

# Trabajo Fin de Grado en Ingeniería de las Tecnologías Industriales

## **Diseño de una heurística constructiva para el problema logístico de reparto híbrido camión-dron**

Autor: Valerio Fernández Ramírez

Tutor: Pedro Luís González Rodríguez

Dpto. Organización Industrial y Gestión de  
Empresas I

Escuela Técnica Superior de Ingeniería

Sevilla, 2026





**GRADO EN INGENIERÍA DE TECNOLOGÍAS INDUSTRIALES  
ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA**

**TRABAJO FIN DE GRADO  
CURSO ACADÉMICO [2025-2026]**

TÍTULO:

**DISEÑO DE UNA HEURÍSTICA CONSTRUCTIVA PARA EL PROBLEMA DE  
REPARTO HÍBRIDO CAMIÓN-DRON**

AUTOR:

**VALERIO FERNÁNDEZ RAMÍREZ**

TUTOR:

**Dr. D. PEDRO LUIS GONZÁLEZ RODRÍGUEZ**

DEPARTAMENTO:

**ORGANIZACIÓN INDUSTRIAL Y GESTIÓN DE EMPRESAS I**

RESUMEN:

Este Trabajo de Fin de Grado aborda el problema de reparto de última milla mediante un sistema híbrido formado por un camión y un dron. El objetivo principal es el desarrollo de una heurística constructiva capaz de generar soluciones iniciales de buena calidad en tiempos de cómputo muy reducidos, atendiendo a la necesidad de obtener buenas soluciones a tiempo real y sirviendo como punto de partida para métodos de optimización y re-scheduling posteriores.

La propuesta se centra en la asignación eficiente de clientes al camión y al dron, teniendo en cuenta las restricciones operativas del sistema y minimizando el tiempo total de reparto (tiempo de misión o makespan). Los resultados obtenidos muestran que la heurística planteada permite obtener soluciones competitivas de forma rápida, lo que la

convierte en una herramienta adecuada para problemas de reparto de última milla a media y gran escala.

## ÍNDICE

---

1	JUSTIFICACIÓN DEL PROYECTO .....	1
1.1	MOTIVACIÓN Y RELEVANCIA .....	1
1.2	BENEFICIOS Y NECESIDAD DEL ESTUDIO .....	2
1.3	OBJETIVO GENERAL.....	3
1.4	OBJETIVOS ESPECÍFICOS.....	3
1.5	ESTRUCTURA DEL DOCUMENTO .....	4
2	ESTADO DEL ARTE .....	7
2.1	REVISIÓN DE ESTUDIOS Y SOLUCIONES EXISTENTES .....	7
3	DESCRIPCIÓN DEL PROBLEMA .....	13
4	PLANTEAMIENTO DE LA HEURÍSTICA.....	16
5	EXPERIMENTACIÓN .....	23
5.1	COMPARACIÓN CON METAHEURÍSTICAS .....	23
5.2	COMPARACIÓN CON PROBLEMA TSP CLÁSICO.....	36
6	CONCLUSIÓN.....	38
7	REFERENCIAS .....	41

# 1 JUSTIFICACIÓN DEL PROYECTO

## 1.1 MOTIVACIÓN Y RELEVANCIA

La distribución de última milla representa actualmente entre el 40% y el 55% del coste total logístico en operaciones urbanas (Khattar, 2023), impulsada por factores como la expansión del comercio electrónico, el aumento del tráfico, la fragmentación de pedidos y la exigencia de entregas inmediatas. A ello se suma la presión regulatoria para adoptar sistemas de transporte más sostenibles, lo que obliga a reducir distancias, emisiones y tiempos sin afectar la calidad del servicio.

En este contexto surge un interés creciente por modelos híbridos basados en la cooperación entre camiones y drones. El dron, como recurso aéreo, permite realizar entregas rápidas en zonas difíciles o dispersas, mientras que el camión actúa como plataforma móvil de lanzamiento y recuperación. Esta cooperación introduce un modelo logístico altamente sincronizado y temporal que supera en complejidad al TSP y a la mayoría de VRP tradicionales.

En la **Figura 1** se puede observar un ejemplo de dron utilizado en servicios de reparto de última milla, desarrollado por la empresa *Meituan*, que ilustra el tipo de vehículos aéreos no tripulados que actualmente se están incorporando a sistemas logísticos para realizar entregas rápidas de pequeños paquetes.



**Figura 1: Dron de entrega Meituan**

**Fuente:** (36KrEnglish, 2024)

El problema asociado, conocido como *Truck-Drone Team Logistic (TDTL)* (González-R et al., 2020), una generalización del conocido problema *Flying Sidekick Traveling Salesman Problem (FSTSP)*, es NP-hard y presenta desafíos adicionales: autonomía limitada del dron, posibilidad de misiones encadenadas y la necesidad de coordinar las rutas para evitar tiempos muertos en cualquiera de los dos vehículos. Dado que el objetivo pasa de minimizar distancia a minimizar el tiempo de la misión (makespan), surge una problemática central para la planificación moderna en última milla.

Más allá de su complejidad teórica, uno de los principales retos en estos sistemas es su aplicabilidad operativa. En entornos reales, las decisiones logísticas deben tomarse en horizontes temporales muy reducidos y bajo condiciones dinámicas, como cambios en la demanda, incidencias de tráfico o restricciones energéticas. Por ello, existe una necesidad creciente de métodos capaces de generar soluciones de buena calidad en tiempos de cálculo muy reducidos, idealmente cercanos al tiempo real. Las aproximaciones exactas, aunque precisas, suelen resultar inviables para instancias de tamaño medio o grande, lo que refuerza el interés por enfoques más ágiles y escalables.

En este sentido, el desarrollo de modelos y algoritmos eficientes no solo tiene valor como herramienta directa de planificación, sino también como componente estructural dentro de metodologías más avanzadas. Una solución rápida y bien estructurada puede emplearse de forma autónoma en entornos donde la rapidez sea prioritaria, o bien integrarse como procedimiento constructivo, heurística inicial o módulo de evaluación dentro de metaheurísticas más complejas. Este doble enfoque aumenta significativamente la utilidad práctica de la investigación, al permitir su aplicación tanto en sistemas operativos reales como en marcos de optimización más amplios.

## **1.2 BENEFICIOS Y NECESIDAD DEL ESTUDIO**

Los métodos exactos como MILP o programación dinámica, aunque teóricamente robustos, solo pueden aplicarse a instancias pequeñas ( $\leq 25-30$  nodos), tamaños muy inferiores a los usados por empresas logísticas reales (150–400 entregas por ruta) (Izquierdo & Mayo, 2021). Esto hace imprescindible

el desarrollo de técnicas heurísticas y metaheurísticas que produzcan soluciones factibles y de calidad en tiempos razonables.

Entre las metaheurísticas existentes diseñadas para resolver el problema TDTL o algunas de sus variantes podemos encontrar ALNS o *Simulated Annealing* (SA), que han demostrado buen rendimiento, pero dependen de forma crítica de la calidad de la solución inicial. Una solución inicial deficiente conduce a exploraciones ineficientes del espacio de búsqueda, generando tiempos elevados o soluciones de baja calidad.

