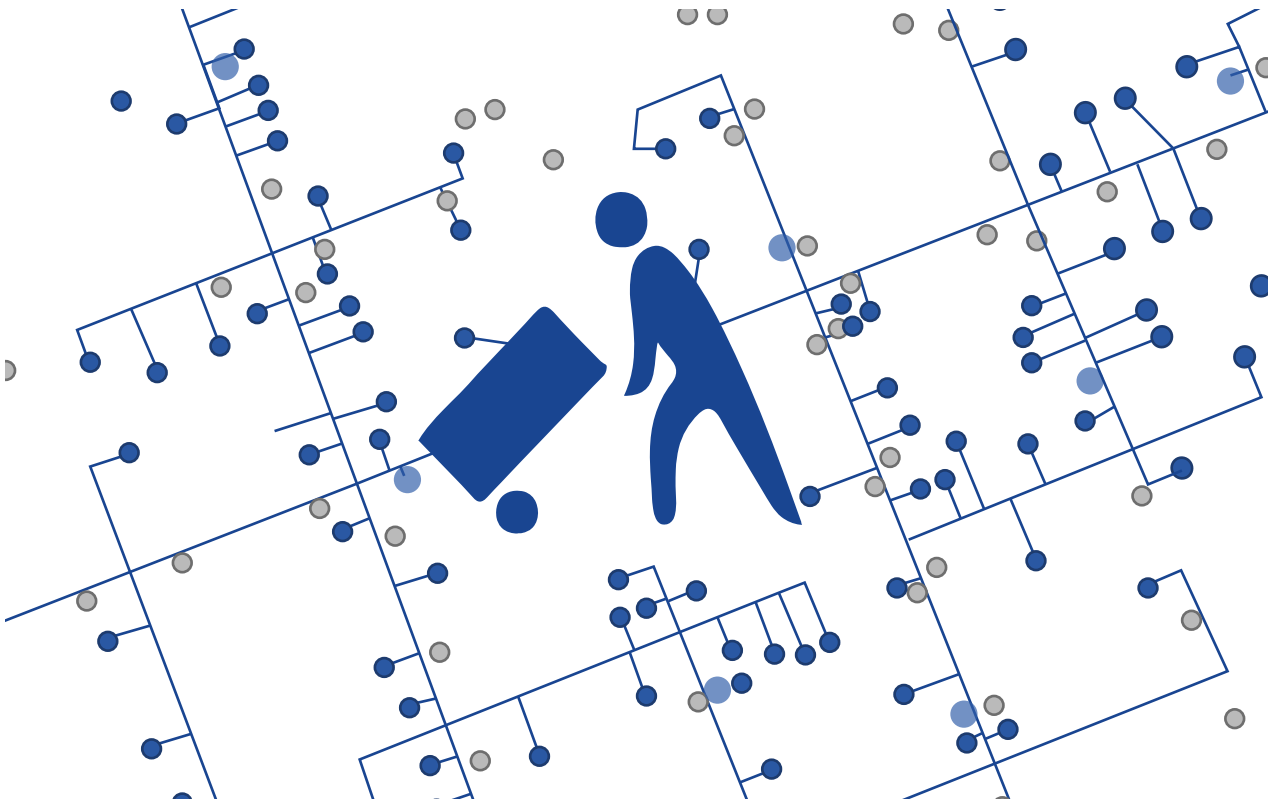


Ubicación óptima de zonas de carga/descarga en la vía pública

Proyecto de Grado
Ingeniería en Computación
Germán Faller



Supervisor: Dr. Ing. Pedro Piñeyro
Co-Supervisor: Dr. Ing. Antonio Mauttone
Usuario responsable: Ing. Matías Gutiérrez
Departamento de Investigación Operativa
Instituto de Computación
Facultad de Ingeniería - UDELAR
Noviembre 2021

Ubicación óptima de zonas de carga/descarga en la vía pública

Germán Faller

Resumen

La actividad económica y el dinamismo social de una ciudad dependen, entre otras cosas, de las operaciones logísticas, en particular de transporte de cargas. En entornos urbanos, un problema concreto es el de las operaciones de carga y descarga de mercancías. Con el fin de garantizar el buen funcionamiento de las mismas, es necesario trabajar en la búsqueda de mejores soluciones, que provean un suministro diario de productos en áreas urbanas de forma eficiente. Una zona de carga y descarga es un espacio reservado de estacionamiento en la vía pública, para realizar operaciones de carga y descarga. Este proyecto estudia el problema de la ubicación óptima de zonas de carga y descarga en entornos urbanos, considerando la posición y necesidad de los locales comerciales. Este puede ser aplicado a las necesidades de operaciones de carga y descarga en una área urbana específica, y en concreto se analiza el caso de una zona de Montevideo, Uruguay. Además, se diseña un modelo optimización basado en una formulación de programación lineal entera mixta, que busca minimizar la distancia entre las zonas y los comercios teniendo en cuenta las necesidades de operaciones de carga y descarga en una área urbana específica, con el fin de determinar donde ubicar las zonas de carga y descarga dentro de un conjunto de opciones conocido. Así mismo, se diseña e implementa un procedimiento de resolución basado en una técnica metaheurística y se validan los resultados frente a un software de optimización comercial. Por último, se desarrolla una herramienta para gestionar los datos y visualizar los resultados de forma amigable. Los resultados experimentales de la metaheurística muestran diferencias menores al 5% en el valor objetivo, comparado con un método exacto, siendo esto alcanzado en un tiempo razonable de cómputo.

Palabras Clave

Logística Urbana, Zonas de carga y descarga, Problema de Localización Asignación, Metaheurísticas, Programación Matemática

Agradecimientos

En primer lugar quiero agradecer al Dr. Ing. Pedro Piñeyro y al Dr. Ing. Antonio Mauttone, quienes con sus conocimientos y apoyo me han guiado durante este último trayecto.

También quiero agradecer especialmente al Ing. Matías Gutiérrez por haber propuesto este desafío, y haber brindado todos los recursos que fueron necesarios para llevar a cabo esta investigación. Además, es quien involucra a la Arq. Cecilia Cuadro y al Arq. Leonardo Rodríguez, ambos pertenecientes a la Unidad de Logística de la Intendencia de Montevideo, a quienes también agradezco por ser partes interesadas en el proyecto.

Asimismo agradezco a los docentes del Instituto de Estructuras y Transporte de la Facultad de Ingeniería Ing. Facundo Sosa y al Ing. Emiliano Gómez por haber facilitado datos reales y actualizados para el caso de estudio del proyecto.

Igualmente quiero agradecer al cuerpo docente de la facultad que de una u otra manera aportaron su ladrillo en esta construcción de conocimiento. También a todo funcionario de la facultad que hizo más ameno el transcurso de esta carrera.

Además quiero dedicar este logro a todas aquellas personas que están o estuvieron en mi vida, y que de diferentes formas me ayudaron a sortear los distintos desafíos a los que me vi enfrentado.

Por último quiero reconocer el inconmensurable soporte brindado por mi compañera, cómplice y amiga Yanina Viera, por todos estos años a mi lado en las buenas pero más aún en las malas.

*“Nada podemos esperar si no
es de nosotros mismos.”*

José Artigas

Índice general

Resumen	I
Agradecimientos	II
Índice General	III
Índice de figuras	IV
Índice de tablas	VI
1. Introducción	1
2. Revisión de literatura	7
3. Modelado	15
3.1. Extensión del problema	16
4. Procedimiento de resolución	21
4.1. Implementación de la heurística	22
5. Arquitectura de la solución	31
5.1. Etapa 1 - Prototipo	31
5.2. Etapa 2 - Implementación de la solución	35
5.3. Etapa 3 - Mejora del Desempeño	37
6. Experimentos computacionales	39
6.1. Validación del modelo	39
6.1.1. Tipo único de vehículo	40
6.1.2. Dos tipos de Vehículos	40
6.2. Evaluación de la metaheurística	41
6.2.1. Validación	42
6.2.2. Comparación contra método exacto	43
6.3. Cambio de plataforma	45
6.4. Caso de estudio	47

6.4.1. Variación de la Demanda	48
6.4.2. Variación en la etapa inicial de la heurística	48
6.4.3. Valoración del estado actual	50
7. Herramienta	53
7.1. Selección de datos de entrada	54
7.2. Configuración	55
7.3. Ejecución	56
7.4. Visualización alternativa	58
7.5. Descarga	59
8. Conclusiones y trabajo futuro	61
8.1. Conclusiones	61
8.2. Trabajo futuro	63
Bibliografía	63
A. Anexo	71

Índice de figuras

1.1. Caso de estudio Ciudad Vieja (Fuente: Carto, 2020).	4
4.1. Diagrama de flujo - VND	22
4.2. Representación de la solución para $n = 5$	23
4.3. Vecindad generada con <i>vns1swap</i>	23
4.4. Vecindad generada con <i>vns2swap</i>	23
5.1. GeoKettle - Procesamiento de los datos de entrada	32
5.2. GeoKettle - Generando los datos de salida	33
5.3. QGIS - Visualización del archivo de salida	34
5.4. GeoKettle - Secuencia de tareas	34
5.5. Vista previa de la herramienta	35
5.6. Modelo - Diagrama de clases	36
6.1. Escenario 1 - cuatro manzanas	39
6.2. Escenario 2 - nueve manzanas	41
6.3. Comparación rendimiento - Cplex Java Rust	46
6.4. Comparación Valor Objetivo - Cplex Java Rust	46
6.5. Escenario 3 - Caso de Estudio	47
6.6. Escenario 3 - Comparación Primera Etapa - Demanda 65%	49
6.7. Escenario 4 - Disposición Actual	50
6.8. Escenario 4 - Asignación de locales para disposición actual	51
6.9. Escenario 4 - Asignación de locales propuesta	51
7.1. Herramienta - Vista Principal	53
7.2. Herramienta - Escenario cargado	54
7.3. Herramienta - Ajustes de mapa	55
7.4. Herramienta - Otros Ajustes	55
7.5. Herramienta - Asignaciones	56
7.6. Herramienta - Anotaciones	57
7.7. Herramienta - Visualización trayecto caminando	58
7.8. QGIS - Importando solución descargada	59

7.9. QGIS - Importando solución descargada - Visualización alternativa . 59

Índice de tablas

6.1. Escenario 1 - Demanda generada	40
6.2. Escenario 1 - Demanda generada 2 niveles	41
6.3. Escenario 2 - Tipos de Vehículos por Zona	42
6.4. Resumen 1.600 Casos	44
6.5. Escenario 3 - Tiempo	48
6.6. Escenario 3 - Valor Obj.	48
6.7. Escenario 3 - Variantes <i>first stage steps</i>	49
A.1. Escenario 1 - Salida	71
A.2. Escenario 1 - Solución	72
A.3. Escenario 1 - Salida 2 niveles	73
A.4. Escenario 2 - Evaluación Metaheurística - Tipo 1	74
A.5. Escenario 2 - Evaluación Metaheurística - Tipo 2	75
A.6. 1.600 Casos - Comparación 1	76
A.7. 1.600 Casos - Comparación 2	78
A.8. 1.600 Casos - Comparación 3	80

Capítulo 1

Introducción

La distribución urbana de mercancías, también llamada distribución capilar o logística de última milla, se refiere al tramo final de la cadena de suministros. Este concepto engloba a todos los movimientos relacionados con actividades comerciales, de abastecimiento y distribución de productos para la industria, además de la comercialización y el consumo de bienes en las ciudades. Representa una parte fundamental para la actividad comercial y de servicio, pero al mismo tiempo trae consigo externalidades como la contribución al congestionamiento del tránsito, que puede llegar a generar hasta un 25 % de las emisiones de gases de efecto invernadero en las áreas metropolitanas, así como, interferir con el resto del transporte urbano respecto al uso del espacio público. La logística de última milla puede representar hasta el 28 % de los costos logísticos totales de las empresas del sector privado. Para que las ciudades sean competentes en este aspecto se necesita que la distribución urbana sea eficiente, ya sea por el aumento de entregas o la reducción de tiempos en las mismas (Banco Interamericano de Desarrollo, 2009).

Dada la importancia de la distribución urbana, esta ha sido ampliamente estudiada a lo largo del tiempo. Diferentes investigaciones han abordado el tema con propósitos económicos (eficiencia en costos y tiempo (Estrada, 2007)), sociales (reducción de los embotellamientos) y ambientales (minimizar emisiones de gases de efecto invernadero). Según van Duin y Quak (2007) el objetivo de los enfoques de planeamiento urbanístico conjuntamente con los ambientales es "proporcionar los medios de oportunidad para satisfacer las necesidades económicas, ambientales y sociales de manera eficiente y equitativa, y reducir al mínimo los efectos adversos, evitables e innecesarios y sus costos asociados, en el espacio correspondiente y escalas de tiempo".

Es necesario comprender los efectos de las operaciones y las actividades en el área urbana para luego relevar su importancia a la hora de pensar en políticas basadas en la planificación del transporte urbano de mercancías.

Las políticas más frecuentes en materia de transporte de carga urbano son: establecimiento de periodos de entrega para las operaciones, políticas de restricciones de tipo de vehículo, políticas de carga y descarga, fiscales y de centros de trasbordo y consolidación. Cada una de ellas tiene diferentes impactos con respecto al bien que se transporta y al uso de canales de distribución. Las propuestas que intentan mejorar la logística en zonas urbanas muchas veces abarcan más de una única política, debido a los efectos que producen en las operaciones de carga de productos (Ballantyne y Lindholm, 2013).

Se puede analizar el problema desde dos perspectivas. La primera abarca políticas basadas en restricciones (Anderson *et al.*, 2005; Browne *et al.*, 2012; Russo y Comi, 2011). Dentro de estas, se hallan las restricciones de horas/vías. Consisten en la prohibición del ingreso de camiones a determinadas zonas de forma permanente o durante cierto periodo de tiempo. Por otro lado se identifican las restricciones de peso/tamaño máximo, que acotan a los tamaños de los vehículos según el área geográfica. Por último, se encuentran las restricciones de zonas de bajas emisiones, en donde se establece una cantidad máxima de emisiones en determinadas zonas geográficas.

La segunda perspectiva está basada en políticas orientadas a mejorar la eficiencia. Aquí se ubican los centros de consolidación (Browne *et al.*, 2012; Goldman y Gorham, 2006; Marinov *et al.*, 2010; Russo y Comi, 2011). Estos son espacios cooperativos de las distintas tiendas de la zona, donde se recibe la mercadería para luego ser distribuida a los distintos locales comerciales en vehículos de menor porte. Otro enfoque de estas políticas reside en la ubicación de las zonas de carga y descarga (Browne *et al.*, 2012; Russo y Comi, 2011), es decir, secciones de la vía pública destinadas exclusivamente para aparcamiento de vehículos mientras realizan tareas de carga o descarga de mercadería. Por otro lado, existen las políticas de vías exclusivas (Russo y Comi, 2011), las cuales son utilizadas exclusivamente por vehículos de transporte de cargas. También se encuentra la política de transportes nocturnos (Browne *et al.*, 2012; Holguín-Veras, 2013), que apuntan a la distribución de cargas en horarios de la noche, donde se prevé menos actividad urbana. Otra de las políticas impulsadas con el fin de mejorar la eficiencia, es el uso de tecnologías de la información. Estas aprovechan los sistemas de información para mejorar las diferentes etapas de la logística urbana según las necesidades que se presenten. Incluso se encuentran los sistemas de capacitación, enfocados en las necesidades de cada actor del sistema logístico perfeccionando técnicamente a los involucrados (Russo y Comi, 2011). Por último se hallan las políticas de buenas prácticas, que se ocupan de desarrollar manuales, actas o agrupaciones con el fin de lograr la interacción entre los actores y la aceptación de acuerdos o prácticas de cada país o ciudad (Browne *et al.*, 2012).

Cuevas *et al.* (2013) realizan un relevamiento de las políticas ya nombradas y las clasifican dentro de determinadas dimensiones: económica, social y ambiental, proporcionando así la sistematización de las investigaciones realizadas con respecto al tema en cuestión.

El proyecto de grado descrito en este informe tiene como eje central el estudio de las actividades de carga y descarga en entornos urbanos. Estas son enmarcadas dentro de las políticas de eficiencia mencionadas anteriormente. Apunta a la mejora en los procesos logísticos, mediante la asignación de zonas exclusivas, con el fin de estacionar y realizar operaciones de carga y descarga (Browne *et al.*, 2012; Muñuzuri *et al.*, 2012; Russo y Comi, 2011).

Una zona de carga y descarga (ZCD) está definida como un lugar en la vía pública destinado para la realización de procesos de esta índole. En otras palabras, son plazas de estacionamiento para que diversos tipos de vehículos (desde camiones hasta motos), tengan la posibilidad de detenerse en ellas, poder descargar bienes y trasladarlos al local predeterminado. Estas zonas también pueden ser usadas para intercambiar bienes entre vehículos. Gracias a la correcta asignación de dichas zonas, la eficiencia en las operaciones de logística capilar mejora (Nourinejad *et al.*, 2014).

Dentro de las ventajas que ofrece esta política se hallan la fácil implementación y los bajos costos. Además, resulta ser más efectiva que las políticas basadas en restricciones, como por ejemplo las de horas/vías o de peso/tamaño máximo, porque busca administrar la logística en zonas urbanas sin alterar la operativa de la actividad (Dablanc *et al.*, 2013).

Motivado por el contexto descrito anteriormente, en este proyecto se propone estudiar el problema de localización y asignación de zonas de carga y descarga, el desarrollo de un modelo de optimización así como un algoritmo que lo resuelva de la manera más eficiente posible. Además, se tiene como objetivo implementar una herramienta que le permita a los tomadores de decisiones trabajar con instancias del problema de forma ágil. Se definieron los siguientes objetivos específicos para este proyecto:

1. Conocer el estado del arte para el problema de localización y asignación, enmarcado en la ubicación de zonas de carga y descarga.
2. Seleccionar y extender un modelo matemático encontrado en la literatura, que se enfoca en la ubicación óptima de zonas de carga y descarga en entornos urbanos.
3. Desarrollar y validar el modelo matemático extendido, con el fin de que pueda ser aplicado a zonas concretas de la ciudad de Montevideo.
4. Implementar un algoritmo que sea capaz de encontrar buenas soluciones utilizando una metaheurística y respetando las restricciones del modelo matemático extendido.

5. Crear una herramienta web para el manejo y presentación de los datos del problema que le permita al usuario resolver diferentes instancias, con diversas configuraciones.

Motivado desde la Unidad de Logística de la Intendencia de Montevideo (de aquí en más, contraparte), este proyecto está enmarcado dentro del planeamiento del uso de espacio público y tiene como enfoque la ubicación óptima de zonas de carga y descarga. De esta manera, se pretende mejorar las características de la distribución urbana de mercancías en la ciudad de Montevideo, más concretamente en el barrio Ciudad Vieja como zona de estudio (ver Figura 1.1) Las propiedades de su infraestructura son: calles estrechas, la presencia de peatonales, y la concentración de zonas comerciales; lo anterior conforma un escenario propenso a que se produzcan maniobras irregulares, tales como estacionar ilegalmente (en doble fila por ejemplo), factor que contribuye a la congestión y al aumento de la inseguridad vial.

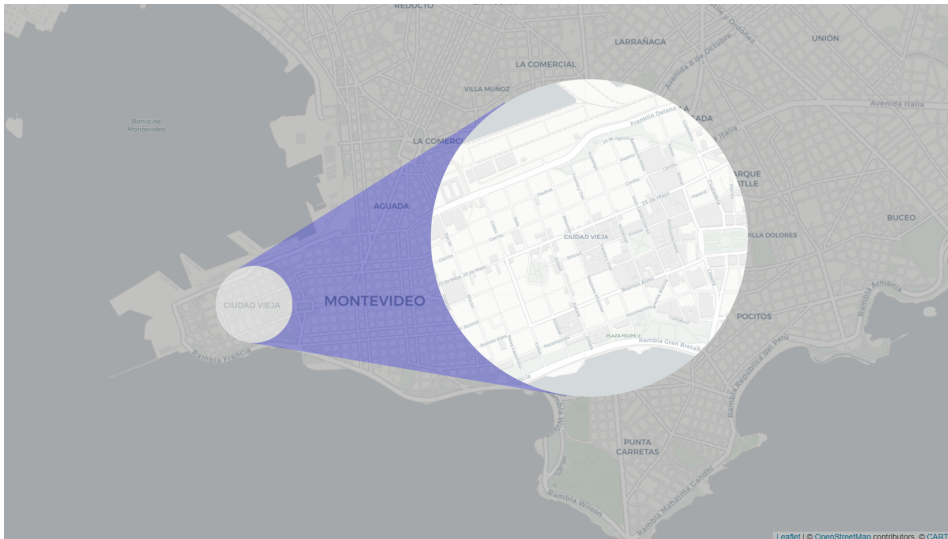


Figura 1.1: Caso de estudio Ciudad Vieja (Fuente: Carto, 2020).

El resto de este documento se estructura de la forma que se describe a continuación. En el Capítulo 2, se define formalmente el problema de ubicación y asignación de zonas de carga y descarga, y se hace una revisión de la literatura existente relacionada con el problema. En el Capítulo 3, se presenta un modelo tomado de la literatura y se extiende convenientemente. El diseño de los algoritmos de resolución desarrollados se presentan en el Capítulo 4. En el Capítulo 5, se presenta la arquitectura y la evolución de la solución informática. En el Capítulo 6 se muestra la validación y evaluación del procedimiento de resolución desarrollado. En el Capítulo 7, se muestra la herramienta implementada. Las conclusiones y trabajo a futuro se encuentran en el Capítulo 8.

Capítulo 2

Revisión de literatura

El problema abordado en este proyecto radica en la localización y asignación de zonas de carga y descarga a locales comerciales en áreas urbanas. Cada local tiene como demanda un tiempo estimado de uso de las zonas con vehículos que le proveen. Por otra parte, cada zona cuenta con una capacidad de tiempo disponible de aparcamiento.

En este capítulo se hace una recopilación de los problemas de localización y una revisión de la literatura relacionada con el problema. Además, se toma un modelo de referencia como base para el trabajo de este proyecto.

El estudio del problema de ubicación de instalaciones (FLP - Facility Location Problem) es una rama de la Investigación de Operaciones que procura ubicar de forma óptima las instalaciones, minimizando los costos asociados al transporte y de instalación, para satisfacer la demanda de un conjunto de clientes. Distintas variantes pueden considerar más elementos del contexto como la densidad poblacional o análisis de mercado. Dentro de estas se enumeran, a efectos de este documento, las más relacionadas.

El problema de ubicación de instalaciones, capacitado (CFLP - Capacitated Facility Location Problem) o no capacitado (UFLP - Uncapacitated Facility Location Problem) es un clásico problema de optimización. En el mismo se determina la cantidad y la ubicación de un conjunto de instalaciones, asignando clientes con cierta demanda a dichas instalaciones, de manera tal que el costo sea mínimo y la demanda sea satisfecha. En este caso los costos a tener en cuenta son, el costo de instalación (dado por abrir una instalación) y el costo de transporte dado por el cliente asignado a dicha instalación. Si una cantidad de clientes cualquiera puede ser asignada a una instalación, se dice que el problema es *no capacitado* (UFLP). Mientras que si la instalación maneja una capacidad acotada para satisfacer la demanda de los clientes, el problema se vuelve *capacitado* (CFLP). Ambos problemas se conoce que son \mathcal{NP} -hard (Cornuéjols *et al.*, 1990).

El problema de ubicación de cobertura máxima (MCLP - Maximal Covering Location Problem) fue propuesto por Church y ReVelle (1974) y a diferencia del clásico FLP, separa los costos de instalación y de transporte, dejando los primeros a criterio del tomador de decisiones asumiendo los costos de construir P instalaciones, mientras que se centra en la óptima ubicación de las mismas de modo tal que estas cubran la máxima demanda en un radio S . Con este enfoque se asume que las instalaciones no tienen una capacidad límite.

Los problemas *p-median* y *p-center* fueron introducidos por Hakimi (1964). Tienen similitudes a MCLP, pero con diferencias en la función objetivo, además de no considerar el radio de cobertura. En *p-median* se busca minimizar la distancia acumulada de los viajes de los clientes a las instalaciones, mientras que en *p-center* se hace foco en el cliente más distante, es decir, se busca minimizar la distancia máxima entre cliente e instalación.

En la literatura se encuentran varios autores que abordan el problema de ubicar zonas de carga y descargas en la vía pública. Proponen diferentes metodologías para llegar a la solución óptima. En este apartado se hará un breve recorrido por los trabajos más relevantes que se relacionan al problema de este proyecto.

