30	50%	11,76
n80w140.005	741	0%	758,50	3%	11,72
n80w160.001	----	----	----	----	----
n80w160.002	----	----	----	----	----
n80w160.003	704	2%	1351,80	49%	11,75
n80w160.004	656	0%	939,30	30%	11,77
n80w160.005	647	0%	678,20	5%	11,68
n80w180.001	----	----	----	----	----
n80w180.002	649	2%	736,70	14%	11,71
n80w180.003	----	----	----	----	----
n80w180.004	659	0%	681,20	3%	11,71
n80w180.005	632	0%	664,70	5%	11,70
n80w200.001	643	3%	1418,00	56%	11,79
n80w200.002	638	0%	701,10	9%	11,71
n80w200.003	679	0%	739,60	8%	11,73
n80w200.004	----	----	----	----	----
n80w200.005	621	0%	630,20	1%	11,68

Fuente: Elaboración Propia.

Tabla A 22: Resultados 80 Ciudades ILS 2.

Instancia	Mínimo	Gap Min ILS	Promedio	Gap prom ILS	Tiempo ejecución
n80w100.001	----	----	----	----	----
n80w100.002	736	2,04%	2118,40	65,96%	27,28
n80w100.003	----	----	----	----	----
n80w100.004	----	----	----	----	----
n80w100.005	----	----	----	----	----
n80w120.001	675	0,00%	780,20	13,48%	27,07
n80w120.002	748	0,00%	1012,90	26,15%	27,16
n80w120.003	702	3,56%	917,00	26,17%	27,17
n80w120.004	645	0,16%	2093,30	69,24%	27,12
n80w120.005	743	0,00%	1645,10	54,84%	27,15
n80w140.001	689	0,00%	693,40	0,63%	27,04
n80w140.002	651	0,00%	961,90	32,32%	27,17
n80w140.003	----	----	----	----	----
n80w140.004	613	0,16%	649,10	5,72%	27,15
n80w140.005	739	0,00%	1004,10	26,40%	27,17
n80w160.001	624	0,00%	839,60	25,68%	27,13
n80w160.002	677	3,40%	1064,90	38,59%	27,25
n80w160.003	704	1,99%	964,50	28,46%	27,14
n80w160.004	668	1,95%	984,20	33,45%	27,17
n80w160.005	647	0,31%	785,90	17,93%	27,11
n80w180.001	697	2,73%	1117,20	39,31%	27,28
n80w180.002	640	1,09%	692,20	8,55%	27,21
n80w180.003	682	0,29%	858,70	20,81%	27,32
n80w180.004	659	0,00%	769,50	14,36%	27,07
n80w180.005	635	0,47%	685,90	7,86%	27,11
n80w200.001	627	0,16%	909,50	31,17%	27,25
n80w200.002	668	4,49%	898,50	28,99%	27,18
n80w200.003	683	0,59%	864,10	21,42%	27,19
n80w200.004	670	0,45%	953,90	30,08%	27,20
n80w200.005	627	0,96%	665,90	6,74%	27,07

Fuente: Elaboración Propia.

Tabla A 23: Resultados 80 Ciudades ILS 3.

Instancia	Mínimo	Gap Min ILS	Promedio	Gap prom ILS	Tiempo ejecución
n80w100.001	----	----	----	----	----
n80w100.002	721	0%	721,00	0%	11,31
n80w100.003	729	0%	729,00	0%	11,31
n80w100.004	----	----	----	----	----
n80w100.005	671	0%	679,00	1%	11,31
n80w120.001	675	0%	675,00	0%	11,29
n80w120.002	748	0%	748,00	0%	11,36
n80w120.003	702	4%	725,00	7%	11,29
n80w120.004	645	0%	645,00	0%	11,29
n80w120.005	743	0%	762,67	3%	11,34
n80w140.001	689	0%	694,88	1%	11,31
n80w140.002	651	0%	665,50	2%	11,32
n80w140.003	696	3%	696,00	3%	11,34
n80w140.004	612	0%	617,50	1%	11,40
n80w140.005	741	0%	759,70	3%	11,33
n80w160.001	666	6%	666,00	6%	11,37
n80w160.002	----	----	----	----	----
n80w160.003	704	2%	704,00	2%	11,37
n80w160.004	668	2%	668,00	2%	11,39
n80w160.005	647	0%	647,00	0%	11,30
n80w180.001	748	9%	748,00	9%	11,37
n80w180.002	658	4%	664,00	5%	11,48
n80w180.003	----	----	----	----	----
n80w180.004	659	0%	676,89	3%	11,49
n80w180.005	635	0%	649,89	3%	11,45
n80w200.001	627	0%	627,00	0%	11,52
n80w200.002	663	4%	678,44	6%	11,64
n80w200.003	679	0%	686,83	1%	11,54
n80w200.004	701	5%	716,33	7%	11,57
n80w200.005	621	0%	636,10	2%	11,52

Fuente: Elaboración Propia.

