

Tipo de documento: Tesis de maestría



Escuela de Negocios. Master in Management + Analytics
**Uso de técnicas de análisis prescriptivo para
optimizar las rutas de distribución de una
PYME**

Autoría: Saavedra Sánchez, Martín
Año: 2024

¿Cómo citar este trabajo?

González Nuñez, E. (2024). "Uso de técnicas de análisis prescriptivo para optimizar las rutas de distribución de una PYME". [Tesis de maestría. Universidad Torcuato Di Tella]. Repositorio Digital Universidad Torcuato Di Tella.
<https://repositorio.utdt.edu/handle/20.500.13098/12938>

El presente documento se encuentra alojado en el Repositorio Digital de la Universidad Torcuato Di Tella bajo una licencia Creative Commons Atribución-No comercial-Compartir igual 4.0 Internacional Dirección: <https://repositorio.utdt.edu>



**UNIVERSIDAD
TORCUATO DI TELLA**

MASTER IN MANAGEMENT + ANALYTICS

**USO DE TÉCNICAS DE ANÁLISIS PRESCRIPTIVO
PARA OPTIMIZAR LAS RUTAS DE DISTRIBUCIÓN
DE UNA PYME**

Martin Saavedra Sanchez

Febrero 2024

Tutor: Juan José Miranda Bront.

Resumen

Es ampliamente conocido el beneficio de la optimización de rutas en grandes empresas de transporte y logística a través de la aplicación de sistemas informáticos. Dado que una gran parte de los transportistas en América Latina, como en otras partes del mundo, son de pequeña envergadura, es importante ayudar a este segmento particular de la industria a reducir los costos de logística como así también el impacto ambiental que esta actividad implica

En este trabajo se aborda el caso de una PYME distribuidora de productos que actualmente se basa en la intuición y la experiencia de sus choferes para el diseño de las rutas. Por medio de la aplicación de técnicas de análisis prescriptivo y heurísticas avanzadas, se busca mejorar la operatoria actual de esta PYME.

Mediante experimentos computacionales utilizando datos reales, se muestra que es posible reducir la duración de las rutas en un 26%, incluyendo una mejor utilización de la capacidad logística, lo que resulta en un potencial ahorro del 34% en los costos logísticos. Además, se analiza la distribución de clientes para identificar oportunidades de mejora en las zonas de entrega y se propone un rediseño de la segmentación de clientes, que brinda la posibilidad de expandir la cartera de clientes manteniendo la misma flota de camiones.

Abstract

The benefit of route optimization in large transportation and logistics companies through the application of computer systems is widely known. Given that a significant portion of carriers in Latin America, as in other parts of the world, are small-scale, it is important to assist this particular segment of the industry in reducing logistics costs as well as the environmental impact associated with this activity.

This study addresses the case of a SME distributing products that currently relies on the intuition and experience of its drivers for route planning. Through the application of prescriptive analysis and advanced heuristic techniques, the aim is to enhance the current operations of this SME.

Through computational experiments using real data, it is shown that it is possible to reduce route durations by 26%, including improved utilization of logistics capacity, resulting in a potential 34% saving in logistics costs. Additionally, the distribution of customers is analyzed to identify improvement opportunities in delivery areas, and a redesign of customer segmentation is proposed, providing the possibility to expand the customer portfolio while maintaining the same fleet of trucks.

Índice

Índice de Tablas.....	4
Índice de Figuras	4
1. Introducción	6
1.1. Contexto	6
1.2. Problema	8
1.3. Objetivo	11
2. Datos	11
3. Metodología	20
3.1. Modelo matemático CVRPTW	21
3.2. Heurísticas y Metaheurísticas	24
3.2.1 Heurística Christofides	25
3.2.2 Cheapest Insertion Heuristic	25
3.2.3. Local Search Algorithms	27
4. Resultados computacionales.....	29
4.1. Escenario 1 (Situación actual)	30
4.1.1. Modelo matemático TSPTW	31
4.1.2. Método heurístico para TSPTW	31
4.2. Escenario 2	38
4.2.1. Método heurístico CVRPTW	38
4.3. Escenario 3	45
4.4. Escenario 4	47
4.4.1. Factibilidad de agrupar zonas	47
4.4.2. Generación de una nueva zonificación	49
5. Análisis de Negocio	54
6. Conclusiones – Posibles mejoras.....	56
7. Referencias	58

Índice de Tablas

Tabla 1. Promedio de camiones utilizados según el día de entrega.....	17
Tabla 2. Cantidad de clientes en zonas de entrega.	17
Tabla 3. Distancia promedio entre clientes	18
Tabla 4. Cantidad de clientes visitados y km recorridos por día de entrega.....	19
Tabla 5. Cantidad de clientes visitados y km recorridos por camión.	19
Tabla 6. Evaluación de heurísticas y metaheurísticas para TSPTW heurístico.	31
Tabla 7. Evaluación de resultados para TSPTW heurístico, tiempos promedio de viaje.....	37
Tabla 8. Evaluación de resultados para TSPTW heurístico, tiempos promedio espera.....	37
Tabla 9. Evaluación de resultados para TSPTW heurístico, kilómetros promedio recorridos....	37
Tabla 10. Evaluación de heurística y metaheurísticas para algoritmo heurístico CVRPTW.	38
Tabla 11. Vehículos Utilizados.	42
Tabla 12. Variación relativa entre promedio de kilómetros recorridos, tiempo total y cantidad de clientes por camión para todas las rutas analizadas.....	42
Tabla 13. Evaluación de resultados CVRPTW, tiempos promedio de viaje respecto a Benchmark.	43
Tabla 14. Evaluación de resultados CVRPTW, tiempos promedio de espera respecto a Benchmark.	43
Tabla 15. Total de clientes con espera y promedio total de espera por recorrido para inicio de recorrido 6hs y 7hs.....	44
Tabla 16. Evaluación resultados CVRPTW, tiempo total recorrido respecto a Benchmark.....	44
Tabla 17. Evaluación resultados CVRPTW, kilómetros recorridos respecto a Benchmark.....	44
Tabla 18. Resultados instancias simuladas.	46
Tabla 19. Proporción adicional de clientes contra Benchmark con 2 y 3 camiones.....	47
Tabla 20. Agrupación de entregas con días consecutivos viernes-sábado.....	48
Tabla 21. Agrupación de entregas con días consecutivos viernes-sábado, resultados.	48
Tabla 22. Distancia promedio de clientes entre clientes para cada zona-cluster con K-Means Restringido.	52
Tabla 23. Solución de CVRTW heurístico para nueva agrupación de zonas.	53
Tabla 24. Matriz de costos operativos de un camión. Con 28 días laborables por mes.....	54

Índice de Figuras

Figura 1. Distribución clientes.....	8
Figura 2. Zonificación de cartera de clientes. Rojo: martes-sábado, Azul: miércoles, Amarillo: jueves, Violeta: viernes, Circulo: Depósito, Círculos: clientes/paradas.	9
Figura 3. Camión utilizado.....	10
Figura 4. Instancia Benchmark correspondiente a un martes. Camiones utilizados: 3, Cliente: 42, total kilómetros recorridos: 115 km, tiempo total de viaje: 230 min.	14
Figura 5. Instancia Benchmark correspondiente a un miércoles. Camiones utilizados: 3, Cliente: 64, total kilómetros recorridos: 232 km, tiempo total de viaje: 374 min.	14
Figura 6. Instancia Benchmark correspondiente a un jueves. Camiones utilizados: 3, Cliente: 65, total kilómetros recorridos: 145 km, tiempo total de viaje: 290 min.	15

Figura 7. Instancia Benchmark correspondiente a un viernes. Camiones utilizados: 3, Cliente: 67, total kilómetros recorridos: 97 km, tiempo total de viaje: 228 min.	15
Figura 8. Instancia Benchmark correspondiente a un sábado. Camiones utilizados: 2, Cliente: 51, total kilómetros recorridos: 72 km, tiempo total de viaje: 172 min.	16
Figura 9. Distancia a depósito.	18
Figura 10. Tiempos de servicio.	20
Figura 11. Instancias correspondientes a un martes; Izq. Benchmark: total kilómetros recorridos: 115 km, tiempo total de viaje: 230 min, Der. Solución TSPTW total kilómetros recorridos: 109 km (-5%), tiempo total de viaje: 222 min (-3%).	32
Figura 12. Instancias correspondientes a un miércoles; Arriba. Benchmark: total kilómetros recorridos: 232 km, tiempo total de viaje: 374 min, Abajo. Solución TSPTW total kilómetros recorridos: 213 km (-8%), tiempo total de viaje: 335 min (-10%).	33
Figura 13. Instancias correspondientes a un jueves; Arriba. Benchmark: total kilómetros recorridos: 148 km, tiempo total de viaje: 292 min, Abajo. Solución TSPTW total kilómetros recorridos: 138 km (-6%), tiempo total de viaje: 263 min (-10%).	34
Figura 14. Instancias correspondientes a un viernes; Arriba. Benchmark: total kilómetros recorridos: 97 km, tiempo total de viaje: 228 min, Abajo. Solución TSPTW total kilómetros recorridos: 84 km (-15%), tiempo total de viaje: 191 min (-16%).	35
Figura 15. Instancias correspondientes a un sábado; Arriba. Benchmark: total kilómetros recorridos: 72 km, tiempo total de viaje: 172 min, Abajo. Solución TSPTW total kilómetros recorridos: 59 km (-18%), tiempo total de viaje: 146 min (-15%).	36
Figura 16. Instancias correspondientes a un martes. Izq. Benchmark: 3 camiones, 115 km totales recorridos, 230 min totales de viaje; Der. Solución CVRPTW: 2 camiones, 75 km (-34%) totales recorridos, 170 min (-26%) totales de viaje.	39
Figura 17. Instancias correspondientes a un miércoles. Arriba. Benchmark: 3 camiones, 232 km totales recorridos, 374 min totales de viaje; Abajo. Solución CVRPTW: 2 camiones, 165 km (-28%) totales recorridos, 291 min (-22%) totales de viaje.	40
Figura 18. Instancias correspondientes a un jueves. Arriba. Benchmark: 3 camiones, 145 km totales recorridos, 290 min totales de viaje; Abajo. Solución CVRPTW: 2 camiones, 113 km (-22%) totales recorridos, 233 min (-19%) totales de viaje.	40
Figura 19. Instancias correspondientes a un viernes. Arriba. Benchmark: 3 camiones, 97 km totales recorridos, 228 min totales de viaje; Abajo. Solución CVRPTW: 2 camiones, 68 km (-30%) totales recorrido, 166 min (-27%) totales de viaje.	41
Figura 20. Instancias correspondientes a un sábado. Izq. Benchmark: 2 camiones, 72 km totales recorridos, 172 min totales de viaje; Der. Solución CVRPTW: 2 camiones, 50 km (-30%) totales recorrido, 125 min (-27%) totales de viaje.	41
Figura 21. Zonas viernes, sábado.	47
Figura 22. Arriba: zonas actuales; Medio: clusters generados con K-Means; Abajo: clusters generados con K-Means restringido.	51
Figura 23. Transición entre zonas actual.	52

1. Introducción

1.1. Contexto

El problema del ruteo vehicular (VRP por sus siglas en inglés) es de gran relevancia en el ámbito empresarial, especialmente para aquellas organizaciones dedicadas a la distribución de productos.

La incorporación de sistemas informáticos para la generación de rutas ha demostrado ser altamente beneficiosa al utilizar métodos computarizados para abordar los desafíos del VRP y lograr una integración completa con los sistemas de información en todas las etapas de planificación y operación. Estas soluciones computacionales han generado ahorros significativos en el transporte a nivel global.

Los algoritmos correspondientes y las herramientas de software utilizadas desempeñan un papel fundamental en la búsqueda de soluciones de alta calidad y factibles para diferentes escenarios del mundo real, todo ello dentro de tiempos de cómputo aceptables. En comparación con enfoques no optimizados, estos métodos permiten obtener importantes reducciones de costos y una mejor utilización de la flota de vehículos. Asimismo, gracias a estas soluciones de planificación, es posible mejorar la automatización, estandarización e integración en los procesos generales de planificación de las organizaciones, lo que a su vez conduce a una planificación más eficiente en términos de tiempo y rentabilidad en comparación con los métodos manuales. (Toth & Vigo, 2014)

El modelo simplificado de VRP se refiere al caso en el cual se cuenta con un solo vehículo, también conocido como problema del viajante de comercio (TSP). Como contrapartida, el caso general incluye múltiples vehículos con distinto tipo de restricciones como, por ejemplo: la capacidad para la carga en cuanto al peso o al volumen máximo de elementos que los vehículos puedan transportar (CVRP), o la ventana de tiempo especificada en la cual los conductores deben recorrer distintas ubicaciones para cumplir con los requerimientos de los clientes (VRPTW) o la combinación de estos (CVRPTW).

La incorporación de estas técnicas de optimización aplicadas al VRP, no sólo impacta en el ámbito empresarial logrando mayor eficacia y rentabilidad, sino que a su vez se traduce en una disminución de las emisiones de gases de efecto invernadero, como el dióxido de carbono (CO₂), producidas por los vehículos. Menos tiempo en la carretera y rutas más eficientes significan una menor quema de combustible y, por lo tanto, una reducción en la huella de carbono de las operaciones de transporte. Asimismo, al implementar técnicas de optimización de ruteo, las

empresas pueden fomentar prácticas más sostenibles en sus operaciones de transporte, lo que actualmente va tomando gran relevancia en la comunidad.

Es notorio y fácilmente apreciable el impacto que tienen estos sistemas al ser aplicados por grandes empresas de transporte y logística, como es el caso de DHL que ha desarrollado e implementado su propio sistema de optimización de ruteo llamado ODD (Optimized Delivery Deployment). Otro caso es el sistema ORION (On-Road Integrated Optimization and Navigation) de UPS, basado en algoritmos avanzados de optimización que permite planificar las rutas de entrega de los conductores de UPS de manera más eficiente. Con un costo de construcción e implementación de más de \$295 millones, se estimó que ORION genera a UPS entre \$300 y \$400 millones de ahorro al año. ORION también contribuye a los esfuerzos de sostenibilidad de UPS al reducir sus emisiones de CO2 en 100.000 toneladas anuales (Holland, 2017).

Las pequeñas y medianas empresas, pese a la falta de capacidades técnicas o recursos financieros para desarrollar o comprar software de generación de rutas, pueden beneficiarse de la aplicación de estas técnicas en su operación diaria. Dado que la mayoría de los transportistas son pequeños en muchos lugares como es el caso de América Latina, es importante que se ayude a este segmento particular de la industria del transporte por carretera (pequeños transportistas) a reducir los costos de logística y su impacto ambiental. (Suzuki, 2015)

En Argentina según lo informado en 2022 por la “Cámara Empresaria de Operadores Logísticos”, la cantidad de transportistas viene incrementándose año a año acumulando un 72% en los últimos 4 años, de los cuales el último año el 74% está en proceso de medición de la huella de carbono, lo que denota un interés por incorporar medidas para su mitigación. Asimismo, el informe muestra que el 91% de los operadores logísticos planifica realizar inversiones para los próximos años en vehículos, equipamiento y sistemas¹.

En este trabajo se aborda el caso de “El Universo del Panadero”, una PYME dedicada a la venta y distribución de materias primas para panaderías y confiterías comprometida en el mercado en el aprovisionamiento de todos los insumos que el industrial panadero necesita para la elaboración de sus productos. Cuenta con un centro de ventas y distribución en la zona de Caseros, brindando atención a la zona noroeste del conurbano bonaerense.

¹ Extraído de: “Informe sobre Operadores Logísticos, Resultados de la encuesta técnica 2022” <https://www.cedol.org.ar/biblioteca-de-logistica.html>

Esta PYME posee una pequeña flota de camiones que a diario distribuye la mercadería vendida. En este trabajo se propone aportar la generación de rutas eficientes explorando técnicas de programación lineal entera (ILP), técnicas heurísticas y metaheurísticas.

1.2. Problema

“El Universo del Panadero” dispone de 3 camiones para dar servicio a alrededor de 395 clientes que van desde panaderías y confiterías hasta mayoristas, industrias y otros comercios, distribuidos geográficamente de la siguiente manera:

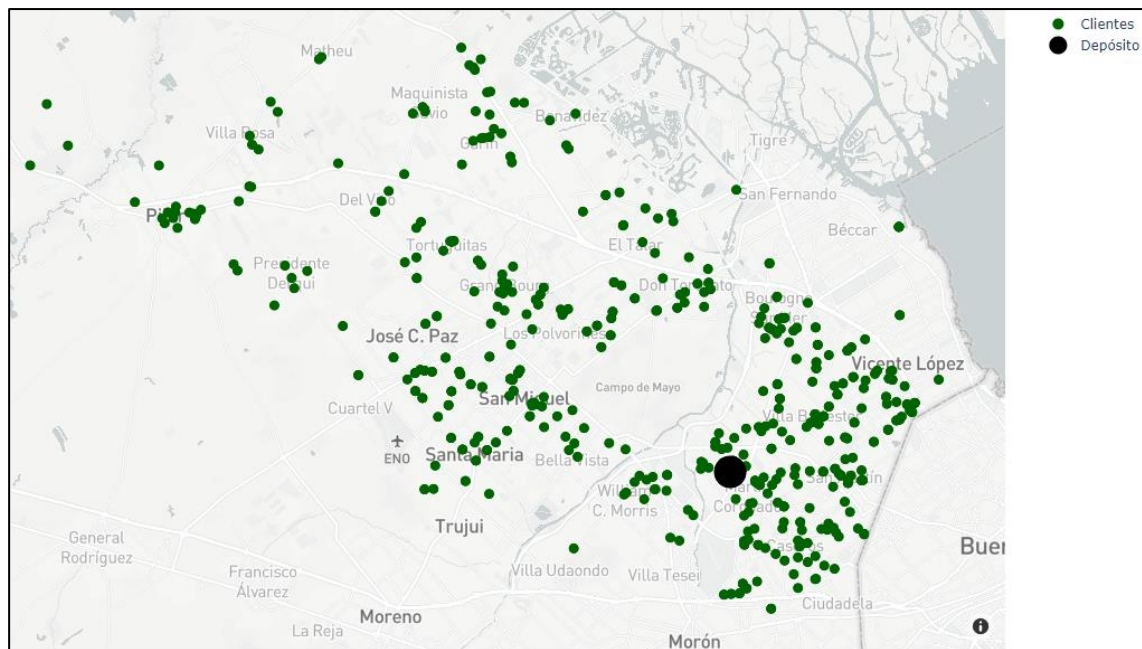


Figura 1. Distribución clientes.