Aiura y Taniguchi (2005) formulan un modelo de planeamiento de espacios en la vía pública para que vehículos de carga y descarga puedan estacionar. El modelo considera el comportamiento de los vehículos particulares así como los de carga y descarga en la ciudad de Kyoto, Japón. La herramienta que propusieron está compuesta de dos sub-modelos. El primero está basado en problemas de ubicación de zonas de carga y descarga; minimiza el costo total, que se compone de multas por retraso, costos de operación, multas por estacionar inapropiadamente, costo de espera en la carga y descarga como también de los vehículos particulares y costos de reparaciones. Mientras que el segundo modelo está basado en la simulación de tráfico de vehículos de carga y descarga, coexistiendo con vehículos particulares en una red de calles en la ciudad. Ambos interactúan en una herramienta con algoritmos genéticos como método de búsquedas. Antes de proponer la solución, se realizó una encuesta a los conductores relacionados a las operaciones de carga y descarga con el fin de entender la situación actual de ese tema. De los resultados obtenidos en esa encuestas y en anteriores, se definieron parámetros, como número y ubicación de zonas disponibles para carga y descarga, tiempo máximo de carga y permanencia, regulaciones existentes a los conductores de autos particulares, distribución y características del tráfico en ese lugar, entre otras.

Dezi *et al.* (2010) se enfocan en la ciudad de Bolonia, Italia, la cual cuenta con una herramienta llamada MerciBo2 para gestionar la distribución de bienes en zonas urbanas. Se definen dentro de la ciudad áreas llamadas zonas de tráfico controlado (limited traffic zone - LTZ). Estas zonas son analizadas y dentro de la información más relevante, se encuentra que el 67% de los vehículos usan menos del 25% de su capacidad y el 12% usan menos del 50% de su capacidad. Igualmente, más del 40% de los vehículos son particulares. Además, dentro de las LTZ se tienen zonas de carga y descarga (Single Commercial Vehicle Loading/Unloading Zone - SCVLZ) y se analiza el tipo de parada de los vehículos comerciales. El 57% de las paradas dentro del LTZ no son permitidas, el 29% estaciona en doble fila, solo el 7% utiliza las SCVLZ, mientras que el 5% estaciona en lugares permitidos y el 2% en zonas privadas. Las razones para no utilizar las SCVLZ son: el 57% dice que la zona ya está ocupada por otro vehículo no comercial, el 29% manifiesta que hace falta más zonas, el 8% se encuentra con la zona ya ocupada por otro vehículo comercial y al 6% le parece que la zona está muy lejos. Después de analizar los datos de uso y tipos de vehículos se inicia un proceso de mejora de la situación. Inicialmente determina los tamaños óptimos para la zona de carga, en este caso tres tamaños. Luego, de la herramienta se puede obtener la cantidad de viajes así como también el tiempo promedio necesario para realizar una carga o descarga, concluyendo el nivel de concurrencia necesario, y por ende, la cantidad de SCVLZ que se van a instalar. Por último para ubicar las zonas, se define una distancia máxima de caminata o radio de acción de la zona, y luego se ubican en dos etapas. Primero se reubican las zonas ya instaladas que presentan solapamientos, es decir, que un negocio se encuentra en el radio de acción de más de una zona. Luego se ubican las nuevas, con una estrategia greedy, ubicando las primeras en zonas de mayor concentración comercial.

Dablanc *et al.* (2013) realizan una recopilación de investigaciones sobre transporte de cargas en áreas urbanas, y las agrupa en tres categorías, que dependen de la estrategia que abordan. El objetivo de ese trabajo además de concentrar la información, es decidir cuál de esas estrategias se adapta mejor al contexto estadounidense. Por un lado están las orientadas a reducir impactos en el medio ambiente, por otro las estrategias vinculadas a los depósitos y finalmente aborda las relacionadas con última milla. Esta última agrupación relacionada con las estrategias de última milla está estrechamente enlazada con el problema abordado en este proyecto de grado. Esta categoría a su vez está dividida en sub-categorías.

1. Etiquetado y programas de certificación: la característica principal de esta estrategia es su voluntariedad. El sector público en conjunto con el privado negocian las metas y reglas operativas, que determinan beneficios para ambas partes mediante certificaciones ecológicas. La desventaja que se identificó de

esta estrategia fue que los grandes transportistas tienen más peso en la fijación de reglas, por lo que algunos sectores de la industria quedan excluidos. Sin embargo, promueve las relaciones colaborativas entre el ámbito público y privado. A medida que se cumplen las metas, se proponen otras y lleva a mejores resultados con el tiempo.

2. Regulaciones de estacionamiento y de tráfico: las autoridades han usado esta herramienta para regular los problemas de última milla porque están bajo su poder. Sin embargo, no tienen control sobre las demandas de cargas y descargas, por lo que resulta no ser una estrategia muy efectiva. Este tipo de regulación suele ser exitosa si se combina con la eficiencia del uso de los recursos viales y teniendo en cuenta la demanda local.
3. Políticas de planeamiento local: esto implica que cada local se haga responsable de los bienes que carga y descarga. Los propietarios pueden verse reacios ante esta política, pero ese costo inicial de desarrollo implica un aumento en el valor de la propiedad, ya que, garantiza una zona de carga y descarga. Esto resulta beneficioso, tanto para la persona que usa el local como para los transportistas.
4. Logísticas de ciudad y programas de consolidación: estas tratan de disminuir la concentración de camiones de carga y descarga en zonas urbanas a través de la instalación de espacios logísticos compartidos.
5. Entregas fuera de horas: el objetivo de esta estrategia es reducir la congestión en horas pico, así como las emisiones. Los transportistas podrían hacer sus entregas más rápido, sin embargo los locales tendrían que tener la capacidad de recibir o entregar bienes fuera del horario comercial.
6. Sistemas inteligentes de transporte: esta estrategia implica a tecnologías que proporcionen en tiempo real información sobre tráfico y estacionamiento, pero también que administren las cargas urbanas a través de aplicaciones automatizadas de cobro de peajes, estacionamiento o control de acceso automatizado. Al inicio es una estrategia muy costosa, pero luego de implementada hace posible la baja de costos de peajes o zonas de acceso.

Frente a un problema con características similares al abordado en este proyecto, se encontró que en el trabajo de Muñuzuri *et al.* (2017) proponen una solución para mejorar el sistema de carga y descarga. En esta oportunidad las pruebas se desarrollaron en la ciudad Sevilla, España. El modelo consta de dos etapas; la primera evalúa el número de zonas de carga y descarga en una determinada calle, con el objetivo de establecer un número de ZCD mínimo para satisfacer una cantidad de locales predeterminados, y en la segunda las distribuye teniendo en cuenta la demanda a satisfacer de determinado local comercial ubicado en esa calle, es decir, busca el mejor lugar para abrir una ZCD. El caso de Sevilla considera una calle (segmento de calle o conjuntos de calles), con un número de comercios establecidos, cada uno

de ellos tiene una demanda promedio diaria, y un promedio de estancia para cada vehículo en la ZCD. Si bien hay factores sociales importantes, por ejemplo: desconocimiento de las reglas, o uso excesivo de las ZCDs, los cuales escapan al modelo y los gobiernos deberían tener en cuenta, este procedimiento sirve de instrumento para un tomador de decisiones, brindando un panorama más claro de la ciudad, los locales y las posibles ZCDs.

Alho *et al.* (2018) analizan el caso de dos zonas de Lisboa, Portugal, ambas con una extensión de 1km^2 aproximadamente. Primero hacen una estimación de la demanda basada en una técnica de machine learning llamada Partition Model, utilizando como variables, la cantidad de viajes semanales, la cantidad de empleados, la superficie, la zona y el rubro del local. Luego analiza la cantidad de zonas de carga y la demanda cubierta en los radios de acción de 25m, 50m y 75m, se eligen diferentes sub conjuntos del total de zonas disponibles y se toman como entrada en un software de simulación. La simulación otorga datos de los vehículos particulares tales como: velocidad promedio, número de paradas, kilómetros recorridos y combustible consumido, para evaluar cómo afectan distintas configuraciones en la zona.

Prata *et al.* (2018) proponen una relajación del problema set-covering y *p-median*, con el objetivo de resolver un problema real en Fortaleza, Brasil. Se plantea un modelo matemático en respuesta al rápido crecimiento de la ciudad. Cuando se realizó el estudio, Fortaleza contaba con 908.074 vehículos, de los cuales el 8% corresponden a vehículos de carga, y su población es de 3,8 millones. Calles angostas, concentración de actividades comerciales, embotellamientos, son algunas de las características del centro de esa ciudad, que dificultan las operaciones de carga y descarga. El modelo matemático que propone maneja dos conjuntos, los clientes y las zonas, y cuenta con dos parámetros importantes: un factor de penalización por no cubrir clientes y el radio que abarcan las zonas. El objetivo es obtener la cantidad óptima de zonas de carga y descarga y de cubrir el número máximo de clientes. Se espera que el modelo sea capaz de determinar el número mínimo de zonas, y que cubra un número máximo de clientes. Luego que se determina un número óptimo de zonas, se pueden elegir las zonas que están más cerca del cliente, con el fin de minimizar el manejo de las cargas.

Por otro parte, se debe tener en cuenta los distintos métodos de resolución del problema de localización y asignación encontrados en la literatura. Habiendo ejemplos basados en:

- Relajación Lagrangeana (Hindi *et al.*, 1998; Li *et al.*, 2017; Pirkul y Jayaraman, 1998), estos están basados en iterar sobre los multiplicadores de Lagrange buscando minimizar la brecha entre una solución factible y la cota inferior del problema relajado.

- Particle Swarm Optimization (PSO) (El-Sherbiny y Alhamali, 2013; Sadeghi-Moghaddam *et al.*, 2019), algoritmo poblacional con búsquedas estocásticas en cada individuo (partícula), variando la velocidad y dirección de búsqueda en cada ciclo, deteniéndose al satisfacer la demanda y comparando las distintas soluciones encontradas.
- Simulated Annealing (SA) (Jayaraman y Ross, 2003; Molla-Alizadeh-Zavardehi *et al.*, 2013; Sadeghi-Moghaddam *et al.*, 2019), donde se introduce una perturbación en los datos. Las perturbaciones son cada vez menor en cada ciclo, buscando así, escapar de óptimos locales.
- Algoritmos Genéticos (GA) (Zhou *et al.*, 2002; Hajiaghahi-Keshteli *et al.*, 2010; Molla-Alizadeh-Zavardehi *et al.*, 2011; Molla-Alizadeh-Zavardehi *et al.*, 2013; El-Sherbiny y Alhamali, 2013; Sadeghi-Moghaddam *et al.*, 2019), otra alternativa poblacional, inspirada en la genética, donde la siguiente generación es construida a partir de cruces de individuos previamente seleccionados, y algunos de ellos, con una ligera perturbación llamada mutación.
- Tabu Search (TS) (Lee y Kwon, 2010), este método, consiste en tener una colección de soluciones ya evaluadas (lista tabú), para que así, cuando en la búsqueda se genere una solución antes evaluada, evitar recorrer el mismo camino.
- Variable Neighborhood Search (VNS) (Molla-Alizadeh-Zavardehi *et al.*, 2013), esta estrategia maneja distintas maneras de recorrer el espacio de soluciones, cada una de estas maneras son llamadas vecindades. Cuando en una vecindad no hay vecinos mejores, se pasa a la siguiente, con el detalle de que al mejorar en una vecindad se vuelve a la inicial.
- Métodos específicos (Sharma *et al.*, 2008; Tsao, 2013), estos se basan en el análisis matemático y desarrollan una solución al problema específico, con el fin de determinar la ubicación de las instalaciones.

De la bibliografía en lo que se refiere al modelado, el trabajo más acorde al problema de este proyecto, es el introducido en Muñuzuri *et al.* (2017). En este proyecto se aborda un problema con características similares, restricciones que son de interés, tales como, cumplir con el total de la demanda y zonas con capacidad limitada. Se cuenta con los siguientes datos acerca del caso de estudio: ubicación de zonas disponibles para instalar una ZCD, ubicación y demanda de los locales comerciales. Es por esto que se tomará como base y se propondrá una extensión del mismo en el Capítulo 3.

Capítulo 3

Modelado

En este capítulo se propone una extensión al modelo de referencia sugerido en el trabajo de Muñuzuri *et al.* (2017) teniendo en cuenta las características que son de interés para este proyecto.

Recapitulando, el problema abordado trata de la localización y asignación de zonas de carga y descarga. Cada local comercial, además de su ubicación, tiene como demanda un tiempo estimado de uso de la zona. Por otra parte, cada zona cuenta con una capacidad de tiempo disponible de aparcamiento. La solución deberá contemplar que la demanda expresada en minutos para las operaciones de carga y descarga de todos los locales sea satisfecha, cuidando de no sobrepasar la capacidad de las zonas. Bajo estas condiciones, Muñuzuri *et al.* (2017) proponen el siguiente modelo matemático. Primero se introduce la notación y luego se presenta el modelo.

Conjuntos:

$\mathcal{I} : \{i \in \mathbf{N} \mid 1 \leq i \leq n\}$ Conjunto de potenciales ZCD

$\mathcal{J} : \{j \in \mathbf{N} \mid 1 \leq j \leq m\}$ Conjunto de locales comerciales

Parámetros:

N : Cantidad de ZCD a abrir

n : Cantidad de potenciales ZCD disponibles, con $N \leq n$

m : Cantidad de locales

C_i : Capacidad de la ZCD i en unidades de tiempo

D_j : Demanda del local j en unidades de tiempo

d_{ij} : Distancia entre la ZCD i y el local j

Variables:

x_{ij} : Cantidad de tiempo asignado a la ZCD i para el local j

y_i : Variable de activación, si la ZCD i está siendo usada

Modelo:

$$\min \sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{J}} d_{ij} \cdot x_{ij} \quad (3.1)$$

$$\text{sujeto a } \sum_{i \in \mathcal{I}} y_i = N \quad (3.2)$$

$$\sum_{i \in \mathcal{I}} x_{ij} = D_j, \quad \forall j \in \mathcal{J} \quad (3.3)$$

$$\sum_{j \in \mathcal{J}} x_{ij} \leq C_i \cdot y_i, \quad \forall i \in \mathcal{I} \quad (3.4)$$

$$y_i \in \{0, 1\}, x_{ij} \geq 0, \quad \forall i \in \mathcal{I}, \forall j \in \mathcal{J} \quad (3.5)$$

En el modelo matemático anterior la función objetivo (3.1), minimiza la suma de los productos entre distancia y demanda, para lograr asignar locales a las ZCD más cercanas. La restricción (3.2) fuerza el número de zonas a activarse, mientras que, las restricciones (3.3) aseguran la satisfacción de la demanda de cada local. En las restricciones (3.4) se limita la cantidad de demanda que se puede asignar a cada zona de carga, incluido el hecho de que una zona de carga solo puede satisfacer la demanda si está activada. Finalmente, las restricciones (3.5) establecen el carácter binario de las variables y_i y el carácter no negativo de x_{ij} .

3.1. Extensión del problema

La definición del modelo propuesto en este proyecto es el resultado de varias instancias de intercambios con la contraparte, participando en el diseño y en la provisión de datos.

A partir del modelo anterior se tomaron las siguientes decisiones:

- Para controlar que un local no sea asignado a una zona remota, se busca que la distancia d_{ij} entre el local y la ZCD asignada no supere una distancia máxima mD previamente determinada.
- El modelo permite repartir la demanda de un local entre varias zonas, ya que los datos de demanda por local están agregados, lo que tiene sentido. De lo contrario, si se usase la información de todas las entregas por local se podría derivar en soluciones poco robustas ante cambios en los datos. Al tener los datos agregados puede que la demanda se fraccione en porciones arbitrariamente pequeñas. Para evitar estas fracciones, se fuerza a que las asignaciones sean de al menos cierto tiempo mT , ya que se asume un tiempo

Modelo:

$$\min \sum_{i=1}^n \sum_{j=1}^m \sum_{k=1}^K d_{ij} \cdot x_{ijk} \quad (3.6)$$

$$\text{sujeto a } \sum_{i=1}^n \sum_{k=1}^K y_{ik} = N \quad (3.7)$$

$$\sum_{i=1}^n x_{ijk} = D_{jk}, \quad \forall j \in \mathcal{J}, \quad \forall k \in \mathcal{K} \quad (3.8)$$

$$\sum_{k=1}^K y_{ik} \leq 1, \quad \forall i \in \mathcal{I} \quad (3.9)$$

$$y_{ik} = 0, \quad \forall k \in \mathcal{G}_i, \quad \forall i \in \mathcal{I} \quad (3.10)$$

$$mT \cdot y_{ik} \leq \sum_{j=1}^m x_{ijk}, \quad \forall i \in \mathcal{I}, \quad \forall k \in \mathcal{K} \quad (3.11)$$

$$\sum_{j=1}^m \sum_{k_1=k}^K x_{ijk_1} \leq C_i \cdot \sum_{k_2=k}^K y_{ik_2}, \quad \forall i \in \mathcal{I}, \quad \forall k \in \mathcal{K} \quad (3.12)$$

$$\sum_{k=1}^K y_{ik} = 1, \quad \forall i \in \mathcal{P} \quad (3.13)$$

$$x_{ijk} \leq C_i \cdot z_{ijk}, \quad \forall i \in \mathcal{I}, \quad \forall j \in \mathcal{J}, \quad \forall k \in \mathcal{K} \quad (3.14)$$

$$x_{ijk} \geq mT \cdot z_{ijk}, \quad \forall i \in \mathcal{I}, \quad \forall j \in \mathcal{J}, \quad \forall k \in \mathcal{K} \quad (3.15)$$

$$z_{ijk} \cdot d_{ij} \leq mD, \quad \forall i \in \mathcal{I}, \quad \forall j \in \mathcal{J}, \quad \forall k \in \mathcal{K} \quad (3.16)$$

$$x_{ijk} \geq 0, \quad \forall i, j, k \quad (3.17)$$

$$y_{ik} \in \{0, 1\}, \quad \forall i, k \quad (3.18)$$

$$z_{ijk} \in \{0, 1\}, \quad \forall i, j, k \quad (3.19)$$

La función objetivo (3.6) minimiza la suma de las distancias de cada viaje (local - ZCD), considerando el tiempo de permanencia en la zona. Las restricciones (3.7), (3.8) se ajustan a partir del modelo de referencia para incluir el tipo de vehículo. Con las restricciones (3.9), (3.10) y (3.11) se busca limitar el tipo de vehículo que será asignado en la zona de descarga. Solo puede asignarse el mayor tipo de vehículo que sea permitido en la ZCD, mientras tenga una demanda mayor a cero. En (3.12) se impide que se asigne demanda para un tipo mayor al tipo de la zona. En (3.13) se fuerza a usar las zonas de descargas prefijadas en \mathcal{P} . Con las restricciones (3.14), (3.15) y (3.16) se activa la variable z_{ijk} , se acota el tiempo mínimo de uso a mT y se limita la distancia a un máximo de mD , respectivamente. Finalmente (3.17), (3.18) y (3.19) son restricciones de dominio de las variables del modelo.

Cabe destacar que el modelo propuesto representa un problema de programación lineal entera mixta (MILP - Mixed Integer Linear Programming) y además presenta características de complejidad iguales a los problemas de localización y asignación mencionados al inicio de este capítulo, siendo estos problemas \mathcal{NP} -hard. En estos casos, cuanto mayor sea el tamaño del problema, el tiempo de cómputo necesario para encontrar una solución óptima se incrementa de forma exponencial. Es por esta razón que dentro de los objetivos de este proyecto está desarrollar un procedimiento de resolución del problema basado en una técnica metaheurística, con el fin de conseguir soluciones de buena calidad en tiempos razonables. Las técnicas metaheurísticas son métodos de solución que coordinan una interacción entre los procedimientos de mejora local y estrategias avanzadas para crear un proceso capaz de escapar de los óptimos locales y realizar una búsqueda robusta de un espacio de soluciones (Gendreau y Potvin, 2010).

Capítulo 4

Procedimiento de resolución

Abordar con métodos exactos el problema de localización y asignación para instancias grandes del mismo suele no ser adecuado. Debido a la naturaleza de este tipo de problema en particular, cuanto mayor es el tamaño, el tiempo de cómputo necesario para encontrar una solución óptima se incrementa exponencialmente. Por ello, es sensato recurrir a un método alternativo a los exactos, con el objetivo de obtener soluciones de buena calidad y en tiempos razonables. Este capítulo propone un procedimiento de resolución basado en la metaheurística Variable Neighborhood Descent (VND).

La búsqueda de vecindad variable (VNS - Variable Neighborhood Search) fue introducida por Mladenović y Hansen (1997). Es un método metaheurístico para resolver problemas de optimización. Este se inicia generando una solución para el problema puntual. Con el fin de mejorar el cubrimiento del espacio de soluciones factibles, tiene en cuenta distintos conceptos de vecindad. Aplicando una serie de transformaciones a una solución dada se generan nuevas soluciones; a cada una de estas transformaciones se le llama vecindad. Luego se aplica un procedimiento de generación aleatoria de soluciones a partir de la vecindad, conocido como “shake”. Por último, se ejecuta una búsqueda local para las soluciones generadas y se compara con la mejor solución conocida. En este punto existen dos estrategias de mejora: aquella que busca la mejor opción de la vecindad, conocida como “best improve”, y la llamada “first improve”, que intenta encontrar la primera opción dentro de la vecindad que mejore la solución actual. Si la solución es mejor, pasa a ser la mejor conocida y se reinicia el procedimiento. De lo contrario, se cambia de vecindad y se sigue buscando.

En este proyecto, se utilizó una variante del VNS, conocida como VND (ver Figura 4.1) introducida por Brimberg *et al.* (2000). La diferencia está, en que, este último no utiliza “shake” ni búsqueda local. Evoluciona tomando la mejor solución de cada vecindad; y al igual que VNS, si esta no es mejor que la mejor conocida, se cambia de vecindad y se sigue buscando.

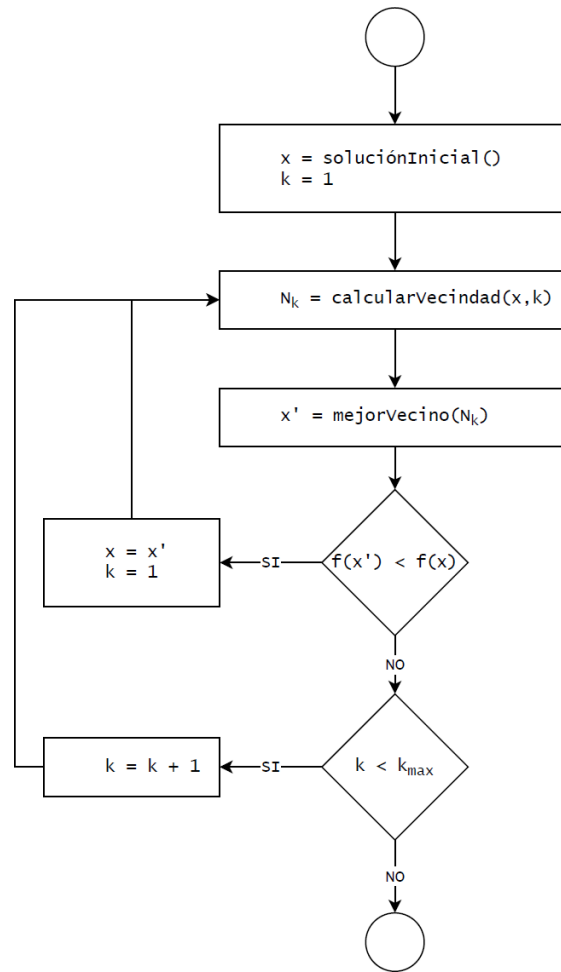


Figura 4.1: Diagrama de flujo - VND
(elaboración propia)

4.1. Implementación de la heurística

En esta subsección se presentan los algoritmos desarrollados para resolver el problema presentado en la Sección 3.1 del Capítulo 3. El procedimiento de resolución está basado en VND, pero antes de entrar en los detalles del algoritmo, se debe definir la representación de la solución y las nociones de vecindad que se emplearan en el mismo.

Representación de la solución

La solución es representada por un conjunto de n valores ordenados, donde la posición del valor está vinculada a una Zona de Carga y Descarga; siendo los valores posibles unos y ceros, correspondiendo el valor uno cuando la ZCD está abierta y cero cuando no (ver Figura 4.2).