Esta limitación revela un vacío claro en la literatura: no existen modelos constructivos y deterministas capaces de generar soluciones iniciales simultáneamente rápidas, escalables a cientos de nodos y estructuralmente informadas por el impacto temporal de cada cliente en el tiempo de la misión.

### **1.3 OBJETIVO GENERAL**

El objetivo del proyecto es desarrollar una heurística constructiva inicial robusta y eficiente para el problema TDTL, capaz de generar soluciones de calidad de forma rápida y escalable. Con ello se busca mejorar el rendimiento de las metaheurísticas avanzadas reduciendo la dependencia de métodos iniciales aleatorios o greedy, y proporcionando una base sólida que optimice de manera significativa los procesos actuales de resolución del problema TDTL. Para ello, diseñaremos una solución inicial “inteligente” que permita introducir mejoras significativas sin incrementar de forma exponencial la complejidad del problema, aplicando ajustes bien fundamentados que optimicen progresivamente la calidad de la solución.

### **1.4 OBJETIVOS ESPECÍFICOS**

Para alcanzar ese objetivo general planteado en el presente Trabajo de Fin de Grado, se abordarán los siguientes puntos:

- **O.E1. Realizar una revisión detallada del estado del arte**, analizando modelos exactos, heurísticos y metaheurísticos existentes, con especial atención a extensiones recientes para identificar limitaciones actuales y justificar la necesidad de una heurística inicial más informada.
- **O.E2. Definir formalmente el problema**, especificando los elementos involucrados, las restricciones operativas del dron y del camión, y el criterio de optimización basado en la minimización del tiempo total de misión.
- **O.E3. Desarrollar la lógica de la heurística propuesta**, explicando los criterios de selección y priorización de clientes en función del impacto sobre la ruta del camión y la sincronización con el dron, así como describir la estructura matemática que sustenta el proceso de decisión y el funcionamiento del algoritmo.
- **O.E4. Realizar una experimentación computacional** que permita validar el uso y la calidad de la heurística, analizando su rendimiento mediante instancias de prueba y evaluando su eficacia y robustez.
- **O.E5. Presentar y analizar los resultados obtenidos**, evaluando la calidad y estabilidad de las soluciones generadas, los tiempos de cómputo y la comparación con enfoques convencionales, determinando así su utilidad como punto de partida para procesos de optimización más avanzados.
- **O.E6. Razonar una conclusión** que sintetice los avances logrados, evalúe el grado de cumplimiento de los objetivos planteados y destaque la contribución de la heurística propuesta al estudio y resolución del TDTL.

## 1.5 ESTRUCTURA DEL DOCUMENTO

El presente Trabajo de Fin de Grado se estructura en varios capítulos que desarrollan de forma progresiva el estudio realizado. En el Capítulo 2 se presenta el estado del arte, donde se revisan los principales trabajos existentes en la literatura sobre el reparto híbrido camión–dron, analizando los modelos exactos, heurísticos y metaheurísticos más relevantes. Continuaremos en el Capítulo 3, donde se describe formalmente el problema abordado, definiendo los elementos del sistema logístico, las hipótesis adoptadas y las restricciones operativas consideradas. A continuación, en el Capítulo 4, se expone el planteamiento de la heurística constructiva propuesta, detallando el funcionamiento del algoritmo,

el pseudocódigo empleado y los criterios utilizados para la asignación de clientes al camión o al dron. Posteriormente, en el Capítulo 5, se desarrolla la experimentación computacional y el análisis de resultados, comparando el rendimiento del método propuesto con otros enfoques presentes en la literatura. Finalmente, en el Capítulo 6, se presentan las conclusiones del trabajo, donde se sintetizan los resultados obtenidos, se evalúa el grado de cumplimiento de los objetivos planteados y se señalan posibles líneas de investigación futura.

## 2 ESTADO DEL ARTE

### 2.1 REVISIÓN DE ESTUDIOS Y SOLUCIONES EXISTENTES

En el ámbito del reparto asistido por drones, el *FSTSP* describe la cooperación entre un camión y un dron que se lanza y recupera en distintos puntos para acelerar las entregas, estableciendo el modelo base para la coordinación terrestre-aérea (Murray & Chu, 2015). Como extensión directa, los mismos autores formulan el *Parallel Drone Scheduling TSP* (PDSTSP), en el cual muchos clientes están suficientemente cerca del depósito para que el dron pueda atenderlos directamente, mientras el camión sirve a los más lejanos, dividiendo así el área de servicio entre ambos vehículos según su rango de operación. Más adelante se introduce el TSP-D (Agatz et al., 2018), una variante donde camión y dron viajan por la misma red y el dron debe despegar y aterrizar en un mismo punto, simplificando restricciones y permitiendo tratar el problema mediante programación dinámica. Finalmente, el *Drone Delivery Problem* (DDP) estudia el escenario donde solo drones realizan las entregas, con rutas que parten y regresan al depósito para recargar baterías, aportando formulaciones MILP para instancias pequeñas y un Simulated Annealing para instancias grandes (Dorling et al., 2017).

A día de hoy, el avance más relevante dentro del ámbito del FSTSP y objeto del estudio llevado a cabo es la extensión conocida como FSTSP-MD con un dron con capacidad de realizar entregas múltiples y en la que el camión comienza y termina su trayectoria en el mismo nodo que funciona como almacén, llamado *depot*. Este modelo reconoce que los drones comerciales actuales ya no están limitados a realizar una única entrega por misión, sino que pueden servir a varios clientes antes de regresar al camión, siempre respetando límites de carga y autonomía.

Como podemos comprobar, se trata de un problema estudiado cuyo interés ha sido creciente en los últimos años debido a su potencial aplicación y a su relevancia en los procesos de reparto de última milla.

En el contexto del problema FSTSP-MD, el trabajo de Schaumann (Schaumann et al., 2025) constituye una de las aportaciones más completas y recientes sobre este problema.

En dicho trabajo, los autores desarrollan una heurística basada en un esquema *order-first, split-second* que introduce un algoritmo de partición tipo shortest path capaz de construir en tiempo polinómico la mejor combinación camión–dron para una secuencia dada de clientes. Esta estructura permite explorar grandes vecindarios del espacio TSP sin necesidad de enumerar todas las misiones posibles del dron, lo que sería exponencial.

Los resultados computacionales del artículo muestran que MD-SPP-H supera a todas las metaheurísticas previas tanto en FSTSP como en FSTSP-MD, resolviendo instancias de hasta 250 clientes con gran calidad y tiempos razonables. Además, los autores proporcionan conclusiones de valor para la gestión logística, destacando que:

- Los drones con capacidad para múltiples entregas reducen significativamente el tiempo de la misión frente al camión solo.
- Los beneficios marginales de aumentar la velocidad o el número de entregas consecutivas del dron presentan rendimientos decrecientes.
- La distribución espacial de los clientes condiciona fuertemente el ahorro operativo.

La relevancia de este trabajo radica en que representa el estado del arte actual, tanto por sus aportaciones metodológicas como por su aproximación realista a las restricciones operativas de los drones comerciales. Su enfoque sirve como base fundamental para proyectos que, como este TFG, buscan desarrollar nuevas heurísticas constructivas iniciales capaces de generar soluciones informadas y escalables para variantes del problema TDTL.