Tabla A 24: Resultados 80 Ciudades ILS 4.

Instancia	Mínimo	Gap Min ILS	Promedio	Gap prom ILS	Tiempo ejecución
n80w100.001	----	----	----	----	----
n80w100.002	736	2%	736,00	2%	25,91
n80w100.003	----	----	----	----	----
n80w100.004	----	----	----	----	----
n80w100.005	----	----	----	----	----
n80w120.001	675	0%	679,50	1%	25,86
n80w120.002	748	0%	748,00	0%	26,00
n80w120.003	702	4%	702,00	4%	25,94
n80w120.004	645	0%	648,40	1%	25,96
n80w120.005	743	0%	757,50	2%	25,87
n80w140.001	689	0%	691,70	0%	25,83
n80w140.002	651	0%	670,60	3%	25,94
n80w140.003	----	----	----	----	----
n80w140.004	613	0%	630,00	3%	25,97
n80w140.005	739	0%	767,00	4%	25,96
n80w160.001	624	0%	645,00	3%	25,98
n80w160.002	677	3%	685,33	5%	25,99
n80w160.003	704	2%	706,86	2%	25,96
n80w160.004	668	2%	676,33	3%	25,98
n80w160.005	647	0%	647,00	0%	25,91
n80w180.001	698	3%	698,00	3%	26,13
n80w180.002	645	2%	656,40	4%	25,96
n80w180.003	682	0%	682,00	0%	26,15
n80w180.004	659	0%	682,38	3%	25,91
n80w180.005	635	0%	657,86	4%	25,94
n80w200.001	627	0%	641,00	2%	26,05
n80w200.002	670	5%	699,50	9%	25,97
n80w200.003	683	1%	684,00	1%	26,09
n80w200.004	670	0%	682,20	2%	26,40
n80w200.005	635	2%	643,89	4%	26,17

Fuente: Elaboración Propia.

Tabla A 25: Resultados 100 ciudades modelo MTZ.

Instancia	BKS	MTZ	Best bound	Gap	Tiempo ejecución
n100w80.001	797	797	797	0,00	1806,18
n100w80.002	790	inf	790	None	3600,01
n100w80.003	829	inf	828	None	3600,01
n100w80.004	854	inf	854	None	3600,01
n100w80.005	759	759	759	0,00	2370,15
n100w100.001	787	inf	787	None	3600,00
n100w100.002	780	inf	780	None	3600,01
n100w100.003	819	829.0	818	1,33%	3600,01
n100w100.004	844	inf	844	None	3600,00
n100w100.005	749	inf	749	None	3600,02
n100w120.001	882	882	882	0,00	54,07
n100w120.002	893	893	893	0,00	2472,73
n100w120.003	909	909	909	0,00	298,92
n100w120.004	923	923	923	0,00	2968,19
n100w120.005	870	870	870	0,00	198,33
n100w140.001	1008	inf	1008	None	3600,01
n100w140.002	1021	1021	1021	0,00	606,67
n100w140.003	844	inf	844	None	3600,01
n100w140.004	854	854	701	17,90%	3600,02
n100w140.005	805	inf	510	None	3600,01
n100w160.001	868	inf	852	None	3600,01
n100w160.002	791	inf	528	None	3600,01
n100w160.003	935	935	935	0,00	2581,83
n100w160.004	814	inf	563	None	3600,01
n100w160.005	917	inf	912	None	3600,01

Fuente: Elaboración Propia.

Tabla A 26: Resultados 100 ciudades modelo GG.