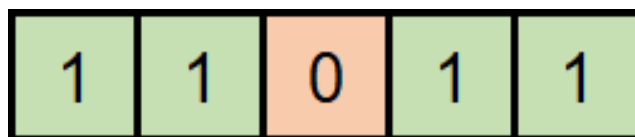


Figura 4.2: Representación de la solución para $n = 5$.
Celdas verdes (1) indican ZCD abierta y
anaranjadas (0) indican que la ZCD está cerrada.

Vecindad

Se hará uso de dos nociones de vecindad. La primera, la cual se llamará *vn_s1swap* (ver Figura 4.3), toma la solución base y por cada ZCD abierta, genera una solución vecina con dicha ZCD cerrada. En otras palabras, son los vecinos con distancia Hamming 1 que tienen menos ZCD abiertas que el vecino original. Observar que si se tiene una solución de n zonas y a están abiertas, se generaran $a - 1$ zonas abiertas.

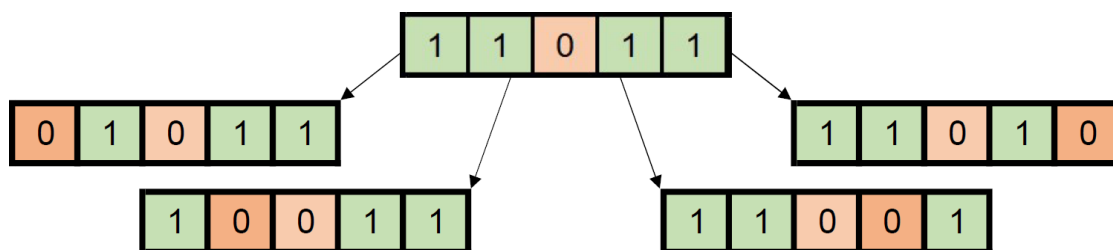


Figura 4.3: Vecindad generada con *vn_s1swap*.

La segunda vecindad llamada *vn_s2swap* (ver Figura 4.4), mantiene el número de ZCD abiertas reubicando una ZCD abierta en la solución base. Teniendo una ZCD abierta, se cierra y se ubica en alguna otra posición que estuviera previamente cerrada. Siguiendo el ejemplo anterior, si se tiene una solución de n zonas y a están abiertas, entonces $(n - a)$ estarán cerradas, habiendo generado $a * (n - a)$ vecinos con a zonas abiertas. Se puede apreciar que el tamaño de esta vecindad es más grande que la de *vn_s1swap*.

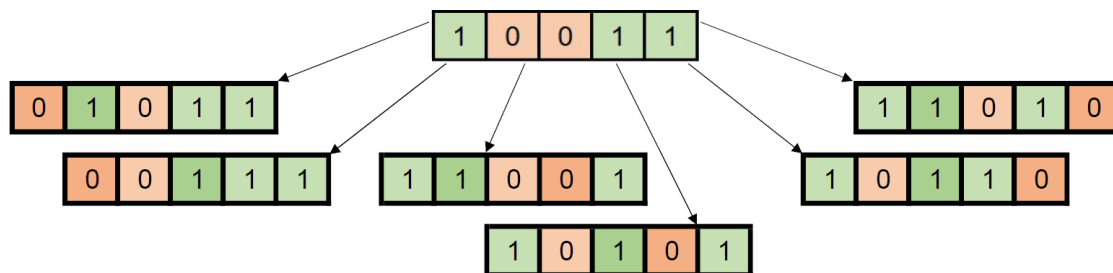


Figura 4.4: Vecindad generada con *vn_s2swap*.

Implementación

Parámetros y Datos de entrada

Para la implementación, se necesitan parámetros y datos de entrada. Los primeros son necesarios para determinar el comportamiento de la heurística, y los segundos son datos específicos de cada instancia del problema, de acuerdo al modelo del Capítulo 3.

Dada la naturaleza del problema y el modo que el algoritmo recorre el espacio de búsqueda, algunas soluciones generadas pueden no cumplir con todas las restricciones impuestas. Para guiar al algoritmo a una solución válida se introduce el parámetro *penalizacion1*. Este es un valor que se multiplica por cada restricción no cumplida que luego se suma al valor objetivo, dejando mejor posicionadas aquellas opciones que respeten más restricciones.

Una vez fijadas las ZCDs abiertas, se deben asignar los locales comerciales a estas ZCDs abiertas. A continuación se propone e implementa una solución heurística para este sub-problema, que es necesaria para calcular el valor objetivo de una solución del problema completo. Esta función de asignación toma fracciones de tiempo de la demanda de los locales comerciales y los asocia a una zona de carga con disponibilidad. El tamaño de dicho paquete es de *chunk size* unidades de tiempo. Más adelante en este capítulo se entrará en detalles de esta función.

Al inicio de la ejecución del algoritmo, la vecindad *vnslswap* genera muchas opciones, donde la mayoría mejoran la solución. A fin de evitar evaluarlas a todas, se optó por una estrategia de “first improve” para las primeras búsquedas en esta vecindad. Es *first stage steps* el parámetro que indica cuántas ocasiones consecutivas se aplicará “first improve” con la vecindad *vnslswap*. A esta serie de búsquedas donde se aplica “first improve” se le llamará *first stage*.

A diferencia de *vnslswap*, *vnsl2swap* potencialmente genera muchas más opciones y pocas de ellas, si no ninguna, mejoran la solución. El parámetro *swap stage mode* tiene tres modos: deshabilitar la vecindad para finalizar antes el algoritmo; habilitar la vecindad y aplicar “first improve” para tener mejoras graduales; y habilitar aplicando “best improve” si se dispone del tiempo para evaluar todas las opciones.

Además, se deben definir los parámetros del modelo, como son: tiempo mínimo de aparcamiento (mT), distancia máxima entre el local y la ZCD (mD), cantidad de zonas a abrir (N), cantidad de tipos de vehículos en el sistema (K) y tipo de vehículo admitido en las zonas (Gs).

Como datos de entrada se tienen, la ubicación y demanda de los locales comerciales por tipo de vehículo, las potenciales ubicaciones de las zonas de carga a abrir y capacidad de las zonas.

Esquema

La estructura principal del algoritmo tiene la siguiente forma:

Algorithm 1: Main()

```

1 datos ← cargarDatos()
2 procesados ← procesarDatos(datos)
3 solucion ← metaheuristica(procesados)
4 salida ← procesarSolucion(solucion)

```

Donde *cargarDatos*() se encarga de leer los parámetros y archivos de entrada. Luego, *procesarDatos*(...) calcula la matriz de distancia y genera el modelo para que la metaheurística opere, *metaheuristica*(...) ejecuta la heurística retornando la mejor solución hallada (de haber una) y por último *procesarSolucion*(...) transforma la solución de la metaheurística en formato de salida estándar, siendo posible visualizarlo en una herramienta o exportarlo para su posterior análisis.

Procedimiento de resolución

Entrando en el detalle del procedimiento de resolución basado en VND, se tiene el pseudocódigo que se muestra a continuación.

Algorithm 2: procedimiento_VND(*datos*)

```

1 best ← solucionInicial()
2 fin ← false
3 v ← 1                                     ▷ Se usa la primera vecindad
4 while !fin do
5   | vecinos ← vecindad(best, v)
6   | mejorVecino ← mejorar(best, vecinos)
7   | if esMejor(mejorVecino, best) then
8   |   | best ← mejorVecino
9   |   | v ← 1                               ▷ Si mejoró la solución se regresa a la primer vecindad
10  | else
11  |   | fin ← v == 2                       ▷ Tengo 2 vecindades
12  |   | v ← v + 1
13  | end
14 end

```

Aquí, *solucionInicial*() genera la solución inicial para la futura búsqueda por exploración del espacio factible. Luego, *vecindad*(...) genera las soluciones vecinas según la solución actual (*best*) y la vecindad que esté activa. Mientras que *mejorar*(...) devuelve alguna solución vecina de la vecindad siempre y cuando sea mejor que la solución actual.

Cálculo de la solución inicial

Para el cálculo de la solución inicial se comienza con todas las ZCD abiertas. Luego, se evalúa esa solución calculando el valor objetivo. Para ello se deberá asignar locales a las ZCD, donde, según el caso, puede que no haya demanda suficiente para todas las ZCD y que después de asignarla se encuentren ZCD sin local comercial asignado, siendo estas cerradas inmediatamente. El procedimiento de asignación es explicado más adelante en este capítulo.

Algorithm 3: *solucionInicial()*

```

1 solucion ← vector(True)
2 asignar(solucion)                                ▷ eventualmente cierra alguna entrada
3 return solucion

```

Avance en la búsqueda

En cada momento se quiere encontrar una nueva solución que mejore la solución actual. En el siguiente algoritmo se describe el método de búsqueda.

Algorithm 4: *mejorar(best, vecinos)*

```

1 firstImprove ← doFirstImprove()                ▷ FirstImprove o BestImprove
2 mejor ← best
3 for v in vecinos do
4   | asignar(v)
5   | if esMejor(v, mejor) then
6   |   | mejor ← v
7   |   | if firstImprove then
8   |   |   | break
9   |   | end
10  | end
11 end
12 return mejor

```

Siendo *doFirstImprove()* una función que consulta el parámetro *first stage steps* y evalúa si se debe hacer un “first improve” o un “best improve”. La función *esMejor(...)* simplemente es una forma de comparar dos soluciones y de determinar cuál es mejor.

Asignación

Para poder calcular el valor objetivo de la solución, es fundamental asignar los locales comerciales a las zonas de carga y descargas abiertas. *asignar(...)* es el procedimiento responsable de realizar esta tarea. En la búsqueda de la mejor solución, muchas opciones son consideradas, siendo necesario realizar esta operación múltiples veces en cada búsqueda. La eficacia del procedimiento de resolución depende en gran medida de este algoritmo en particular.

Algorithm 5: *asignar(solucion)*

```

1  $[K, Gs, mT, mD, N, chunksize] \leftarrow leerParametros()$ 
2  $zcds \leftarrow obtenerPotencialesZCD()$  ▷ Potenciales ZCD
3  $locales \leftarrow obtenerLocalesComerciales()$  ▷ Distintos Comercios
4  $asignaciones \leftarrow empty()$ 
5  $zcdAux \leftarrow prepararZCD(zcds, solucion)$ 
6 for  $k \leftarrow K$  to 0 do
7    $localesAux \leftarrow prepararLocales(locales, k)$ 
8   while  $!localesAux.empty()$  do
9     for  $local \in localesAux$  do
10       $zcd \leftarrow mejorZcd(local, zcdAux)$ 
11       $asignacion \leftarrow asociar(local, zcd)$ 
12       $asignaciones.add(asignacion)$ 
13       $actualizarLocal(local)$ 
14       $actualizarZcd(zcd)$ 
15      if  $demandaSatisfecha(local)$  then
16         $localesAux.remove(local)$ 
17      end
18      if  $capacidadSaturada(zcd)$  then
19         $zcdAux.remove(zcd)$ 
20      end
21    end
22  end
23 end
24  $procesarAsignaciones(asignaciones, solucion)$ 

```

Este algoritmo tiene características de un método Greedy, ya que, toma en cada paso la mejor opción, con la esperanza de obtener buenos resultados. Como se adelantó previamente en este capítulo, el tamaño de estos pasos está dado por el parámetro *chunk size*, siguiendo una analogía de trayecto, la distancia a recorrer lo impone la instancia del problema con su demanda total. Siendo la demanda invariante en un escenario, se puede ajustar la granularidad con la que se asigna la demanda con *chunk size*. Es por esto que para valores más bajos, la asignación da más pasos, pero es más precisa a costa de un mayor tiempo de procesamiento, en comparación con valores más altos donde la asignación es más rápida pese a alejarse

del óptimo.

Su construcción se basa en algunas funciones auxiliares, tales como:

- *leerParametros()* retorna los distintos parámetros de la ejecución.
- *obtenerPotencialesZCD()* y *obtenerLocalesComerciales()* se encargan de recuperar la información de las zonas de carga y los locales comerciales de la memoria del sistema.
- *prepararZCD(...)* calcula, basándose en la solución evaluada y las ZCD del sistema, con qué ZCD se debe trabajar, debido a que *solución* es quien almacena las ZCD disponibles.
- *prepararLocales(...)*, devuelve todos los locales comerciales que tengan una demanda de tipo *k*.
- *mejorZcd(...)*, busca la mejor ZCD disponible teniendo en cuenta la distancia a *local*, la demanda, y los parámetros *mT*, *mD*, *chunk size*, además de la capacidad de las ZCDs, que está siendo ocupada en cada iteración. Cuando no es posible encontrar una ZCD, se devuelve un error y esta solución es descartada. Esto significa que la solución (ZCD abiertas) no es factible, o que el procedimiento heurístico no pudo encontrar una asignación factible.
- *asociar(...)* crea un elemento *asignación*, teniendo en cuenta el tipo de vehículo, la demanda, y la distancia entre las coordenadas.
- *actualizarLocal(...)* modifica el *local* restándole la demanda insatisfecha.
- *actualizarZcd(...)* modifica la *zcd* para marcar como cubierta cierta capacidad.
- *demandaSatisfecha(...)* retorna *true* si el *local* tiene toda su demanda satisfecha, entonces puede ser removido de la lista de pendientes.
- *capacidadSaturada(...)* retorna *true* si la *zcd* cuenta con una capacidad disponible menor a *mT*, entonces puede ser removida de las ZCD disponibles, ya que, no hay manera de asignar esa fracción a un local.
- *procesarAsignaciones(...)* agrupa las asignaciones que corresponden al mismo par ZCD-Local, calcula el valor objetivo y se lo asigna a *solución*.

En este capítulo se han visto los diferentes componentes necesarios para resolver el problema planteado en el Capítulo 3, con base en la metaheurística de VND. El Capítulo 5 abordará los detalles de las distintas fases de la implementación.

Capítulo 5

Arquitectura de la solución

En este capítulo se describe la arquitectura de la solución propuesta para abordar el problema de localización y asignación de zonas de carga y descarga (ZCD) a locales comerciales en áreas urbanas mediante la implementación de la metaheurística VND vista en el Capítulo 4.

En las diferentes etapas del desarrollo de la solución se contó con diferentes datos. Para cada etapa se describe: el objetivo de la etapa, los datos con los que se contó y las herramientas o tecnologías utilizadas para cumplir el objetivo.

Es importante tener en cuenta que los datos disponibles están almacenados en archivos espaciales, conocidos como “shapefile”. Estos son bases de datos capaces de almacenar información espacial, como lo son las coordenadas de los locales comerciales o potenciales zonas de carga y descarga. Además pueden contener otra información, por ejemplo, la demanda del local. Dichos archivos son la principal vía de acceso a datos, así como de salida, de la misma manera que son el formato de visualización y comunicación con el usuario.

5.1. Etapa 1 - Prototipo

El objetivo de la primera etapa es validar el modelo matemático propuesto en el Capítulo 3, mediante un caso de estudio reducido, que consta de solamente nueve manzanas. Los locales comerciales son la totalidad de padrones en el área seleccionada, sin discriminar las viviendas de los comercios. Por otro lado, las posibles zonas carga y descarga son las del caso de estudio original, es decir, la zona comercial de la Ciudad Vieja. Como no se cuenta con una demanda por locales, la misma es estimada. Para esta etapa se utilizó una herramienta de extracción, transformación y carga de datos (ETL - Extract, Transform and Load) y un paquete de software (kit) de programación lineal para resolver el modelo de forma exacta, además de un lenguaje de programación imperativo para generar los valores de la demanda en el formato de entrada del kit antes mencionado.

Para crear la matriz de distancia entre los locales y las zonas de carga y descarga se utiliza GeoKettle¹, una herramienta de ETL, capaz de integrar distintas fuentes, realizar operaciones y trabajar con bases de datos geospaciales. Con GeoKettle se puede acceder fácilmente a los datos de los shapefiles, además de operar con estos; permite crear una matriz de distancia en pocos pasos.

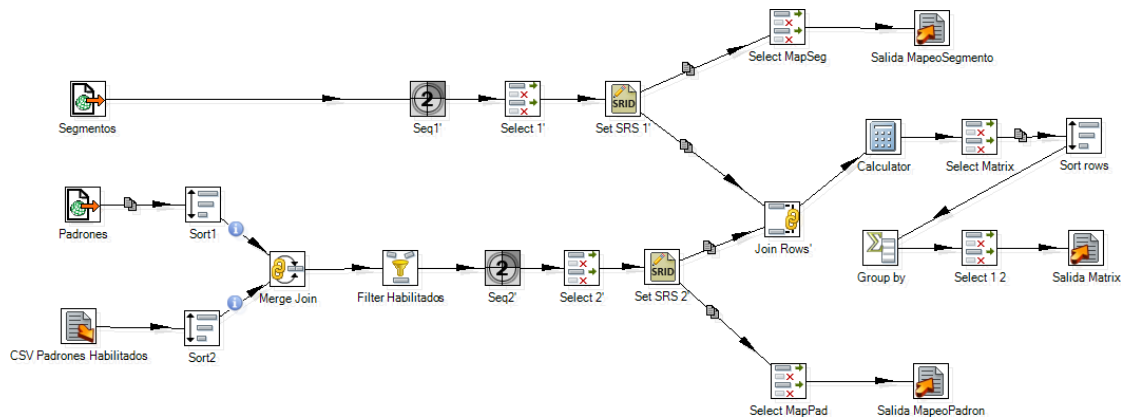


Figura 5.1: GeoKettle - Procesamiento de los datos de entrada

Se presenta un esquema del flujo de transformación de datos (ver Figura 5.1) para crear la matriz de distancia. Este consta de los siguientes de pasos:

- Se comienza leyendo los datos de los locales comerciales y potenciales zonas de carga y descarga.
- De contar con la información, se filtran los padrones del escenario que no están habilitados, quedándose con los habilitados.
- Se calcula un producto cartesiano *ZCD - Local Comercial* para hacer un cálculo de las distancias.
- Por último se da formato a los datos para generar la matriz.

Una vez creada la matriz de distancia, es necesario generar la demanda y procesar todos los datos con el fin de realizar la integración con el kit de programación lineal. Para esta tarea se utiliza GNU Octave², un programa y lenguaje de programación para realizar cálculos numéricos.

La demanda de los locales comerciales es generada con una distribución Normal de media 15 unidades y desviación estándar 5. Además, para introducir casos particulares, se le aumentó a los locales 15 unidades de demanda con una probabilidad del 15%. Mientras que la capacidad de las ZCDs es fija en 150 unidades, un valor arbitrario teniendo en cuenta la demanda de los locales y la cantidad máxima de zonas.

Además Octave se encarga de leer la matriz de distancia, y generar un archivo de datos con el formato adecuado para ser resuelto de forma exacta.

¹https://live.osgeo.org/archive/10.5/es/overview/geokettle_overview.html

²<http://gnu.org/software/octave/>

Para resolver el modelo se utiliza el paquete de optimización GLPK ³, que utiliza como formato de entrada el lenguaje de modelado MathProg. El modelo matemático utilizado es el descrito en el Capítulo 3 y los datos son los generados en el paso anterior. De encontrar una solución óptima, a través de GLPK se generará un archivo de texto de salida, donde cada línea tiene la información de asignación: local, ZCD, demanda asignada y tipo vehículo.

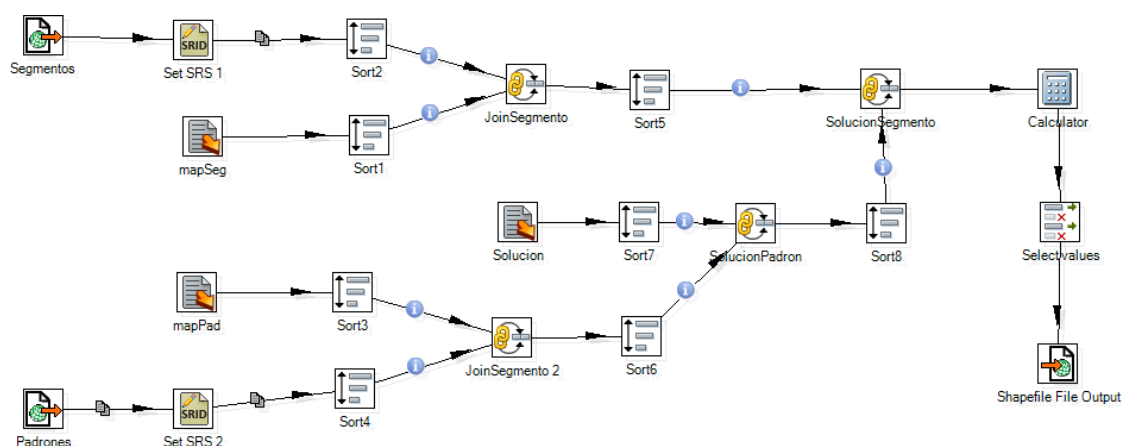


Figura 5.2: GeoKettle - Generando los datos de salida

Finalmente, se recurre nuevamente a GeoKettle (ver Figura 5.2) para procesar la salida de GLPK y generar un nuevo shapefile con segmentos de asignación (Local-ZCD). Es necesario volver a cargar los shapefiles originales para recuperar los datos espaciales, se combinan con el archivo de salida de la ejecución de GLPK y se genera un nuevo shapefile con la información de las asignaciones. Este tipo de archivos permite su importación en software de información geográfica como QGIS ⁴ para su visualización (ver Figura 5.3).

³<http://gnu.org/software/glpk/>

⁴<http://qgis.org/>

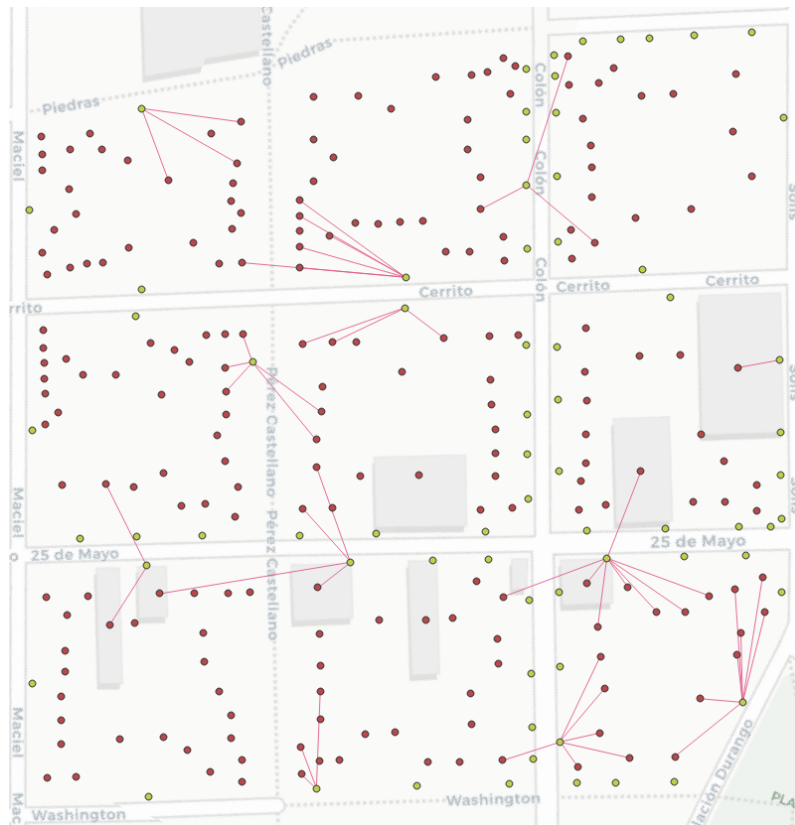


Figura 5.3: QGIS - Visualización del archivo de salida

Además, GeoKettle permite encadenar una secuencia de tareas (ver Figura 5.4), ahorrándose la ejecución manual de cada una de las fases desarrolladas en esta etapa, algo muy útil cuando se necesita reiterar los experimentos.