El reparto de mercadería se realiza de martes a sábado, donde cada día tiene asignado una zona previamente delimitada, basada principalmente en los límites municipales. Estas zonas adjudicadas a cada día de la semana, a su vez, están particionadas en 3 sub-zonas a ser atendidas por cada uno de los tres camiones. Los límites de estas zonas permanecen estables, creciendo en cantidad de clientes a medida que la fuerza de ventas capture nuevos. Los tres camiones reparten diariamente en simultaneo en la misma zona con el objetivo de poder redistribuir la carga en caso de que uno de ellos quede inhabilitado.

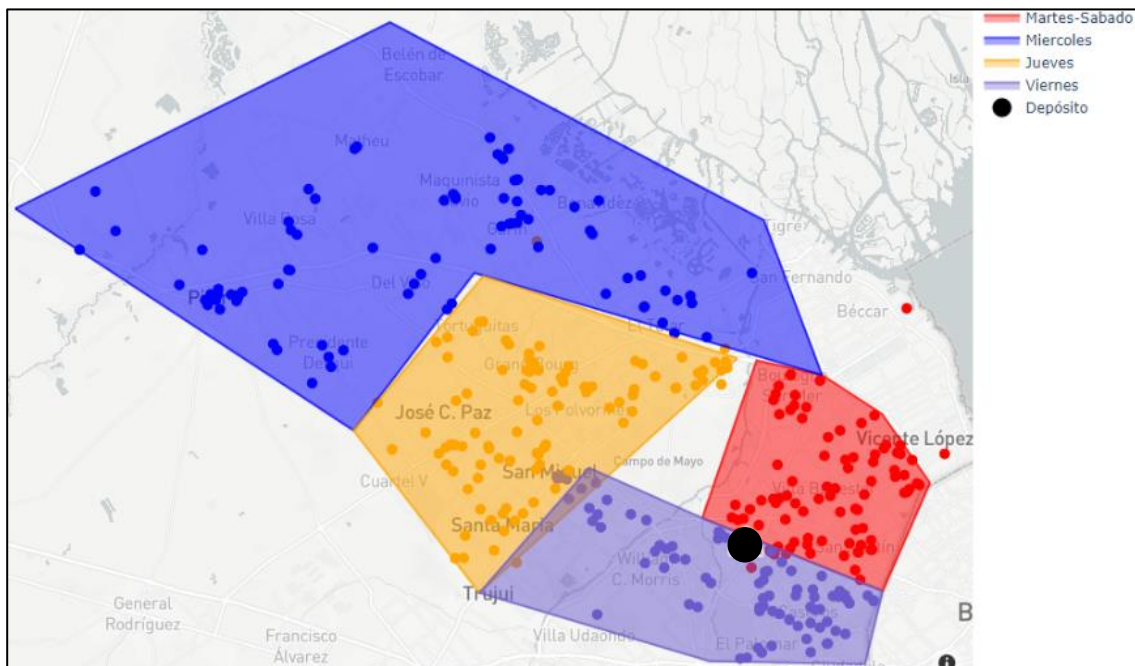


Figura 2. Zonificación de cartera de clientes. Rojo: martes-sábado, Azul: miércoles, Amarillo: jueves, Violeta: viernes, Circulo: Depósito, Círculos: clientes/paradas.

De lunes a viernes, la fuerza de venta levanta pedidos de zonas delimitadas para ser entregadas al siguiente día.

Cada camión cuenta con un chófer y un ayudante que es el encargado de descargar y entregar la mercadería. Actualmente, la programación de las rutas es delegada a los choferes. Al inicio de la jornada laboral, mientras se termina de cargar los camiones, reciben el listado de los clientes que deben visitar sin un orden establecido previamente y con los productos que deben despachar según la zonificación preestablecida. De forma manual, siguiendo su intuición y experiencia, los choferes establecen el orden en que se realizarán las entregas respetando las ventanas horarias de cada cliente.

La tarea del ruteo, delegada a los choferes, si bien es eficaz ya que al final del día se realizan todas las entregas pautadas, no es del todo eficiente en cuanto al ordenamiento de clientes, distancia total, tiempos de la ruta y carga de trabajo adicional al chofer. Esto motiva a la empresa a buscar otros tipos de métodos para resolver el problema de ruteo. Por otro lado, la empresa se encuentra continuamente en la búsqueda de nuevos clientes dentro de las zonas de entrega, por lo cual necesita prever a cuantos clientes más podría brindar servicio con rutas optimizadas sin necesidad de sumar nuevos camiones.

Para poder abordar el problema del ruteo, primero se deben identificar los elementos que los componen:

- Clientes:
 - Ubicación: es necesario contar con la ubicación (latitud y longitud) de los clientes en donde se realizará cada entrega. Esto es indispensable para calcular la matriz de distancias y tiempos de viaje entre clientes y entre clientes y depósito. La matriz de tiempos se desprende de la matriz de distancia al aplicar un factor proporcional a la velocidad promedio de los camiones.
 - Ventana de tiempo: las panaderías y confiterías tienen un horario de recepción de materia prima que va desde las 6 de la mañana hasta las 13 horas, mientras que el resto de los clientes tienen una ventana horaria de recepción que va desde las 8 de la mañana hasta las 13hs.
 - Demanda: productos demandados por cada cliente, medida en kilogramos. Esto será una restricción del problema ya que la cantidad total demandada no debe superar a la capacidad de los camiones.
 - Tiempo de servicio: se refiere al tiempo necesario para descargar la mercadería en cada cliente.
- Depósito; “El Universo del Panadero”, cuenta con un local en la zona de Caseros, con oficinas y espacio para almacenar mercadería con y sin refrigeración.
 - Ubicación: será el punto de inicio, y el destino.
 - Ventana de tiempo: tiene horario de atención de 6hs a 15hs.
- Camiones: actualmente se cuenta con tres camiones Mercedes Benz con equipo de refrigeración.



Figura 3. Camión utilizado

- Capacidad: los camiones tienen una capacidad máxima de 7.000 kilos, que no debe ser superada por la demanda total de los clientes.
- Tiempo de trabajo: será el horario laboral del chofer y ayudante que es de 6hs a 15hs.

1.3. Objetivo

Explorando técnicas de optimización de ruteo de vehículos, aplicada a la operación del “El Universo del Panadero”, se busca en primera instancia cuantificar el potencial de ahorro en costos logísticos, como distancia recorrida (consumo de combustible) y alivianar la carga de trabajo de los choferes, respecto a la operatoria actual. Por otro lado, se evalúa la escalabilidad de los métodos utilizados en diferentes escenarios para entender a cuántos clientes se podría abastecer sin necesidad de incrementar la flota. Por último, se analiza la distribución de clientes para identificar oportunidades de mejorar la zonificación actual.

En síntesis, se busca dar respuesta a las preguntas planteadas en los siguientes escenarios:

- **Escenario 1(situación actual):** ¿Qué potencial de ahorro aportaría la utilización de técnicas de análisis prescriptivo para resolver las instancias respetando la segmentación por zonas actual? (en la cual un determinado chofer tiene asignada una zona para cada día)
- **Escenario 2:** ¿Qué potencial de ahorro aportaría la utilización de técnicas de análisis prescriptivo para resolver las instancias si se modificara la zonificación actual y permitiendo la asignación libre de clientes a camiones?
- **Escenario 3:** ¿A cuántos clientes se podría abastecer con la misma cantidad de camiones, modificando eventualmente la segmentación actual?
- **Escenario 4:** Analizar posibles cambios en la zonificación diaria actual para, por ejemplo, reducir la cantidad de días de entrega necesarios, generar zonas balanceadas, expandir la operación o absorber nuevos clientes.

La generación diaria del ruteo se realizará de forma rápida y práctica, por medio de un ejecutable que tomará el listado de clientes a ser visitados y los productos a entregar, ofreciendo una solución razonable en un corto tiempo para el problema de ruteo, asignando clientes a camiones y determinando un orden de visita.

2. Datos

En el apartado anterior se mencionaron los elementos necesarios para poder modelar el problema de ruteo que son:

- Clientes:
 - Ubicación: latitud y longitud.
 - Ventana de tiempo: rango horario en horas.

- Demanda: medida en kilogramos.
- Tiempo de servicio: se desprenderá de la demanda, medida en minutos.
- Depósito:
 - Ubicación.
 - Ventana de tiempo.
- Camiones:
 - Capacidad.
 - Tiempo de trabajo.

Para desarrollar y evaluar un método de generación de rutas que sea preciso y representativo de la operación, se necesita contar con los datos mencionados anteriormente para una considerable cantidad de días. Estos datos permiten realizar estimaciones que reflejen la variabilidad intrínseca de la operación. Al conjunto de datos relacionados a cada día de entregas se lo llamará instancia.

En este sentido “El Universo del Panadero” brindó, de diferentes fuentes, una serie de datos para poder reconstruir las instancias. Estas son:

- Ventas por camión (descargado de CRM, formato Excel): contiene un listado desordenado de clientes que visitó cada camión con el detalle de los productos entregados. Se cuenta con los siguientes atributos.
 - Fecha de entrega.
 - Identificador del cliente.
 - Tipo de cliente.
 - Dirección.
 - Productos, Cantidades.
 - Camiones asignados.

De este archivo, se pueden determinar los clientes visitados y el tiempo requerido para descargar la mercadería en cada punto de entrega, estimado en función de los kilos, y las ventanas horarias dependiendo del tipo de cliente, pueden ser de 6hs a 13hs o 8hs a 13hs.

Una de las dificultades que presenta este archivo es la normalización de las direcciones, para poder determinar las coordenadas. Una vez normalizadas las direcciones se construyó un script en Python que realiza solicitudes a la API de Google llamada "Google

Maps Geocoding API”². Esta API permite a los desarrolladores convertir direcciones postales en información geográfica precisa, como las coordenadas geográficas (latitud y longitud) y otros detalles relacionados con la dirección.

De este archivo se extraen los datos necesarios para poder reconstruir las instancias, con excepción del orden en que se visitó a cada uno de los clientes.

- Seguimiento vehicular. Por cuestiones de seguridad y gestión de la flota, la empresa tiene contratado un servicio de monitoreo satelital, por el cual cada camión cuenta con un rastreador satelital GPS, que informa cada cierta cantidad de minutos las coordenadas y fecha-hora, a un servidor del cual se puede exportar la ruta de cada camión. Esta información es posible exportarla de la plataforma web de gestión en formato KML³, un formato de archivo basado en XML diseñado principalmente para representar información geoespacial en tres dimensiones.

Combinando las ubicaciones de cada cliente, para cada día, extraídas del archivo de ventas y los datos del seguimiento vehicular, se pudo determinar el orden en el que un camión visitó a cada cliente un determinado día. Esto se logró identificando el punto geolocalizado, reportado por el GPS, más cercano a la ubicación de los clientes y por medio de la meta data asociada a cada reporte se extrajo la hora más próxima a la visita del camión.

Con estas dos fuentes de datos, se pudo reconstruir las instancias que por un lado servirán de input para el método de optimización y por otro lado se tendrá la planificación ejecutada en cada día, incluyendo la cantidad de kilómetros recorridos por cada camión que servirá como Benchmark para poder comparar las soluciones arrojadas por el método de optimización contra un escenario real.

² <https://developers.google.com/maps/documentation/geocoding/overview?hl=es-419>

³ <https://developers.google.com/kml?hl=es-419>

A continuación, se ilustran los recorridos realizados por los camiones, para distintos días de entrega, con distintas métricas asociadas.

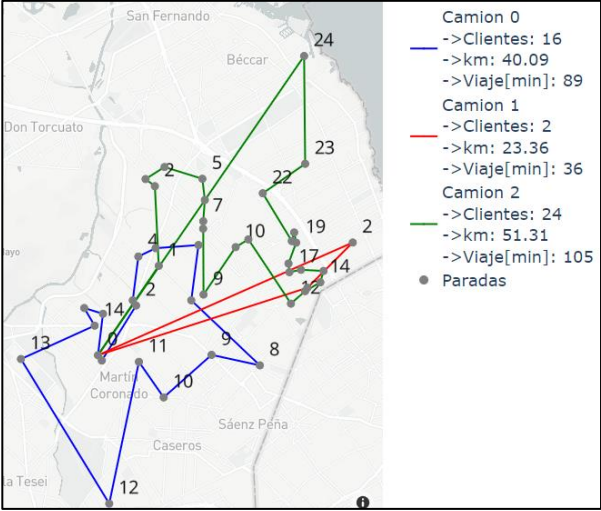


Figura 4. Instancia Benchmark correspondiente a un martes. Camiones utilizados: 3, Cliente: 42, total kilómetros recorridos: 115 km, tiempo total de viaje: 230 min.

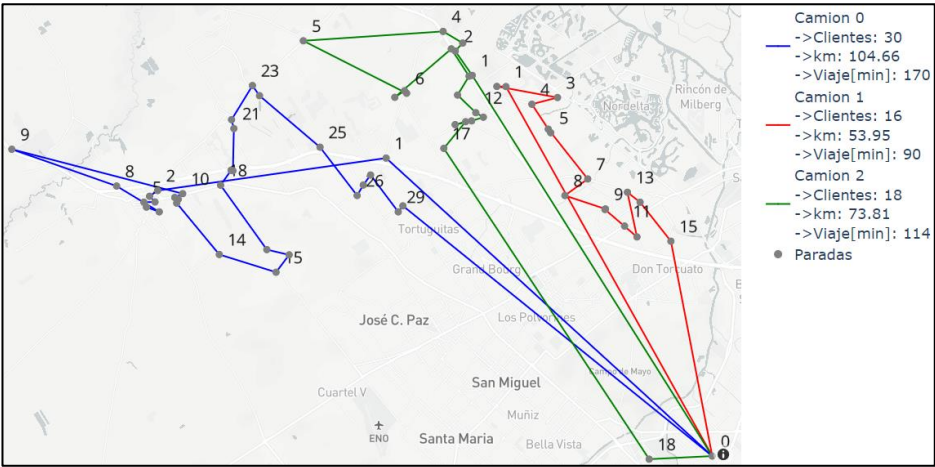


Figura 5. Instancia Benchmark correspondiente a un miércoles. Camiones utilizados: 3, Cliente: 64, total kilómetros recorridos: 232 km, tiempo total de viaje: 374 min.

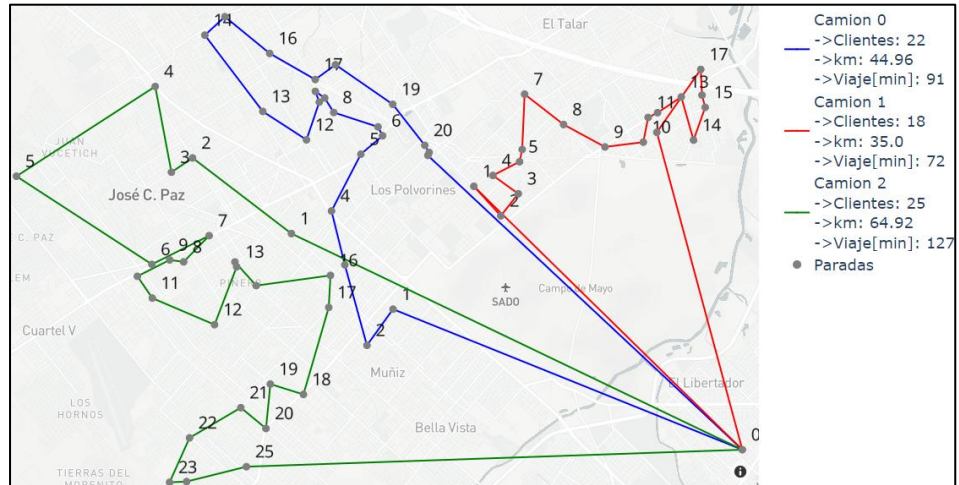


Figura 6. Instancia Benchmark correspondiente a un jueves. Camiones utilizados: 3, Cliente: 65, total kilómetros recorridos: 145 km, tiempo total de viaje: 290 min.