La literatura coincide en que la solución inicial influye fuertemente en la calidad final alcanzada por las metaheurísticas. Por ello existe una oportunidad clara de mejora: diseñar una heurística inicial rápida pero más inteligente en la selección de los clientes a asignar al dron.

Dentro del conjunto de trabajos analizados en este proyecto destaca el modelo TDTL (González-R et al., 2020).

Su aportación principal reside en el desarrollo de una metaheurística basada en un enfoque Iterated Greedy (IG) combinada con SA, capaz de obtener soluciones de calidad incluso en escenarios de gran tamaño donde los métodos exactos resultan computacionalmente inviables.

Uno de los aspectos más significativos del trabajo es la insistencia en la necesidad de construir una solución inicial robusta y eficaz antes de aplicar la metaheurística. Los autores demuestran que la calidad del punto de partida condiciona directamente el rendimiento global del algoritmo, tanto en velocidad como en la profundidad de la mejora alcanzable. Para ello, proponen un procedimiento específico que parte de una ruta inicial basada en un TSP aproximado mediante 2-opt, que posteriormente se transforma en una solución factible mediante la identificación sistemática de los puntos de sincronización entre camión y dron utilizando Dijkstra. Aunque el procedimiento es eficiente, no resulta escalable para más de 250 nodos.

En esta misma línea, el trabajo (Agatz et al., 2018) constituye otra referencia esencial dentro del estado del arte, especialmente por su formalización del problema y por la claridad de su enfoque route-first, cluster-second. En su propuesta, desarrollada para el problema TSP-D en el que el dron debe despegar y aterrizar en un mismo nodo, la construcción de una solución inicial se basa en generar en primer lugar un tour TSP completo, obtenido mediante algoritmos heurísticos o exactos, que actúa como secuencia estructural sobre la que se realiza posteriormente la asignación de clientes al dron. Esta partición se lleva a cabo mediante el Greedy Partitioning Heuristic, que respeta el orden del tour inicial y decide de forma eficiente qué entregas puede asumir el dron sin comprometer la viabilidad temporal del sistema. Aunque su solución inicial es conceptualmente sencilla, el estudio demuestra que partir de una secuencia bien definida permite explorar de manera más eficaz el espacio de soluciones y facilita la posterior mejora mediante búsquedas locales. Este planteamiento refuerza la idea central de este TFG: la necesidad de diseñar heurísticas constructivas iniciales que no solo sean rápidas, sino también suficientemente inteligentes como para proporcionar, desde el comienzo, una estructura de ruta que permita al algoritmo aprovechar todo el potencial del dron en etapas posteriores.

En el método propuesto por (Ha et al., 2017) para la solución del problema TSP-D, la heurística GRASP construye cada solución inicial partiendo de un TSP

convencional sin incorporar aún ninguna consideración propia del TSP-D. A partir de esta ruta básica, el algoritmo aplica un procedimiento de split que transforma el TSP en una solución en la que se introduce el dron, y posteriormente ejecuta diversos operadores de búsqueda local para refinarla. Sin embargo, este split presenta una complejidad computacional elevada, concretamente  $O(n^4)$ , debido a la construcción del grafo auxiliar y al cálculo exhaustivo de todos los arcos y posibles entregas con dron, lo que limita su escalabilidad y aumenta significativamente el coste por iteración. Por tanto, los resultados del GRASP podrían mejorar de forma significativa si partiera de una solución inicial más informada y adaptada, reduciendo la dependencia de un TSP convencional y aprovechando mejor el elevado coste computacional del procedimiento de split.

En la literatura relacionada con este problema también hemos podido encontrar una metaheurística que no depende de la solución inicial (Windras Mara et al., 2022). En el artículo, los autores señalan que se empieza desde una solución inicial aleatoria, ya que el algoritmo ALNS, en cada iteración, destruye gran parte de la solución recogida para reconstruirla utilizando reinserciones guiadas. Si bien esta metaheurística da buenos resultados, no ha sido probada en instancias de más de 200 nodos, por lo que su rendimiento en problemas de mayor escala todavía no ha sido verificado.

A la vista de la revisión de la literatura, a través de las aportaciones más relevantes, podemos realizar la clasificación que se muestran en la siguiente tabla:

Autor	Problema estudiado	Enfoque o método	Idea principal	Limitaciones
(Murray & Chu, 2015)	FSTSP	Formulación matemática + heurística	Introducen el problema de cooperación camión–dron para reparto de última milla	Solo permite una entrega por misión del dron
(Dorling et al., 2017)	Drone Delivery Problem (DDP)	MILP + Simulated Annealing	Modelan el reparto realizado exclusivamente por drones considerando consumo energético	Escalabilidad limitada para grandes instancias
(Agatz et al., 2018)	TSP-D	Route-first, cluster-second	Construcción de una ruta TSP inicial sobre la que se asignan clientes al dron	Simplifica algunas restricciones operativas del dron
(Ha et al., 2017)	TSP-D	GRASP + búsqueda local	Generación de soluciones iniciales a partir de un TSP seguido de un procedimiento split	Procedimiento split computacionalmente costoso
(González-R et al., 2020)	TDTL	Iterated Greedy + Simulated Annealing	Metaheurística capaz de resolver instancias de mayor tamaño.	Dependencia de la calidad de la solución inicial
(Windras Mara et al., 2022)	FSTSP-MD	ALNS	Estrategia adaptativa basada en destrucción y reinserción de nodos	Escasa validación en instancias de gran tamaño
(Schaumann et al., 2025)	FSTSP-MD	Order-first split-second heuristic	Uso de un algoritmo de partición tipo shortest path para generar combinaciones camión–dron eficientes	Coste computacional mayor que heurísticas simples

**Tabla 1**

**Fuente:** *Elaboración propia*

Tras la revisión de la literatura, la **Tabla 1** resume las principales contribuciones existentes en el ámbito del reparto híbrido camión–dron. Como puede observarse, los primeros trabajos se centraron en la formulación y modelado del problema, introduciendo variantes como el FSTSP o el TSP-D que establecen las bases para la coordinación entre vehículos terrestres y drones. Posteriormente, la investigación se ha orientado hacia el desarrollo de algoritmos

capaces de resolver instancias de mayor tamaño, destacando especialmente el uso de metaheurísticas como *IG*, *SA* o *Adaptive Large Neighborhood Search*. Estos métodos han demostrado ser capaces de obtener soluciones de alta calidad para problemas complejos, aunque su rendimiento suele depender en gran medida de la calidad de la solución inicial y del ajuste de sus parámetros.

En este contexto, surge la necesidad de desarrollar heurísticas constructivas que permitan generar soluciones iniciales de forma rápida y eficiente, manteniendo al mismo tiempo una estructura adecuada para su posterior mejora mediante metaheurísticas. Este objetivo constituye precisamente la motivación principal del presente trabajo.

### 3 DESCRIPCIÓN DEL PROBLEMA

El problema abordado en este trabajo consiste en determinar las rutas de un camión y un dron que cooperan para visitar un conjunto de clientes, partiendo de un nodo origen y finalizando en un nodo destino. Este tipo de problemas surge en escenarios donde la capacidad del dron para operar de forma autónoma está limitada principalmente por la duración de la batería, lo que hace necesaria su coordinación con un vehículo terrestre que actúe como base móvil.