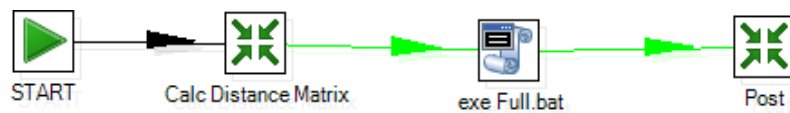


Figura 5.4: GeoKettle - Secuencia de tareas

5.2. Etapa 2 - Implementación de la solución

En esta etapa se consideran diferentes escenarios, desde nueve manzanas hasta el caso de estudio completo. Utilizando tanto demanda real como demanda estimada para realizar diferentes experimentos. Una vez validado el modelo en la Etapa 1, se buscó mantener una arquitectura simple. Para eso fue necesario de un lenguaje de programación que permita desarrollar todas las funcionalidades cubiertas antes por Geokettle, Octave y QGIS. Sumado a la necesidad de sustituir GLPK por la heurística diseñada en el Capítulo 4. Para esta tarea, Java (en su versión 11) es el lenguaje de programación seleccionado, por la experiencia previa en su uso, además de ofrecer todas las bibliotecas necesarias para abordar las tareas necesarias. El conjunto de tecnologías utilizado es:

- *geoTools* (versión 23.0): utilizado para todo lo relacionado con el manejo de datos geoespaciales.
- *spring-boot* (versión 2.3.0.RELEASE): es un framework web que disponibiliza la aplicación en la red.
- *vaadin* (versión 14): es una biblioteca de interfaz gráfica para web.
- *leafletJS* (versión 1.7): ofrece mapas interactivos.

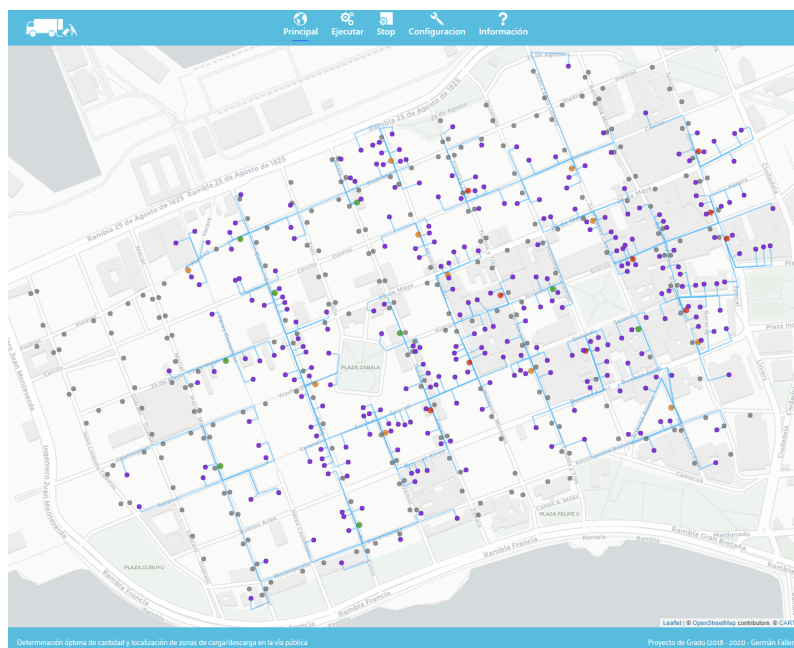


Figura 5.5: Vista previa de la herramienta

Gracias a estas bibliotecas, es posible construir una herramienta (ver Figura 5.5) donde el usuario puede: cargar distintos escenarios al sistema, configurar los distintos parámetros del algoritmo, ejecutar la heurística, ver los resultados, además de exportar la solución en un shapefile para un posterior análisis. Más detalles de la herramienta se brindarán en el Capítulo 7.

Inicialmente en esta etapa la asignación de zonas a locales, se decidió resolver de forma exacta con GLPK. Para esto fue necesario contar con dos bibliotecas extras, *de.xypron.linopt/linopt* (compilación propia) que permite escribir en Java el problema de asignación y *org.gnu.glpk/glpk-java* (version 1.12) que hace de interfaz con GLPK. Esta versión fue descartada, ya que las llamadas a GLPK debían ser secuenciadas (una al terminar la otra), generando un cuello de botella, dado que cada llamada era relativamente lenta. Al escribir el método *asignar(...)* (Algoritmo 5) visto en el Capítulo 4 se pudo observar una mejora sustancial en los tiempos de ejecución, con el agregado de que se podía hacer distintas ejecuciones de *asignar(...)* en paralelo a costa de perder precisión en la asignación.

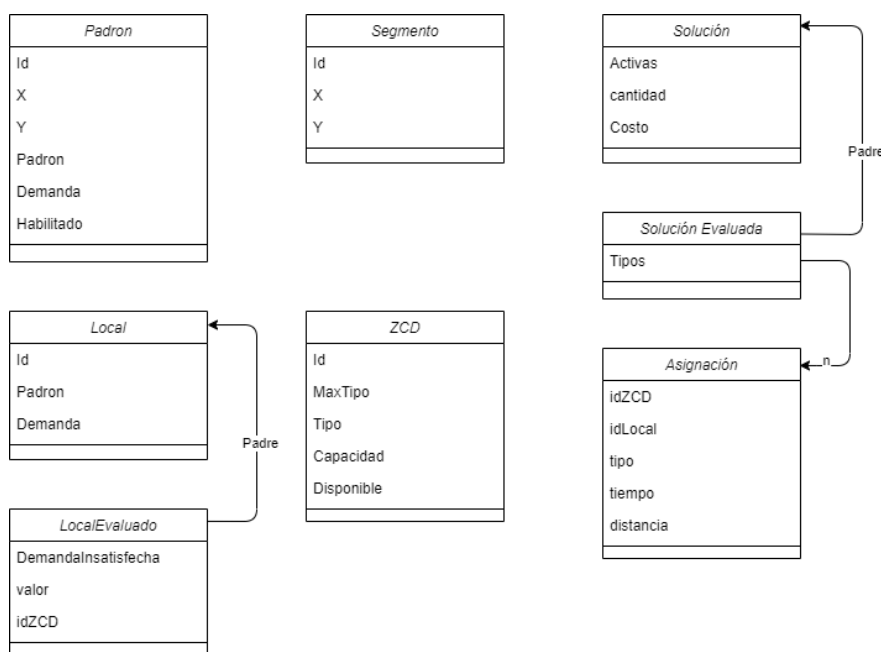


Figura 5.6: Modelo - Diagrama de clases

La solución implementada cuenta con cuatro módulos y el modelo de datos (ver Figura 5.6). El módulo *loader* responsable de leer y procesar la fuente de datos; el módulo *distanceMatrix* que almacena la matriz de distancia y ofrece métodos de fácil acceso dado un local y una zona; y los módulos *heuristic* y *evaluator* donde el primero es el corazón de la búsqueda basado en el Algoritmo 2 y el segundo gestiona las asignaciones según el Algoritmo 5

En el modelo de datos se puede notar clases similares entre sí. La razón de esto es que, tanto *Padron* como *Segmento* almacenan los datos originales e información inmutable a lo largo de toda la ejecución. Mientras que *Local* es una versión simplificada de *Padron*, ya que será un objeto duplicado múltiples veces, manteniendo la mínima información necesaria para el cálculo. Luego, *LocalEvaluado* y *ZCD* son objetos intermedios para calcular la asignación; además de descartables, tienen atributos que solo son necesarios al momento de hacer el cálculo, luego estos son

prescindibles. Asimismo se tiene la clase *Solución*, esta es gestionada por *heuristic*, la cual decide qué zonas *activas* tendrá esta solución. El *costo* aunque no se conoce en un principio, es necesario a este nivel para hacer rápidas comparaciones en *heuristic* sin entrar en detalles. *Solución Evaluada* tiene un conjunto de *Asignaciones* porque es el resultado de calcular las asignaciones de una solución; además, referencia la solución original para poder corregir el costo.

5.3. Etapa 3 - Mejora del Desempeño

Luego de desarrollada toda la solución, se incluyen los datos reales del caso de estudio, y se comparan los resultados frente a CPLEX⁵, un software comercial de optimización. Debido a los costos de licenciamiento de este software de optimización, se apunta a que el usuario pueda contar con una heurística, lo más eficaz y eficiente posible.

Con el fin de disminuir los tiempos de búsqueda de la heurística se decide implementar parte de la solución con Rust⁶, un lenguaje de programación conocido, entre otras ventajas, por su rendimiento. Una comparativa Java - CPLEX y Java - Rust se puede encontrar en el Capítulo 6.

A pesar de que la sección que más impacta en los tiempos de ejecución es el Algoritmo 5 (*asignar(...)*), se codifica toda la heurística con el lenguaje Rust (llámese Backend), dejando implementado en Java lo referente a la interfaz gráfica (llámese Frontend), comunicando ambos sistemas via API Rest. Para desarrollar el backend se utiliza:

- *Rust* (versión 1.45): lenguaje de programación base.
- *rocket* (versión 0.4.5): framework web que disponibiliza la API Rest.
- *shapefile* (versión 0.2.0): relacionado con manejo de datos geoespaciales.
- *rayon* (versión 1.3.0): biblioteca para ejecuciones en paralelo.

Por otra parte, el frontend mantiene las mismas dependencias pero con menos responsabilidades, siendo necesario implementar un conector para consumir los datos del backend.

En esta última versión se logra una mejora de rendimiento destacable, aun así, los resultados en el valor objetivo no varían (o las variaciones son marginales) pues el recorrido en el espacio de búsqueda se mantiene y es determinista.

En este capítulo se presentaron las distintas herramientas y tecnologías utilizadas a lo largo de todo el proyecto. En el Capítulo 6 se describirán distintas ejecuciones, comparaciones y análisis de los resultados obtenidos en las distintas etapas del desarrollo de la heurística.

⁵<https://www.ibm.com/analytics/cplex-optimizer>

⁶<https://www.rust-lang.org/>

Capítulo 6

Experimentos computacionales

El objetivo de este capítulo es presentar los resultados de las ejecuciones que fueron realizadas en las diferentes etapas del proyecto. Todas las ejecuciones fueron realizadas en una computadora personal con un procesador Intel Core i7-9700K 3.60 GHz, 64 bit, 64 GB RAM, con Windows 10 Pro (20H2).

Para cada ejecución, se presenta: el objetivo de la prueba, el escenario, la fuente de los datos, resultados y posterior análisis.

6.1. Validación del modelo

La validación del modelo consiste en resolver una instancia del problema con GLPK y verificar que el resultado de la ejecución es coherente con el esperado. En este caso se tomó un área de cuatro manzanas de Ciudad Vieja, Montevideo (ver Figura 6.1).



Figura 6.1: Escenario 1 - cuatro manzanas
Puntos verdes: posibles ZCD - Puntos violetas: Locales comerciales

La matriz de distancia se calculó a partir de los datos geo-espaciales del caso de estudio en el área limitada, usando para el cálculo la distancia euclídea.

6.1.1. Tipo único de vehículo

Una primera prueba consiste en evaluar un escenario con un único tipo de vehículo. En esta, la demanda de los Locales Comerciales fue generada con una distribución Normal de media 15 unidades temporales y desviación estándar 5. Además, para introducir casos anormales, con una probabilidad del 15 % se le aumentó a los locales 15 unidades de demanda (ver Tabla 6.1).

La capacidad de las ZCDs fue generada a partir de la demanda, calculando la demanda agregada (suma de todas las demandas), y a este valor se lo dividió entre 10. Visto de otro modo, si no se contaran con las restricciones del modelo, bastarían 10 ZCDs para satisfacer toda la demanda. Esto resulta en una capacidad constante de 70 unidades para todas las ZCDs.

En el modelo se utilizaron los siguientes parámetros:

- Cantidad de ZCD: 21
- Cantidad de Locales: 44
- Cantidad de tipos de vehículos: 1
- ZCD a abrir: 15
- Distancia máxima: 115 metros
- Tiempo mínimo de asignación: 10 unidades temporales

Local	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
Demanda	16	18	20	0	0	16	0	30	12	14	18	30	0	14	11	14	20	19	24	17	13	0
Local	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44
Demanda	31	36	30	24	21	13	13	14	11	19	13	11	16	38	15	20	18	16	0	11	16	16

Tabla 6.1: Escenario 1 - Demanda generada

Luego de la ejecución (ver Anexo Tabla A.1 y Tabla A.2) se observó:

- La cantidad de ZCD utilizadas fue de 15.
- La demanda de todos los locales fue atendida.
- En ningún caso la capacidad de la ZCD fue excedida.
- La distancia máxima de asignación fue de 114 metros.
- El tiempo mínimo asignado fue de 10 unidades temporales.

6.1.2. Dos tipos de Vehículos

La segunda prueba consiste en evaluar un escenario con dos tipos de vehículos.

Se recuerda que los tipos de vehículos respetan una distribución jerárquica; a mayor tipo, más grande es el vehículo.

Para contemplar a los tipos de vehículos se utilizó la misma área, pero, se introdujeron cambios en los parámetros del modelo, se generó una nueva demanda para los locales y se calculó una nueva capacidad para las ZCD.

Los cambios para esta ejecución fueron:

- Cantidad de tipos de vehículos: 2
- ZCDs que aceptan el vehículo tipo 2: [1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21]
- ZCDs que solo aceptan el vehículo tipo 1: [2, 4, 6, 8, 10, 12, 14, 16, 18, 20]
- Demanda generada utilizando media 20, desviación estándar 10. En esta oportunidad, se tomó el cociente entre el valor generado y el tipo de demanda. Los vehículos tipo 1 son más demandados que los tipo 2 (ver Tabla 6.2).
- Capacidad de la ZCD fija: 136

Local	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
Dem(1)	18	18	12	12	46	0	38	30	44	14	30	12	52	62	28	12	26	22	0	0	50	16
Dem(2)	0	0	12	15	18	0	25	11	11	15	13	28	16	11	0	11	11	19	11	10	0	12
Local	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44
Dem(1)	52	30	0	0	16	14	66	50	26	0	14	22	20	30	22	20	0	28	0	12	12	26
Dem(2)	0	26	0	0	11	0	0	11	10	0	0	10	0	0	0	0	0	16	0	18	0	14

Tabla 6.2: Escenario 1 - Demanda generada 2 niveles

Luego de la ejecución (ver Anexo Tabla A.3) se observó:

- Se respetan las restricciones mencionadas en la Subsección 6.1.1.
- No hay asignación de demanda tipo 2 a ZCD que solo acepte tipo 1.

En conclusión, los resultados observados cumplieron con las restricciones establecidas en el problema, mostrando coherencia entre el resultado esperado y el obtenido.

6.2. Evaluación de la metaheurística

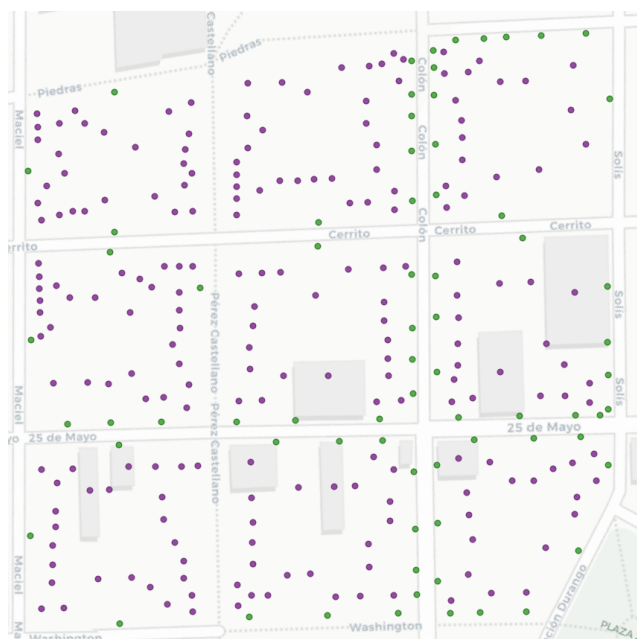


Figura 6.2: Escenario 2 - nueve manzanas
Puntos verdes: posibles ZCD - Puntos violetas: Locales comerciales

La evaluación de la metaheurística se divide en dos partes: la validación, que consiste en verificar que los resultados son coherentes; y la comparación, donde se compara con un método de solución exacta para un mismo escenario. En esta instancia se tomó un área de nueve manzanas de Ciudad Vieja, Montevideo (ver Figura 6.2).

La matriz de distancia se calculó a partir de los datos geo-espaciales del caso de estudio en el área limitada.

6.2.1. Validación

La validación de la metaheurística consiste en verificar que el resultado de la ejecución es coherente con el esperado.

Se utilizó el esquema para dos tipos de vehículos; el tipo de vehículo que acepta cada Zona de Carga fue elegido aleatoriamente. La instancia resulta en 36 zonas tipo 1 y 35 zonas tipo 2 (ver Tabla 6.3).

Tipo 1	1, 5, 9, 10, 12, 14, 15, 19, 20, 22, 24, 25, 27, 28, 32, 34, 36, 38, 40, 41, 45, 46, 49, 50, 52, 54, 55, 56, 60, 61, 62, 63, 64, 67, 69, 71
Tipo 2	2, 3, 4, 6, 7, 8, 11, 13, 16, 17, 18, 21, 23, 26, 29, 30, 31, 33, 35, 37, 39, 42, 43, 44, 47, 48, 51, 53, 57, 58, 59, 65, 66, 68, 70

Tabla 6.3: Escenario 2 - Tipos de Vehículos por Zona

La demanda de los locales comerciales para cada tipo de vehículo fue generada con una distribución Normal de media 20 y desviación estándar 10. A su vez, se tomó el cociente entre este valor y el tipo de demanda, con el objetivo de generar menos demanda para los tipos de vehículo mayores. Además, para introducir casos anormales, con una probabilidad del 15% se le aumentó a los locales 15 unidades de demanda.

Por otro lado, la capacidad de las ZCDs fue generada a partir de la demanda, calculando la demanda agregada (suma de todas las demandas), y a este valor se lo dividió entre 20. Visto de otro modo, si no se contara con las restricciones del modelo, bastarían 20 ZCDs para satisfacer toda la demanda. Esto implica una capacidad constante de 555 unidades para todas las ZCDs.

En la configuración de la metaheurística se utilizaron los siguientes parámetros:

- Cantidad de ZCD: 71
- Cantidad de Locales: 213
- Cantidad de tipos de vehículos: 2
- ZCD a abrir: 25
- Distancia máxima: 115 metros
- Tiempo mínimo de asignación: 10 unidades temporales

- *first stage steps*: 0
- *swap stage mode*: “*first improve*”

Analizando los resultados (ver Tabla A.4 y Tabla A.5) se observó:

- La cantidad de ZCD utilizadas fue de 25.
- La demanda de todos los locales fue atendida.
- En ningún caso la capacidad de la ZCD fue excedida.
- La distancia máxima de asignación fue de 112 metros.
- El tiempo mínimo asignado fue de 10 unidades temporales.
- No hay asignación de demanda tipo 2 a ZCD que solo acepte tipo 1.

En conclusión, los resultados observados cumplieron con las restricciones establecidas en el problema, mostrando coherencia entre el resultado esperado y el obtenido.

6.2.2. Comparación contra método exacto

La comparación de la metaheurística frente a un método exacto sirve tanto para evaluar la calidad de las soluciones obtenidas, así como para comparar tiempos de ejecución. En esta oportunidad CPLEX 12,10,0,0 fue el software utilizado para las pruebas con método exacto. Se desarrolló el modelo en OLP (lenguaje de programación lineal) para ser ejecutado en el mismo ambiente.

Se planteó un plan de pruebas, tomando distintas configuraciones de escenario para el área dada (ver Figura 6.2). Estas configuraciones son combinaciones de las distintas variantes descritas a continuación:

- dos grupos de Zonas de Carga y Descarga:
 - habilitando que todas las ZCD aceptan vehículos tipo 2.
 - el 50 % de las ZCD aceptan solo vehículos tipo 1 y el resto aceptan vehículos tipo 2.
- cuatro formas de generar la demanda:
 - distribución Normal de media 15 y desviación estándar 2.
 - distribución Normal de media 15 y desviación estándar 5.
 - distribución Normal de media 20 y desviación estándar 5.
 - distribución Normal de media 20 y desviación estándar 10.
- dos formas de generar la capacidad de las ZCD:
 - dividiendo la demanda agregada entre 15.
 - dividiendo la demanda agregada entre 20.
- diez cantidades de ZCD a abrir; en esta variante se buscan soluciones desde 20 zonas habilitadas hasta 29 zonas utilizadas inclusive.

De esta manera, se utilizan todas las combinaciones de las variantes presentadas, produciendo 160 configuraciones. Para cada una de ellas se generaron 10 instancias utilizando semillas distintas, obteniendo en total 1.600 instancias diferentes del problema. Cabe la aclaración de que la aleatoriedad se encuentra en la generación de los datos de entrada y no en la metaheurística.

En el caso del método exacto se configuró para que ejecute un máximo de 180 segundos (3 minutos). En dicho tiempo, devolverá (si existe) la mejor solución encontrada hasta ese momento.

Una vez que se realizaron las pruebas y se procesaron las salidas de las ejecuciones, se prosiguió a analizar los resultados obtenidos (ver Tabla 6.4). Se destacan tres conjuntos de pruebas según su característica: la primera a analizar (Caso 1), es aquella que la metaheurística no pudo resolver, es decir, no pudo determinar una solución para el problema. La siguiente a analizar (Caso 2), es la que resultó más desafiante para la metaheurística. Por último, un caso menos restrictivo (Caso 3), para analizar condiciones más holgadas.

	Caso 1	Caso 2	Caso 3
Cantidad de ZCD a abrir	20	21	24
Cociente Capacidad	20	20	15
# Casos	80	80	80
# Ejecuciones exacto con tiempo excedido	4	44	0
# Ejecuciones exacto sin solución	76	0	0
Exacto GAP Máximo	3,32 %	0,85 %	0,13 %
Exacto GAP Promedio	1,54 %	0,08 %	0,03 %
Exacto Tiempo Máximo (s)	180	183	16,7
Exacto Tiempo Promedio (s)	180	133	7,7
# Ejecuciones metaheurística sin solución	80	0	0
Metaheurística GAP Máximo	N/A	5,23 %	0,96 %
Metaheurística GAP Promedio	N/A	3,37 %	0,41 %
Metaheurística Tiempo Máximo (s)	N/A	3,1	2,5
Metaheurística Tiempo Promedio (s)	N/A	1,77	1,4

Tabla 6.4: Resumen 1.600 Casos

Se define como GAP a la diferencia relativa porcentual del valor objetivo de una solución encontrada con una cota inferior. Para el exacto, la cota es calculada por el software comercial. Para la metaheurística, la cota es la solución del exacto.

El caso que la metaheurística no pudo resolver sucede cuando, se busca abrir 20 ZCDs con la capacidad que resulta del cociente de la demanda agregada y 20, independientemente del resto de variantes, dejando así, 80 ejecuciones sin éxito. Mientras que el método exacto, logró resultados sobre 4 de las 80 ejecuciones, consumiéndose la ventana de tiempo (180s) asignada y logrando un GAP máximo de 3,3 % (ver Tabla A.6).

Con respecto al caso que busca abrir 21 ZCDs, con la capacidad que resulta del cociente de la demanda agregada y 20, se visualiza que ambos métodos se desempeñaron correctamente (ver Tabla A.7). El método exacto consumió su cuota de tiempo en el 55 % de las ejecuciones, logrando en el peor caso un GAP de 0,85 %. Mientras que la metaheurística mantuvo una distancia de entre 2 % y 5 % de la mejor solución encontrada por el exacto dando resultados en entre 1 segundo y 3 segundos, esto es, como máximo en un 10 % del tiempo utilizado por el método exacto.

Por último, se representa un caso más relajado. El caso que busca abrir 24 ZCDs, con la capacidad que resulta del cociente de la demanda agregada y 15. Ambos métodos ejecutaron holgadamente (ver Tabla A.8). El método exacto manejó tiempos de entre 3,4 y 16,7 segundos, hallando la solución óptima en todos los casos. Mientras que la metaheurística ejecutó entre 0,8 y 2,5 segundos, tomando en el peor caso, un 40 % del tiempo que le tomó al método exacto para un mismo escenario, resultando en una distancia al valor óptimo de en promedio 0,4 % y peor caso 1 %.

Como conclusión de la evaluación de la metaheurística, se puede decir que se comporta como es esperado en este problema. En cuanto al desempeño, se mantiene en el 5 % del mejor valor encontrado por el exacto en el peor caso, lo cual es aceptable en este tipo de aplicaciones. Con respecto a los tiempos de ejecución ha mostrado valores alentadores.

6.3. Cambio de plataforma

El objetivo de esta sección es el de justificar, mediante pruebas de rendimiento, el cambio de plataforma que se hizo de Java a Rust. Para esto se utilizó el escenario y el plan de pruebas de la Sección 6.2.2 y se compararon ambas versiones.

Luego de la ejecución de los 1.600 casos, se aprecian diferencias en el rendimiento de las versiones, tardando para la prueba completa: 33 minutos la versión Rust, 120 minutos (cuatro veces más) la versión Java, y 560 minutos (dieciocho veces más) el método exacto.