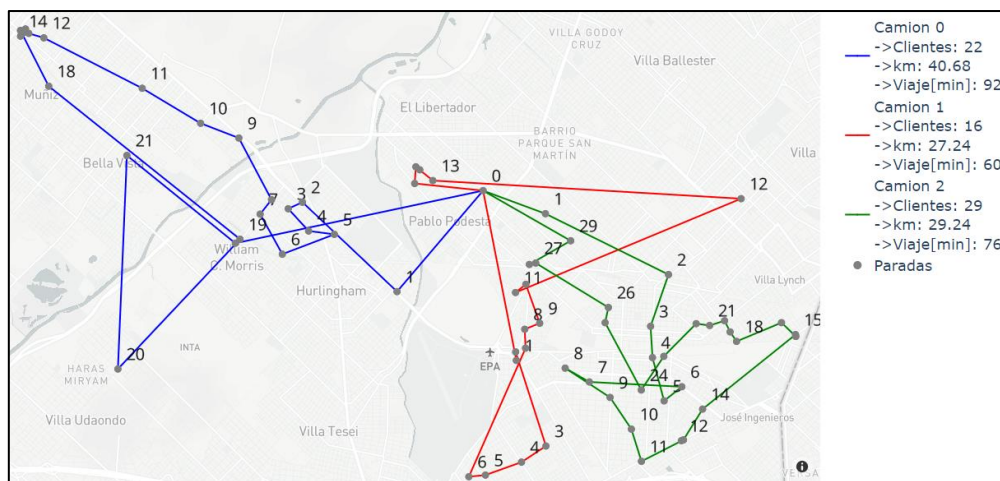


Figura 7. Instancia Benchmark correspondiente a un viernes. Camiones utilizados: 3, Cliente: 67, total kilómetros recorridos: 97 km, tiempo total de viaje: 228 min.

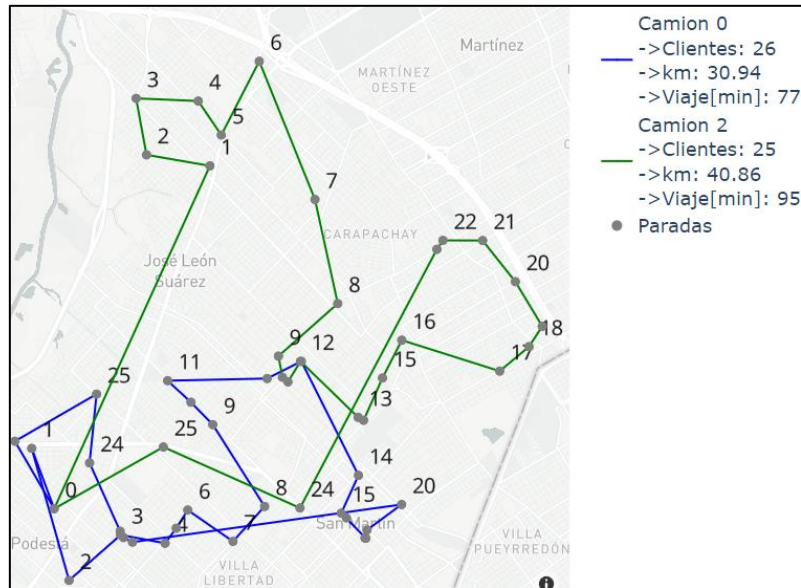


Figura 8. Instancia Benchmark correspondiente a un sábado. Camiones utilizados: 2, Cliente: 51, total kilómetros recorridos: 72 km, tiempo total de viaje: 172 min.

A partir de los recorridos ilustrados en las figuras anteriores y sus métricas asociadas, se puede ver que los días miércoles y jueves se realizan los recorridos más alejados del depósito abarcando zonas más extensas con mayor dispersión de clientes, lo cual se ve reflejado en el total de kilómetros recorridos y tiempo de viaje. Por otro lado, se observa que no todos los días se visitan la misma cantidad de clientes, los cuales varían entre 42 y 67. También se detecta que no todos los días se utilizaron los tres camiones de la flota.

Si bien no es un aspecto de mejora en el cual se enfoca este trabajo, se puede ver que el martes tiene una ruta con sólo 2 clientes, en otras palabras, se observan rutas desbalanceadas.

En las cinco figuras anteriores, principalmente en la Figura 7, se evidencia que hay cruces entre trayectorias de un mismo camión, lo que provoca ineficiencia.

Análisis cuantitativo

Como la operación organiza la entrega por zonas dedicada a cada día de la semana (de martes a sábado), se organiza la información en camiones asignados y se calculan las métricas de interés.

Uso de camiones

No todos los días se utilizan los 3 camiones de la flota, esto se puede deber a una baja demanda o a que alguno de los camiones o choferes se encuentran inhabilitados. Para cada día de entregas se calcula el mínimo, máximo y promedio de camiones utilizados.

Día	Mínimo Camiones Utilizados	Promedio Camiones Utilizados	Máximo Camiones Utilizados
martes	2	2,2	3
miércoles	3	3,0	3
jueves	3	3,0	3
viernes	2	2,8	3
sábado	1	1,8	2

Tabla 1. Promedio de camiones utilizados según el día de entrega.

Para los miércoles y jueves se utilizaron los 3 camiones, los martes y viernes se utilizaron entre 2 y 3 camiones y para los sábados se utilizaron entre 1 y 2.

La cantidad de camiones disponibles en cada instancia será un input del algoritmo a implementar, para poder evaluar si éste ofrece soluciones con la misma cantidad o menor de camiones, lo que se puede traducir en ahorro.

Cartera de clientes

En la Figura 2, se muestra por medio de polígonos las zonas de entrega establecidas y con círculos los clientes, los cuales se extrajeron de los datos provistos.

En total se contabilizó 395 clientes de los cuales el 90% de estos clientes son visitados 1 vez a la semana, por lo general el mismo día de la semana. A continuación, se muestra la cantidad total de clientes que se visitaron según el día de entrega. Esto no quiere decir que todos ellos fueron visitados un mismo día.

Día Entrega	Cantidad de Clientes
martes	72
miércoles	87
jueves	100
viernes	90
sábado	88

Tabla 2. Cantidad de clientes en zonas de entrega.

En su gran mayoría, alrededor del 75% de los clientes son panaderías y confiterías y el 25% restante se distribuye entre mayoristas, industrias y otros comercios

Complementando la Figura 2 con el mapa de zonas correspondiente a cada día y con la distribución de los clientes, se calcula la distancia promedio de cada cliente al depósito y la distancia promedio entre clientes (DPC).

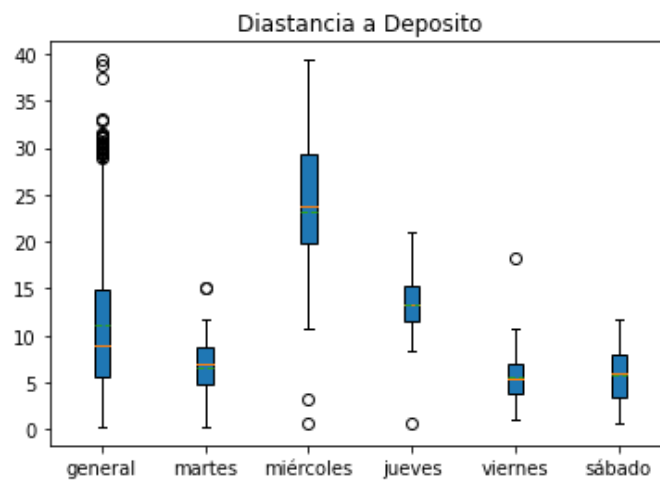


Figura 9. Distancia a depósito.

Día Entrega	DPC
martes	5,6
miércoles	12,1
jueves	9,0
viernes	5,7
sábado	5,2
Promedio	10,3

Tabla 3. Distancia promedio entre clientes

Tanto en la Figura 8 como en la Tabla 3, se puede apreciar al igual que en el mapa de la Figura 2, que la zona de entrega correspondiente a los miércoles y jueves es la más distante al depósito. Particularmente para la zona del miércoles se pudo ver tanto visualmente como a partir de los datos que posee una mayor dispersión de clientes ya que es la zona más alejada del centro metropolitano, con menor densidad de población.

Análisis de los recorridos

A continuación, se cuantificará la cantidad mínima, promedio y máximo de clientes visitados y la distancia total de la ruta medida en kilómetros. Se puede ver que los kilómetros recorridos coinciden con la tendencia del gráfico de cajas anterior donde las zonas de entrega de los miércoles y jueves, al estar más alejados del depósito son los días en donde los recorridos son más largos, así mismo la distancia entre clientes de estas zonas son mayores.

Día Entrega	Cantidad Clientes Zona	Cantidad Clientes			Distancia Ruta [km]		
		Min	Promedio	Max	Min	Promedio	Max
martes	72	21	39	48	70	90	115
miércoles	87	60	63	66	226	238	258
jueves	100	65	67	69	145	151	167
viernes	90	40	58	68	60	88	102
sábado	88	23	43	56	28	63	75

Tabla 4. Cantidad de clientes visitados y km recorridos por día de entrega.

Como se comentó anteriormente y queda en evidencia en la tabla anterior, todos los clientes de una misma zona no son visitados el mismo día, cubriendo sólo entre el 47% y 70% de los clientes. Por otro lado, se detectan zonas de entrega con diferencias considerables en la cantidad promedio de clientes visitados, por ejemplo, el miércoles cuenta con un 72% más de clientes que el martes.

Por último, las mismas métricas expuestas en la tabla anterior, sólo para los camiones.

Camión	Cantidad Clientes			Distancia Ruta [km]		
	Min	Promedio	Max	Min	Promedio	Max
1	2	16	32	16	40	69
2	3	24	31	11	51	112
3	9	24	33	26	52	96
Total	2	33	33	11	47	112

Tabla 5. Cantidad de clientes visitados y km recorridos por camión.

De la tabla anterior se puede ver que, si bien la cantidad máxima de clientes visitados es similar, el camión 1 visita en promedio 50% menos de clientes y recorre entre 28% y 30% kilómetros menos.

Estas métricas, servirán como Benchmark, para comparar las soluciones arrojadas por los algoritmos desarrollados más adelante.

Tiempos de servicio

El tiempo de servicio de cada cliente depende de la cantidad de ítems que serán entregados, pero principalmente del peso equivalente de cada uno de ellos. Por ello en base a la experiencia del encargado del despacho, se estimó el tiempo de servicio como una función escalonada, en función de los kilos a ser entregados.

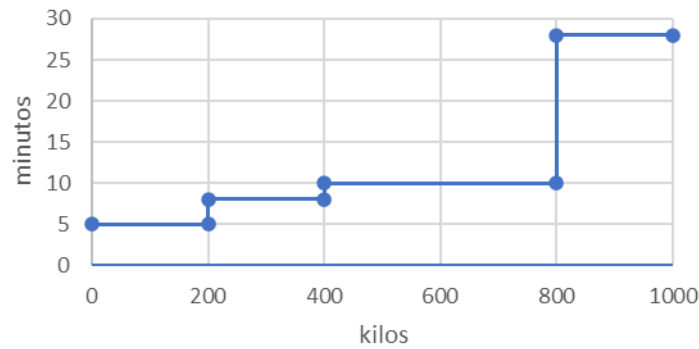


Figura 10. Tiempos de servicio.

3. Metodología

Para dar respuesta a las preguntas planteadas en el apartado 1.3, se formaliza el problema mediante modelos ILP, y luego se resuelve con métodos heurísticos, que brindan soluciones cercanas a la óptima en tiempos de cómputo razonables.

La herramienta que se utiliza para resolver el problema de generación de rutas por medio de métodos heurísticos es Google OR-Tools (Operations Research Tools)⁴. OR-Tools es una biblioteca de código abierto desarrollada por Google que proporciona una amplia gama de algoritmos y herramientas para la optimización combinatoria.

Los pasos para seguir para abordar cada una de las etapas planteadas anteriormente son los siguientes:

1. Definición del problema: Se deben establecer los parámetros y restricciones del problema, como la cantidad de vehículos disponibles, las ubicaciones de los clientes, las demandas de cada cliente, las ventanas de tiempo para las visitas, las capacidades de los vehículos, la función objetivo que en este caso es minimizar el tiempo de viaje total de la ruta, la matriz de costos y tiempos de viaje.

⁴ <https://developers.google.com/optimization/routing>

2. Implementación: por medio de código de programación Python e incorporando bibliotecas específicas se definirán las variables, la función objetivo y las restricciones del algoritmo. Se configurarán los parámetros del algoritmo de optimización, como el tipo de heurística a utilizar, los límites de tiempo o iteraciones, etc.
3. Ejecución y calibración: Se ejecuta el algoritmo de optimización con diferentes tipos de heurísticas y metaheurísticas, escogiendo aquella que mejor resultados arroje. Si es necesario, se pueden realizar ajustes en los parámetros del algoritmo o en la formulación del problema para obtener soluciones satisfactorias.

Dentro de las heurísticas implementadas por el paquete de bibliotecas OR-Tools se evalúan las siguientes:

- First Unbound Min Value.
- Local Cheapest Insertion.
- Christofides.
- Savings.

A partir de ahora se denominarán de la siguiente manera: FUMV, LCI, C y S respectivamente.

Así mismo se evaluarán las siguientes metaheurísticas de búsqueda local:

- Tabu Search.
- Guided Local Search.

A partir de ahora se denominarán de la siguiente manera: TS y GLS respectivamente.

A continuación, se describe el modelo CVRPTW matemático de ILP en su forma completa y se describen algunas de las heurísticas y metaheurística que se utilizarán para hallar soluciones.

3.1. Modelo matemático CVRPTW

El CVRPTW puede describirse como el siguiente problema teórico de grafos, como desarrolla (Toth & Vigo, 2014). Sea el grafo $G = (V, A)$, donde $V = \{0, \dots, n + 1\}$, es el conjunto de vértices y A , es el conjunto de arcos. G es un grafo dirigido completo, excepto porque no hay arcos que lleguen al vértice 0, ni arcos que salen del vértice $n+1$. Sea $N = V \setminus \{0, n + 1\}$ que corresponde al conjunto de clientes, mientras que los vértices 0 y $n + 1$ corresponden al depósito.

Un costo no negativo, c_{ij} , está asociado con cada arco $(i, j) \in A$ y representa el costo de viaje incurrido para ir del vértice i al vértice j . Generalmente, el uso de los arcos de bucle, (i, i) , no está permitido y esto se impone definiendo $c_{ii} = +\infty$ para todo $i \in V$. El problema que se aborda en este trabajo es llamado simétrico, ya que $c_{ij} = c_{ji}$ para todo $(i, j) \in A$. De la misma

forma, un tiempo de viaje no negativo, t_{ij} , está asociado con cada arco $(i, j) \in A$ y representa el tiempo de viaje incurrido para ir del vértice i al vértice j . El uso de los arcos de bucle, (i, i) , no está permitido y esto se impone definiendo $t_{ii} = +\infty$ para todo $i \in V$. Ya que el problema que se aborda en este trabajo es simétrico, $t_{ij} = t_{ji}$ para todo $(i, j) \in A$.

En el depósito hay disponible un conjunto de K vehículos idénticos, cada uno con capacidad C . Todas las rutas de vehículos factibles corresponden a caminos en G que comienzan desde el nodo 0 y terminan en el nodo $n + 1$.

Cada cliente i ($i = 1, \dots, n$) tiene asociada una demanda no negativa conocida, d_i , para ser entregada. A su vez cada cliente i está asociado con un intervalo de tiempo $[a_i, b_i]$, llamado ventana de tiempo, en donde:

- a_i (inicio de la ventana de tiempo): representa el momento más temprano en que un vehículo puede comenzar a dar servicio al cliente i . Si un vehículo llega antes de este tiempo, debe esperar hasta a_i para iniciar el servicio.
- b_i (fin de la ventana de tiempo): representa el momento más tarde en que un vehículo puede comenzar a atender al cliente i . El servicio debe comenzar antes de este tiempo para ser válido. Si un vehículo llega después del instante b_i , no se le permite comenzar el servicio y la ruta no sería factible.

El instante de tiempo en que los vehículos salen del depósito, y un tiempo adicional de servicio s_i para cada cliente, están dados. El servicio de cada cliente debe comenzar dentro de la ventana de tiempo asociada y el vehículo debe detenerse en la ubicación del cliente durante s_i instantes de tiempo, pudiendo finalizar luego de b_i . Una ventana de tiempo también está asociada a los nodos 0 y $n+1$, es decir, $[a_0, b_0] = [a_{n+1}, b_{n+1}] = [E, L]$, donde E y L representan la salida más temprana posible del depósito y la última llegada posible al depósito, respectivamente. Además, se definen demandas y tiempos de servicio cero para estos dos nodos, es decir, $d_0 = d_{n+1} = s_0 = s_{n+1} = 0$.

Normalmente, las matrices de costo y tiempo de viaje coinciden, y las ventanas de tiempo se definen suponiendo que todos los vehículos abandonan el depósito en el instante 0. Esto significa que el valor de E es 0, lo que simplifica el cálculo del tiempo de viaje y permite una planificación más directa desde el inicio del horizonte de tiempo.

El CVRPTW consiste en encontrar una colección de a lo sumo K circuitos simples con mínimo costo, y tal que:

- (i) cada circuito empieza y termina en el depósito;
- (ii) cada cliente es visitado por exactamente un circuito;
- (iii) la suma de las demandas de los vértices visitados por un circuito no excede la capacidad del vehículo, C ; y
- (iv) para cada cliente i , el servicio comienza dentro de la ventana de tiempo, $[a_i, b_i]$, y el vehículo se detiene durante s_i instantes de tiempo para descargar la mercadería.