Instancia	BKS	GG	Best bound	Gap	Tiempo ejecución
n100w80.001	797	797	797	0,00	1678,97
n100w80.002	790	inf	790	None	3600,00
n100w80.003	829	829	829	0,00	1682,35
n100w80.004	854	854	854	0,00	1324,64
n100w80.005	759	759	759	0,00	1089,00
n100w100.001	787	797	787	1,25%	3600,01
n100w100.002	780	inf	780	None	3600,00
n100w100.003	819	819	818	0,12%	3600,01
n100w100.004	844	844	844	0,00	1374,32
n100w100.005	749	749	749	0,00	441,78
n100w120.001	882	882	882	0,00	471,15
n100w120.002	893	893	893	0,00	1496,70
n100w120.003	909	909	909	0,00	176,66
n100w120.004	923	926	923	0,32%	3600,03
n100w120.005	870	870	870	0,00	1344,49
n100w140.001	1008	1008	1008	0,00	617,53
n100w140.002	1021	1021	1021	0,00	365,84
n100w140.003	844	844	844	0,00	445,34
n100w140.004	854	854	540	36,77	3600,02
n100w140.005	805	805	805	0,00	2838,92
n100w160.001	868	868	852	1,84%	3600,02
n100w160.002	791	791	521	34,17%	3600,08
n100w160.003	935	935	486	48,00%	3600,02
n100w160.004	814	inf	559	None	3600,01
n100w160.005	917	917	917	0,00	577,75

Fuente: Elaboración Propia.

Tabla A 27: Resultados 100 Ciudades ILS 1.

Instancia	Mínimo	Gap Min ILS	Promedio	Gap prom ILS	Tiempo ejecución
n100w80.001	----	----	----	----	----
n100w80.002	----	----	----	----	----
n100w80.003	----	----	----	----	----
n100w80.004	----	----	----	----	----
n100w80.005	----	----	----	----	----
n100w100.001	----	----	----	----	----
n100w100.002	----	----	----	----	----
n100w100.003	----	----	----	----	----
n100w100.004	----	----	----	----	----
n100w100.005	----	----	----	----	----
n100w120.001	882	0%	1779,70	50%	15,20
n100w120.002	893	0%	893,30	0%	15,16
n100w120.003	909	0%	912,50	0%	15,45
n100w120.004	923	0%	1210,90	24%	15,31
n100w120.005	870	0%	870,00	0%	15,24
n100w140.001	1008	0%	1018,80	1%	15,11
n100w140.002	1021	0%	1185,90	14%	15,63
n100w140.003	844	0%	844,00	0%	15,36
n100w140.004	854	0%	1181,90	28%	15,89
n100w140.005	805	0%	1270,00	37%	15,54
n100w160.001	872	0%	1013,80	14%	15,81
n100w160.002	791	0%	879,60	10%	15,47
n100w160.003	935	0%	935,00	0%	15,00
n100w160.004	814	0%	943,00	14%	15,59
n100w160.005	917	0%	3529,70	74%	15,94

Fuente: Elaboración Propia.

Tabla A 28: Resultados 100 Ciudades ILS 2.

Instancia	Mínimo	Gap Min ILS	Promedio	Gap prom ILS	Tiempo ejecución
n100w80.001	----	----	----	----	----
n100w80.002	----	----	----	----	----
n100w80.003	----	----	----	----	----
n100w80.004	----	----	----	----	----
n100w80.005	----	----	----	----	----
n100w100.001	----	----	----	----	----
n100w100.002	----	----	----	----	----
n100w100.003	----	----	----	----	----
n100w100.004	----	----	----	----	----
n100w100.005	----	----	----	----	----
n100w120.001	882	0%	3046,10	71%	36,46
n100w120.002	893	0%	893,00	0%	36,36
n100w120.003	909	0%	1116,70	19%	36,78
n100w120.004	923	0%	1429,50	35%	36,34
n100w120.005	870	0%	906,60	4%	35,80
n100w140.001	1008	0%	1015,50	1%	35,35
n100w140.002	1021	0%	1228,60	17%	36,03
n100w140.003	844	0%	1289,10	35%	35,52
n100w140.004	854	0%	2060,50	59%	36,87
n100w140.005	805	0%	1454,60	45%	36,66
n100w160.001	872	0%	1011,90	14%	36,52
n100w160.002	791	0%	803,20	2%	36,24
n100w160.003	935	0%	955,60	2%	35,73
n100w160.004	814	0%	831,00	2%	37,29
n100w160.005	917	0%	2120,20	57%	37,37

Fuente: Elaboración Propia.

Tabla A 29: Resultados 100 Ciudades ILS 3.