Si se analiza más en detalle, desglosado por cantidad de ZCD a abrir (ver Figura 6.3), se observa cómo ambas versiones mantienen el margen de rendimiento entre ellas en todo momento. Mientras que el método exacto, conforme se exige más el problema, aumenta exponencialmente los tiempos de ejecución.

Además de la ya mencionada mejora en promedio de rendimiento que se obtiene con el cambio de plataforma, cabe mencionar que las mejoras caso a caso varían entre dos y siete veces, independientemente de la configuración de la prueba.

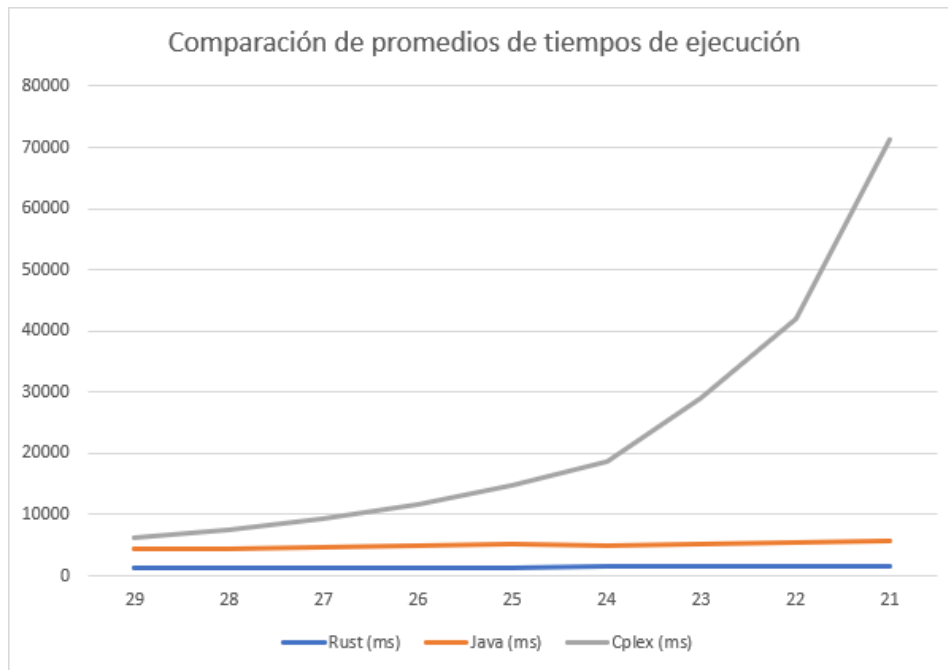


Figura 6.3: Comparación rendimiento - Cplex Java Rust
 Tiempos según cantidad de ZCD a abrir
 Menos es mejor, tiempos en ms

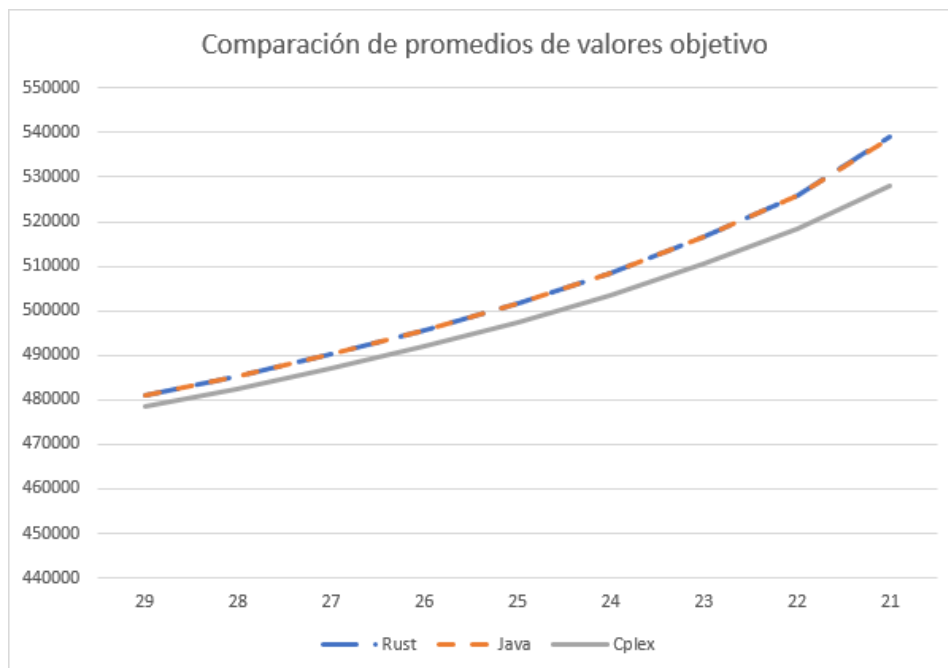


Figura 6.4: Comparación Valor Objetivo - Cplex Java Rust
 Valor objetivo según cantidad de ZCD a abrir
 Menos es mejor

Por otro lado, analizando los valores objetivos (ver Figura 6.4) no se aprecian diferencias entre ambas soluciones. Mientras que en comparación con el método exacto se mantiene un margen de entre 0,5% y 2% en el promedio de las ejecuciones.

Si bien el cambio de plataforma trae una mejora lineal en los tiempos de ejecución, en escenarios grandes la mejora no es significativa frente al método exacto. En casos sencillos ofrece soluciones en el entorno de 1 segundo. Esto último es interesante para un tomador de decisiones que debe realizar distintas combinaciones de parámetros y analizar cada caso.

6.4. Caso de estudio

En este proyecto el caso de estudio se ubica en el barrio Ciudad Vieja de la ciudad de Montevideo, Uruguay. Este cuenta con 290 locales comerciales y 286 posibles Zonas de Carga y Descarga.

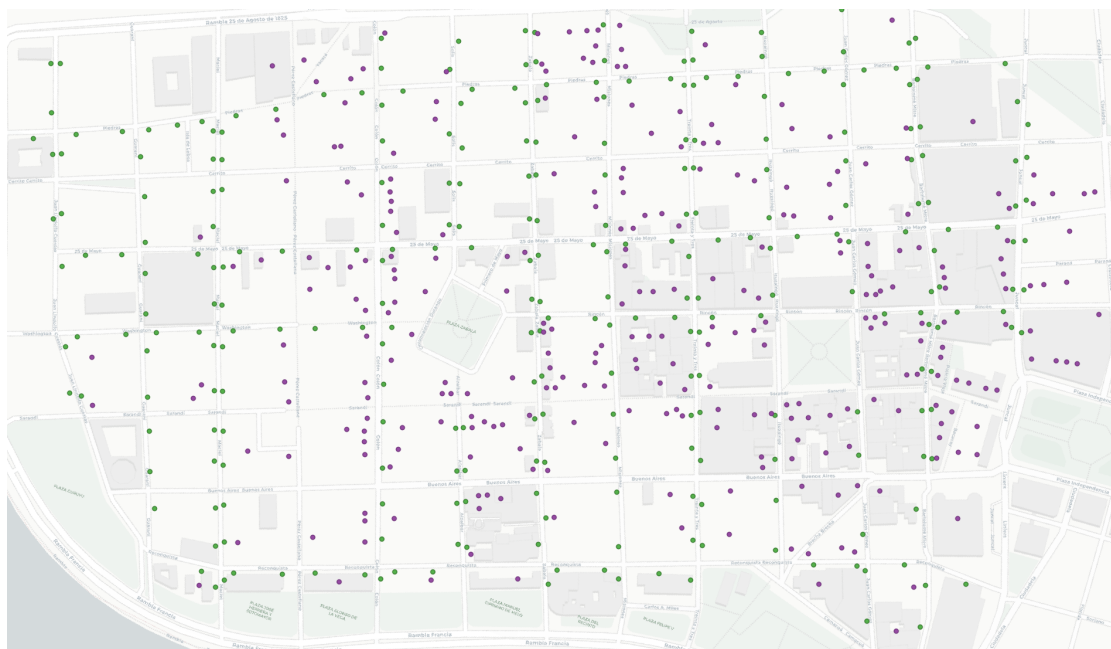


Figura 6.5: Escenario 3 - Caso de Estudio
Puntos verdes: posibles ZCD - Puntos violetas: Locales comerciales

La matriz de distancia se calculó a partir de los datos geo-espaciales del caso de estudio en el área limitada, usando para el cálculo la distancia Manhattan.

La demanda de los locales fue proporcionada por el Instituto Nacional de Logística¹, siendo un estimado del tiempo, en minutos, que necesitan los vehículos de carga estar aparcados en cada local.

En este escenario, se maneja solo un tipo de vehículo, ya que los datos no brindan más información. Se configuró la capacidad de las Zonas de Carga para que estén disponibles 24h (1440m).

¹<http://inalog.org.uy/>

6.4.1. Variación de la Demanda

Se dispone realizar variaciones en la demanda real, para observar el comportamiento de la metaheurística. Se generan tres escenarios: el primero tiene en cuenta 100 % de la demanda real, el segundo 90 % de la demanda real y por último 65 % de la demanda real. Para demanda 100 % se busca abrir como máximo 40 Zonas de carga y descarga; luego, para demanda 90 %, se busca abrir 35 ZCD; finalmente, para demanda 65 %, se busca abrir 30 ZCD. La cantidad de ZCD a abrir se estima en base a la demanda de los locales y la disponibilidad de la ZCD (24h)

Tiempo (s)				
Demanda	100 %	90 %	65 %	
Exacto	66	180	113	
Metaheurística	94	64	37	
T_e/T_m	1,42	0,35	0,32	

Tabla 6.5: Escenario 3 - Tiempo

Valor Obj.				
Demanda	100 %	90 %	65 %	
Exacto	3.561.864	3.629.326	2.647.422	
Metaheurística	3.775.603	3.804.195	2.681.859	
$(m - e)/e$	0,0600	0,0482	0,0130	

Tabla 6.6: Escenario 3 - Valor Obj.

Se observa cómo a mayor demanda la metaheurística tarda más en devolver un resultado (ver Tabla 6.5), esto se explica por que hay menos espacios de tiempo libre en las ZCD. En comparación con el método exacto, en general la metaheurística es tres veces más rápida. Excepcionalmente el método exacto se comportó muy bien cuando el problema está más acotado. En la tabla (ver Tabla 6.6) se relaciona el valor objetivo con las demandas, comparando el método exacto contra la metaheurística y se muestra la proporción del error. En esta se aprecia cómo el error baja de 0,06 a 0,013 cuando la demanda es más holgada.

6.4.2. Variación en la etapa inicial de la heurística

La heurística cuenta con dos vecindades, *vns1swap* y *vns2swap*, además se identificó a las primeras búsquedas como *first stage* donde se aplicará la estrategia “first improve” con la vecindad *vns1swap*, la cantidad de búsquedas que incluyen esta etapa esta determinada por el parámetro *first stage steps*.

A continuación se analizara el comportamiento al variar el parámetro *first stage steps* con los valores de: 0 (FI-0), 20 (FI-20) y 100 (FI-100).

Para esta comparación se tomó el 65 % de la demanda.

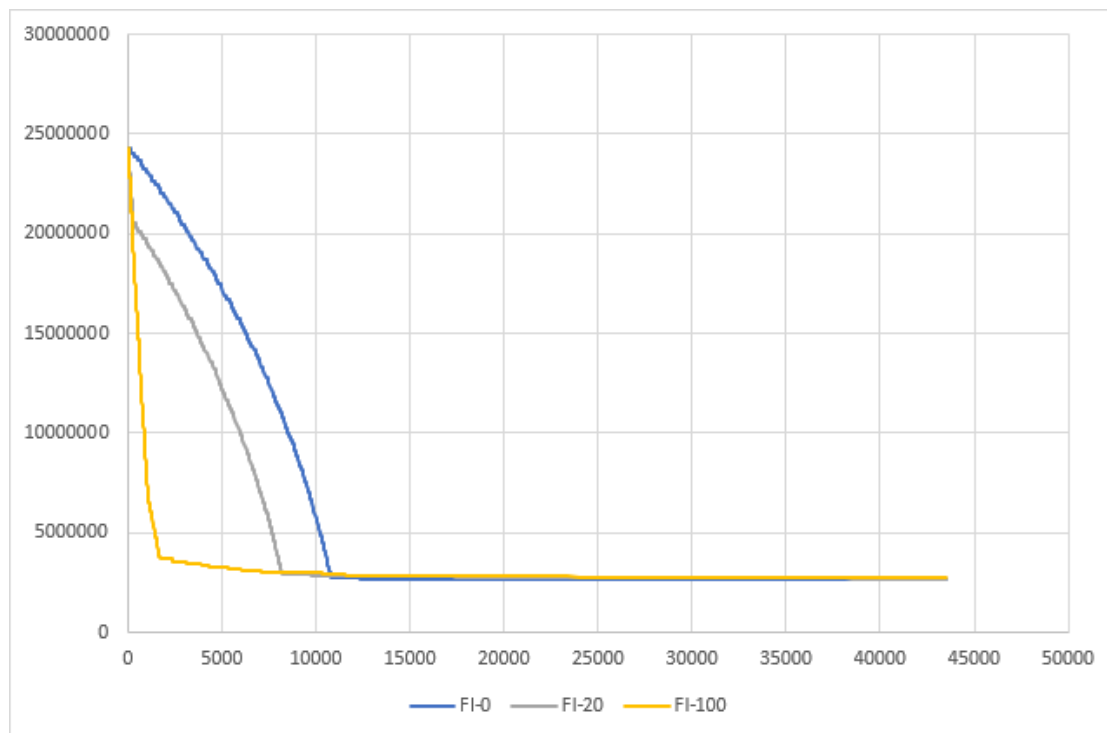


Figura 6.6: Escenario 3 - Comparación Primera Etapa - Demanda 65 %
Tiempo (ms) \times Valor Objetivo

En la gráfica (ver Figura 6.6) se puede ver cómo se consigue el mismo valor objetivo en menos tiempo conforme aumenta el valor del parámetro *first stage steps*.

Se define *intemediate stage* al conjunto de búsquedas que suceden a *first stage* y aplican “best improve” en la vecindad *vns1swap*, y se define *final stage* al conjunto de búsquedas que aplican sobre la vecindad *vns2swap*; cabe aclarar que siguiendo el procedimiento VNS después de cada mejora con *vns2swap* se aplica *vns1swap*, aunque sin éxito la gran mayoría de las veces por la naturaleza del problema.

	Exacto	Stages (FI-0)			Stages (FI-20)			Stages (FI-100)		
		First	Intermediate	Final	First	Intermediate	Final	First	Intermediate	Final
Tiempo (s)	113,00	0,01	10,81	22,83	0,27	8,18	38,52	1,09	1,65	43,52
V. Obj.	2.647.422	24.266.116	2.743.048	2.680.686	20.524.018	2.949.083	2.681.859	6.495.319	3.822.680	2.715.183
%T	–	0,01	9,57	20,21	0,23	7,24	34,09	0,97	1,46	38,51
%V	–	916,59	103,61	101,26	775,25	111,39	101,30	245,35	144,39	102,56

Tabla 6.7: Escenario 3 - Variantes *first stage steps*

Se puede observar que al finalizar *intemediate stage* los valores objetivos se sitúan más lejos del óptimo según aumenta *first stage steps*. Además de que, cuanto mayor sea *first stage steps*, más tarda *final stage* en dar respuesta (ver Tabla 6.7). Puede ser de interés evitar *final stage* si se quieren ver distintos resultados rápidamente. Para una ejecución con *first stage steps* = 20 se ve en la tabla cómo en los primeros 8 segundos se consigue el 91 % del resultado, mientras que se tarda 30 segundos más en conseguir el 9 % restante.

6.4.3. Valoración del estado actual

Actualmente en Ciudad Vieja se encuentran instaladas 36 zonas de carga y descarga distribuidas en el barrio (ver Figura 6.7). El objetivo de esta prueba es el de evaluar la disposición actual, con la función objetivo planteada en el Capítulo 3.

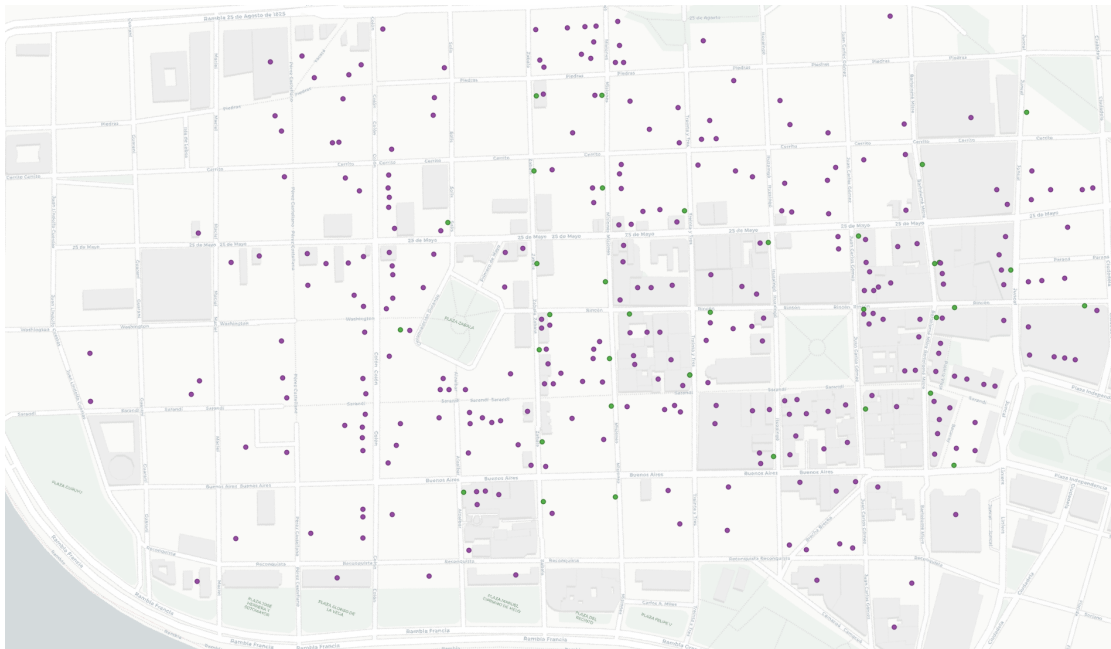


Figura 6.7: Escenario 4 - Disposición Actual
Puntos verdes: Actuales ZCD - Puntos violetas: Locales comerciales

La información de la ubicación de la disposición actual fue tomada del Sistema de Información Geográfica² de la Intendencia de Montevideo.

Para esta evaluación se tomaron algunas consideraciones:

- Se liberó la restricción de distancia máxima, configurando una suficientemente grande para que no afecte en las decisiones de asignación. Esto se debe a que, con la disposición actual, hay locales comerciales, los cuales, su distancia a la ZCD más cercana es más de 500m, pudiendo hacer que el algoritmo no encuentre una solución al problema.
- Se tomó la cantidad de zonas a abrir en 36 para coincidir con la disposición actual.
- Se configuró una capacidad de 24h para las zonas de carga.
- Se consideró el 90 % de la demanda actual, ya que 36 zonas con capacidad de 24h no satisfacen la demanda al 100 %.

²<https://sig.montevideo.gub.uy/>

Una vez hecha la evaluación de la disposición actual (ver Figura 6.8), se tiene como resultado un valor de 7.435.959 para la función objetivo. Luego se ejecuta la metaheurística buscando abrir la misma cantidad de zonas (36) (ver Figura 6.9), teniéndose un valor de 3.483.197 en la función objetivo. La diferencia de valor puede ser explicada por las grandes distancias a las que se encuentra la ZCD más cercana para algunos locales comerciales, evidenciándose una posibilidad de mejora.

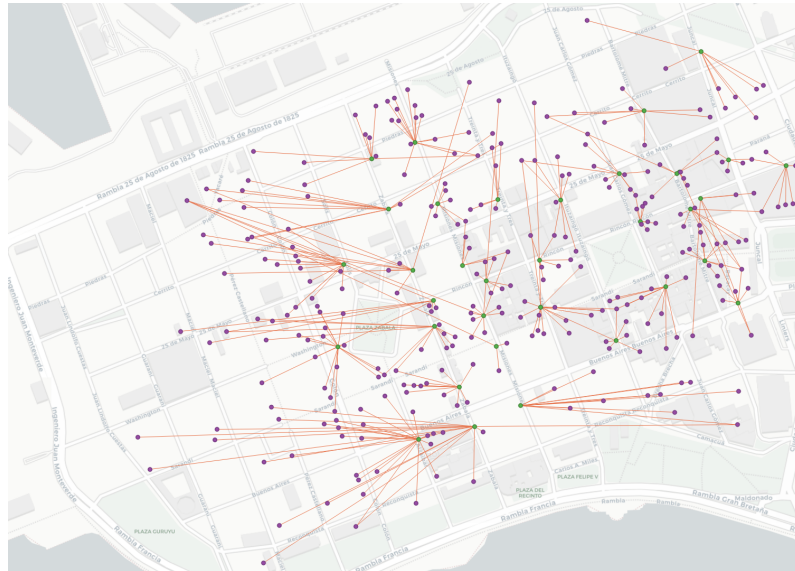


Figura 6.8: Escenario 4 - Asignación de locales para disposición actual
Puntos verdes: Actuales ZCD - Puntos violetas: Locales comerciales
Valor objetivo: 7.435.959

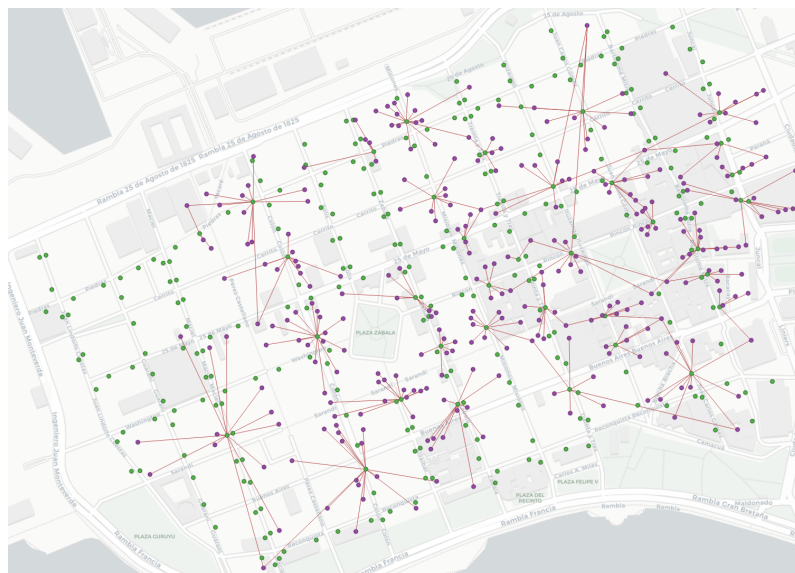


Figura 6.9: Escenario 4 - Asignación de locales propuesta
Puntos verdes: Actuales ZCD - Puntos violetas: Locales comerciales
Valor objetivo: 3.483.197

Capítulo 7

Herramienta

En este capítulo se presentará la herramienta informática desarrollada y se introducirán las características principales de la misma. Gracias a esta herramienta, el usuario podrá: cargar distintos escenarios al sistema, configurar los distintos parámetros del algoritmo, configurar ajustes de la herramienta, ejecutar la heurística, ver los resultados, además de exportar la solución en un shapefile para un posterior análisis.