A continuación, se presenta una formulación de programación matemática para el CVRPTW que involucra dos tipos de variables: variables de flujo x_{ijk} , $(i, j) \in A$, $k \in K$, igual a 1 si el arco (i, j) es utilizado por el vehículo k y 0 en caso contrario, y variables de tiempo w_{ik} , $i \in V$, $k \in K$, que especifican el inicio del servicio en el nodo i cuando el vehículo k realiza el servicio.

$$\min \sum_{k \in K} \sum_{(i,j) \in A} c_{ij} x_{ijk} \quad (1)$$

Sujeto a

$$\sum_{k \in K} \sum_{j \in \Delta^+(i)} x_{ijk} = 1 \quad \forall i \in N, \quad (2)$$

$$\sum_{j \in \Delta^+(0)} x_{0jk} = 1 \quad \forall k \in K, \quad (3)$$

$$\sum_{i \in \Delta^-(j)} x_{ijk} - \sum_{i \in \Delta^+(j)} x_{jik} = 0 \quad \forall k \in K, j \in N, \quad (4)$$

$$\sum_{i \in \Delta^-(n+1)} x_{i,n+1,k} = 1 \quad \forall k \in K, \quad (5)$$

$$x_{ijk}(w_{ik} + s_i + t_{ij} - w_{jk}) \leq 0 \quad \forall k \in K, (i, j) \in A, \quad (6)$$

$$a_i \sum_{j \in \Delta^+(i)} x_{ijk} \leq w_{ik} \leq b_i \sum_{j \in \Delta^+(i)} x_{ijk} \quad \forall k \in K, i \in N, \quad (7)$$

$$E \leq w_{ik} \leq L \quad \forall k \in K, i \in \{0, n+1\}, \quad (8)$$

$$\sum_{i \in N} d_i \sum_{j \in \Delta^+(i)} x_{ijk} \leq C \quad \forall k \in K, \quad (9)$$

$$x_{ijk} \in \{0,1\} \quad \forall k \in K, (i, j) \in A, \quad (10)$$

La función objetivo (1) de esta formulación expresa el costo total. Las restricciones (2) limitan la asignación de cada cliente a exactamente una ruta de vehículo. A continuación, restricciones (3)-(5) caracterizan el flujo en el camino que debe seguir el vehículo k . Además, las restricciones (6)-(8) y (9) garantizan la viabilidad con respecto a las consideraciones de tiempo y capacidad,

respectivamente. Notar que, para un k dado, las restricciones (7) fuerzan a $w_{ik} = 0$ siempre que el vehículo k no visite al cliente i . Finalmente, las condiciones (10) imponen condiciones binarias sobre las variables de flujo.

Las condiciones binarias (10) permiten linealizar las restricciones (6) como:

$$w_{ik} + s_i + t_{ij} - w_{jk} \leq (1 - x_{ijk})M_{ij} \quad \forall k \in K, (i, j) \in A, \quad (6a)$$

donde M_{ij} son constantes grandes. Además, M_{ij} puede ser reemplazada por $\max\{b_i + s_i + t_{ij} - a_j, 0\}$, $(i, j) \in A$ y las restricciones (6) o (6a) sólo necesitan aplicarse para los arcos $(i, j) \in A$ tal que $M_{ij} > 0$; de lo contrario, cuando $\max\{b_i + s_i + t_{ij} - a_j, 0\} = 0$, estas restricciones se satisfacen para todos los valores de w_{ik} , w_{jk} , y x_{ijk} .

Por otro lado, es posible minimizar la cantidad de vehículos utilizados, lo que puede generar una disminución de costo. Este es un costo único y está relacionado con los costos fijos a la hora de utilizar un vehículo. Esto se puede modelar agregando el siguiente termino a la función objetivo (Ioannou, 2001):

$$+ \sum_{k=1}^K W_k Z_k$$

Con la siguiente restricción:

$$\begin{aligned} x_{ijk} &\leq Z_k \quad \forall (i, j) \in A, \\ Z_k &\in \{0, 1\}, \quad \forall k \in V.E \end{aligned}$$

Y reemplazando la restricción (3) por la siguiente:

$$\sum_{j \in \Delta^+(0)} x_{0jk} = Z_k \quad \forall k \in K$$

W_k es el costo fijo de utilizar el vehículo k , y Z_k es igual a 1 si el vehículo k este activo de lo contrario 0.

3.2. Heurísticas y Metaheurísticas

Como se mencionó anteriormente el problema de generación de rutas VRP presenta un gran desafío: hallar soluciones óptimas para problemas con muchos clientes es difícil y demanda mucho tiempo. Por eso, la práctica exige buscar heurísticas y metaheurísticas. Estos son algoritmos que se utilizan para encontrar soluciones de buena calidad eficientemente, aunque generalmente sin garantía de optimalidad, que puedan resolver casos reales rápido y de forma flexible.

Desde los comienzos del problema, se han propuesto muchas heurísticas. Unas construyen soluciones paso a paso, otras las mejoran, y las más recientes, llamadas metaheurísticas, combinan distintas ideas para lograr soluciones muy buenas en segundos, acercándose al 1% de la mejor opción conocida. (Toth & Vigo, 2014)

A continuación, se describen algunas de las heurísticas y metaheurísticas que se utilizan en la resolución del problema, con excepción de la heurística First Unbound Min Value. La documentación de esta biblioteca la describe como: Selecciona el primer nodo con un sucesor no vinculado y lo conecta al primer nodo disponible⁵. Pero por falta de bibliografía que desarrolle esta heurística, se decidió no incluir una sección que la describa.

3.2.1 Heurística Christofides

Los algoritmos heurísticos, caracterizados por su tasa de crecimiento polinomial en función del número de variables, son herramientas valiosas para ofrecer soluciones aproximadas a problemas combinatorios. En este contexto, surge la interrogante sobre cuál es la peor relación posible (R_w) entre el valor obtenido mediante la heurística y el valor de la solución óptima. En (Rosenkrantz D. J., 1974) se exploraron diversas heurísticas para el TSP, concluyendo que, para el mejor de los algoritmos investigados, R_w tiende a 2 a medida que n (el número de ciudades en el TSP) tiende a infinito.

Posteriormente (Christofides, 1976) propuso un algoritmo heurístico para abordar el TSP en instancias con n ciudades, cuya matriz de costos satisface la desigualdad triangular. La estrategia del algoritmo involucra la determinación de un Árbol de Expansión Mínima (MST) del grafo G que define el TSP, seguido por la identificación de una matching perfecto de costo mínimo en un subgrafo inducido de G . Un análisis en el peor de los casos de esta heurística revela que la relación entre la solución obtenida y la solución óptima del TSP es estrictamente inferior a $3/2$, lo que representa una mejora significativa del 50%.

Existen varios algoritmos eficientes de orden $O(n^2)$ para determinar la MST de un grafo. Sin embargo, el algoritmo más conocido para calcular matching de cardinalidad máxima tiene una tasa de crecimiento $O(n^3)$. Por consiguiente, la tasa de crecimiento global del algoritmo propuesto por Christofides es $O(n^3)$.

3.2.2 Cheapest Insertion Heuristic

La Heurística de Inserción más Económica (CIH, por sus siglas en inglés) es una heurística bien conocida de la clase "Insertion" utilizada para resolver problemas TSP y otros problemas de

⁵ https://developers.google.com/optimization/routing/routing_options?hl=es-419

optimización combinatoria. Se destaca por ser rápida, generar soluciones adecuadas, ser fácil de implementar y sencilla de extender para manejar restricciones complicadas. En este apartado, se explica el algoritmo CIH y cómo funciona.

En (Rosenkrantz D. J., 1977) se analizan algoritmos de esta clase. Los cuales inician el proceso de construcción del recorrido con un "recorrido semilla" que consta de un solo nodo con un bucle propio o de un bucle que involucra sólo dos nodos. (El método de selección de estos dos nodos iniciales es en sí mismo una cuestión de elección).

Dado un conjunto N de nodos, es útil definir el costo, $c(i, k, j)$ de insertar un nuevo nodo, k , en un recorrido entre dos nodos, i y j , que anteriormente eran adyacentes en el recorrido:

$$c(i, k, j) = d(i, k) + d(k, j) - d(i, j)$$

Para el caso CIH, el siguiente nodo, del conjunto N , en unirse al recorrido, T , es el que minimiza $c(i, k, j)$ entre todos los nodos $k \in (N - T)$ y para todos los pares consecutivos de nodos $(i, j) \in T$. La ubicación donde el nodo seleccionado se inserta es la que minimiza $c(i, k, j)$. El procedimiento se repite hasta que todos los nodos se hayan insertado en T .

El peor de los casos de esta heurística revela que la relación entre la solución obtenida y la solución óptima del TSP es menor a 2. Mientras que la complejidad computacional es $O(n^3)$, pero con una programación cuidadosa puede ser $O(n^2 \log n)$.

3.2.3. Savings

La heurística Savings de Clarke y Wright, (G. Clarke, 1964), es una estrategia efectiva para resolver el problema de ruteo de vehículos (VRP). Esta técnica se basa en el concepto de "ahorro", que representa la reducción en la distancia total recorrida al combinar rutas individuales de los clientes en rutas conjuntas.

El algoritmo procede de la siguiente manera:

- **Solución Inicial:** Se comienza con una solución inicial donde cada cliente es atendido por un vehículo separado que regresa al depósito después de la entrega. Esto forma un conjunto de rutas individuales, cada una comenzando y terminando en el depósito.
- **Cálculo de Ahorros:** Para cada par de clientes i y j , se calcula un valor de ahorro S_{ij} . El ahorro se define como la reducción en la distancia total al atender a i y j en una misma ruta en lugar de rutas separadas. Matemáticamente, esto se expresa como:

$$S_{ij} = d_{0i} + d_{0j} - d_{ij}$$

Donde d_{0i} y d_{0j} son las distancias desde el depósito a los clientes i y j , respectivamente, y d_{ij} es la distancia entre los clientes i y j .

- **Priorización de Ahorros:** Los valores de ahorro se ordenan en orden descendente. Esto prioriza las combinaciones de rutas que ofrecen los mayores ahorros.
- **Construcción de Rutas:** Comenzando por el mayor ahorro, se combinan las rutas de los clientes correspondientes siempre que la capacidad del vehículo y otras restricciones no sean violadas. Es decir, se agrega el cliente j a la ruta del cliente i si esto resulta en un ahorro y es factible en términos de capacidad y otras limitaciones.
- **Iteración y Finalización:** Este proceso se repite iterativamente hasta que no sea posible realizar más combinaciones que generen ahorros adicionales.

Esta heurística es simple y rápida, proporcionando soluciones buenas y prácticas para problemas reales de ruteo de vehículos. Aunque no garantiza una solución óptima, es muy eficaz en términos de tiempo computacional y facilidad de implementación, lo que la hace una herramienta valiosa en la logística y distribución.

3.2.4. Local Search Algorithms

Como se describe en (Toth & Vigo, 2014), Local Search es una técnica de búsqueda heurística que se utiliza para resolver problemas de optimización combinatoria, como el VRP. En Local Search, se parte de una solución inicial y se exploran soluciones vecinas en un espacio de búsqueda. La idea es mejorar iterativamente la solución actual moviéndose a una solución vecina que tenga un mejor valor de función objetivo.

La exploración de soluciones vecinas se realiza mediante la aplicación de operadores de vecindad, que definen cómo se generan las soluciones vecinas a partir de la solución actual. Por ejemplo, en el VRP, un operador de vecindad podría consistir en intercambiar dos clientes entre dos rutas diferentes.

La característica distintiva de Local Search es que se enfoca en mejorar la solución actual sin preocuparse por explorar todo el espacio de búsqueda. Esto significa que puede encontrar soluciones de alta calidad en un tiempo razonable, pero también puede quedarse atrapado en óptimos locales.

Para evitar quedar atrapado en óptimos locales, se pueden utilizar técnicas como la perturbación, que consiste en introducir cambios aleatorios en la solución actual para escapar de los óptimos locales. También se pueden utilizar técnicas de diversificación, que consisten en explorar diferentes regiones del espacio de búsqueda para encontrar soluciones alternativas.

Local Search es la base de diversos métodos metaheurístico para problemas de optimización combinatoria. A continuación, se describen brevemente dos de los utilizados en este trabajo.

Tabú Search

Tabú Search (TS), vea (Glover, 1977) y (Gendreau, 2010), es un método de optimización que se utiliza para resolver problemas combinatorios, como el VRP. En TS, se parte de una solución inicial y se exploran soluciones vecinas en un espacio de búsqueda. La característica distintiva de TS es el uso de una lista tabú que registra movimientos previamente realizados para evitar repeticiones y promover la exploración de nuevas soluciones.

Durante la búsqueda, las soluciones que comparten ciertas características con la solución actual se marcan como tabú, lo que impide que se vuelvan a explorar en el corto plazo. Esto fomenta la diversificación de la búsqueda y evita que el algoritmo se estanque en óptimos locales.

Algoritmo TS:

Notación:

- S , solución actual.
- S^* , mejor solución conocida.
- f^* , valor de S^* .
- $N(S)$, vecindario de S .
- $\tilde{N}(s)$, el subconjunto “admisibles” de $N(S)$ (es decir, no tabú o permitido por aspiración)

Inicialización:

- Escoger o construir una solución inicial S_0 .
 - Asignar: $S := S_0, f^* := f(S_0), S^* := S_0, T := \emptyset$

Búsqueda:

- Mientras no se cumpla el criterio de terminación, hacer:
 - Seleccionar S en $\underset{S' \in \tilde{N}(S)}{\operatorname{argmin}}[f(S')]$;
 - Si $f(S) < f^*$, entonces asignar $f^* := f(S), S^* := S$;
 - Registrar tabú para el movimiento actual en T (eliminar la entrada más antigua si es necesario).

Criterio de terminación:

Los criterios de detención más comúnmente utilizados en TS son:

- Después de un número fijo de iteraciones (o una cantidad fija de tiempo de CPU).
- Después de un cierto número de iteraciones sin una mejora en el valor de la función objetivo (el criterio utilizado en la mayoría de las implementaciones);
- Cuando el objetivo alcanza un valor umbral preespecificado.

Guided Local Search

Guided Local Search (GLS), vea (Voudouris, 1999), es un método de optimización desarrollado como resultado de un proyecto de investigación con el objetivo de extender la red neuronal GENET para abordar problemas de satisfacción de restricciones parciales y optimización combinatoria.

Para aplicar GLS, es necesario definir las características del problema. Por ejemplo, en el TSP, una característica podría ser si la ruta candidata viaja inmediatamente de la ciudad A la ciudad B. GLS asocia un costo y una penalización con cada característica. Los costos a menudo pueden definirse tomando los términos y sus coeficientes de la función objetivo. Por ejemplo, en el TSP, el costo de la característica mencionada puede ser simplemente la distancia entre las ciudades A y B. Las penalizaciones se inicializan en 0 y solo se incrementarán cuando la búsqueda local alcance un óptimo local. Dada una función objetivo g que asigna a cada solución candidata s un valor numérico, GLS define una función h que será utilizada por la búsqueda local (reemplazando g):

$$h(s) = g(s) + \lambda \times \sum_i (p_i \times I_i(s)),$$

donde s es una solución candidata, λ es un parámetro del algoritmo GLS, i abarca las características, p_i es la penalización para la característica i (todas las p_i se inicializan en 0) y I_i es una indicación de si exhibe la característica i :

$$I_i(s) = 1 \text{ Si } s \text{ exhibe la característica } i; \text{ de lo contrario } 0.$$

Sobre la base de los algoritmos de búsqueda local, GLS les ayuda a escapar de los óptimos locales de la siguiente manera. Cada vez que el algoritmo de búsqueda local se asienta en un óptimo local, GLS aumenta la función de costo añadiendo penalizaciones a características seleccionadas. La novedad de GLS radica principalmente en la forma en que selecciona las características a penalizar. La intención es penalizar "características desfavorables" o características que

"importan más" cuando una búsqueda local se asienta en un óptimo local. Una característica con un alto costo tiene más impacto en el costo total. Otro factor que debe considerarse es el valor de penalización actual de esa característica. La utilidad de penalizar la característica i , $util_i$, bajo un óptimo local, s_* , se define de la siguiente manera:

$$util_i(s_*) = I_i(s_*) \times \frac{c_i}{1 + p_i}$$

donde c_i es el costo y p_i es el valor de penalización actual de la característica i . En otras palabras, si una característica no se exhibe en el óptimo local (indicado por I_i), entonces la utilidad de penalizarla es 0. Cuanto mayor sea el costo de esta característica (cuanto mayor será c_i), mayor será la utilidad de penalizarla. Además, cuanto mayor sea el número de veces que se ha penalizado (cuanto mayor sea p_i), menor será la utilidad de penalizarla nuevamente. En un óptimo local, la característica con el mayor valor de utilidad será penalizada. Cuando se penaliza una característica, su valor de penalización siempre se incrementa en 1. La escala de la penalización se ajusta por λ .

Al considerar el costo y la penalización actual al seleccionar la característica a penalizar, GLS enfoca su esfuerzo de búsqueda en áreas más prometedoras del espacio de búsqueda: áreas que contienen soluciones candidatas que exhiben "buenas características", es decir, características que implican un menor costo. Por otro lado, las penalizaciones ayudan a prevenir que la búsqueda dirija todo el esfuerzo a una región particular del espacio de búsqueda.

4. Resultados computacionales

4.1. Escenario 1 (Situación actual)

En este apartado se busca darle respuesta a la pregunta planteada en el escenario 1 enunciado en el apartado 1.3. Como se mencionó anteriormente, en la operatoria actual para cada día de entrega los tres camiones reparten en zonas contiguas que serán respetadas.