Instancia	Mínimo	Gap Min ILS	Promedio	Gap prom ILS	Tiempo ejecución
n100w80.001	----	----	----	----	----
n100w80.002	----	----	----	----	----
n100w80.003	----	----	----	----	----
n100w80.004	----	----	----	----	----
n100w80.005	----	----	----	----	----
n100w100.001	----	----	----	----	----
n100w100.002	----	----	----	----	----
n100w100.003	----	----	----	----	----
n100w100.004	----	----	----	----	----
n100w100.005	767	2%	767,00	2%	15,84
n100w120.001	882	0%	882,00	0%	15,45
n100w120.002	893	0%	893,00	0%	15,47
n100w120.003	909	0%	909,00	0%	15,73
n100w120.004	923	0%	952,00	3%	15,77
n100w120.005	870	0%	870,00	0%	15,62
n100w140.001	1008	0%	1008,00	0%	15,48
n100w140.002	1021	0%	1021,00	0%	16,15
n100w140.003	844	0%	844,00	0%	16,07
n100w140.004	854	0%	854,00	0%	16,63
n100w140.005	805	0%	807,40	0%	16,43
n100w160.001	872	0%	873,43	1%	16,10
n100w160.002	791	0%	802,71	1%	16,11
n100w160.003	935	0%	935,00	0%	15,31
n100w160.004	814	0%	831,00	2%	16,32
n100w160.005	917	0%	917,00	0%	16,37

Fuente: Elaboración Propia.

Tabla A 30: Resultados 100 Ciudades ILS 4.

Instancia	Mínimo	Gap Min ILS	Promedio	Gap prom ILS	Tiempo ejecución
n100w80.001	----	----	----	----	----
n100w80.002	----	----	----	----	----
n100w80.003	----	----	----	----	----
n100w80.004	----	----	----	----	----
n100w80.005	----	----	----	----	----
n100w100.001	----	----	----	----	----
n100w100.002	----	----	----	----	----
n100w100.003	----	----	----	----	----
n100w100.004	----	----	----	----	----
n100w100.005	----	----	----	----	----
n100w120.001	882	0,00%	882,00	0,00%	32,47
n100w120.002	893	0,00%	893,00	0,00%	32,37
n100w120.003	909	0,00%	909,00	0,00%	32,51
n100w120.004	923	0,00%	944,67	2,29%	32,30
n100w120.005	870	0,00%	870,00	0,00%	32,12
n100w140.001	1008	0,00%	1008,00	0,00%	32,26
n100w140.002	1021	0,00%	1021,00	0,00%	32,51
n100w140.003	844	0,00%	844,00	0,00%	32,38
n100w140.004	854	0,00%	854,00	0,00%	32,64
n100w140.005	805	0,00%	807,25	0,28%	32,49
n100w160.001	872	0,46%	882,00	1,59%	32,75
n100w160.002	791	0,00%	803,20	1,52%	32,45
n100w160.003	935	0,00%	935,00	0,00%	32,17
n100w160.004	814	0,00%	828,83	1,79%	32,84
n100w160.005	917	0,00%	917,00	0,00%	33,31

Fuente: Elaboración Propia.

UNIVERSIDAD DE CONCEPCIÓN – FACULTAD DE INGENIERÍA

RESUMEN MEMORIA DE TÍTULO

Departamento	Departamento de Ingeniería Civil Industrial
Carrera	Ingeniería Civil Industrial
Nombre del memorista	José Tomás Garcés Guzmán
Título de la memoria	Un algoritmo para resolver el problema del vendedor viajero con ventanas de tiempo para servicios postales
Fecha presentación oral	13/10/2022
Profesor(es) Guía	Carlos Contreras Bolton
Profesor(es) Revisor(es)	Rosa Medina
Concepto	
Calificación	
Resumen	
	<p>Esta memoria de título tiene como objetivo estudiar el problema del vendedor viajero con ventanas de tiempo aplicado al caso de los servicios postales, y elaborar una metaheurística para resolver el problema. Este problema es una variante del problema clásico del vendedor viajero que consiste en minimizar el costo de una ruta hamiltoniana con una restricción adicional, que consiste en la existencia de ventanas de tiempo para cada uno de los destinos en la ruta, generándose un aumento en la dificultad en la resolución. Esta dificultad se debe</p>

a que al llegar a un destino de forma prematura, se genera un aumento en el costo de la ruta y además, el arribo posterior al cierre de la ventana de tiempo produce una ruta no factible. En los servicios postales por lo general existen dos tipos de entregas, la paquetería y las cartas, para ambos casos la industria busca optimizar las rutas de entrega para reducir los costos operativos y de mano de obra. Además, existen muchos casos en los que el cliente tiene un tiempo acotado para recibir su entrega. Para resolver este problema se propone un algoritmo basado en una búsqueda local iterada. Este algoritmo trabaja con penalizaciones, perturbaciones y búsquedas locales. Los resultados de la metaheurística muestran buenos rendimientos hasta las instancias de 60 nodos.