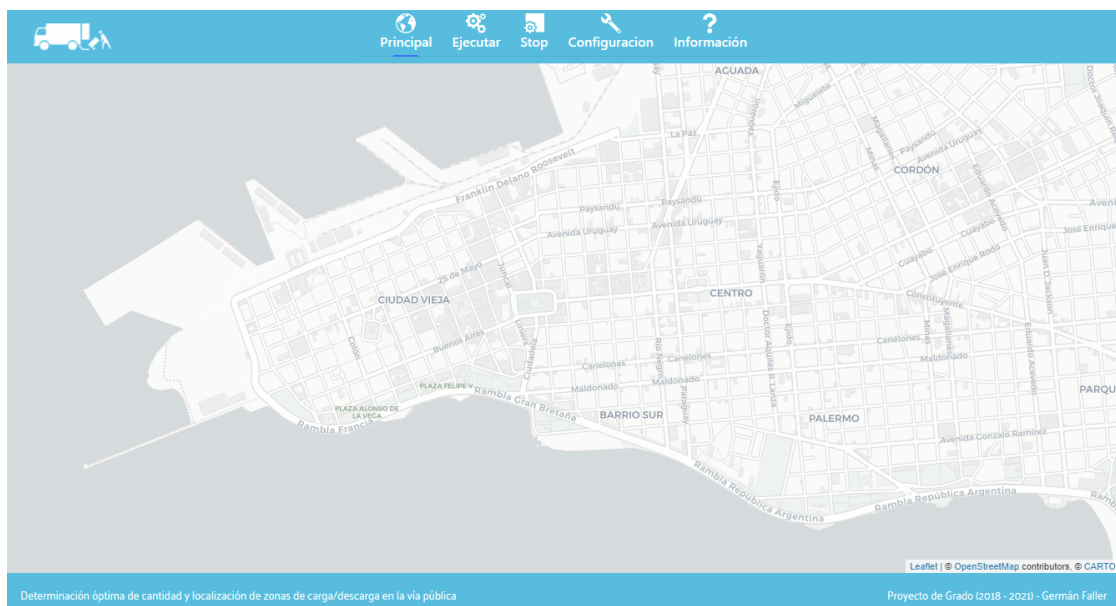


Figura 7.1: Herramienta - Vista Principal

7.1. Selección de datos de entrada

Como se ha dicho anteriormente, el formato de entrada y salida son shapefiles. Para seleccionar un escenario puntual se deberán colocar dos archivos en una ubicación específica relativa al proyecto.

- El primero perteneciente a la localización de locales comerciales, ubicado en `./data/padrones`.
- El segundo refiere a la localización de potenciales Zonas de Carga y Descarga, ubicado en `./data/segmentos`.

De esta manera, por ejemplo, se puede ver el comportamiento en Punta Carretas entre las 11:00 y las 15:00 si se dispusiera de esos datos.

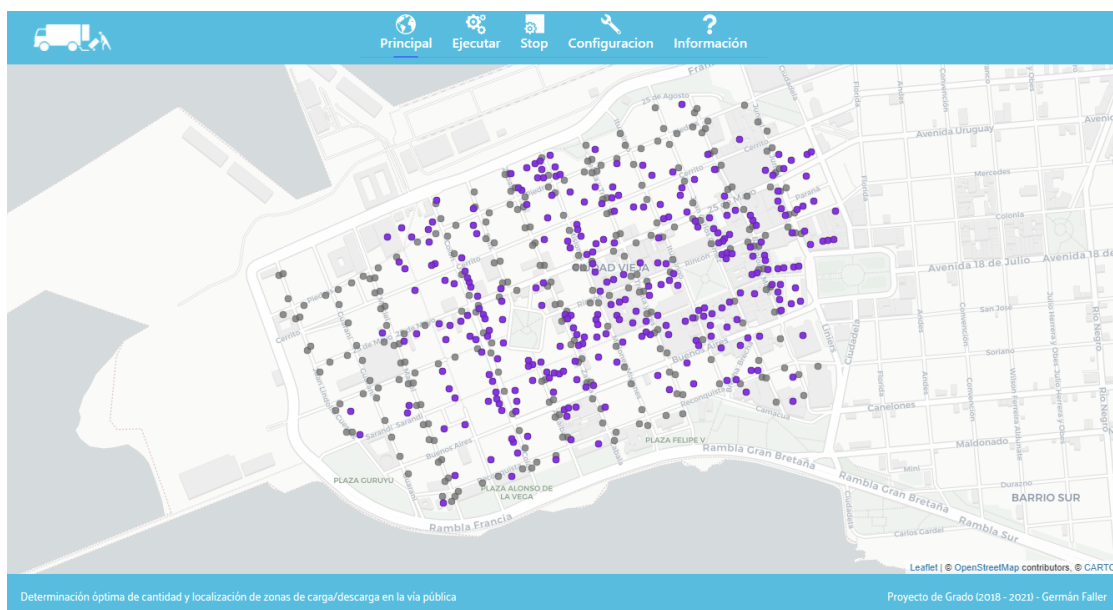


Figura 7.2: Herramienta - Escenario cargado
Puntos grises: Potenciales ZCD - Puntos violetas: Locales comerciales

En este proyecto se trabajará sobre el barrio Ciudad Vieja de la ciudad de Montevideo. Una vez cargado el escenario, la herramienta mostrará la ubicación de los locales comerciales como puntos violetas, así como las potenciales zonas de carga y descarga en gris (ver Figura 7.2).

7.2. Configuración

En el apartado de configuración es posible realizar varios ajustes, de la heurística, del problema y de la herramienta.

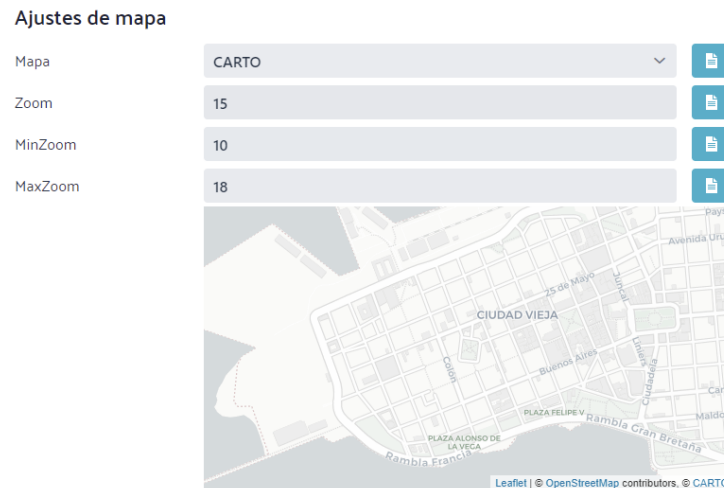


Figura 7.3: Herramienta - Ajustes de mapa



Figura 7.4: Herramienta - Otros Ajustes

Dentro de los ajustes que se pueden realizar, se tienen los ajustes de mapa (ver Figura 7.3), pudiéndose seleccionar el proveedor de mapas, como pueden ser Google Maps, OpenStreetMap, Carto, etc. Además es configurable: el zoom inicial de la herramienta; el zoom mínimo, es decir qué tan lejos se puede ver el mapa; y el zoom máximo, que tan cerca se puede ver el mapa.

Del mismo modo, dentro de configuración se encuentran los parámetros de la heurística y del problema (ver Figura 7.4). Estos pueden ser cambiados antes de cada ejecución para analizar distintas variantes del problema o del algoritmo.

7.3. Ejecución

En la barra superior se encuentra un botón para iniciar la ejecución del algoritmo. Una vez finalizada la ejecución se agregará a la derecha un resumen con información, la guía de colores, cantidad de zonas por tipo de vehículo y valor objetivo de la solución (ver Figura 7.5).

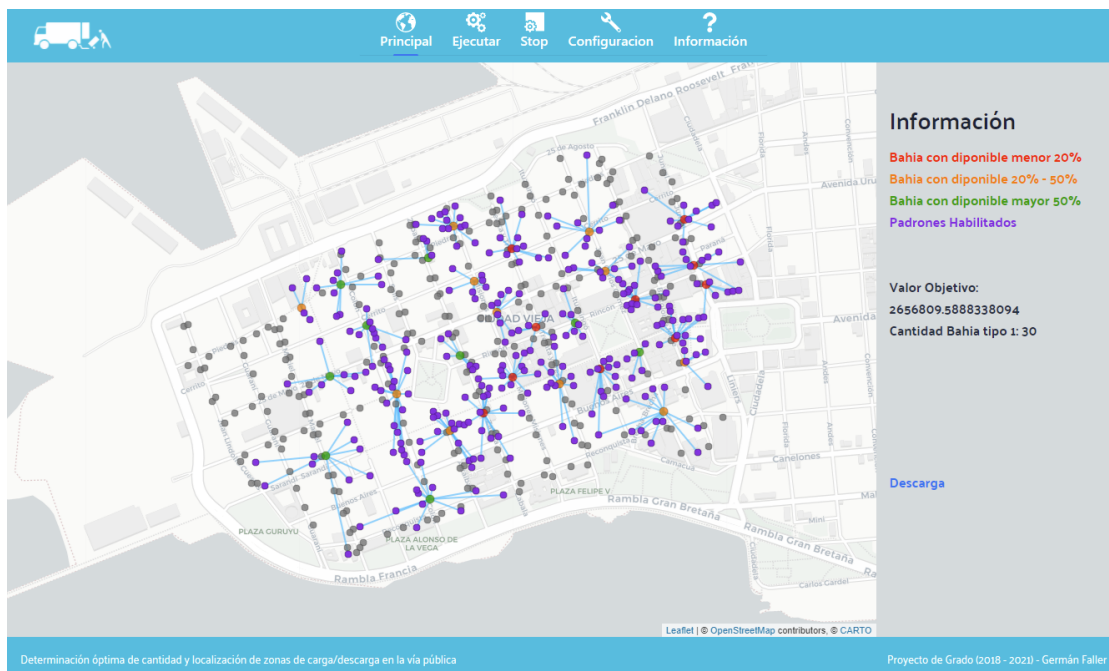


Figura 7.5: Herramienta - Asignaciones

En el mapa se incluye líneas de asignación que unen el local comercial con la ZCD asignada. Se pintan con un código de colores las ZCD utilizadas en función del tiempo disponible luego de la asignación de demanda:

- *rojo*: ZCD con su capacidad disponible menor a un 20 %.
- *naranja*: ZCD con su capacidad disponible entre 20 % y 50 %.
- *verde*: ZCD con su capacidad disponible mayor a un 50 %.

Otra funcionalidad con la que se cuenta son las anotaciones en los elementos del mapa (ver Figura 7.6) para tener más información de la solución expuesta. Seleccionando un local comercial se puede ver: la identificación y la demanda según tipo de vehículo. De la misma manera para una ZCD muestra: el identificador, la capacidad total, la capacidad disponible, el tipo máximo de vehículo que esa ZCD permite y el tipo de vehículo que finalmente se le asignó. Por último, el segmento entre el local comercial y la ZCD muestra: cuanto tiempo se asignó, la distancia a recorrer y el tipo de vehículo en la asignación.

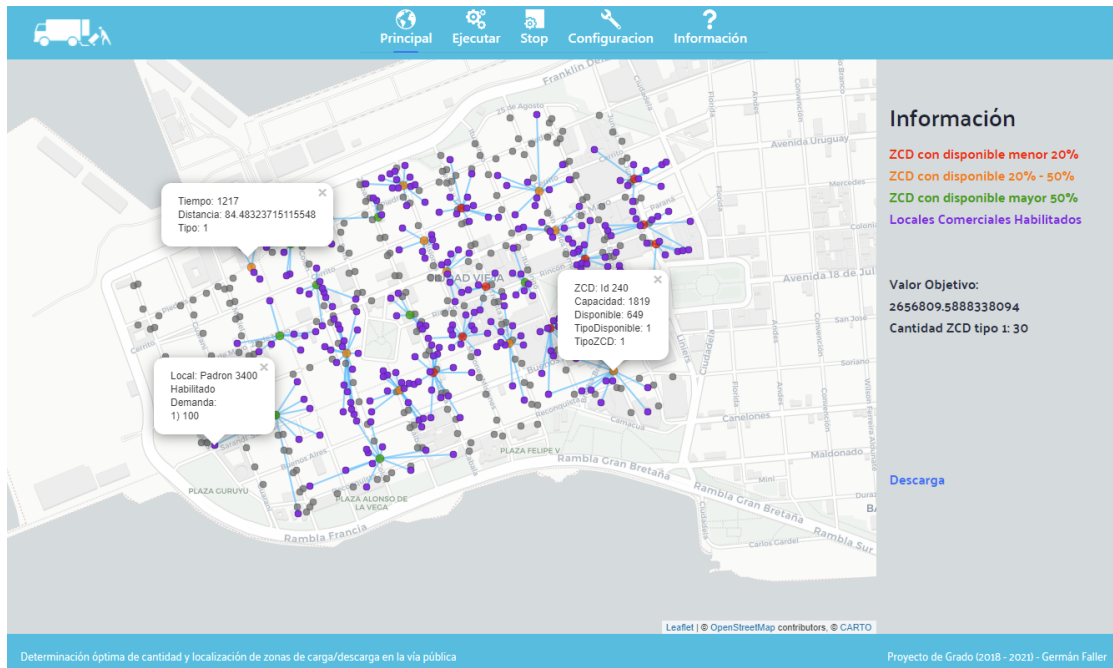


Figura 7.6: Herramienta - Anotaciones

7.4. Visualización alternativa

Una alternativa de visualización se logra utilizando un servicio de terceros para dibujar los trayectos más realistas (ver Figura 7.7), sugiriendo la caminata del peatón entre los dos puntos (asignaciones de locales a ZCD). En este caso, se utilizaron los servicios de Google para calcular el trayecto. Como desventaja se puede notar la superposición de los distintos trayectos dificultando seleccionarlos para acceder a la información de distancia y tiempo asignado. Además este servicio tiene un costo asociado, teniendo un conjunto limitado de usos gratuitos, los cuales han sido utilizados construyendo la herramienta.

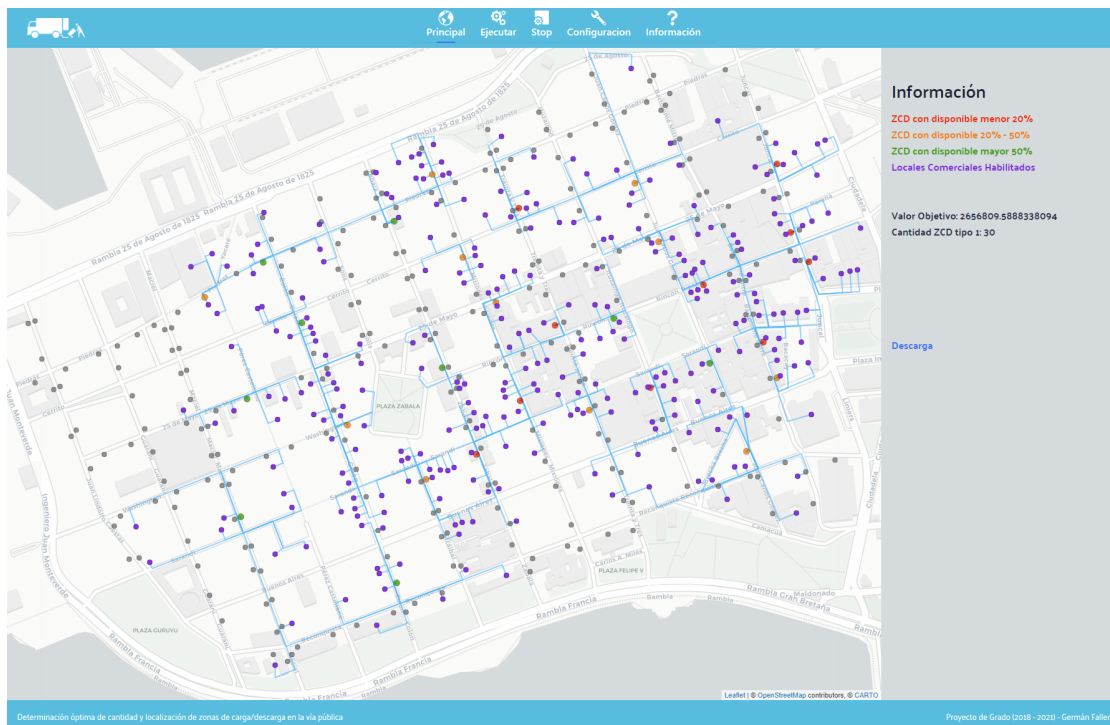


Figura 7.7: Herramienta - Visualización trayecto caminando

7.5. Descarga

Una vez ejecutado el algoritmo y una solución es presentada, se tiene la opción de exportarla. De esta manera se puede importar en QGIS esta solución.

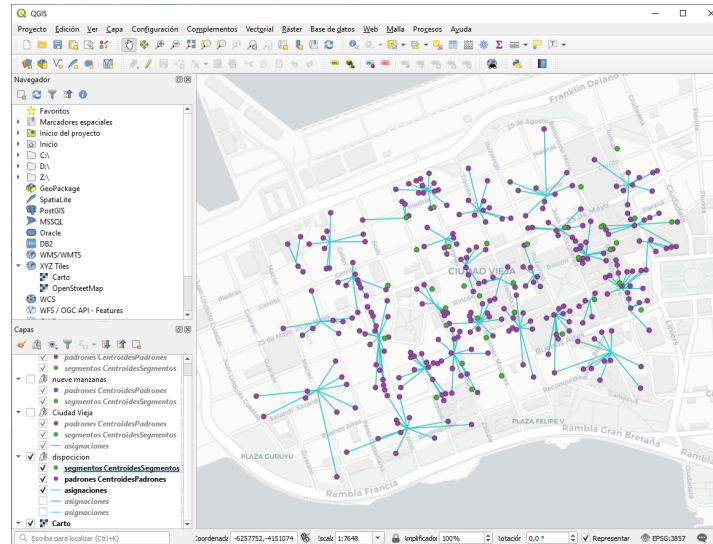


Figura 7.8: QGIS - Importando solución descargada

Con esta funcionalidad se le da la opción al tomador de decisiones de guardar y comparar distintos resultados. Tanto la visualización original (ver Figura 7.8) como la visualización alternativa (ver Figura 7.9) son exportables, pudiendo ambas ser importadas en QGIS.

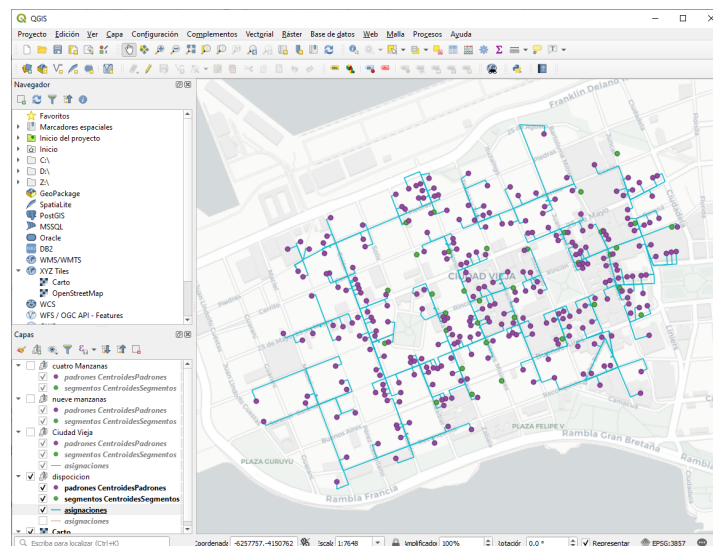


Figura 7.9: QGIS - Importando solución descargada - Visualización alternativa

Concluyendo así la herramienta y acercando la implementación de modelos y algoritmos a usuarios encargados de la toma de decisiones, que no son expertos en áreas de optimización y algoritmia.

Capítulo 8

Conclusiones y trabajo futuro

En este capítulo se presentan las conclusiones del proyecto realizado, así como líneas de trabajo futuro que surgen de extensiones posibles del trabajo llevado a cabo.

8.1. Conclusiones

Este proyecto abordó un problema dentro del planeamiento del uso de espacio público presentado por integrantes de la Unidad de Logística de la Intendencia de Montevideo. En este problema se busca mejorar las características de la distribución urbana de mercancías en la ciudad de Montevideo, más concretamente en el barrio Ciudad Vieja como zona de estudio. Se clasificó como un problema de *localización* para las zonas de carga y descarga, y *asignación* de las mismas a locales comerciales.

Se recopilaron y analizaron distintos trabajos encontrados en la literatura sobre problemas similares al abordado en el proyecto. Se tomaron en cuenta distintos trabajos para tratar el mismo problema en distintos países del mundo, por ejemplo: Francia, España, Italia, Brasil, entre otros. Se hizo especial énfasis en aquellas propuestas que tienen un enfoque de modelado matemático y optimización. En este sentido, se tomó como referencia el modelo de Muñuzuri *et al.* (2017), que frente a un problema con características similares, proponen una solución para mejorar el sistema de carga y descarga en la ciudad de Sevilla, España.

Se extendió el modelo de referencia restringiendo la distancia que un local comercial puede estar de una zona de carga asignada y definiendo un máximo para esta. Se limitó el tiempo de uso de la ZCD, dado que se asume un mínimo por viaje, evitando así, posibles fraccionamientos poco realistas. Se agregó el concepto de tipo de vehículo, considerando una cantidad de tipos, los cuales respetan una distribución jerárquica; a mayor tipo, más grande es el vehículo. Otra característica que se tuvo en cuenta, fue el tipo máximo de vehículo por ZCD, además de realizar los ajustes necesarios en el modelo para incluir esta nueva característica.

Para desarrollar el procedimiento de resolución para el problema abordado, se tomó como base la metaheurística VND (Variable Neighborhood Descent). Se introdujo la representación de la solución como un conjunto de valores binarios ordenados, donde la posición del valor está vinculada a si está cerrada (0) o abierta (1) una determinada Zona de Carga y Descarga. Se definieron dos nociones de vecindad: una que principalmente busca bajar la cantidad de ZCD utilizadas. Mientras que la otra, busca reubicar las ZCD ya seleccionadas. También se diseñó e implementó un algoritmo de asignación de ZCD a Locales Comerciales, que en cada paso, para cada local, le asigna una fracción de la demanda a la ZCD más cercana con disponibilidad.

Se realizaron pruebas durante todo el proyecto. Luego de extender el modelo, en cuanto a restricciones, los resultados obtenidos coincidieron con los esperados. De la misma manera, al validar los algoritmos desarrollados, se controló que las soluciones devueltas cumplieran con las restricciones del problema. El procedimiento de resolución se evaluó frente a un software comercial de optimización (CPLEX) manteniéndose por debajo del 5 % del valor óptimo en el peor caso. Con respecto a los tiempos de ejecución en instancias grandes del problema, se obtuvieron resultados muy por debajo del método exacto.

Se implementó una herramienta para el manejo y presentación de los datos del problema que le permite al tomador de decisiones gestionar diferentes aspectos de las instancias. Toma los datos del escenario y los muestra en pantalla en un mapa interactivo que ofrece información del mismo. Permite ajustar diferentes parámetros del escenario, del modelo, del algoritmo y de la herramienta en sí. Además, ofrece la posibilidad de ejecutar el procedimiento de resolución, mostrando en el mapa la solución retornada junto a información de interés. Como funcionalidad extra, es posible exportar la solución para ser almacenada, comparada o incluso importada en herramientas de manejo de información geográfica.

Por lo antes mencionado, se entiende entonces que los objetivos planteados para este proyecto fueron cumplidos satisfactoriamente,

8.2. Trabajo futuro

A lo largo del desarrollo de este proyecto fueron surgiendo oportunidades para seguir investigando, y ya sea por el alcance definido en los objetivos, o bien, por el tiempo que insume ahondar en ciertos aspectos, estas son sugeridas como trabajo a futuro.

Para lograr buenos resultados, el método de resolución depende fuertemente de los datos proporcionados. Mejorar en calidad y detalles los datos que se cuentan de la demanda de los locales, basado en encuestas o formularios web, aportaría a la precisión de los resultados.

Poder contar con una retroalimentación por parte de los usuarios brindaría información del uso actual del sistema, horas de mayor concurrencia, zonas más ocupadas, zonas libres, incluso denuncias por uso indebido de la zona de carga. Para esto, una aplicación móvil con seguimiento GPS podría ser una opción.

En cuanto al algoritmo, se encontraron ejemplos de combinaciones de metaheurísticas como puede ser VNS con Simulated Annealing lo cual sería interesante analizar y comparar. Una variante posible de la solución propuesta es tomar los parámetros de penalización por incumplimiento de restricciones y la granularidad con la que se asigna la demanda de un local a una ZCD, que actualmente son fijos para toda la ejecución y calcularlos en función de la evolución de la ejecución.

El valor objetivo depende en gran medida de la asignación de ZCD a locales comerciales. En este sentido, se podría mejorar el algoritmo planteado, para que sea capaz de reconocer y evaluar soluciones que no son consideradas actualmente, logrando así reducir la brecha actual del 5% (en el peor caso) que hay con un método exacto.

Por otro lado, es posible mejorar la usabilidad de la herramienta, permitiendo cargar los escenarios desde la herramienta en lugar de que estos datos sean tomados desde un directorio puntual. Además se podría mejorar la configuración clasificando los distintos parámetros y agregando una descripción a los mismos. La vista del mapa puede ser mejorada en diseño y experiencia de usuario para hacerla más intuitiva al usuario final. Si bien la herramienta permite exportar la solución, actualmente no tiene opción de importar una. Para finalizar, puede ser de interés restringir el acceso a usuarios autorizados, debiéndose implementar un mecanismo de autenticación.