Dado que un único dron no puede atender a todos los clientes debido a su autonomía limitada, el camión asume el papel de estación móvil de intercambio de baterías. Tradicionalmente, muchos trabajos han considerado al camión como un mero vehículo de soporte, encargado únicamente de transportar el dron y servir como punto de lanzamiento y recuperación (Poikonen, 2017). Sin embargo, en aplicaciones reales resulta natural permitir que el camión también participe activamente en el servicio, atendiendo directamente a parte de los clientes.

Asimismo, gran parte de la literatura asume que la ruta del camión está total o parcialmente predefinida, ya sea por restricciones operativas o por la necesidad de que ciertos clientes sean atendidos obligatoriamente por el vehículo terrestre (Murray & Chu, 2015). Aunque este enfoque puede ser adecuado en problemas de reparto con cargas heterogéneas, resulta menos representativo en contextos donde todos los clientes pueden ser atendidos indistintamente por cualquiera de los dos recursos, como ocurre en escenarios de reparto ligero o misiones de inspección. Por ello, en este trabajo se adopta un enfoque más general, en el que tanto el camión como el dron pueden servir a cualquier cliente, y las decisiones de enrutamiento y sincronización se determinan de forma conjunta.

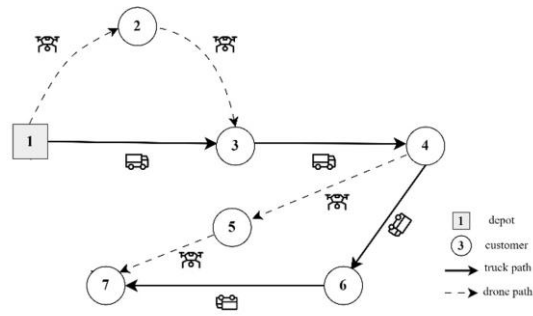
Cabe destacar que en una parte importante de la literatura sobre problemas de reparto camión–dron se asume que la ruta del vehículo terrestre es cerrada, es decir, que el recorrido comienza y finaliza en el mismo nodo, generalmente denominado *depot*. Este planteamiento resulta habitual en muchas variantes del TSP y sus extensiones, ya que representa escenarios en los que el vehículo debe regresar al punto de partida tras completar todas las entregas. Sin embargo, en el presente trabajo se adopta un enfoque distinto, considerando una

ruta abierta en la que el nodo de inicio y el nodo de finalización no necesariamente coinciden. Esta formulación permite modelar de forma más flexible ciertos escenarios logísticos en los que el punto de salida y el punto de llegada del sistema pueden ser diferentes, manteniendo al mismo tiempo la coordinación entre el camión y el dron a lo largo de toda la misión.

Bajo este marco general, el objetivo es minimizar el tiempo total de finalización de la misión (makespan), garantizando la sincronización entre ambos vehículos y la viabilidad energética de las rutas del dron.

A continuación, se establecen las hipótesis bajo las cuales se formula el problema:

1. Se desprecia cualquier limitación de distancia o tiempo para el camión; puede completar su ruta sin necesidad de repostaje.
2. El dron solo puede reunirse con el camión en nodos cliente. En estos nodos se lleva a cabo el reemplazo de la batería, cuyo tiempo se considera despreciable. Si uno de los dos vehículos llega antes, debe esperar al segundo.
3. Se asume un número ilimitado de baterías disponibles. Cada vez que el dron se sincroniza con el camión, comienza su siguiente trayecto con la batería al máximo nivel de carga.
4. El dron puede visitar múltiples clientes en cada uno de sus viajes, siempre que la energía restante sea suficiente para completar la secuencia de visitas y alcanzar el siguiente punto de encuentro con el camión.
5. El dron puede viajar sobre el camión durante cualquier parte de la ruta del camión. En ese caso, no consume batería.
6. Las distancias entre nodos se modelan mediante la distancia euclidiana, y tanto el camión como el dron se desplazan a velocidad constante.
7. Se asume un consumo lineal de energía para el dron en función de la distancia recorrida. No se consideran efectos no lineales asociados al despegue, aterrizaje o variación de carga útil.
8. Cada cliente debe ser visitado exactamente una vez, ya sea por el camión o por el dron.



**Figura 2: Ejemplo ruta camión + dron**

**Fuente:** (Taihelong, 2025)

En la **Figura 2** se muestra un ejemplo ilustrativo de una ruta coordinada entre un camión y un dron, donde ambos vehículos colaboran para atender a distintos clientes, sincronizando sus trayectorias en determinados nodos de encuentro. Podemos observar un ejemplo de ruta abierta.

## 4 PLANTEAMIENTO DE LA HEURÍSTICA

La solución inicial se obtiene mediante una heurística constructiva en dos fases. En primer lugar, se calcula una ruta base resolviendo un problema TSP clásico considerando únicamente al camión. A partir de esta ruta, en segundo lugar, se van reasignando determinados clientes al dron siguiendo un criterio de “lejanía” y respetando siempre la autonomía máxima del dron.

Para el planteamiento vamos a hacer uso de un pseudo código y, para que no haya duda, vamos a explicarlo con texto e imágenes.

### Pseudocódigo

*Algoritmo 1: Heurística Constructiva Inicial TDTL*

**Input:**  $N = \{1, \dots, n\}$   
 $0 = \text{depot\_origen}$   
 $n+1 = \text{depot\_destino}$   
 $d(i,j)$   
 $E$

**Output:** RT  
 $RD = \{\text{trip1}, \text{trip2}, \dots\}$

```

1  $\sigma \leftarrow \text{SolveTSP}(N \cup \{0, n+1\})$ 
2  $RT \leftarrow \sigma$ 
3  $RD \leftarrow \emptyset$ 

4 for  $i \in N$  do
5    $X[i] \leftarrow d(0,i) + d(i,n+1)$ 
6 end for

7  $L \leftarrow \text{sort\_descending\_by\_value}(N, X)$ 
8  $\text{studied}[i] \leftarrow \text{false}, \forall i \in N$ 

9 for  $M$  in  $L$  do
10   if  $\text{studied}[M] = \text{true}$  then
11     continue
12   end if
13
14    $A \leftarrow \text{predecessor\_of}(M, RT)$ 
15    $C \leftarrow \text{successor\_of}(M, RT)$ 
16
17    $\text{trip\_candidate} \leftarrow \text{BuildBetterDroneTrip}(A, M, C, RD, RT)$ 
18
19   if  $\text{trip\_candidate}$  is feasible then
20      $RD \leftarrow \text{UpdateDroneTrips}(RD, \text{trip\_candidate})$ 
21      $RT \leftarrow \text{UpdateTruckRoute}(RT, \text{trip\_candidate})$ 

```

```

22 | end if
23 |
24 | studied[M] ← true
25 end for

26 return (RT, RD)

```

Para ayudar a visualizar la heurística, vamos a explicarla con texto e imágenes:

### Fase 1: ruta base del camión

1. Se resuelve un TSP sobre todos los clientes, suponiendo que solo existe el camión.
2. Se obtiene una secuencia:

$$R = [O, n_1, n_2, \dots, n_k, F]$$

donde  $O$  es el nodo origen y  $F$  el nodo final.