Se cuenta con una instancia a resolver por cada camión con el listado de clientes y las ventanas de tiempo. Tanto el modelo matemático como el método heurístico tiene como función objetivo minimizar el tiempo de viaje total de la ruta, sujeto a las ventanas de tiempo de los clientes. En este caso no se consideran las restricciones de capacidad ya que como premisa se sabe que en estas instancias la capacidad de los camiones no es superada. En total se cuenta con 63 instancias a resolver.

4.1.1. Modelo matemático TSPTW

El modelo TSPTW es una simplificación del CVRPTW expuesto en el apartado 3.1, que se obtiene al considerar K igual a 1 (un único vehículo). Por otro lado, no serán necesarias las restricciones de capacidad (9) o estas se consideran infinitas, ya que como dato se sabe que las capacidades de los camiones no se supera.

4.1.2. Método heurístico para TSPTW

Se implementa y configura un algoritmo heurístico TSPTW para 1 vehículo con la herramienta OR-Tools, incluyendo restricciones de ventana de tiempo.

Para hallar la mejor de las 8 combinaciones de heurísticas y metaheurísticas se va a ejecutar el algoritmo con 67 instancias correspondientes a 26 días. Se evalúan los resultados cuantificando cantidad de instancias con solución factibles (es decir, que se cumple la ventana de tiempo de los clientes) y se utiliza como métrica de performance los kilómetros recorridos, el tiempo de viaje(t_{viaje}), tiempos de espera(t_{espera}) y tiempos totales de la ruta(t_{total}) que incluyen los tiempos de servicio, que al ser iguales en todas las ejecuciones no son mostrados.

First Solution Strategy	Local Search Metaheuristic	Instancias Resueltas	Promedio km recorridos	Promedio de t_{viaje}	Promedio de t_{espera}	Promedio Tiempo total
C	GLS	100%	43,6	82,72	74,06	278,57
C	TB	100%	43,9	83,09	76,13	281,01
FUMV	GLS	100%	43,6	82,72	70,34	274,85
FUMV	TB	100%	44,1	83,27	72,85	277,91
LCI	GLS	100%	43,6	82,72	68,84	273,34
LCI	TB	100%	43,7	83,03	69,91	274,73
S	GLS	100%	43,6	82,72	76,16	280,67
S	TB	100%	43,8	83,00	75,69	280,48

Tabla 6. Evaluación de heurísticas y metaheurísticas para TSPTW heurístico.

Como se puede ver las combinaciones que mejor resultado otorgan son las que tienen Local Cheapest Insertion como heurística de first solution strategy y Guided Local Search como metaheurística de búsqueda local.

Soluciones

A continuación, se muestra una solución para cada día de entrega, obtenidas mediante la heurística Local Cheapest Insertion y metaheurísticas Guided Local Search.

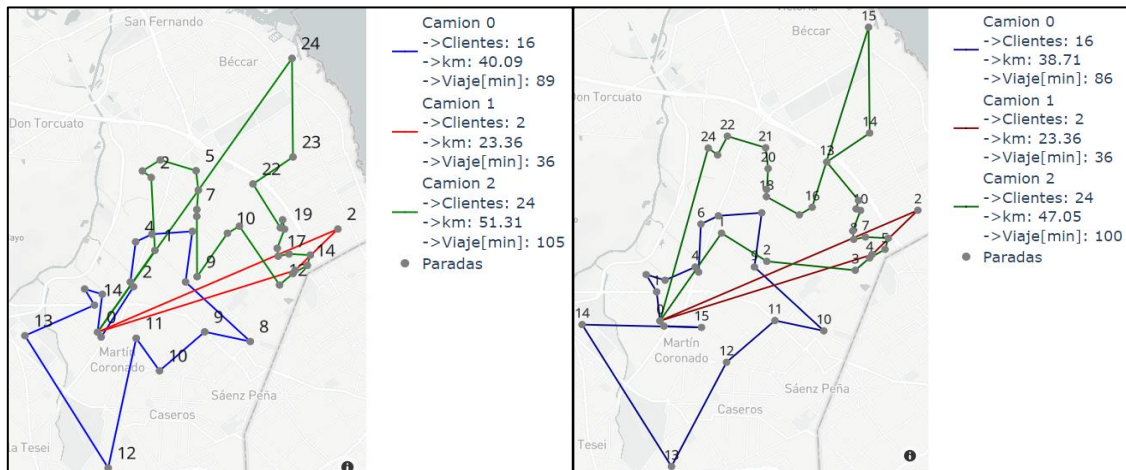


Figura 11. Instancias correspondientes a un martes; Izq. Benchmark: total kilómetros recorridos: 115 km, tiempo total de viaje: 230 min, Der. Solución TSPTW total kilómetros recorridos: 109 km (-5%), tiempo total de viaje: 222 min (-3%).

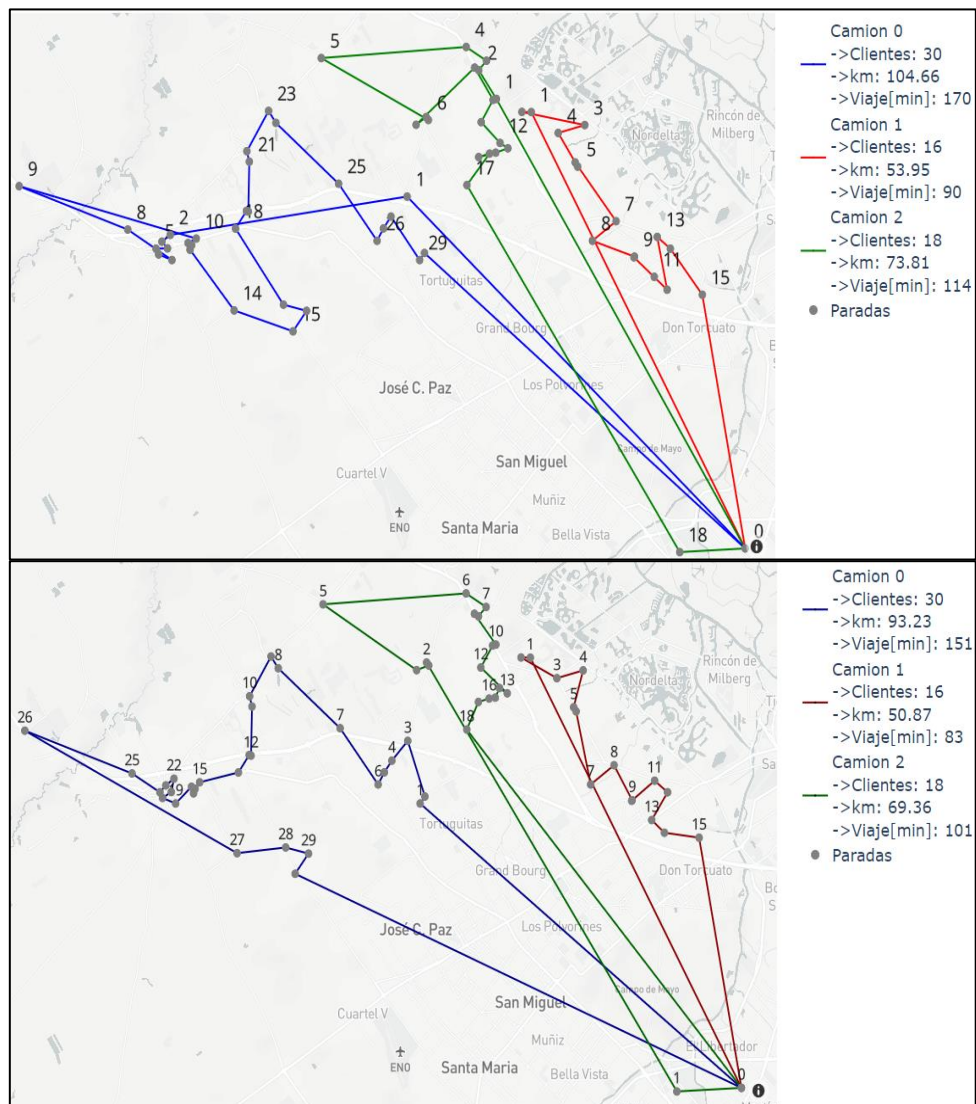


Figura 12. Instancias correspondientes a un miércoles; Arriba. Benchmark: total kilómetros recorridos: 232 km, tiempo total de viaje: 374 min, Abajo. Solución TSPTW total kilómetros recorridos: 213 km (-8%), tiempo total de viaje: 335 min (-10%).

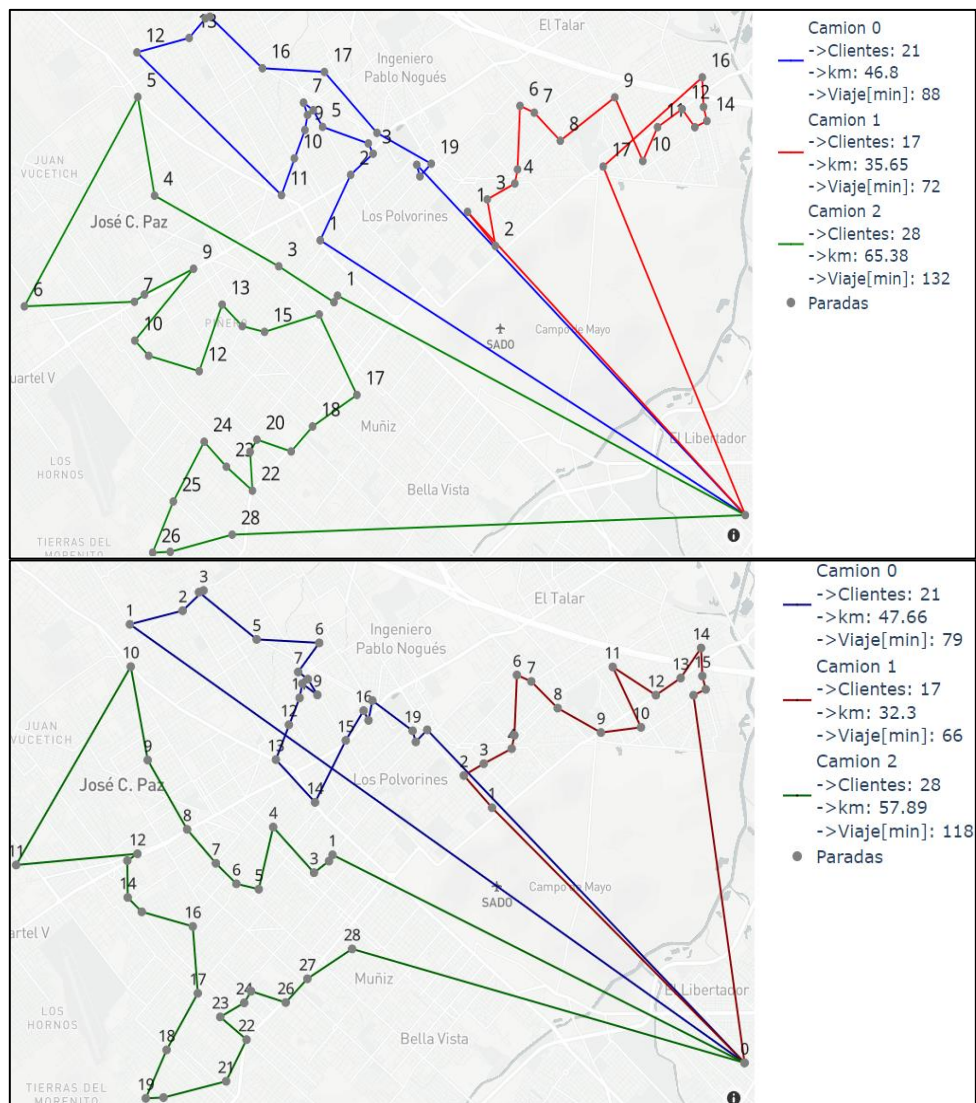


Figura 13. Instancias correspondientes a un jueves; Arriba. Benchmark: total kilómetros recorridos: 148 km, tiempo total de viaje: 292 min, Abajo. Solución TSPTW total kilómetros recorridos:138 km (-6%), tiempo total de viaje:263 min (-10%).

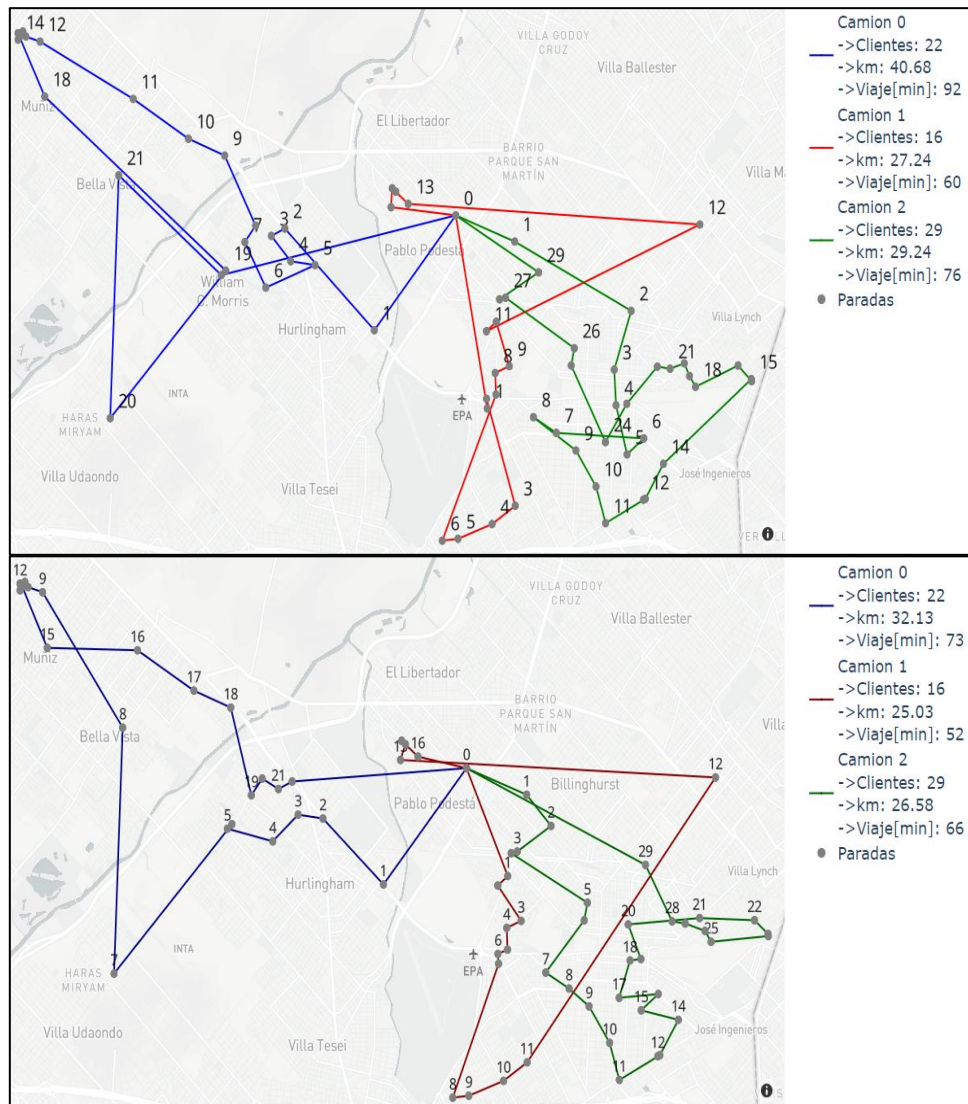


Figura 14. Instancias correspondientes a un viernes; Arriba. Benchmark: total kilómetros recorridos: 97 km, tiempo total de viaje: 228 min, Abajo. Solución TSPTW total kilómetros recorridos: 84 km (-15%), tiempo total de viaje: 191 min (-16%).