Bibliografía

- Aiura, N. & Taniguchi, E. (2005). Planning on-street loading-unloading spaces considering the behaviour of pickup-delivery vehicles. *Journal of the Eastern Asia Society for Transportation Studies*, 6, 2963-2974. <https://doi.org/10.11175/easts.6.2963>
- Alho, A. R., de Abreu e Silva, J., de Sousa, J. P. & Blanco, E. (2018). Improving mobility by optimizing the number, location and usage of loading/unloading bays for urban freight vehicles. *Transportation Research Part D: Transport and Environment*, 61, 3-18. <https://doi.org/10.1016/j.trd.2017.05.014>
- Anderson, S., Allen, J. & Browne, M. (2005). Urban logistics—how can it meet policy makers’ sustainability objectives? [Sustainability and the Interaction Between External Effects of Transport (Part Special Issue, pp. 23-99)]. *Journal of Transport Geography*, 13(1), 71-81. <https://doi.org/10.1016/j.jtrangeo.2004.11.002>
- Ballantyne, E. & Lindholm, M. (2013). Identifying the Need for Freight to be Included in Local Authority Transport Planning (J. Gonzalez-Feliu, F. Semet & J.-L. Routhier, Eds.). En J. Gonzalez-Feliu, F. Semet & J.-L. Routhier (Eds.), *Sustainable Urban Logistics: Concepts, Methods and Information Systems*. Berlin, Heidelberg, Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-31788-0_3
- Banco Interamericano de Desarrollo. (2009). *Logística urbana: Los desafíos de la Distribución Urbana de Mercancías*. Recuperado el 1 de julio de 2020, desde <https://publications.iadb.org/es/publicacion/14260/logistica-urbana-los-desafios-de-la-distribucion-urbana-de-mercancias>
- Brimberg, J., Hansen, P., Mladenović, N. & Taillard, E. D. (2000). Improvements and Comparison of Heuristics for Solving the Uncapacitated Multisource Weber Problem. *Operations Research*, 48(3), 444-460. <https://doi.org/10.1287/opre.48.3.444.12431>
- Browne, M., Allen, J., Nemoto, T., Patier, D. & Visser, J. (2012). Reducing Social and Environmental Impacts of Urban Freight Transport: A Review of Some Major Cities [Seventh International Conference on City Logistics which was

- held on June 7- 9,2011, Mallorca, Spain]. *Procedia - Social and Behavioral Sciences*, 39, 19-33. <https://doi.org/10.1016/j.sbspro.2012.03.088>
- Carto. (2020). *Carto*. Recuperado el 1 de julio de 2020, desde <https://carto.com/attribution>
- Church, R. & ReVelle, C. (1974). The maximal covering location problem. *Papers of the Regional Science Association*, 32(1), 101-118. <https://doi.org/10.1007/bf01942293>
- Cornuéjols, G., Nemhauser, G. L. & Wolsey, L. A. (1990). The uncapacitated facility location problem (P. B. Mirchandani & R. L. Francis, Eds.). En P. B. Mirchandani & R. L. Francis (Eds.), *Discrete Location Theory*. <https://api.semanticscholar.org/CorpusID:8880493>
- Cuevas, A., Giesen, R. & Muñoz, J. C. (2013). Revisión crítica a políticas de logística urbana sustentable. *Congreso Chileno de Ingeniería de Transporte*, (16). <https://revistas.uchile.cl/index.php/CIT/article/view/28404>
- Dablanc, L., Giuliano, G., Holliday, K. & O'Brien, T. (2013). Best Practices in Urban Freight Management : Lessons from an International Survey. *Transportation Research Record*, 22p. <https://hal.archives-ouvertes.fr/hal-00854997>
- Dezi, G., Dondi, G. & Sangiorgi, C. (2010). Urban freight transport in Bologna: Planning commercial vehicle loading/unloading zones. *Procedia - Social and Behavioral Sciences*, 2(3), 5990-6001. <https://doi.org/10.1016/j.sbspro.2010.04.013>
- El-Sherbiny, M. M. & Alhamali, R. M. (2013). A hybrid particle swarm algorithm with artificial immune learning for solving the fixed charge transportation problem. *Computers & Industrial Engineering*, 64(2), 610-620. <https://doi.org/10.1016/j.cie.2012.12.001>
- Estrada, M. Á. (2007). *Análisis de estrategias eficientes en la logística de distribución de paquetería* (Tesis doctoral). UPC, Departament d'Infraestructura del Transport i del Territori. <http://hdl.handle.net/2117/93936>
- Gendreau, M. & Potvin, J.-Y. (2010). *Handbook of Metaheuristics*. Springer US. <https://doi.org/10.1007/978-1-4419-1665-5>
- Goldman, T. & Gorham, R. (2006). Sustainable urban transport: Four innovative directions [Sustainable Cities]. *Technology in Society*, 28(1-2), 261-273. <https://doi.org/10.1016/j.techsoc.2005.10.007>
- Hajiaghahi-Keshteli, M., Molla-Alizadeh-Zavardehi, S. & Tavakkoli-Moghaddam, R. (2010). Addressing a nonlinear fixed-charge transportation problem using a spanning tree-based genetic algorithm. *Computers & Industrial Engineering*, 59(2), 259-271. <https://doi.org/10.1016/j.cie.2010.04.007>

- Hakimi, S. L. (1964). Optimum Locations of Switching Centers and the Absolute Centers and Medians of a Graph. *Operations Research*, 12(3), 450-459. <http://www.jstor.org/stable/168125>
- Hindi, K., Basta, T. & Pieńkosz, K. (1998). Efficient solution of a multi-commodity, two-stage distribution problem with constraints on assignment of customers to distribution centres. *International Transactions in Operational Research*, 5(6), 519-527. [https://doi.org/10.1016/S0969-6016\(98\)00039-2](https://doi.org/10.1016/S0969-6016(98)00039-2)
- Holguín-Veras, J. (2013). The good, the bad, and the ugly: lessons from the off-peak delivery project in New York. *Urban Freight for Livable Cities*, 118-125. [https://www.sciencedirect.com/science/refhub/S1361-9209\(16\)30471-0/h0130](https://www.sciencedirect.com/science/refhub/S1361-9209(16)30471-0/h0130)
- Jayaraman, V. & Ross, A. (2003). A simulated annealing methodology to distribution network design and management. *European Journal of Operational Research*, 144(3), 629-645. [https://doi.org/10.1016/S0377-2217\(02\)00153-4](https://doi.org/10.1016/S0377-2217(02)00153-4)
- Lee, Y. H. & Kwon, S. G. (2010). The hybrid planning algorithm for the distribution center operation using tabu search and decomposed optimization. *Expert Systems with Applications*, 37(4), 3094-3103. <https://doi.org/10.1016/j.eswa.2009.09.020>
- Li, X., Wei, K., Aneja, Y., Tian, P. & Cui, Y. (2017). Matheuristics for the single-path design-balanced service network design problem. *Computers & Operations Research*, 77, 141-153. <https://doi.org/10.1016/j.cor.2016.07.010>
- Marinov, M., Zunder, T. & Dewan, I. (2010). Urban freight consolidation concepts: is there something missing? *Transport Problems : an International Scientific Journal*, 5, 87-94.
- Mladenović, N. & Hansen, P. (1997). Variable neighborhood search. *Computers & Operations Research*, 24(11), 1097-1100. [https://doi.org/10.1016/S0305-0548\(97\)00031-2](https://doi.org/10.1016/S0305-0548(97)00031-2)
- Molla-Alizadeh-Zavardehi, S., Hajiaghaei-Keshteli, M. & Tavakkoli-Moghaddam, R. (2011). Solving a capacitated fixed-charge transportation problem by artificial immune and genetic algorithms with a Prüfer number representation. *Expert Systems with Applications*, 38(8), 10462-10474. <https://doi.org/10.1016/j.eswa.2011.02.093>
- Molla-Alizadeh-Zavardehi, S., Sadi Nezhad, S., Tavakkoli-Moghaddam, R. & Yazdani, M. (2013). Solving a fuzzy fixed charge solid transportation problem by metaheuristics. *Mathematical and Computer Modelling*, 57(5), 1543-1558. <https://doi.org/10.1016/j.mcm.2012.12.031>
- Muñuzuri, J., Cortés, P., Grosso, R. & Guadix, J. (2012). Selecting the location of minihubs for freight delivery in congested downtown areas [City Logistics]. *Journal of Computational Science*, 3(4), 228-237. <https://doi.org/10.1016/j.jocs.2011.12.002>

- Muñuzuri, J., Cuberos, M., Abaurrea, F. & Escudero, A. (2017). Improving the design of urban loading zone systems. *Journal of Transport Geography*, *59*, 1-13. <https://doi.org/10.1016/j.jtrangeo.2017.01.004>
- Nourinejad, M., Wenneman, A., Habib, K. N. & Roorda, M. J. (2014). Truck parking in urban areas: Application of choice modelling within traffic microsimulation. *Transportation Research Part A: Policy and Practice*, *64*, 54-64. <https://doi.org/10.1016/j.tra.2014.03.006>
- Pirkul, H. & Jayaraman, V. (1998). A multi-commodity, multi-plant, capacitated facility location problem: formulation and efficient heuristic solution. *Computers & Operations Research*, *25*(10), 869-878. [https://doi.org/10.1016/S0305-0548\(97\)00096-8](https://doi.org/10.1016/S0305-0548(97)00096-8)
- Prata, B., Oliveira, L. & Holanda, T. (2018). Locating on-street loading and unloading spaces by means of mixed integer programming. *TRANSPORTES*, *26*(1), 16-30. <https://doi.org/10.14295/transportes.v26i1.1051>
- Russo, F. & Comi, A. (2011). Measures for Sustainable Freight Transportation at Urban Scale: Expected Goals and Tested Results in Europe. *Journal of Urban Planning and Development*, *137*(2), 142-152. [https://doi.org/10.1061/\(asce\)up.1943-5444.0000052](https://doi.org/10.1061/(asce)up.1943-5444.0000052)
- Sadeghi-Moghaddam, S., Hajiaghahi-Keshteli, M. & Mahmoodjanloo, M. (2019). New approaches in metaheuristics to solve the fixed charge transportation problem in a fuzzy environment. *Neural Computing and Applications*, *31*(1), 477-497. <https://doi.org/10.1007/s00521-017-3027-3>
- Sharma, M. J., Moon, I. & Bae, H. (2008). Analytic hierarchy process to assess and optimize distribution network. *Applied Mathematics and Computation*, *202*(1), 256-265. <https://doi.org/10.1016/j.amc.2008.02.008>
- Tsao, Y.-C. (2013). Distribution center network design under trade credits. *Applied Mathematics and Computation*, *222*, 356-364. <https://doi.org/10.1016/j.amc.2013.07.028>
- van Duin, J. H. R. & Quak, H. J. (2007). City logistics: A chaos between research and policy making? A review, En *Urban Transport XIII: Urban Transport and the Environment in the 21st Century*, WIT Press. <https://doi.org/10.2495/ut070141>
- Zhou, G., Min, H. & Gen, M. (2002). The balanced allocation of customers to multiple distribution centers in the supply chain network: a genetic algorithm approach. *Computers & Industrial Engineering*, *43*(1), 251-261. [https://doi.org/10.1016/S0360-8352\(02\)00067-0](https://doi.org/10.1016/S0360-8352(02)00067-0)

Apéndice A

Anexo

ZCD	Local	Tipo	Asignación	Distancia
4	28	1	13	38,66
4	29	1	13	71,52
4	42	1	11	42,75
6	36	1	38	90,67
6	43	1	16	113,59
7	37	1	15	95,17
7	38	1	20	25,04
7	39	1	18	17,33
7	40	1	16	32,93
8	16	1	14	26,70
8	26	1	24	24,11
9	17	1	20	17,66
10	18	1	19	21,93
10	23	1	20	102,62
11	19	1	24	29,76
11	20	1	17	89,96
12	21	1	13	114,00
12	23	1	11	69,17
12	24	1	16	51,10
12	25	1	30	37,53
14	6	1	16	35,76
14	8	1	30	52,78
14	10	1	14	72,81
14	12	1	10	113,02
15	9	1	12	31,87
15	11	1	18	66,01
15	12	1	20	90,22
15	24	1	20	32,44
17	2	1	18	44,56
17	3	1	20	28,92
18	1	1	16	16,84
18	14	1	14	48,39
18	15	1	11	20,64
18	44	1	16	24,61
19	27	1	21	19,05
19	35	1	16	8,70
20	34	1	11	25,91
21	30	1	14	73,77
21	31	1	11	46,41
21	32	1	19	29,67
21	33	1	13	14,70

Tabla A.1: Escenario 1 - Salida

La Tabla A.1 muestra las asignaciones ZCD-Local, donde la primera y segunda columna son los identificadores de la ZCD y del local respectivamente. Luego le siguen el tipo de vehículo, unidades temporales asignadas y la distancia entre ambos puntos.

Local	ZCD															Total	D
	4	6	7	8	9	10	11	12	14	15	17	18	19	20	21		
1												16				16	16
2																18	18
3												18				20	20
6										16						16	16
8									30							30	30
9										12						12	12
10										14						14	14
11											18					18	18
12										10	20					30	30
14												14				14	14
15												11				11	11
16				14												14	14
17					20											20	20
18						19										19	19
19							24									24	24
20							17									17	17
21								13								13	13
23							20									31	31
24								16		20						36	36
25								30								30	30
26				24												24	24
27														21		21	21
28	13															13	13
29	13															13	13
30															14	14	14
31															11	11	11
32															19	19	19
33															13	13	13
34														11		11	11
35													16			16	16
36		38														38	38
37			15													15	15
38			20													20	20
39			18													18	18
40			16													16	16
42	11															11	11
43		16														16	16
44												16				16	16
Total	37	54	69	38	20	39	41	70	70	70	38	57	37	11	57	708	
C	70	70	70	70	70	70	70	70	70	70	70	70	70	70	70		

Tabla A.2: Escenario 1 - Solución

La Tabla A.2 es una matriz que muestra la asignación y compara con los resultados esperados. En las columnas se tienen las ZCD, mientras que en las filas se encuentran los locales, cada valor en la matriz es la asignación de unidades temporales. Las dos últimas columnas muestran el total de demanda asignado a cada local y la demanda del local. Por otro lado las dos últimas filas muestran el total de tiempo que cada zona tiene en uso, comparándolo con la capacidad de las mismas.

ZCD	Local	Tipo	Asignación	Distancia	ZCD	Local	Tipo	Asignación	Distancia
4	28	1	14	38,66	5	42	2	18	36,95
4	29	1	66	71,52	7	40	2	16	32,93
5	42	1	12	36,95	9	16	2	11	42,08
6	16	1	12	36,64	9	17	2	11	17,66
6	36	1	30	90,67	11	18	2	19	25,49
6	43	1	12	113,59	11	19	2	11	29,76
7	37	1	22	95,17	11	20	2	10	89,96
7	38	1	20	25,04	11	22	2	12	105,60
7	40	1	28	32,93	13	4	2	15	19,12
9	17	1	26	17,66	13	5	2	18	20,52
11	18	1	22	25,49	13	7	2	25	61,54
12	21	1	50	114,00	13	11	2	13	104,06
12	22	1	16	87,20	15	8	2	11	29,98
12	23	1	52	69,17	15	9	2	11	31,87
12	24	1	18	51,10	15	10	2	15	50,00
13	4	1	12	19,12	15	12	2	28	90,22
13	5	1	46	20,52	15	24	2	26	32,44
14	7	1	38	46,30	17	3	2	12	28,92
14	8	1	30	52,78	17	13	2	16	88,45
14	9	1	11	68,91	19	27	2	11	19,05
14	10	1	14	72,81	19	34	2	10	37,53
14	11	1	30	88,82	21	14	2	11	59,12
14	12	1	12	113,02	21	30	2	11	73,77
15	9	1	33	31,87	21	31	2	10	46,41
15	24	1	12	32,44	21	44	2	14	46,45
17	2	1	18	44,56					
17	3	1	12	28,92					
17	13	1	52	88,45					
18	1	1	18	16,84					
18	14	1	62	48,39					
18	15	1	28	20,64					
18	44	1	26	24,61					
19	27	1	16	19,05					
19	35	1	20	8,70					
20	34	1	22	25,91					
21	30	1	50	73,77					
21	31	1	26	46,41					
21	33	1	14	14,70					

Tabla A.3: Escenario 1 - Salida 2 niveles

La Tabla A.3 muestra las asignaciones ZCD-Local, discriminadas por tipo de vehículo. La sección de la izquierda agrupa las asignaciones de vehículos tipo 1, mientras que la sección de la derecha contiene las de tipo 2. En las secciones, la primera y segunda columna son los identificadores de la ZCD y del local respectivamente. Luego le siguen el tipo de vehículo, unidades temporales asignadas y la distancia entre ambos puntos.

ZCD	LC	Asig.	Dist.	ZCD	LC	Asig.	Dist.	ZCD	LC	Asig.	Dist.
11	92	32	33,723	57	15	18	26,589	53	102	34	104,201
66	72	28	41,092	45	65	48	65,567	13	126	12	68,978
12	7	42	51,897	7	164	32	23,786	33	122	36	34,527
62	31	36	33,695	57	16	58	28,989	65	59	20	28,991
61	80	22	38,862	62	35	34	47,014	60	40	22	19,295
11	94	22	45,091	53	104	30	109,398	60	39	16	8,822
17	157	32	45,238	33	124	72	65,310	42	148	48	37,666
12	9	22	26,150	12	49	22	33,208	48	134	32	59,555
45	141	24	21,288	60	38	36	39,977	45	66	28	52,539
48	188	22	34,143	48	152	62	25,569	53	106	28	81,370
53	156	30	54,154	47	138	60	41,128	65	62	30	54,118
42	118	32	63,035	47	191	30	65,798	29	199	34	32,399
29	172	14	65,964	53	86	26	33,023	12	51	34	44,032
65	63	10	71,134	62	63	24	73,381	57	17	14	19,354
61	78	66	22,324	13	101	16	98,438	66	64	60	69,673
26	113	40	23,192	47	136	24	15,301	7	166	38	8,551
20	182	56	35,734	20	180	26	18,172	61	77	38	15,188
29	203	28	41,324	29	201	34	32,030	59	27	56	23,554
61	55	68	45,602	53	170	18	34,457	59	53	28	28,580
65	57	54	29,774	11	89	36	43,771	60	47	24	44,103
29	210	26	29,185	17	159	54	81,843	58	212	30	58,010
45	145	14	38,500	8	108	42	44,889	62	28	16	63,245
35	209	38	45,768	20	177	46	32,145	7	165	24	13,166
59	28	30	29,607	45	143	24	12,429	26	115	38	24,161
35	194	28	53,742	59	54	36	23,750	58	101	10	89,376
12	12	16	50,751	11	93	60	38,540	60	43	44	35,536
11	98	10	55,944	66	71	38	30,090	7	160	14	105,073
18	206	52	21,761	26	117	42	42,146	13	130	38	28,006
8	107	42	59,581	58	23	62	26,317	33	61	50	44,175
58	2	48	68,266	17	184	32	33,431	58	24	24	33,643
7	163	32	36,392	35	196	58	28,859	20	179	68	11,812
47	193	16	76,486	59	25	68	18,696	42	150	26	54,219
58	21	40	36,490	65	56	18	42,559	8	168	50	28,528
11	91	22	28,002	62	36	14	47,095	17	186	60	59,413
12	48	38	42,175	58	1	36	53,706	33	121	32	14,657
61	76	56	29,422	29	174	14	54,542	62	34	62	22,777
18	154	38	47,296	53	88	32	37,663	7	207	44	25,109
60	46	20	38,394	42	213	18	41,628	65	208	64	37,576
29	198	34	39,528	29	205	12	61,563	7	109	30	37,665
18	111	66	47,721	12	5	14	68,939	13	125	28	69,144
48	190	20	71,531	53	103	42	110,673	53	84	10	36,844
13	129	44	36,033	7	81	40	19,909	47	135	28	44,877
58	211	24	48,855	60	41	40	17,534	42	147	58	51,757
48	153	34	14,090	53	85	26	24,381	58	20	38	52,187
8	167	24	14,241	8	169	28	41,590	12	204	12	42,564
26	114	64	18,271	12	50	72	65,695	17	158	64	80,661
20	181	76	25,652	42	149	28	25,742	60	44	26	44,392
47	137	30	26,543	48	133	44	35,335	12	4	18	78,425
12	8	38	27,723	62	33	22	17,337	57	13	34	27,794
48	187	50	38,086	12	11	34	35,288	58	22	38	33,697
53	171	14	38,209	18	131	26	36,119	45	142	34	15,234
53	87	22	27,305	66	69	22	42,424	29	173	30	48,964
18	183	30	21,281	33	123	48	50,558	45	67	64	37,120
57	18	30	28,702	13	100	34	91,623	66	73	28	36,310
61	79	40	27,871	29	200	60	29,421	35	195	54	23,537
12	10	24	30,259	7	162	54	44,614	60	42	54	26,565
45	68	38	40,935	60	45	66	54,839	12	6	32	65,485
45	140	32	43,057	59	26	28	17,428	11	95	56	45,018
47	192	36	44,018	60	37	36	51,521	8	83	34	52,695
48	151	30	49,305	11	90	18	30,105	13	98	14	73,327
20	178	40	27,885	26	116	36	26,758	62	29	44	53,721
45	144	48	28,433	8	82	38	23,531	29	202	10	31,291
29	175	14	38,581	42	119	22	47,207	45	146	44	54,850
12	52	18	54,537	18	110	40	18,997	65	60	18	39,197
65	58	54	19,548	20	176	34	47,263	7	161	18	63,755
47	139	48	60,975	48	189	26	51,854	62	30	26	36,842
13	99	62	82,430	11	97	62	53,121	62	32	60	23,639
66	70	16	35,429	57	20	30	40,664	48	132	62	28,776
17	185	32	36,179	66	74	68	53,179	18	155	44	50,216
57	19	30	39,435	12	3	32	75,950	33	120	26	26,971
35	197	52	40,650	57	14	36	18,303				
13	96	22	52,137	13	128	32	37,044				

Tabla A.4: Escenario 2 - Evaluación Metaheurística - Tipo 1

ZCD	LC	Asig.	Dist.	ZCD	LC	Asig.	Dist.	ZCD	LC	Asig.	Dist.
29	202	19	31,291	48	151	14	49,305	29	198	36	39,528
47	137	17	26,543	65	39	12	109,433	17	158	26	80,661
26	112	29	38,586	8	108	30	44,889	65	34	21	36,439
18	183	18	21,281	18	131	21	36,119	58	211	20	48,855
59	49	18	53,528	11	100	10	74,239	65	58	42	19,548
58	104	10	102,838	53	84	16	36,844	42	119	10	47,207
57	13	21	27,794	59	54	16	23,750	48	189	17	51,854
57	14	37	18,303	66	71	23	30,090	65	79	31	70,393
26	180	16	31,662	29	200	12	29,421	57	19	34	39,435
65	60	14	39,197	29	175	11	38,581	13	96	12	52,137
26	116	16	26,758	11	4	16	82,284	26	114	23	18,271
17	185	10	36,179	53	87	17	27,305	66	70	25	35,429
33	120	10	26,971	29	173	31	48,964	48	187	20	38,086
59	124	16	47,686	53	88	24	37,663	47	192	15	44,018
66	65	15	69,862	66	73	36	36,310	47	139	30	60,975
11	97	22	53,121	11	95	23	45,018	58	2	35	68,266
59	52	11	36,338	35	195	16	23,537	8	167	14	14,241
18	110	18	18,997	59	47	19	101,402	11	102	30	89,220
57	20	14	40,664	58	21	17	36,490	11	92	22	33,723
57	18	18	28,702	53	85	16	24,381	11	91	11	28,002
42	146	14	66,318	11	90	12	30,105	57	44	11	83,360
59	50	15	61,612	48	134	12	59,555	65	36	13	49,342
59	26	14	17,428	7	109	10	37,665	48	132	36	28,776
42	144	13	92,734	33	122	29	34,527	59	5	22	82,347
65	40	12	112,765	59	126	16	33,760	18	155	13	50,216
66	69	22	42,424	65	78	11	75,940	53	106	17	81,370
66	64	13	69,673	26	179	23	30,226	66	63	13	69,736
57	43	21	92,216	42	148	16	37,666	29	199	31	32,399
8	82	32	23,531	17	159	14	81,843	65	80	20	59,402
7	166	35	8,551	58	22	12	33,697	11	89	14	43,771
59	27	13	23,554	17	157	32	45,238	47	141	10	91,472
11	94	23	45,091	29	174	28	54,542	53	156	17	54,154
53	86	18	33,023	29	172	11	65,964	59	6	30	60,084
65	59	18	28,991	48	188	16	34,143	58	23	38	26,317
11	101	27	81,055	35	196	12	28,859	26	113	11	23,192
53	170	11	34,457	42	118	32	63,035	7	164	12	23,786
33	32	16	32,202	59	7	30	65,925	48	152	14	25,569
7	162	28	44,614	47	193	19	76,486	47	138	18	41,128
57	11	24	44,227	8	83	14	52,695	7	160	13	105,073
29	201	28	32,030	65	55	11	66,655	29	210	25	29,185
33	61	28	44,175	35	209	25	45,768	18	206	13	21,761
7	161	29	63,755	35	194	13	53,742	17	186	18	59,413
18	182	33	39,310	57	45	22	72,913	11	93	13	38,540
57	15	11	26,589	29	203	12	41,324	7	165	13	13,166
58	103	34	90,763	65	37	19	58,410	17	184	15	33,431
57	12	13	42,077	26	181	14	39,141	26	178	16	46,299
26	117	12	42,146	48	133	17	35,335	18	111	16	47,721
66	143	15	94,839	65	208	29	37,576	57	42	18	101,187
8	168	14	28,528	42	150	13	54,219	66	68	22	58,551
33	33	33	45,739	8	107	32	59,581	42	145	15	82,668
7	207	33	25,109	7	81	13	19,909	13	129	19	36,033
66	77	41	82,301	33	121	27	14,657	58	24	18	33,643
66	74	19	53,179	59	125	10	33,594	48	190	16	71,531
13	130	23	28,006	66	67	17	64,179	66	76	13	68,066
57	10	19	46,609	57	17	14	19,354	47	140	17	69,704
58	3	16	78,025	18	154	21	47,296	59	29	15	40,684
13	127	28	51,072	59	51	28	42,114	65	57	30	29,774
33	123	13	50,558	47	135	13	44,877	53	171	13	38,209
59	204	19	84,863	42	147	13	51,757	57	8	20	49,144
59	25	14	18,696	11	98	11	55,944				
42	149	24	25,742	66	66	17	64,188				

Tabla A.5: Escenario 2 - Evaluación Metaheurística - Tipo 2

La Tabla A.4 y la Tabla A.5 muestran las asignaciones ZCD-Local, discriminadas por tipo de vehículo. La Tabla A.4 agrupa las asignaciones de vehículos tipo 1, mientras que la Tabla A.5 contiene las de tipo 2. En cada tabla, la primera y segunda columna son los identificadores de la ZCD y del local respectivamente. Luego le siguen unidades temporales asignadas y la distancia entre ambos puntos.