### Fase 2: Identificación de los nodos más adecuados para el servicio mediante dron

1. Para cada nodo cliente  $i$  exceptuando el último se calcula:

$$X_i = d(O, i) + d(i, F)$$

Donde  $d(\cdot, \cdot)$  es la distancia entre nodos. Este valor mide la distancia euclídea de  $i$  respecto al origen y al final; los nodos con mayor  $X_i$  son candidatos prioritarios a ser servidos por el dron.

2. Se ordenan los nodos en orden descendente de  $X_i$ , formando la lista  $L$ .  $M$  es el primer **nodo no estudiado** de  $L$ . Un nodo  $X_i$  se considerará estudiado si y solo si se ha intentado asignar a dron.

### Fase 3: Identificación de los nodos más adecuados para el servicio mediante dron

Se recorre esta lista y, para cada nodo  $M$  se procede como sigue:

Sea  $A$  el nodo anterior a  $B$  (eventualmente  $M$ ) en la ruta original del TSP y  $C$  el posterior, la secuencia actual es:

$$\dots \rightarrow A \rightarrow B \rightarrow C \rightarrow \dots$$

**Caso general:** ni  $A$  ni  $C$  forman parte actualmente de una ruta del dron.

Se comprueba si el dron puede realizar el trip:

$$A \rightarrow B \rightarrow C$$

cumpliendo la restricción de autonomía:

$$d(A, B) + d(B, C) \leq Q$$

donde  $Q$  es la autonomía máxima del dron. Si se cumple, el nodo  $B$  pasa a ser atendido por el dron, el camión deja de visitarlo y el trip  $A-B-C$  se añade como ruta del dron. El arco  $A-C$  se asigna como ruta del camión. El nodo  $M$  se considerará estudiado.

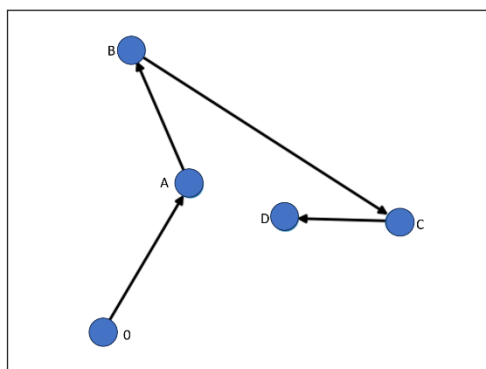


Figura 3: Ejemplo de ruta TSP

Fuente: *Elaboración propia*

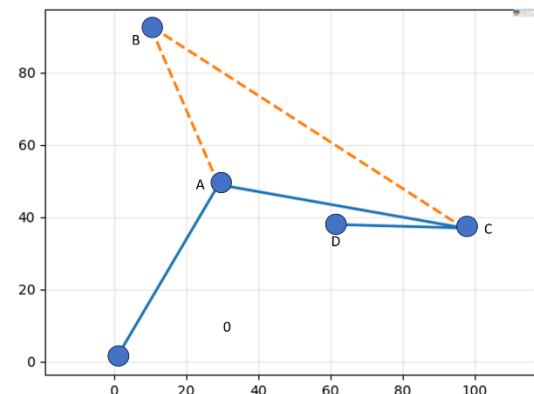


Figura 4: Primera asignación a dron

Fuente: *Elaboración propia*

**Caso especial I:** el nodo anterior a  $M$  en el TSP ya pertenece a una ruta del dron. En este caso se intenta extender esa ruta para incluir también  $M$ . Por ejemplo, si el dron ya recorre:

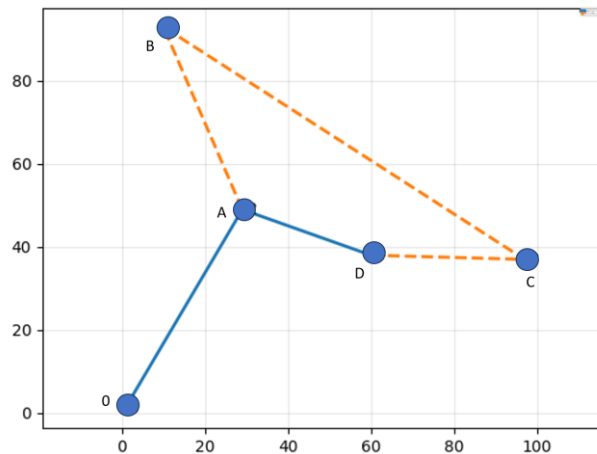
$$\text{Trip}_D = A \rightarrow B \rightarrow C$$

Y el camión realiza el recorrido:

$$\text{Trip}_T = 0 \rightarrow A \rightarrow C \rightarrow D$$

Se **estudiará** el siguiente nodo de la lista  $L$ , eventualmente  $C$ . Se comprueba si la ruta extendida incluyendo el nuevo nodo (en el ejemplo,  $A \rightarrow B \rightarrow C \rightarrow D$ ) sigue siendo factible respecto a la autonomía, siendo  $D$  el nuevo nodo de reunión de dron y camión.

Si la nueva ruta es factible, se actualiza el trip del dron y el nodo añadido deja de ser visitados por el camión.



**Figura 5: Ejemplo unión de rutas**

Fuente: *Elaboración propia*

**Caso especial II:** el nodo posterior  $B$  ya pertenece a una ruta del dron. En este caso, del mismo modo, se intenta extender esa ruta para incluir también el nodo  $A$ , eventualmente  $M$ . Por ejemplo, si el dron ya recorre:

$$\text{Trip}_D = A \rightarrow B \rightarrow C$$

Y el camión realiza el recorrido:

$$\text{Trip}_T = 0 \rightarrow A \rightarrow C$$

Se estudiará el siguiente nodo en la lista  $L$ , eventualmente  $A$ . Se comprueba si la ruta extendida  $0 \rightarrow A \rightarrow B \rightarrow C$  es factible respecto a la autonomía, siendo  $C$  el nuevo nodo de reunión camión-dron.

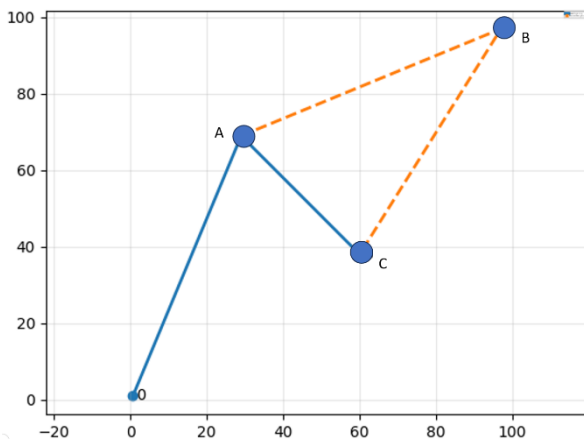


Figura 6: Primera asignación a dron

Fuente: *Elaboración propia*

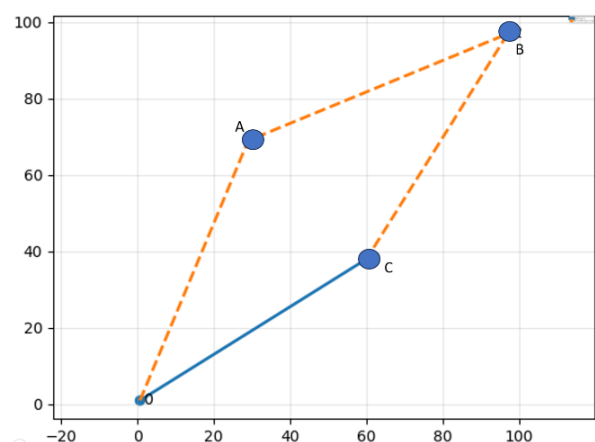


Figura 7: Segunda asignación a dron

Fuente: *Elaboración propia*

**Caso especial III:** Tanto el nodo anterior  $A$  como el nodo posterior  $C$  están asignados a distintas rutas de dron. De esta forma, el dron realiza los viajes:

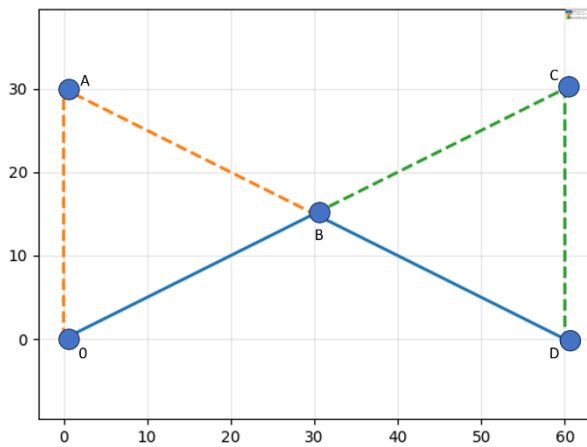
$$\text{Trip}_{D1} = 0 \rightarrow A \rightarrow B$$

$$\text{Trip}_{D_2} = B \rightarrow C \rightarrow D$$

Realizando el camión el siguiente trayecto:

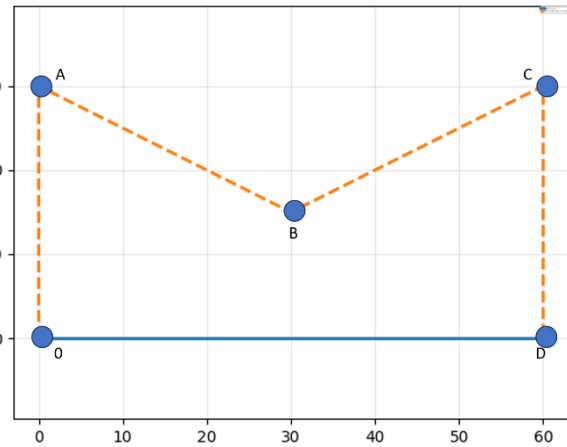
$$\text{Trip}_T = 0 \rightarrow B \rightarrow D$$

En este caso, nuevamente, intentaremos asignar el nodo  $B$ , eventualmente  $M$ , unificando las dos rutas de dron, si la batería lo permite, de la siguiente forma:



**Figura 7: Dos rutas de dron separadas**

Fuente: *Elaboración propia*



**Figura 8: Asignación de nodo intermedio**

Fuente: *Elaboración propia*

La heurística constructiva propuesta presenta una estructura computacional eficiente, adecuada para la generación de soluciones iniciales de calidad en instancias de tamaño medio y grande. El procedimiento se compone de cuatro etapas principales: (i) obtención de una ruta base mediante la resolución del TSP considerando únicamente al camión, (ii) cálculo de un indicador de relevancia para cada nodo, (iii) ordenación de los nodos según dicho indicador y (iv) reasignación iterativa de nodos al dron mediante comprobaciones de factibilidad energética.

En primer lugar, la ruta base del camión se obtiene aplicando una heurística clásica para el TSP, como 2-opt, cuyo coste computacional práctico es  $O(n^2)$ . A continuación, el cálculo del valor  $X_i$  para cada nodo —empleado como criterio de priorización para la reasignación— requiere únicamente una operación

constante por nodo, por lo que esta etapa presenta complejidad  $O(n)$ . Posteriormente, la ordenación de los nodos según sus valores  $X_i$  tiene un coste  $O(n \log n)$ .

La fase principal del algoritmo consiste en intentar la reasignación de cada nodo al dron según el orden establecido. En el caso general, la verificación de si el dron puede realizar el recorrido  $A \rightarrow M \rightarrow B$  se efectúa en tiempo constante, pues solo implica comparar la distancia total del trip con la autonomía disponible. En el caso especial, en el que uno de los nodos adyacentes ya pertenece a una ruta del dron, es necesario evaluar la posible ampliación de dicha ruta. En el peor caso, esta extensión requiere recorrer los nodos de la ruta del dron, operación que es acotada por  $O(n)$ . Como este proceso se repite para cada uno de los  $n$  nodos, la complejidad total de la fase de reasignación es  $O(n^2)$ .

Sumando los costes computacionales de todas las etapas y considerando que tanto  $O(n)$  como  $O(n \log n)$  son asintóticamente dominados por  $O(n^2)$ , la complejidad total de la heurística puede expresarse como:

$$T(n) = O(n^2)$$

Este resultado muestra que el procedimiento es **polinómico, escalable y computacionalmente ligero**, lo que permite su aplicación eficiente como generador de soluciones iniciales dentro de esquemas de optimización más avanzados, como metaheurísticas iterativas o híbridas.

## 5 EXPERIMENTACIÓN

### 5.1 COMPARACIÓN CON METAHEURÍSTICAS

Con el objetivo de evaluar el comportamiento del método propuesto y situarlo en el contexto del estado del arte, en este apartado se presenta una comparación computacional frente a dos de los enfoques más representativos y competitivos de la literatura reciente. Puesto que no existen trabajos previos que permitan comparar directamente distintas soluciones iniciales para este problema, resulta de interés analizar el rendimiento de la solución inicial propuesta frente a metaheurísticas altamente competitivas, con el fin de evaluar su calidad relativa y su potencial. Este análisis se realiza siendo conscientes de la dificultad inherente de alcanzar resultados comparables, debido a la diferencia conceptual y de complejidad entre una heurística constructiva y una metaheurística, estas últimas diseñadas específicamente para explorar de forma más intensiva el espacio de soluciones. Adicionalmente, se ha llevado a cabo una comparación en términos de tiempo de misión y tiempo de cómputo frente al problema clásico del *TSP*, con el objetivo de analizar tanto la eficiencia computacional del método como los beneficios operativos derivados de la incorporación de drones en el sistema de distribución. Para garantizar una evaluación homogénea y objetiva, todos los métodos han sido contrastados sobre una batería de instancias de referencia de tipos *uniform*, *single-center* y *double-center* que podemos encontrar en GitHub (<https://github.com/schasarah/FSTSP-MD-Heuristic>).

Estas instancias son propuestas por los autores en el trabajo *Traveling Salesman Problem with Drone* (Agatz et al., 2018) con el objetivo de realizar pruebas computacionales que permitan evaluar el comportamiento de distintos métodos de resolución bajo diferentes configuraciones espaciales de los clientes. En particular, se definen varios tipos de instancias artificiales que representan distintos patrones de distribución geográfica, lo que permite analizar la robustez y el rendimiento de los algoritmos en escenarios variados.