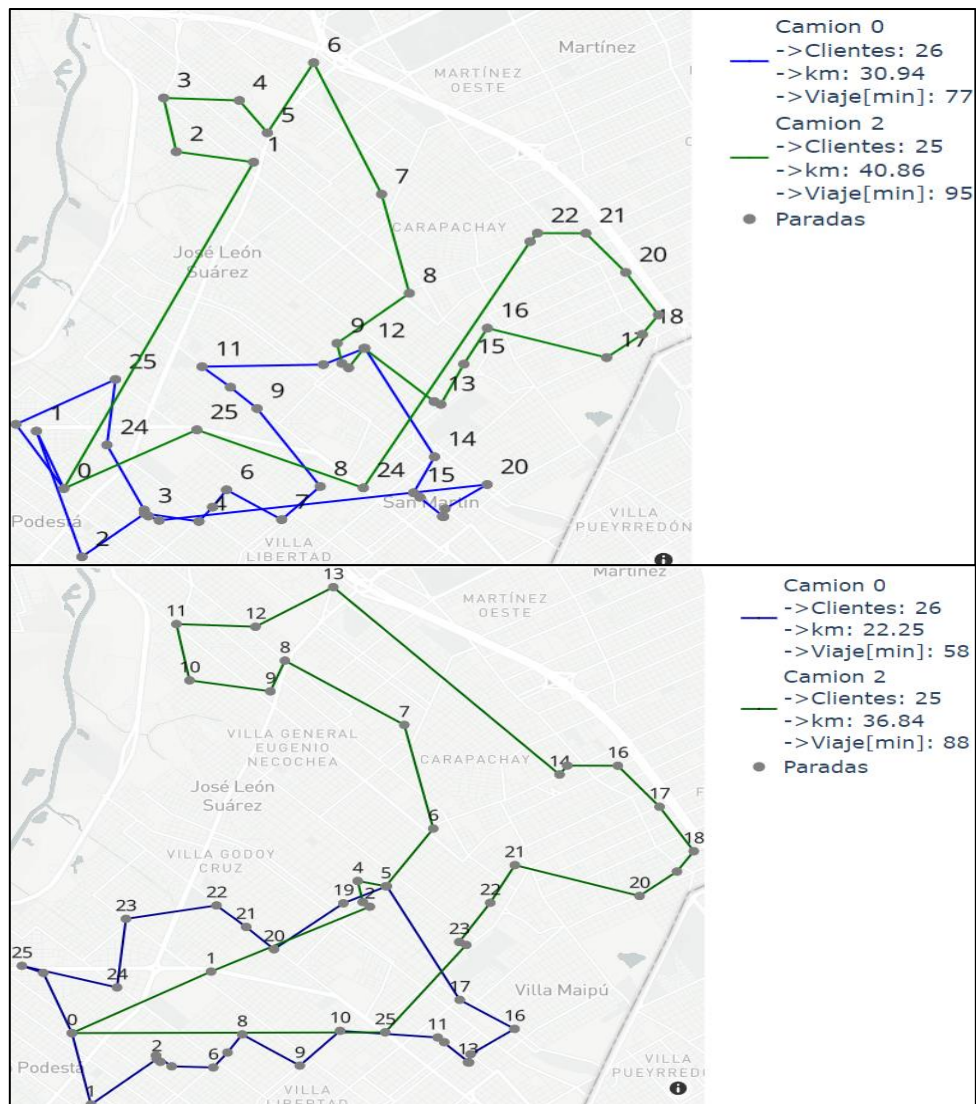


Figura 15. Instancias correspondientes a un sábado; Arriba. Benchmark: total kilómetros recorridos: 72 km, tiempo total de viaje: 172 min, Abajo. Solución TSPTW total kilómetros recorridos: 59 km (-18%), tiempo total de viaje: 146 min (-15%).

En las instancias graficadas anteriormente se puede observar que las soluciones arrojadas por el algoritmo TSPTW genera rutas de mejor calidad, reduciendo los cruces entre la trayectoria de un mismo camión y recorridos con mayor fluidez. Se detecta que los kilómetros recorridos se reducen entre 7% y 18% y mientras que el tiempo de viaje se reduce entre un 3% y un 16%.

A continuación, se comparan los resultados de todas las instancias evaluadas por el algoritmo TSPTW contra las instancias Benchmark, utilizando como métricas de evaluación el promedio de los tiempos de viaje, los tiempos de espera y kilómetros recorridos para cada día de la semana.

En primera instancia se comparan los tiempos promedio de viaje (t_{viaje}) medidos en minutos.

Tiempo Viaje	martes	miércoles	jueves	viernes	sábado	Totales
Benchmark	84	126	99	73	81	94
TSPTW	76	112	89	63	70	83
Mejora%	9%	11%	10%	15%	14%	12%

Tabla 7. Evaluación de resultados para TSPTW heurístico, tiempos promedio de viaje.

Las instancias solucionadas con TSPTW reduce el tiempo de viaje en un 12%.

De la misma forma se comparan los tiempos ociosos, que es el tiempo que debe esperar el camión previo la ventana de tiempo de los clientes. Dentro de los datos provistos, se cuenta con la hora de inicio de cada ruta por lo que se pudo calcular el tiempo promedio de espera.

Tiempos de Espera	martes	miércoles	jueves	viernes	sábado	Totales
Benchmark	87	87	94	53	70	77
TSPTW	74	78	72	58	65	69
Mejora%	15%	11%	24%	11%	7%	11%

Tabla 8. Evaluación de resultados para TSPTW heurístico, tiempos promedio espera.

Se puede observar en las instancias Benchmark, así como en las resueltas por medio TSPTW heurístico que se realiza una única espera por recorrido, es decir, es necesario esperar en un único cliente. Esto se debe a la holgura de las ventanas de tiempo.

En las instancias resueltas mediante la heurística Local Cheapest Insertion y metaheurísticas Guided Local Search para TSPTW, el tiempo de espera se reduce en un 11%.

También se computaron los kilómetros promedio recorridos y se comparó contra el Benchmark.

Km Recorridos	martes	miércoles	jueves	viernes	sábado	Totales
Benchmark	40	79	50	31	34	48
TSPTW	34	70	44	27	29	42
Mejora%	14%	12%	12%	14%	16%	13%

Tabla 9. Evaluación de resultados para TSPTW heurístico, kilómetros promedio recorridos.

En promedio el TSPTW heurístico genera un ahorro de 13% en kilómetros promedio recorridos respecto al Benchmark.

Con el objetivo de evaluar el impacto de las restricciones de ventanas de tiempo en la generación de las rutas, se generó una heurística que resuelva un TSP. Al comparar los resultados del VRP

contra el TSPTW no se observan diferencias en las métricas evaluadas. Se infiere que esto se debe a la holgura de las ventanas de tiempo de los clientes.

4.2. Escenario 2

En este apartado se aborda el escenario en el cual se rompe la zonificación diaria actual, en la que para cada día un camión brinda servicio dentro de una determinada zona.

Se cuenta con 26 instancias que corresponden a la cantidad de días de entrega provistos por “El Universo del Panadero”. Cada instancia consta de un número de clientes a visitar, con sus ubicaciones, demanda, ventanas de tiempo y la cantidad de camiones que se utilizaron. Este escenario se aborda como un problema CVRPTW, que al momento de resolverlo se hace una libre asignación de camión-cliente, respetando las restricciones de capacidad de los camiones, las ventanas de tiempo de los clientes y como función objetivo se especifica la minimización del tiempo total de viaje y camiones utilizados.

El modelo matemático CVRPTW fue expuesto en el apartado 3.1. A continuación se resolverá este escenario mediante métodos heurísticos.

4.2.1. Método heurístico CVRPTW

Para hallar la mejor de las 8 combinaciones de heurísticas y metaheurísticas se va a ejecutar el algoritmo heurístico CVRPTW con las 26 instancias para las 8 combinaciones.

Se evalúan los resultados cuantificando cantidad de instancias con solución y se utiliza como métrica de performance el tiempo de viaje(t_{viaje}) y los kilómetros recorridos. Los promedios fueron calculados sobre las instancias con solución.

First Solution Strategy	Local Search Metaheuristic	Instancias Solucionadas	Promedio km recorridos	Promedio de Tiempo Viaje
C	GLS	100%	90	183
C	TB	100%	91	186
FUMV	GLS	96%	89	183
FUMV	TB	96%	92	187
LCI	GLS	100%	89	183
LCI	TB	100%	91	185
S	GLS	96%	89	183
S	TB	96%	90	185

Tabla 10. Evaluación de heurística y metaheurísticas para algoritmo heurístico CVRPTW.

Se puede ver que, para las ejecuciones con algoritmos con heurísticas de primera solución, Savings y First Unbound Min Value resolvieron 25 de las 26 instancias. Mientras que para las ejecuciones resueltas con Christofides y Local Cheapest Insertion resolvió el total de las instancias. Para estas últimas dos, los resultados arrojados son similares. Se continuará el análisis con la combinación de Local Cheapest Insertion como estrategia de primera solución y Guided Local Search como metaheurística de búsqueda local.

Soluciones

A continuación, se muestran soluciones para 6 días de la semana haciendo el paralelismo contra las instancias de Benchmark de referencia.

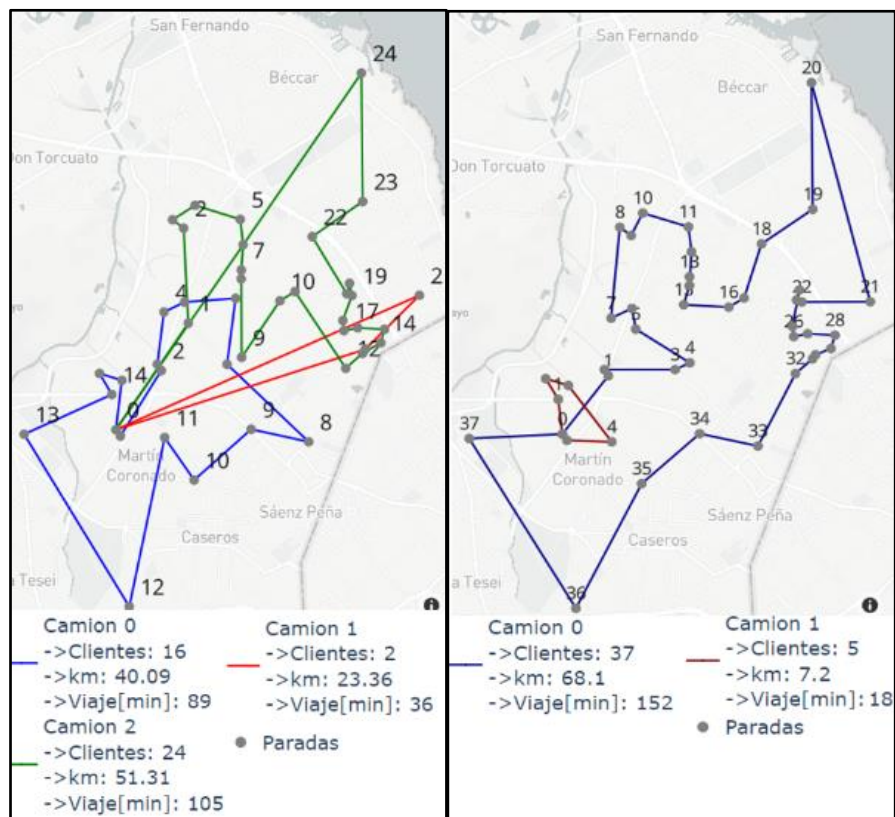


Figura 16. Instancias correspondientes a un martes. Izq. Benchmark: 3 camiones, 115 km totales recorridos, 230 min totales de viaje; Der. Solución CVRPTW: 2 camiones, 75 km (-34%) totales recorridos, 170 min (-26%) totales de viaje.

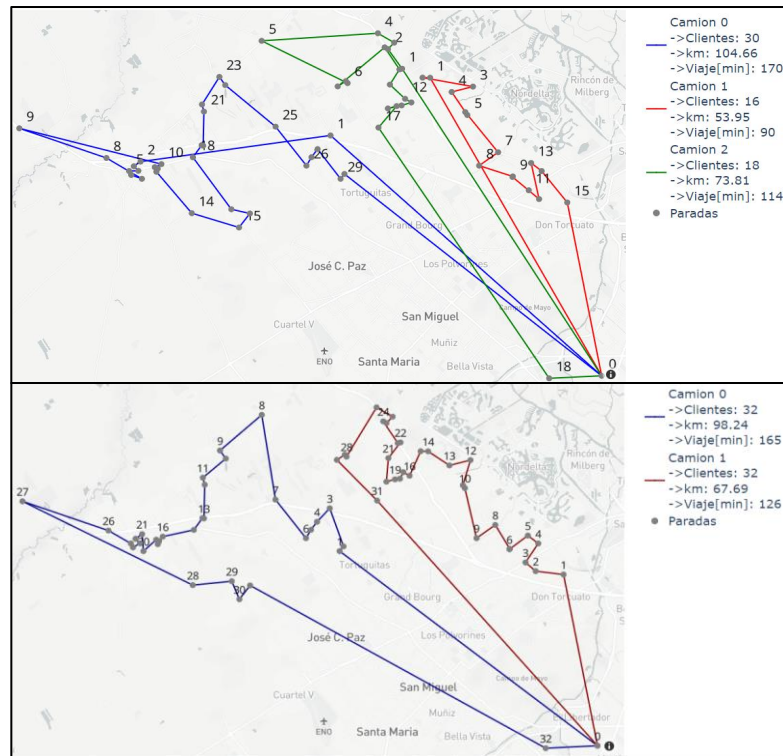


Figura 17. Instancias correspondientes a un miércoles. Arriba. Benchmark: 3 camiones, 232 km totales recorridos, 374 min totales de viaje; Abajo. Solución CVRP: 2 camiones, 165 km (-28%) totales recorridos, 291 min (-22%) totales de viaje.

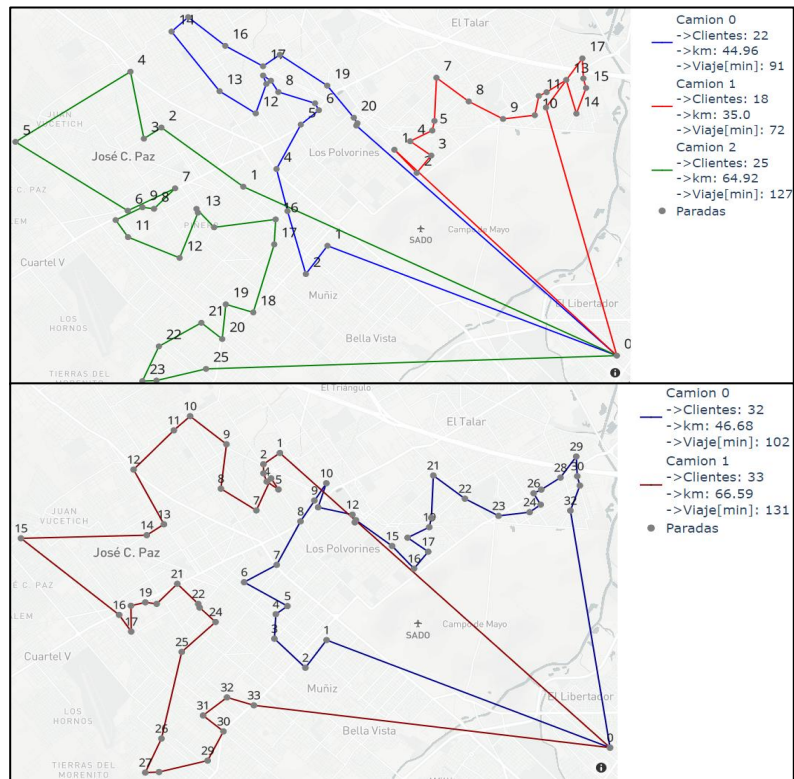


Figura 18. Instancias correspondientes a un jueves. Arriba. Benchmark: 3 camiones, 145 km totales recorridos, 290 min totales de viaje; Abajo. Solución CVRP: 2 camiones, 113 km (-22%) totales recorridos, 233 min (-19%) totales de viaje.

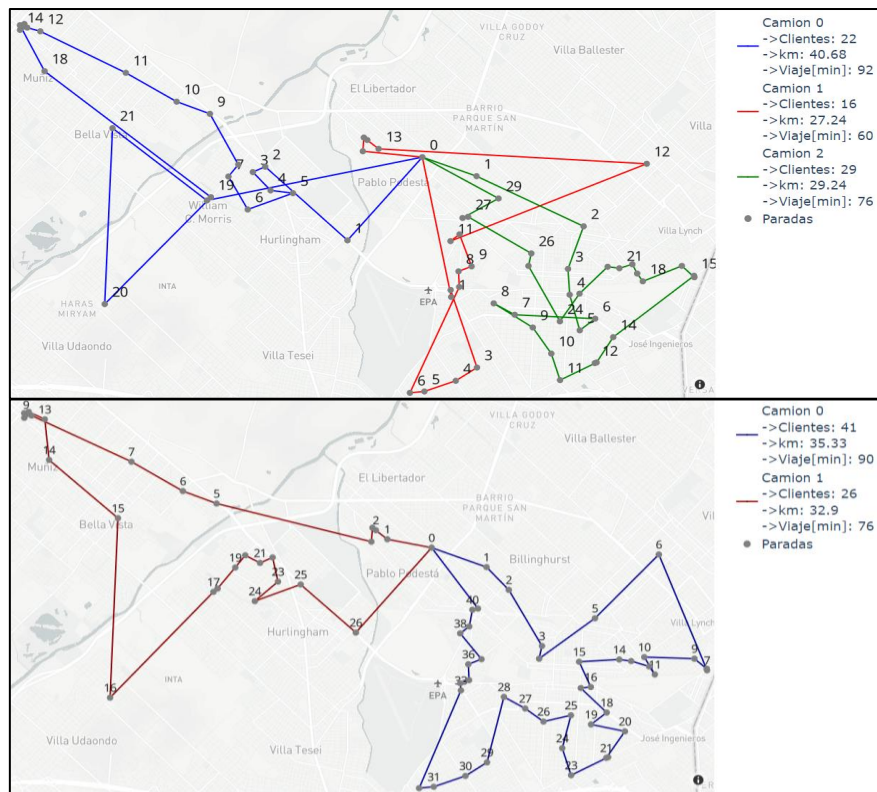


Figura 19. Instancias correspondientes a un viernes. Arriba. Benchmark: 3 camiones, 97 km totales recorridos, 228 min totales de viaje; Abajo. Solución CVRPTW: 2 camiones, 68 km (-30%) totales recorrido, 166 min (-27%) totales de viaje.