Semilla	N	Capacidad	Dem. Media	Dem. desv	Tipos	Valor	Tiempo(ms)	Gap(%)
17851	20	20	15	2	1	0	0	0
17851	20	20	15	2	2	0	0	0
17851	20	20	15	5	1	484077,9	180250	1,26
17851	20	20	15	5	2	439501,8	180170	1,54
17851	20	20	20	5	1	0	0	0
17851	20	20	20	5	2	0	0	0
17851	20	20	20	10	1	0	0	0
17851	20	20	20	10	2	0	0	0
27965	20	20	15	2	1	0	0	0
27965	20	20	15	2	2	0	0	0
27965	20	20	15	5	1	0	0	0
27965	20	20	15	5	2	0	0	0
27965	20	20	20	5	1	0	0	0
27965	20	20	20	5	2	0	0	0
27965	20	20	20	10	1	0	0	0
27965	20	20	20	10	2	0	0	0
30927	20	20	15	2	1	0	0	0
30927	20	20	15	2	2	0	0	0
30927	20	20	15	5	1	0	0	0
30927	20	20	15	5	2	0	0	0
30927	20	20	20	5	1	0	0	0
30927	20	20	20	5	2	0	0	0
30927	20	20	20	10	1	0	0	0
30927	20	20	20	10	2	0	0	0
32454	20	20	15	2	1	0	0	0
32454	20	20	15	2	2	0	0	0
32454	20	20	15	5	1	0	0	0
32454	20	20	15	5	2	0	0	0
32454	20	20	20	5	1	0	0	0
32454	20	20	20	5	2	0	0	0
32454	20	20	20	10	1	0	0	0
32454	20	20	20	10	2	0	0	0
40367	20	20	15	2	1	0	0	0
40367	20	20	15	2	2	0	0	0
40367	20	20	15	5	1	0	0	0
40367	20	20	15	5	2	0	0	0
40367	20	20	20	5	1	0	0	0
40367	20	20	20	5	2	0	0	0
40367	20	20	20	10	1	0	0	0
40367	20	20	20	10	2	0	0	0
47971	20	20	15	2	1	0	0	0
47971	20	20	15	2	2	0	0	0
47971	20	20	15	5	1	0	0	0
47971	20	20	15	5	2	0	0	0
47971	20	20	20	5	1	0	0	0
47971	20	20	20	5	2	0	0	0
47971	20	20	20	10	1	0	0	0
47971	20	20	20	10	2	0	0	0
63755	20	20	15	2	1	0	0	0
63755	20	20	15	2	2	0	0	0
63755	20	20	15	5	1	0	0	0
63755	20	20	15	5	2	0	0	0
63755	20	20	20	5	1	0	0	0
63755	20	20	20	5	2	0	0	0
63755	20	20	20	10	1	0	0	0
63755	20	20	20	10	2	0	0	0
72073	20	20	15	2	1	0	0	0
72073	20	20	15	2	2	0	0	0
72073	20	20	15	5	1	0	0	0
72073	20	20	15	5	2	0	0	0
72073	20	20	20	5	1	0	0	0
72073	20	20	20	5	2	0	0	0
72073	20	20	20	10	1	0	0	0
72073	20	20	20	10	2	0	0	0
75537	20	20	15	2	1	0	0	0
75537	20	20	15	2	2	0	0	0
75537	20	20	15	5	1	0	0	0
75537	20	20	15	5	2	0	0	0
75537	20	20	20	5	1	0	0	0
75537	20	20	20	5	2	0	0	0
75537	20	20	20	10	1	641020,6	180220	0,07
75537	20	20	20	10	2	597899,1	180250	3,32
98771	20	20	15	2	1	0	0	0
98771	20	20	15	2	2	0	0	0
98771	20	20	15	5	1	0	0	0
98771	20	20	15	5	2	0	0	0
98771	20	20	20	5	1	0	0	0
98771	20	20	20	5	2	0	0	0
98771	20	20	20	10	1	0	0	0
98771	20	20	20	10	2	0	0	0

Tabla A.6: 1.600 Casos - Comparación 1

La Tabla A.6 muestra un resumen de distintas ejecuciones realizadas por el método exacto, estas buscan abrir 20 ZCDs con la capacidad que resulta del cociente de la demanda agregada y 20, utilizando una demanda emulada.

Las columnas son: la semilla utilizada para la generación de datos, la cantidad de ZCDs a abrir (N), el factor capacidad o cociente, la demanda media utilizada, la desviación de la demanda, la configuración de tipos de vehículos (siendo 1 que todas las zonas acepten vehículos tipo 2, mientras que el valor 2 significa que el 50% de las ZCD aceptan solo vehículos tipo 1), el valor objetivo obtenido, el tiempo utilizado y por último el GAP.

Semilla	Dem. Media	Dem. desv	Tipos	Exacto			Metaheurística		Análisis	
				Valor	Tiempo(ms)	Gap(%)	Valor	Tiempo(ms)	Valor(%)	Tiempo(%)
17851	15	2	1	482851,1	177050	0,01	493724,831	1640	2,25	0,93
17851	15	2	2	439311,5	180130	0,63	451033,5206	1858	2,67	1,03
17851	15	5	1	464402,5	180390	0,02	479563,1084	1221	3,26	0,68
17851	15	5	2	424828,3	180200	0,85	440832,0002	1265	3,77	0,70
17851	20	5	1	624556,1	49110	0,01	636912,5155	1891	1,98	3,85
17851	20	5	2	567859,6	180130	0,05	583839,9396	2040	2,81	1,13
17851	20	10	1	607025,1	32160	0,01	624092,5755	1786	2,81	5,55
17851	20	10	2	555284,7	180130	0,56	570811,7779	2022	2,80	1,12
27965	15	2	1	501500,7	180450	0,02	519023,1404	1774	3,49	0,98
27965	15	2	2	459098,4	180200	0,02	479557,5201	1930	4,46	1,07
27965	15	5	1	487309	95200	0,01	502943,2901	1400	3,21	1,47
27965	15	5	2	446646,7	109760	0,01	467123,8227	1624	4,58	1,48
27965	20	5	1	643870,6	80660	0,01	661997,0599	2132	2,82	2,64
27965	20	5	2	589144,5	61700	0,05	611571,7419	2353	3,81	3,81
27965	20	10	1	628808,5	38230	0,01	644187,4969	1883	2,45	4,93
27965	20	10	2	577000,7	52140	0,02	599901,7262	1581	3,97	3,03
30927	15	2	1	499515,4	182760	0,04	519074,2399	1829	3,92	1,00
30927	15	2	2	459074,3	180230	0,61	479964,4185	1631	4,55	0,90
30927	15	5	1	489331,2	182800	0,02	512199,4232	1072	4,67	0,59
30927	15	5	2	451454	180140	0,07	467330,2151	1587	3,52	0,88
30927	20	5	1	649994,6	180250	0,01	670848,9831	2265	3,21	1,26
30927	20	5	2	596734,5	180080	0,75	611836,4378	2393	2,53	1,33
30927	20	10	1	646204,9	180390	0,02	667947,4689	1829	3,36	1,01
30927	20	10	2	593816,7	89530	0,01	612697,332	2492	3,18	2,78
32454	15	2	1	502234,1	180340	0,02	518609,7124	1897	3,26	1,05
32454	15	2	2	458702,9	180170	0,49	473144,4885	1596	3,15	0,89
32454	15	5	1	490317,7	180190	0,03	511138,0734	1491	4,25	0,83
32454	15	5	2	449717,2	180160	0,09	468043,1148	1646	4,07	0,91
32454	20	5	1	652238,8	180110	0,24	675347,5552	1876	3,54	1,04
32454	20	5	2	594123,4	112010	0,01	612559,7978	2582	3,10	2,31
32454	20	10	1	643743,9	46020	0,01	668772,6227	1761	3,89	3,83
32454	20	10	2	592730,8	63990	0,08	615933,846	1594	3,91	2,49
40367	15	2	1	500313,8	181190	0,06	524416,5915	1549	4,82	0,85
40367	15	2	2	451649,8	180110	0,04	466451,3748	1891	3,28	1,05
40367	15	5	1	493213,1	180340	0,06	513539,1465	1388	4,12	0,77
40367	15	5	2	442275,2	180160	0,06	455251,9537	1686	2,93	0,94
40367	20	5	1	647645,9	114380	0,01	673610,657	1378	4,01	1,20
40367	20	5	2	583136,4	180220	0,37	606096,5215	2655	3,94	1,47
40367	20	10	1	646486,2	183800	0,03	677581,9171	1892	4,81	1,03
40367	20	10	2	581467,9	82970	0,05	598727,5962	2428	2,97	2,93
47971	15	2	1	496546,1	57920	0,01	507430,7317	1373	2,19	2,37
47971	15	2	2	447864,2	180160	0,02	459761,1175	1252	2,66	0,69
47971	15	5	1	488362,5	62190	0,01	504863,9912	1259	3,38	2,02
47971	15	5	2	440701,3	180190	0,03	451545,87	2101	2,46	1,17
47971	20	5	1	649730,2	74140	0,02	667837,2329	1905	2,79	2,57
47971	20	5	2	586643,3	59970	0,01	600619,8686	3093	2,38	5,16
47971	20	10	1	647010,5	21190	0,02	662463,7287	1915	2,39	9,04
47971	20	10	2	585452,6	56190	0,01	600796,1173	2015	2,62	3,59
63755	15	2	1	497640,9	85630	0,01	517803,454	1251	4,05	1,46
63755	15	2	2	456783,8	180170	0,02	473223,3932	2084	3,60	1,16
63755	15	5	1	486052,3	74470	0	502612,5475	1839	3,41	2,47
63755	15	5	2	446035,6	108590	0,01	464740,3769	1551	4,19	1,43
63755	20	5	1	644149,5	35310	0	672024,132	1624	4,33	4,60
63755	20	5	2	591981,8	180140	0,01	610208,2372	2666	3,08	1,48
63755	20	10	1	636758,5	32950	0,02	660370,3835	1403	3,71	4,26
63755	20	10	2	582468,8	180190	0,08	604597,8045	1788	3,80	0,99
72073	15	2	1	493979,7	180470	0,03	507378,037	1465	2,71	0,81
72073	15	2	2	454303	180170	0,02	468938,6713	1313	3,22	0,73
72073	15	5	1	475475,4	180310	0,06	498069,9445	1163	4,75	0,65
72073	15	5	2	437711,8	180190	0,09	451470,4566	2012	3,14	1,12
72073	20	5	1	632298,3	47890	0,02	648854,0556	2537	2,62	5,30
72073	20	5	2	581739,6	180170	0,34	598119,3092	2132	2,82	1,18
72073	20	10	1	618730,9	180980	0,03	638385,0587	1882	3,18	1,04
72073	20	10	2	567787,7	134560	0,02	592928,1181	1275	4,43	0,95
75537	15	2	1	487685	181030	0,03	508241,876	1570	4,22	0,87
75537	15	2	2	438367,7	38440	0,01	453125,9125	1347	3,37	3,50
75537	15	5	1	473086,6	180140	0,05	482771,4427	2340	2,05	1,30
75537	15	5	2	426045,8	98480	0,01	441082,2674	1537	3,53	1,56
75537	20	5	1	627929,2	180470	0,02	652853,6074	1491	3,97	0,83
75537	20	5	2	566183	73470	0,01	595782,0442	1598	5,23	2,18
75537	20	10	1	619258,5	138340	0,01	650826,612	1158	5,10	0,84
75537	20	10	2	557934,3	180220	0,02	577001,4243	1320	3,42	0,73
98771	15	2	1	500191,9	92250	0,01	514741,7082	1398	2,91	1,52
98771	15	2	2	454004	180200	0,05	465996,3022	1543	2,64	0,86
98771	15	5	1	485458	180060	0,02	498985,6917	1534	2,79	0,85
98771	15	5	2	441876,5	180300	0,05	450370,8709	1880	1,92	1,04
98771	20	5	1	642527,7	73330	0,01	659184,8133	1599	2,59	2,18
98771	20	5	2	584749,9	180250	0,02	597201,5322	2769	2,13	1,54
98771	20	10	1	627180,9	19670	0,01	645184,8695	1731	2,87	8,80
98771	20	10	2	573434,3	105000	0,02	589070,2342	1447	2,73	1,38

Tabla A.7: 1.600 Casos - Comparación 2
Factor Capacidad: 20 - ZCD a abrir: 21

La Tabla A.7 muestra un resumen de distintas ejecuciones además de un análisis, en este caso se busca abrir 21 ZCDs con la capacidad que resulta del cociente de la demanda agregada y 20, utilizando una demanda emulada.

Las columnas son: la semilla utilizada para la generación de datos, la demanda media utilizada, la desviación de la demanda, la configuración de tipos de vehículos (siendo 1 que todas las zonas acepten vehículos tipo 2, mientras que el valor 2 significa que el 50% de las ZCD aceptan solo vehículos tipo 1); las siguientes tres columnas refieren al método exacto siendo: valor objetivo, tiempo y GAP; las siguientes dos columnas refieren a la Metaheurística mostrando el valor objetivo y el tiempo utilizado; la siguiente columna muestra el porcentaje de variación en el valor objetivo y la última columna muestra el porcentaje de tiempo que uso la Metaheurística tomando como total el tiempo utilizado por el exacto.

Semilla	Dem. Media	Dem. desv	Tipos	Exacto			Metaheurística		Análisis	
				Valor	Tiempo(ms)	Gap(%)	Valor	Tiempo(ms)	Valor(%)	Tiempo(%)
17851	15	2	1	442931,7	5170	0	445208,7973	950	0,51	18,38
17851	15	2	2	407252,2	12360	0,02	409956,6355	862	0,66	6,97
17851	15	5	1	427380,4	4440	0	429279,7382	1032	0,44	23,24
17851	15	5	2	395753,2	5390	0	397779,9229	1429	0,51	26,51
17851	20	5	1	574357,2	6700	0	576893,5248	1526	0,44	22,78
17851	20	5	2	530573,2	11730	0	533366,1641	1938	0,53	16,52
17851	20	10	1	560608	4670	0	563020,7879	1112	0,43	23,81
17851	20	10	2	519451,9	9480	0,05	519806,2687	1613	0,07	17,01
27965	15	2	1	463695,3	4950	0,01	465508,7055	959	0,39	19,37
27965	15	2	2	427005,4	10810	0,09	428871,9802	848	0,44	7,84
27965	15	5	1	451452,7	5250	0,01	453487,2501	1263	0,45	24,06
27965	15	5	2	416920,9	9330	0,06	418662,4137	1224	0,42	13,12
27965	20	5	1	595554	4830	0,01	597829,413	1698	0,38	35,16
27965	20	5	2	549156,5	11000	0,04	551725,6934	1332	0,47	12,11
27965	20	10	1	583256,9	4800	0	586124,3289	1859	0,49	38,73
27965	20	10	2	538433,4	8730	0,08	542160,402	984	0,69	11,27
30927	15	2	1	460188,1	7270	0,01	460921,3331	1233	0,16	16,96
30927	15	2	2	424699,4	10020	0,04	427457,2944	969	0,65	9,67
30927	15	5	1	451743,6	4810	0,01	452955,2169	1407	0,27	29,25
30927	15	5	2	419717,7	6340	0,02	421003,5272	1838	0,31	28,99
30927	20	5	1	597874,7	4390	0	600034,0482	1227	0,36	27,95
30927	20	5	2	551240,5	8280	0,02	553931,6894	1795	0,49	21,68
30927	20	10	1	594569,3	3580	0	595983,8391	1396	0,24	38,99
30927	20	10	2	551629	8520	0,01	552019,3073	2455	0,07	28,81
32454	15	2	1	463456,9	3410	0,01	464938,0056	1132	0,32	33,20
32454	15	2	2	424824,1	9580	0,13	425457,7245	1219	0,15	12,72
32454	15	5	1	451932,2	4830	0,01	453092,1507	1184	0,26	24,51
32454	15	5	2	415923,5	14330	0,01	416328,7937	1807	0,10	12,61
32454	20	5	1	598804,9	3560	0	600707,2597	1514	0,32	42,53
32454	20	5	2	550320,1	11880	0,02	552172,5439	1561	0,34	13,14
32454	20	10	1	594784,7	4740	0,07	596550,2692	1162	0,30	24,51
32454	20	10	2	547059,5	11580	0,03	548138,2055	1397	0,20	12,06
40367	15	2	1	459369,5	7420	0,01	461250,1234	1194	0,41	16,09
40367	15	2	2	418448,1	10550	0,11	420186,2187	840	0,42	7,96
40367	15	5	1	452977,1	5770	0,01	456839,0583	1061	0,85	18,39
40367	15	5	2	411323,4	7110	0,04	412209,4492	1021	0,22	14,36
40367	20	5	1	595862,5	6160	0,05	598207,8675	1526	0,39	24,77
40367	20	5	2	542691,6	9700	0,03	544516,2691	1070	0,34	11,03
40367	20	10	1	596065,9	5970	0	600235,2816	1196	0,70	20,03
40367	20	10	2	541421,5	7090	0	543510,2932	1468	0,39	20,71
47971	15	2	1	460913,5	5970	0,02	462433,9081	961	0,33	16,10
47971	15	2	2	416629,9	12660	0,07	419012,0102	1567	0,57	12,38
47971	15	5	1	457049,9	7630	0,03	458389,6226	1013	0,29	13,28
47971	15	5	2	409534,7	4730	0	410298,0235	1178	0,19	24,90
47971	20	5	1	604276,9	4700	0	606321,5862	1216	0,34	25,87
47971	20	5	2	546271,4	8250	0,02	547756,0322	2206	0,27	26,74
47971	20	10	1	606583,5	6750	0	607501,2135	1178	0,15	17,45
47971	20	10	2	543442,7	4440	0	546611,5132	1335	0,58	30,07
63755	15	2	1	459875,4	5670	0	461220,5329	1268	0,29	22,36
63755	15	2	2	422962,4	16700	0	424584,3329	2131	0,38	12,76
63755	15	5	1	447420,4	4720	0,05	449659,0269	1429	0,50	30,28
63755	15	5	2	412105,8	7660	0,01	414935,0867	1273	0,69	16,62
63755	20	5	1	593926,4	4200	0	595804,805	1584	0,32	37,71
63755	20	5	2	546401,3	7440	0	548634,6663	2130	0,41	28,63
63755	20	10	1	583811,9	4230	0,02	587162,2506	1439	0,57	34,02
63755	20	10	2	536953,1	10360	0,02	538090,5073	1712	0,21	16,53
72073	15	2	1	457343,6	8200	0,01	458821,8898	1467	0,32	17,89
72073	15	2	2	419720,9	11910	0,03	423359,2546	1677	0,87	14,08
72073	15	5	1	438704,7	5230	0,05	440410,8522	1159	0,39	22,16
72073	15	5	2	404879,9	4530	0	408679,2683	1706	0,94	37,66
72073	20	5	1	584246,5	8030	0,07	585941,8895	1566	0,29	19,50
72073	20	5	2	538504,3	9830	0,01	541648,6764	2461	0,58	25,04
72073	20	10	1	569277,3	5810	0,02	571137,5352	1092	0,33	18,80
72073	20	10	2	528098,6	4490	0	530446,3557	1615	0,44	35,97
75537	15	2	1	448979,2	4700	0,11	449986,0795	992	0,22	21,11
75537	15	2	2	410421,6	10940	0,06	413418,0019	1258	0,73	11,50
75537	15	5	1	435961,7	6940	0,04	437423,2139	886	0,34	12,77
75537	15	5	2	398490,1	9410	0,07	401050,6911	1540	0,64	16,37
75537	20	5	1	579848,4	8090	0,13	580852,8316	1235	0,17	15,27
75537	20	5	2	529994,3	13340	0,02	535085,2474	1561	0,96	11,70
75537	20	10	1	571388,9	8970	0,03	572784,9709	1068	0,24	11,91
75537	20	10	2	521218,7	10220	0,04	525702,9612	2132	0,86	20,86
98771	15	2	1	461011,5	6190	0	462043,068	969	0,22	15,65
98771	15	2	2	422447,7	12880	0	424316,691	1160	0,44	9,01
98771	15	5	1	447496,2	6500	0,05	448809,7506	1601	0,29	24,63
98771	15	5	2	409684,8	11410	0,04	410897,2064	1634	0,30	14,32
98771	20	5	1	591416,8	5060	0	593073,8027	1216	0,28	24,03
98771	20	5	2	542473,4	9640	0,1	545117,5795	1440	0,49	14,94
98771	20	10	1	578498,6	8030	0,02	580682,0722	1126	0,38	14,02
98771	20	10	2	528460,1	12980	0,01	531438,1717	1240	0,56	9,55

Tabla A.8: 1.600 Casos - Comparación 3
Factor Capacidad: 15 - ZCD a abrir: 24

La Tabla A.8 muestra un resumen de distintas ejecuciones además de un análisis, en este caso se busca abrir 24 ZCDs con la capacidad que resulta del cociente de la demanda agregada y 15, utilizando una demanda emulada.

Las columnas son: la semilla utilizada para la generación de datos, la demanda media utilizada, la desviación de la demanda, la configuración de tipos de vehículos (siendo 1 que todas las zonas acepten vehículos tipo 2, mientras que el valor 2 significa que el 50% de las ZCD aceptan solo vehículos tipo 1); las siguientes tres columnas refieren al método exacto siendo: valor objetivo, tiempo y GAP; las siguientes dos columnas refieren a la Metaheurística mostrando el valor objetivo y el tiempo utilizado; la siguiente columna muestra el porcentaje de variación en el valor objetivo y la última columna muestra el porcentaje de tiempo que uso la Metaheurística tomando como total el tiempo utilizado por el exacto.