En todos los casos, las ubicaciones de los clientes se generan en un plano bidimensional, asumiendo que el tiempo de viaje del camión es proporcional a la distancia euclídea entre puntos.

En las instancias de tipo *uniform*, las coordenadas de cada cliente se generan de forma independiente siguiendo una distribución uniforme discreta en el intervalo  $[0, 100]$  para ambos ejes. Este tipo de instancia da lugar a una distribución homogénea de los puntos en el espacio, sin ningún tipo de agrupación o estructura definida. Por tanto, representa un escenario base en el que los clientes están dispersos de manera completamente aleatoria.

Por otro lado, las instancias *single-center* introducen una mayor estructura espacial, simulando entornos en los que existe un único centro de actividad, como podría ser el centro de una ciudad. En este caso, las ubicaciones se generan en coordenadas polares, donde el ángulo se selecciona de forma uniforme en el intervalo  $[0, 2\pi]$  y la distancia al centro sigue una distribución normal con media 0 y desviación típica 50. Posteriormente, estas coordenadas se transforman a coordenadas cartesianas. Este procedimiento provoca que exista una mayor concentración de puntos en torno al origen, generando una distribución radial más realista que la uniforme.

Finalmente, las instancias *double-center* extienden el caso anterior con el objetivo de representar entornos con dos núcleos de población o actividad. Para ello, las ubicaciones se generan inicialmente como en el caso *single-center*, pero posteriormente cada punto se traslada en el eje  $x$  una distancia de 200 unidades con probabilidad 0.5. Como resultado, se obtienen dos agrupaciones de clientes, una centrada en el origen y otra desplazada, lo que permite modelar escenarios más complejos con múltiples centros urbanos.

Las instancias consideradas abarcan distintos tamaños del problema, desde casos de dimensión moderada hasta instancias de gran escala, lo que permite analizar tanto la calidad de las soluciones obtenidas como la escalabilidad y eficiencia computacional de cada enfoque. En particular, la comparación se centra en el tiempo total de misión como medida principal de desempeño, así como en los tiempos de cómputo requeridos para la obtención de las soluciones.

Para comparar los resultados de las siguientes instancias se ha utilizado una velocidad de dron igual al doble de la velocidad del camión. La autonomía de la batería del dron se fija mediante el parámetro  $Q$ , definido como:

$$Q = 2 \times \frac{\sum Td}{|Td|}$$

Es decir, la batería se establece como el doble del tiempo medio de vuelo del dron entre pares de nodos. De esta forma, el valor de (Q) representa aproximadamente la capacidad necesaria para que el dron pueda realizar un trayecto medio de ida y vuelta entre clientes, proporcionando una autonomía representativa del tamaño medio de las distancias presentes en la instancia analizada.

Antes de comenzar a comparar resultados vamos a visualizar un ejemplo del formato en el que vamos a conseguir la solución para una instancia sencilla como *uniform-1-n5*.

Siendo:

- u = nodo de lanzamiento de dron en caso de haberlo.
- v = nodo de recogida de dron en caso de haberla.
- S = nodos visitados exclusivamente por el dron.

El camión realizará la ruta (u → v) en cada intervalo.

El dron realizará la ruta (u → [S] → v) en caso de que S no esté vacío. En el otro caso, el dron irá encima del camión.

*Ejemplo:*

Makespan total final: 233.0011

Intervalos (u, v, S): [(0, 1, [2]), (1, 3, []), (3, 4, [])]

### SOLUCIÓN ESTRUCTURADA:

Intervalo 0: 0 -> 1

Tiempo intervalo: [0.0000, 92.5226]

Camión: ruta = [0, 1] tiempo=92.5226

Dron: ruta = [0, 2, 1] tiempo=51.8588

Intervalo 1: 1 -> 3

Tiempo intervalo: [92.5226, 195.9876]

Camión: ruta = [1, 3] tiempo=103.4650

Dron: no se lanza en este intervalo

Intervalo 2: 3 -> 4

Tiempo intervalo: [195.9876, 233.0011]

Camión: ruta = [3, 4] tiempo=37.0135

Dron: no se lanza en este intervalo

Una vez visto el formato de las soluciones, vamos a comparar los resultados de esta heurística constructiva con los obtenidos por (Schaumann et al., 2025) y (González-R et al., 2020) en las instancias ya mencionadas.

Para el cálculo del *Relative Percentage Deviation* (RPD) se ha utilizado la siguiente expresión:

$$RPD(\%) = \frac{Z_{VFR} - Z_{best}}{Z_{best}} \times 100$$

donde  $Z_{VFR}$  representa el valor de la función objetivo obtenido por el método propuesto en este trabajo, y  $Z_{best}$  corresponde al mejor valor de la función objetivo alcanzado por las metaheurísticas consideradas en la comparación. Este indicador permite medir, en términos porcentuales, la desviación relativa de la solución propuesta respecto a la mejor solución conocida, facilitando así la evaluación de su calidad.

## INSTANCIAS UNIFORM

En primer lugar, vamos a probar la heurística en las instancias tipo *uniform*. Para estas instancias, las metaheurísticas comparadas han estado ejecutándose durante 600 segundos por cada instancia. Marcaremos en **negrita** las soluciones en las que se supera a alguna de las dos metaheurísticas:

n	ID INSTANCIA	MD-SPP-H	IGH	VFR	Tiempo (s)	RPD	$\overline{RPD}$
50	71-n50	282.7	325.3	<b>315.15</b>	0.0176	11.48	19.88
	72-n50	302.5	316	329.51	0.024	8.93	
	73-n50	294.2	322.5	338.6	0.044	15.09	
	74-n50	289	291.7	388	0.02	34.26	
	75-n50	298	320.7	348.91	0.026	17.08	
	76-n50	285	306.6	345.6	0.023	21.26	
	77-n50	301.7	317.2	384	0.029	27.28	
	78-n50	278.4	291.3	321	0.027	15.30	
	79-n50	273.8	311.7	351.06	0.0286	28.22	
	80-n50	271.8	286.5	326	0.033	19.94	
75	81-n75	321.7	368.7	422	0.045	31.18	21.73
	82-n75	303.6	321.8	386	0.04	27.14	
	83-n75	296.4	325.6	357.08	0.061	20.47	
	84-n75	321.8	388.4	391.26	0.066	21.58	
	85-n75	344.9	395.7	412.15	0.0558	19.50	
	86-n75	326	371.2	379.8	0.066	16.50	
	87-n75	328.8	276.1	363.11	0.077	10.43	
	88-n75	330.5	360.7	422	0.067	27.69	
	89-n75	298.2	348.7	378.23	0.11	26.84	
	90-n75	322.5	336.1	374	0.0521	15.97	
100	91-n100	359.4	403.8	421	0.1713	17.14	22.64
	92-n100	345.4	409.8	<b>396.6</b>	0.099	14.82	
	93-n100	339.6	394.9	427.11	0.121	25.77	
	94-n100	344.9	382	434.2	0.133	25.89	
	95-n100	352	421.9	<b>413.95</b>	0.1681	17.60	
	96-n100	353	423.5	471.6	0.1486	33.60	
	97-n100	356.8	387.4	445.4	0.16	24.83	
	98-n100	353	419.4	<b>393.9</b>	0.09	11.59	
	99-n100	350.5	424.1	446.9	0.1	27.50	
	100-n100	356.5	397.4	455	0.201	27.63	
175	101-n175	432.3	511.6	545.34	0.43	26.15	22.60
	102-n175	425.7	530.7	<b>521.99</b>	0.54	22.62	
	103-n175	445.2	499.3	516.62	0.506	16.04	
	104-n175	436.2	500.4	538.3	0.49	23.41	
	105-n175	441.4	523.8	549.35	0.3022	24.46	
	106-n175	446.8	480.7	536.2	0.664	20.01	
	107-n175	447.5	519.8	524	0.664	17.09	