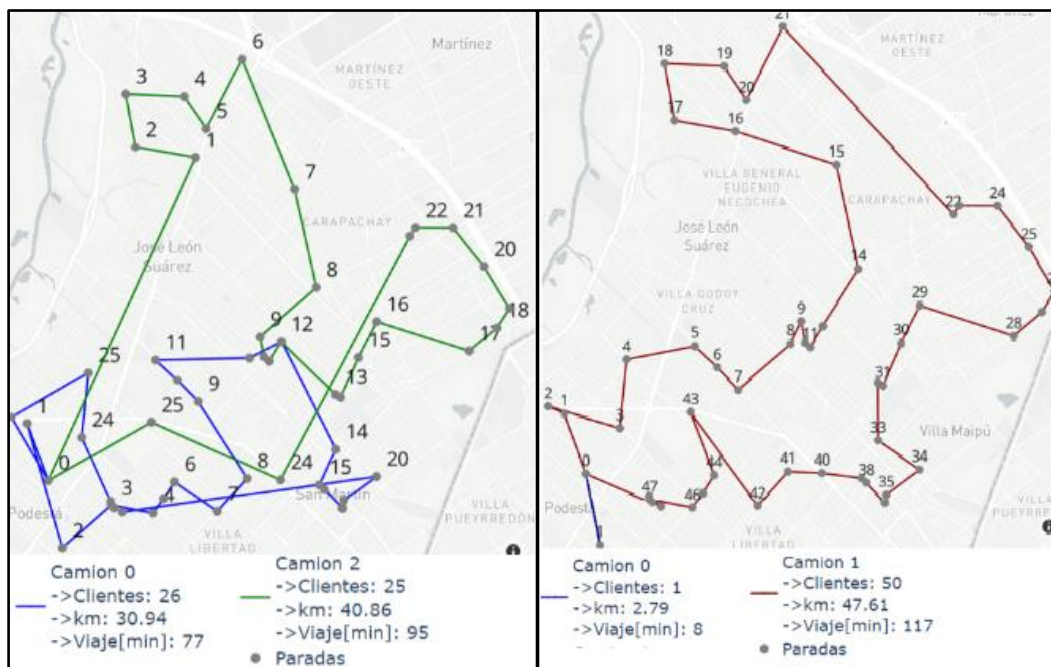


Figura 20. Instancias correspondientes a un sábado. Izq. Benchmark: 2 camiones, 72km totales recorridos, 172min totales de viaje; Der. Solución CVRPTW: 2 camiones, 50km (-30%) totales recorrido, 125min (-27%) totales de viaje.

En las instancias graficadas anteriormente se puede observar que las soluciones arrojadas por el algoritmo CVRPTW genera rutas de mejor calidad, reduciendo los cruces entre la trayectoria de un mismo camión dando como resultados recorridos con mayor fluidez. Por otra parte, se reducen los kilómetros recorridos entre un 22% y un 34% como así también el tiempo de viaje entre en un 19% y en un 27%.

También se puede ver que las soluciones arrojadas por el algoritmo CVRPTW heurístico, utilizan menor cantidad de camiones que en la operación actual. Como se puede ver en el siguiente cuadro el 62% de las instancias Benchmark utilizaron 3 camiones, mientras que las soluciones del algoritmo CVRPTW heurístico no asigno en ningún caso 3 camiones, el 77% de las instancias se resolvieron con 2 camiones.

Vehículos Utilizados	Benchmark		CVRPTW	
3	16	62%	0	0%
2	9	35%	20	77%
1	1	4%	6	23%
Total	26		26	

Tabla 11. Vehículos Utilizados.

Esto se ve reflejado también en el promedio de kilómetros recorridos, tiempo total y cantidad de clientes por camión para todas las rutas analizadas. Dado que la solución provista por la heurística CVRPTW utiliza menos vehículos, estos incrementarán su utilización ya que absorben la carga de trabajo de los vehículos excluidos.

Tiempo Viaje	Km recorridos	Tiempo total	Cantidad Clientes
Benchmark	47	254	21
CVRPTW	52	295	31
Variación Relativa	11%	16%	48%

Tabla 12. Variación relativa entre promedio de kilómetros recorridos, tiempo total y cantidad de clientes por camión para todas las rutas analizadas.

A continuación, se comparan los resultados arrojados por la heurística CVRPTW contra las instancias Benchmark, utilizando como métricas de evaluación los tiempos de viaje, los tiempos de servicio, los tiempos de espera y la distancia promedio recorrida para cada día de la semana, teniendo en cuenta todos los vehículos utilizados.

En primera instancia se comparan los tiempos de viaje medidos en minutos. Como se puede ver en la siguiente tabla, las soluciones arrojadas por el algoritmo CVRPTW heurístico reducen el tiempo de viaje entre un 13% y un 40%, con un promedio de 23%.

Tiempo Viaje	martes	miércoles	jueves	viernes	sábado	Totales
Benchmark	188	336	320	264	171	241
CVRPTW	145	290	232	158	114	186
Mejora%	23%	13%	28%	40%	33%	23%

Tabla 13. Evaluación de resultados CVRPTW, tiempos promedio de viaje respecto a Benchmark.

De la misma forma se comparan los tiempos ociosos, que es el tiempo que debe esperar el camión previo la ventana de tiempo de los clientes.

Tiempo Espera	martes	miércoles	jueves	viernes	sábado	Totales
Benchmark	123	113	103	60	64	89
CVRPTW	75	68	68	81	61	70
Mejora%	39%	40%	34%	34%	5%	21%

Tabla 14. Evaluación de resultados CVRPTW, tiempos promedio de espera respecto a Benchmark.

Complementando la tabla anterior, se analiza la cantidad de veces que es necesario esperar antes del inicio de la ventana de tiempo de los clientes para las instancias resueltas por el CVRPTW. En la tabla 15 se muestra, la cantidad total de clientes con espera y el promedio de espera en minutos cuando el inicio del recorrido es 6hs como es el caso actual. Se puede ver que la cantidad total de esperas son 44 con un promedio de 70 minutos.

Adicionalmente se observa que estas esperas suceden, en su mayoría, antes del primer cliente, esto se puede adjudicar a la holgura de las ventanas de tiempo de los clientes y la relación con las ventanas de tiempo del depósito, cabe pensar que retrasando el inicio del recorrido se reducirían la cantidad total y los minutos de espera. En la tabla 15 se agrega la cantidad total de clientes con espera y el promedio de espera si el inicio del recorrido comienza 1 hora más tarde.

Inicio de recorrido	Total Clientes Con Espera	Promedio Total Espera por recorrido[min]
6hs	44	70
7hs	31	26

Tabla 15. Total de clientes con espera y promedio total de espera por recorrido para inicio de recorrido 6hs y 7hs.

Se puede observar que, con el inicio del recorrido a las 7hs, la cantidad de clientes con espera se reduce un 30% y el promedio total de espera se reduce un 63%.

Si bien la reducción de los tiempos de espera está fuera del foco de este trabajo, en base a lo analizado se podría evaluar retrasar los inicios de las rutas para evitar esperas.

A continuación, se detallan los tiempos promedio del total de los recorridos en minutos. Esto incluye el tiempo de viaje, el tiempo de servicio y el tiempo de espera para todos los vehículos utilizados. Se puede ver que este tiempo se reduce entre un 12% y un 30%, con un promedio de 20%.

Tiempo Total Ruta	martes	miércoles	jueves	viernes	sábado	Totales
Benchmark	552	778	811	683	526	645
CVRPTW	473	757	726	526	409	578
Mejora%	14%	3%	10%	10%	22%	10%

Tabla 16. Evaluación resultados CVRPTW, tiempo total recorrido respecto a Benchmark.

Por último, se evalúa el promedio de kilómetros recorridos para cada día. El algoritmo heurístico CVRPTW reduce los kilómetros recorridos entre un 3% y un 22%, con un promedio de 10%.

Km Recorridos	martes	miércoles	jueves	viernes	sábado	Totales
Benchmark	90	207	180	125	71	124
CVRPTW	64	170	118	68	46	92
Mejora%	29%	18%	35%	46%	35%	26%

Tabla 17. Evaluación resultados CVRPTW, kilómetros recorridos respecto a Benchmark.

4.3. Escenario 3

En este apartado se analiza a cuantos clientes se le podría dar servicio utilizando el método heurístico CVRPTW descrito en la Sección 4.2.1.

Para cada día de la semana (martes a sábado) se generan instancias, comenzando con 70 clientes y se fue incrementando de a 10 hasta finalizar en 150. Se utilizaron clientes extraídos de los datos provistos por la empresa y clientes ficticios generados aleatoriamente respetando los límites de la zonificación de cada día. Para cada zona se estableció un listado de clientes reales y ficticios, aleatoriamente se seleccionaron clientes de este listado hasta alcanzar la cantidad deseada. Para estimar la demanda de los clientes reales, se calculó el promedio y el desvío estándar de cada cliente a partir de los datos, para que al momento de seleccionar uno de estos producir una demanda acorde. Para crear la demanda de los clientes ficticios, se seleccionó aleatoriamente el promedio y la desviación estándar de un cliente real.

Para cada escenario con una determinada cantidad de clientes, se proponen tres instancias diferentes en cuanto a la ubicación de cada cliente y su demanda, a ser resueltos con un máximo de tres camiones.

Dentro de este análisis en el cual se ejecutó el CVRPTW heurístico generado anteriormente utilizando Local Cheapest Insertion como estrategia de primera solución, se identificó que para ciertas instancias no se encontró solución factible, mientras que utilizando la estrategia Christofides se consiguió una solución factible. Por lo que se modificó el script para que en caso de no hallar una solución con la primera estrategia se vuelva a ejecutar con la segunda.

En la siguiente tabla se analizan la cantidad de instancias resueltas y la cantidad de camiones utilizados, teniendo como referencia la cantidad máxima de clientes observados en las instancias Benchmark. Se reportan las ejecuciones con al menos una de las tres instancias con solución. Cabe aclarar que, dado que el método utilizado es heurístico, que no se haya encontrado una solución no quiere decir que no existan soluciones factibles, pero sirve como una aproximación a la hora de evaluar una posible expansión.

Día entrega	Máxima Cantidad Clientes Benchmark	Cantidad Clientes Simulados	Instancias Resueltas	Camiones Utilizados
martes	48	70	3	2
		80	3	2
		90	3	3
		100	3	3
miércoles	66	70	3	2
		80	3	2
		90	3	3
		100	3	3
		110	3	3
		120	2	3
jueves	69	70	3	2
		80	3	2
		90	3	2
		100	3	3
		110	3	3
		120	3	3
		130	1	3
viernes	68	70	3	2
		80	3	2
		90	3	3
		100	3	3
		110	3	3
		120	3	3
		130	2	3
		140	2	3
		150	1	3
sábado	56	70	3	2
		80	3	2
		90	3	2
		100	3	2
		110	3	3
		120	3	3
		130	3	3
		140	1	3

Tabla 18. Resultados instancias simuladas.

A partir de este ejercicio se puede concluir que para cada día de la semana hay capacidad para aumentar la cantidad de clientes manteniendo la cantidad actual de camiones, e incluso prescindiendo de uno.

A continuación, se muestra proporcionalmente el potencial adicional de clientes que se podría abastecer con 2 y 3 camiones contra el Benchmark.

Día Entrega	Proporción adicional de clientes contra Benchmark	
	Con 2 camiones	Con 3 camiones
martes	69%	110%
miércoles	23%	38%
jueves	32%	75%
viernes	19%	78%
sábado	80%	130%

Tabla 19. Proporción adicional de clientes contra Benchmark con 2 y 3 camiones.

Se puede observar que en promedio con 2 camiones se podría dar servicio a 45% más de clientes, mientras que con 3 camiones un 86% más.

4.4. Escenario 4

En este apartado se analizan posibles cambios en la zonificación diaria actual, para por ejemplo reducir la cantidad de días de entrega necesarios.

Se proponen dos enfoques. En el primero se evalúa la factibilidad de agrupar zonas de entrega como están definidas actualmente, y en el segundo se consideran técnicas de clusterización de aprendizaje no supervisado.

4.4.1. Factibilidad de agrupar zonas

Viendo que existe la posibilidad de incrementar la cantidad de clientes a visitar, se plantea un escenario en el cual se agrupan las entregas de días y zonas consecutivas, como puede ser el caso de los viernes y sábados.

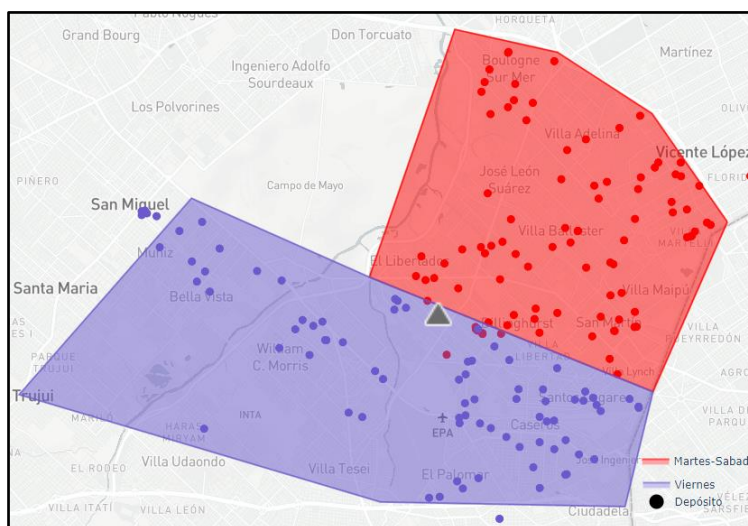


Figura 21. Zonas viernes, sábado.

En el siguiente cuadro se agrupó las instancias consecutivas para los viernes y sábados disponibles. La demanda total no excede la capacidad de los 3 camiones y la cantidad de clientes a visitar parece estar dentro de los límites, analizados previamente.

Días Agrupados	Clientes	Demanda Total
viernes 02-09 + sábado 03-09	105	9.840
viernes 09-09 + sábado 10-09	87	9.565
viernes 16-09 + sábado 17-09	109	13.846
viernes 23-09 + sábado 24-09	115	10.423
viernes 30-09 + sábado 01-09	124	11.467
viernes 07-09 + sábado 08-09	63	6.477

Tabla 20. Agrupación de entregas con días consecutivos viernes-sábado.

Para examinar la factibilidad de estos escenarios se generan instancias agrupando estos días y se los evalúa con el CVRPTW heurístico.

Se encontró que el CVRPTW heurístico ofrece soluciones factibles para estos escenarios. A continuación, se comparan los kilómetros recorridos de las instancias Benchmark contra las arrojadas por el CVRPTW heurístico. Se puede ver que aporta una mejora promedio de 25% respecto a la situación actual.

Días Agrupados	Km recorridos Benchmark	Km recorridos CVRPTW	Mejora %
viernes 02-09 + sábado 03-09	152	117	23%
viernes 09-09 + sábado 10-09	112	86	23%
viernes 16-09 + sábado 17-09	174	120	31%
viernes 23-09 + sábado 24-09	177	140	21%
viernes 30-09 + sábado 01-09	171	116	30%
viernes 07-09 + sábado 08-09	122	97	20%

Tabla 21. Agrupación de entregas con días consecutivos viernes-sábado, resultados.

Se podría considerar factible reducir a 4 los días de entrega, lo que implicaría un ahorro en la operación entre otros beneficios.

4.4.2. Generación de una nueva zonificación

En este apartado se analiza si mediante técnicas de clustering propias de aprendizaje no supervisado es posible mejorar las zonas utilizadas actualmente y que permita lograr zonas de entrega equilibradas.

Clustering se refiere a una serie de técnicas cuyo objetivo es encontrar subgrupos o clústers en un conjunto de datos. Cuando se agrupan las observaciones de un conjunto de datos, se busca dividirlos en grupos distintos para que las observaciones dentro de cada grupo sean bastante similares entre sí, mientras que las observaciones en diferentes grupos sean bastante diferentes entre sí.

Como se desarrolla en (James, 2013), la agrupación en clústeres de K- Means es un popular algoritmo de aprendizaje automático no supervisado que se utiliza para dividir un conjunto de datos en K clústeres distintos y que no se superponen. El algoritmo es simple y se usa ampliamente en diversos campos, como la segmentación de imágenes, la extracción de datos y el reconocimiento de patrones.

El objetivo de la agrupación de K- Means es dividir las observaciones en K grupos de modo que la variación total dentro del grupo, sumada en todos los K grupos, sea lo más pequeña posible.

Para lograr este objetivo, el algoritmo K- Means sigue un procedimiento iterativo simple. El algoritmo comienza seleccionando aleatoriamente K observaciones del conjunto de datos para que sirvan como centroides iniciales de los K grupos. Luego, el algoritmo asigna cada observación al centroide más cercano, según la distancia euclidiana entre la observación y el centroide. Una vez que todas las observaciones se han asignado a un grupo, el algoritmo actualiza los centroides de cada grupo tomando la media de todas las observaciones asignadas a ese grupo. Luego, el algoritmo repite este proceso hasta que los centroides ya no cambian o se alcanza un número máximo de iteraciones.

Una de las principales ventajas de la agrupación de K- Means es su simplicidad y velocidad. El algoritmo es fácil de implementar y puede manejar grandes conjuntos de datos de manera eficiente.

Este algoritmo está implementando en la librería scikit-learn para Python. Para este caso particular, se generan 4 clúster utilizando como input el conjunto de latitud y longitud de cada cliente, para encontrar 4 grupos que correspondan a 4 días de entrega.

Se fijó una semilla para poder obtener la misma solución en sucesivas corridas. La asignación del algoritmo es la que se muestra en la figura 21.

En la tabla 21 se reporta la distancia promedio entre clientes (DPC) dentro del mismo cluster. Se observa que, la distancia promedio entre clientes dentro del mismo cluster se reduce en la clusterización realizada por medio de K-Means. Esto permite rutas mejor balanceadas en cuanto a distancia total recorrida y demanda.

Una limitación de este algoritmo es que no permite determinar una cantidad máxima y mínima de puntos por cluster, incluso es posible que se generen clusters vacíos en ciertas casuísticas.

Como resultado del ejercicio del apartado 4.3 se deduce que con 3 camiones se le podría dar servicio a entre 100 y 130 clientes, sería útil que el algoritmo de clusterización tenga en cuenta esta limitación y pueda generar cluster acordes a estos límites, y de esta forma tener zonas de entrega equilibradas.

El algoritmo planteado en el artículo (Bradley, 2000), propone una solución al límite mínimo y máximo de puntos para cada clúster por medio de restricciones. El agrupamiento restringido de K-Means es una modificación del algoritmo tradicional de K-Means que aborda algunas de sus limitaciones, al agregar restricciones al problema de optimización. Se ha demostrado que los métodos propuestos, en este artículo, para agregar restricciones son efectivos para evitar soluciones locales con clústeres vacíos o con muy pocos puntos y al mismo tiempo fijar un límite máximo.

Este algoritmo está implementado en la librería para Python k-means-constrained, que utiliza el recurso SimpleMinCostFlow de las herramientas de investigación de operaciones de Google (OR-Tools).

Como se propuso anteriormente con el algoritmo K-Means tradicional, se generan 4 clústers restringiendo la cantidad entre 70 y 100 clientes por clúster. De esta forma se generan 4 grupos que correspondan a 4 días de entrega.

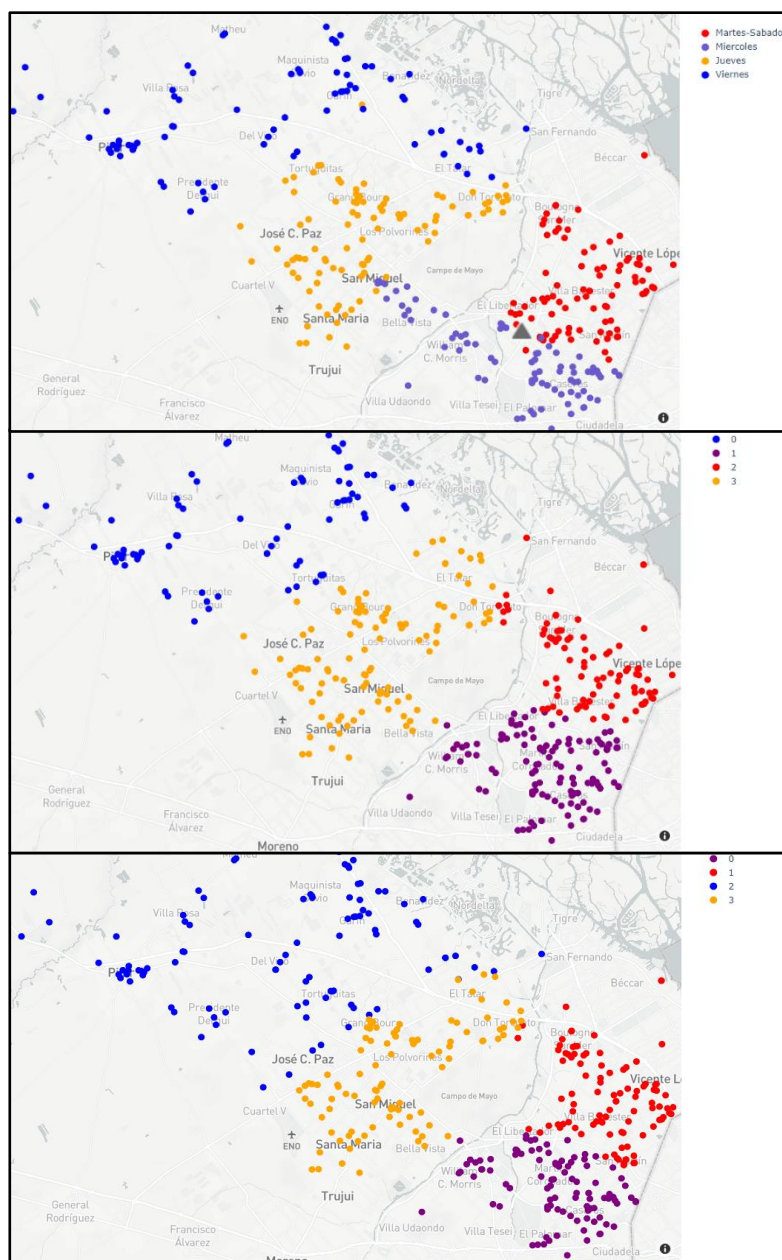


Figura 22. Arriba: zonas actuales; Medio: clusters generados con K-Means; Abajo: clusters generados con K-Means restringido.

Se calcula la distancia promedio de clientes entre clientes (DPC) dentro del mismo cluster y lo se lo compara contra los clústers del K-Means.

Zonas actuales	Clientes	DPC	Cluster K-Means	Clientes	DPC	Cluster K-Means Restringido	Clientes	DPC
martes-sábado	93	5,4	0	75	8,5	0	71	5,4
miércoles	83	12,1	1	106	5,6	1	92	3,5
jueves	98	9,0	2	62	4,4	2	100	8,1
viernes	83	5,7	3	114	8,5	3	94	10,3
Promedio		8,0	Promedio		6,7	Promedio		6,8

Tabla 22. Distancia promedio de clientes entre clientes para cada zona-cluster con K-Means Restringido.

Se observa que la distancia promedio entre clientes dentro del mismo cluster se reduce en la clusterización realizada por medio de K-Means restringido y que se respeta el límite máximo de clientes por cluster, no se aprecia diferencias significativas con el K-Means simple.

Tanto para el agrupamiento generado por el algoritmo K-Means y K-Means restringido, se crearon zonas que tienen similitud con la zonificación actual. El aspecto en el que se nota una mejora en ambas agrupaciones K-Means es en la transición de las zonas correspondientes a jueves-viernes y jueves-miércoles, como se muestra a continuación.

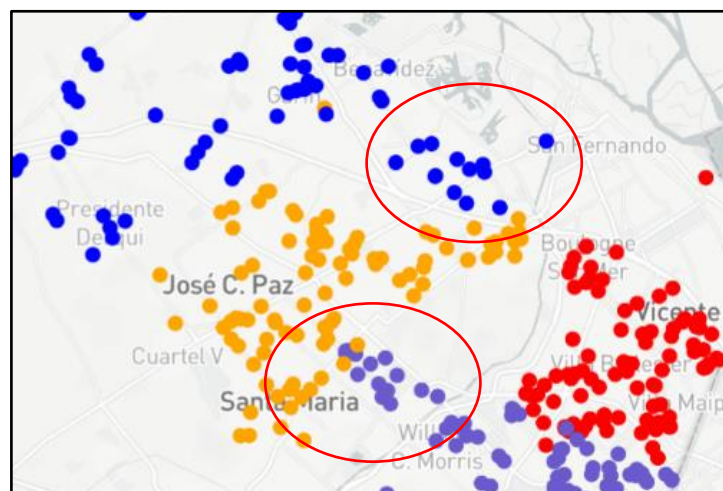


Figura 23. Transición entre zonas actual.

Esto que se ve reflejado en la DPC, ya que los puntos marcados se encuentran notoriamente alejados del resto de los otros puntos de este cluster, lo que hace aumentar el DPC.

Se evalúa la factibilidad de generación de rutas con el CVRTW heurístico para las agrupaciones realizadas anteriormente. Se generaron 4 instancias correspondientes a los 4 clústeres generados, con el total de clientes correspondientes a cada uno de estos, con una demanda

definida en función del promedio de la demanda de cada cliente, a resolver con 3 camiones y respetando las ventanas de tiempo. Se encontró una solución factible para estas 4 instancias. A continuación, se reportan el total de kilómetros recorridos, tiempo de viaje y el promedio de vehículos utilizados. Se observan resultados similares en ambas agrupaciones.

Agrupación	Km recorrido	Tiempo de Viaje[min]	Promedio vehículos Utilizados
K-Mean	515	1.050	2,25
K-Mean Restringido	532	1.070	2,50

Tabla 23. Solución de CVRTW heurístico para nueva agrupación de zonas.

Tanto en el apartado anterior como en este, se comprobó que ya sea agrupando días de entrega (viernes y sábado) como creando nuevos clusters es posible darle servicio a la misma cantidad de clientes en 4 días de entrega en vez de 5, dejando todavía capacidad para más clientes.

5. Análisis de Negocio

En este apartado se condensa lo expuesto anteriormente para dar respuesta a lo planteado en la Sección 1.3.

Para complementar lo desarrollado en este trabajo y poder evaluar un ahorro económico en cada uno de los escenarios, “El Universo del Panadero” proveyó una matriz de costos operativos, fijos y variables relacionados a un camión. Dentro de los costos fijos se tiene en cuenta seguro, servicio de seguimiento vehicular, patente, salario del chofer, salario del ayudante y cargas sociales. Dentro de los costos variables se contempla costo de combustible y de mantenimiento. Los costos son expresados en dólares teniendo en cuenta el tipo de cambio correspondiente al periodo analizado.

Costos camión	Mensual	Diario
Costos Fijos	US\$ 1.955	US\$ 70
Costos variables	US\$ 288	US\$ 10
TOTAL	US\$ 2.243	US\$ 80

Tabla 24. Matriz de costos operativos de un camión. Con 28 días laborables por mes.

Como se mostró en la Tabla 5, un camión recorre en promedio 47 km por día, por lo que se podría estimar que el costo de combustible por kilómetro recorrido es de u\$0.22.

- **Escenario 1(situación actual):** ¿Qué potencial de ahorro aportaría la utilización de técnicas de análisis prescriptivo para resolver las instancias respetando la segmentación por zonas actual? (en la cual un determinado chofer tiene asignada una zona para cada día)

En este escenario el ahorro se traduce en reducción de costos variables que son proporcionales a los kilómetros recorridos por cada camión.

Si cada camión recorre 47 kilómetros diarios, con un costo de US\$ 0,22 por kilómetro recorrido, y una reducción aportada por el método heurístico TSPTW de un 13%, equivaldría a un ahorro mensual (28 días) de US\$ 37 o US\$ 450 anual.

- **Escenario 2:** ¿Qué potencial de ahorro aportaría la utilización de técnicas de análisis prescriptivo para resolver las instancias si se modificara la zonificación actual y permitiendo la asignación libre de clientes a camiones?

Este escenario es el que se abordó con el CVRPTW heurístico, en donde el ahorro no solo viene por la reducción de kilómetros recorridos, sino que se agrega un ahorro por la disminución en la cantidad de vehículos necesarios.

Como se vio Tabla 11, el método CVRTW heurístico resolvió las 26 instancias de prueba con 2 o menos camiones, por lo que se podría prescindir de un camión lo que generaría un ahorro mensual de US\$ 2.243 o US\$ 26.918 anual.

Por otro lado, respecto a la reducción de kilómetros recorridos, como se ve en la Tabla 16, se redujo un 26%, lo que implica un ahorro mensual (28 días) de US\$ 75 o US\$ 898 anual.

Sumando el ahorro en costos fijos y variable se alcanzaría a US\$ 2.318 mensual o US\$ 27.817 anual, lo que implica un ahorro total de 34%.

- **Escenario 3:** ¿A cuántos clientes se podría abastecer con la cantidad de camiones actual, rompiendo la segmentación?

Contestar a esta interrogante reviste una significativa importancia, ya que en cierta medida influye en el alcance de crecimiento que puede experimentar la empresa. Conocer cuántos clientes puede atender diariamente un vehículo se convierte en un dato de relevancia clave.

Como se mostró en la Tabla 18, en la situación actual con 3 camiones rompiendo la segmentación y resolviendo el problema del ruteo con el método CVRPTW heurístico, se podría dar servicio en promedio a 86% más de clientes.

- **Escenario 4:** Analizar posibles cambios en la zonificación diaria actual, para por ejemplo reducir la cantidad de días de entrega necesarios.

Tanto en el Escenario 3 como en el Escenario 4, se comprobó que ya sea agrupando días de entrega (viernes y sábado) como generando nuevos clusters es posible darle servicio a la misma cantidad de clientes en 4 días de entrega en vez de 5, quedando todavía capacidad para más clientes. Lo que reduciría entre 4 y 5 días de entregas al mes lo que podría generar un ahorro mensual de US\$ 721 o US\$ 8.652 anual.

6. Conclusiones – Posibles mejoras

Según lo desarrollado en este trabajo, poner en funcionamiento un sistema que resuelva el problema de ruteo, por medio de heurísticas avanzadas como las implementadas en las bibliotecas de OR-Tools, puede resultar en un ahorro considerable de los costos operativos.

Principalmente se recomienda romper con la zonificación diaria actual, en la que cada camión tiene definida una zona dentro de cada día de entrega, y así permitirle al método una libre asignación de clientes a vehículos.

Por otro lado, a raíz de los análisis realizados, se sugiere evaluar la agrupación de zonas ya sea agrupando días de entrega (viernes y sábado) como generando nuevos clusters, para así reducir los días de entrega.

Como se comentó anteriormente, todos los días por la tarde un empleado administrativo descarga un listado de clientes a ser visitados y los productos a entregar, preasignados a cada camión. En la mañana siguiente los choferes toman estos listados y siguiendo su intuición y experiencia, establecen el orden en que se realizarán las entregas, respetando las ventanas horarias de cada cliente.

De implementarse este sistema, la generación diaria del ruteo sería de forma rápida y práctica, por medio de un ejecutable que consumiría el listado de clientes a ser visitados y los productos a entregar, asignando clientes a camiones y determinando un orden de visita, lo que no tomaría más de 2 minutos, lo que resulta insignificante contra el tiempo que necesita el chofer para armar por sí solo las rutas, permitiéndole estar más enfocado tanto en la conducción como en el vehículo.

En el desarrollo de este trabajo se construyó una matriz de distancia geodésica simétrica, una posible mejora consistiría en construir la matriz de distancia y tiempo necesarios por medio de la API de "Google Maps" que genera estas matrices teniendo en cuenta el camino efectivo entre paradas y el estado actual del tránsito.

Otro aspecto que se podría mejorar sería tener en cuenta la regularidad del servicio. La regularidad del servicio se refiere a los lapsos de tiempo en los que un cliente recibe atención a lo largo de diferentes días. Por ejemplo, si el cliente A es atendido en una semana entre las 8 y las 10 de la mañana, y en otra semana entre las 9 y las 11 de la mañana, el "intervalo de atención" abarca de 8 a 11 horas. Cuanto más breve sea este intervalo, el cliente A percibirá una mayor consistencia en el servicio. Es importante destacar que este intervalo de atención semanal

difiere de las "ventanas de tiempo", las cuales se consideran como información inicial del problema. En cambio, los intervalos de atención surgen del análisis de los resultados de las visitas diarias a lo largo de las semanas. En resumen, los beneficios de mantener una consistencia en el servicio se reflejan principalmente en la satisfacción del cliente, un aspecto esencial para las empresas.

De los resultados arrojados por el método heurístico CVRPTW, se observa que en ocasiones se generan rutas desbalanceadas entre camiones, como se puede ver en la Figura 23, en donde el algoritmo asignó 5 clientes a un camión y 37 a otros. Este es otro aspecto interesante para ser analizados en posibles mejoras.

Otra herramienta complementaría que se podría desarrollar es un script que tome la cartera de clientes y genere la cantidad de clusters necesarios, por medio del algoritmo K-Means restringido. Esto podría ser útil en caso de que la empresa incorpore nuevos clientes o zonas de entrega.

7. Referencias

- Bradley, P. S., Bennett, K. P., & Demiriz, A. (2000). Constrained k-means clustering. Microsoft Research, Redmond, 20(0), 0.
- Christosofides, N. (1976). Worst-Case Analysis of a New Heuristic for the Travelling Salesman Problem. *Operations Research Forum*.
- Gendreau, M., & Potvin, J. Y. (Eds.). (2010). Handbook of metaheuristics (Vol. 2, p. 9). New York: Springer.
- Glover, F. (1977). Future paths for integer programming and links to artificial intelligence. *Computers & operations research*, 533–549.
- Holland, C., Levis, J., Nuggehalli, R., Santilli, B., & Winters, J. (2017). UPS optimizes delivery routes. *Interfaces*, 47(1), 8-23.
- Ioannou, G., Kritikos, M., & Prastacos, G. (2001). A greedy look-ahead heuristic for the vehicle routing problem with time windows. *Journal of the Operational Research Society*, 52(5), 523-537.
- James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). An introduction to statistical learning (Vol. 112, p. 18). New York: springer.
- Rosenkrantz, D. J., Stearns, R. E., & Lewis, P. M. (1974, October). Approximate algorithms for the traveling salesperson problem. In 15th Annual Symposium on Switching and Automata Theory (swat 1974) (pp. 33-42). IEEE.
- Rosenkrantz, D. J., Stearns, R. E., & Lewis, II, P. M. (1977). An analysis of several heuristics for the traveling salesman problem. *SIAM journal on computing*, 6(3), 563-581.
- Suzuki, Y., & Kabir, Q. S. (2015). Green vehicle routing for small motor carriers. *Transportation Journal*, 54(2), 186-212.
- Toth, P., & Vigo, D. (Eds.). (2014). Vehicle routing: problems, methods, and applications. Society for industrial and applied mathematics.
- Voudouris, C., & Tsang, E. (1999). Guided local search and its application to the traveling salesman problem. *European journal of operational research*, 113(2), 469-499.