	108-n175	458.5	534.7	540	0.51	17.78	
	109-n175	435.2	606.7	566	0.6371	30.06	
	110-n175	418.4	508.6	537.4	0.44	28.44	
250	111-n250	494.9	607	629	1.543	27.10	28.46
	112-n250	494.4	600.3	623.9	0.9541	26.19	
	113-n250	494.9	603.3	663.8	1.12	34.13	
	114-n250	502.5	596.8	650.47	0.9952	29.45	
	115-n250	501.2	603.5	650	1.1	29.69	
	116-n250	494.5	589.8	634.4	1.16	28.29	
	117-n250	518.3	640.8	676.1	1.18	30.45	
	118-n250	484.4	584.1	599.8	1.12	23.82	
	119-n250	527.1	621.7	682.29	1.216	29.44	
	120-n250	493.5	611.6	622.04	1.06	26.05	
		9-n500	-	-	913.45	10.08	
	Tiempo medio de ejecución (s)	600	600	0.36481		23.06339	

Tabla 2

Fuente: *Elaboración propia*

La **Tabla 2** recoge los resultados obtenidos al resolver distintas instancias del problema mediante tres enfoques diferentes: dos métodos heurísticos de mayor complejidad (MD-SPP-H e IGH) y el método propuesto en este trabajo (VFR). Como puede observarse, el método VFR presenta, en general, valores de la función objetivo superiores a los obtenidos por los otros enfoques, lo que indica soluciones de menor calidad en términos estrictamente óptimos.

No obstante, un análisis conjunto de los resultados y de los tiempos medios de ejecución pone de manifiesto una diferencia clave entre los métodos. Mientras que MD-SPP-H e IGH requieren tiempos de ejecución del orden de 600 segundos, el método propuesto obtiene sus soluciones en un tiempo medio de 0.1408 segundos, lo que supone una reducción de varios órdenes de magnitud.

Estos resultados evidencian que el enfoque desarrollado en este trabajo prioriza la eficiencia computacional, ofreciendo soluciones iniciales de buena calidad en tiempos muy reducidos. Aunque dichas soluciones no alcanzan el nivel de optimización de métodos más complejos, sí constituyen una base sólida y coherente sobre la que aplicar técnicas de mejora posteriores.

Un análisis más detallado de la **Tabla 2** a través del indicador RPD (*Relative Percentage Deviation*) permite contextualizar mejor el rendimiento del método propuesto. Aunque la heurística desarrollada presenta desviaciones positivas respecto a los mejores métodos conocidos, los valores de RPD obtenidos se mantienen en rangos moderados, incluso para instancias de gran tamaño. Esto indica que, si bien las soluciones no alcanzan el óptimo proporcionado por metaheurísticas avanzadas, se mantienen relativamente próximas en términos relativos.

En este sentido, el método propuesto presenta un perfil claramente orientado a entornos operativos reales: soluciones cercanas al estado del arte (según el RPD) obtenidas en tiempos prácticamente instantáneos. Este compromiso calidad–tiempo es precisamente el tipo de solución que se demanda en contextos industriales, donde la planificación debe realizarse bajo restricciones temporales estrictas y con capacidad de respuesta ante cambios dinámicos.

Por tanto, lejos de interpretarse como una limitación, los valores de RPD obtenidos refuerzan la validez del enfoque desarrollado. La heurística debe complementar metaheurísticas, actuando como generador de soluciones iniciales eficientes que permitan mejorar la aplicabilidad práctica de los métodos existentes dentro de un marco de ingeniería real.

La utilización de una solución inicial de calidad generada en un tiempo tan reducido permitiría acelerar notablemente la convergencia de estos métodos, mejorando la calidad final de la solución sin incurrir en los elevados costes computacionales asociados a una inicialización más compleja.

Por tanto, los resultados obtenidos refuerzan el interés del enfoque desarrollado, ya que demuestra ser una herramienta eficaz para generar soluciones rápidas y razonablemente buenas, especialmente indicada para entornos donde el tiempo de cálculo es un factor crítico o como etapa previa dentro de esquemas híbridos heurístico-metaheurísticos.

Para ilustrar el funcionamiento de la heurística constructiva propuesta, se presentan a continuación, en la **Figuras 9**, un ejemplo en el que apreciar los recorridos de camión y dron, así como los nodos de lanzamiento (*launch nodes*) y los nodos de recuperación (*rendezvous nodes*).

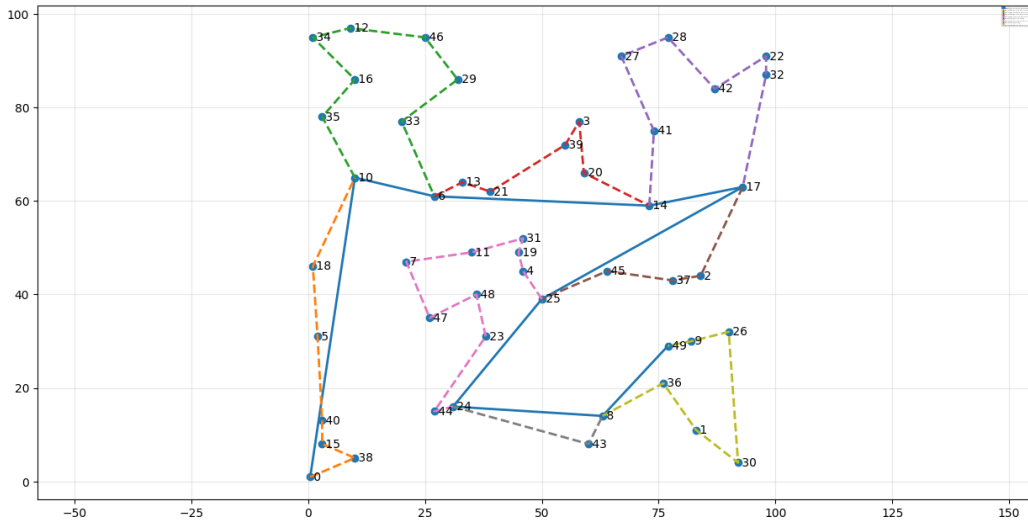


Figura 9: Uniform 77-n50

Fuente: *Elaboración propia*

INSTANCIAS SINGLE-CENTER

Seguiremos con la comparación y ahora contrastaremos los resultados en instancias tipo *single-center*:

n	ID INSTANCIA	IGH	VFR	Tiempo IGH (s)	Tiempo VFR (s)	RPD	$\overline{RPD}$
50	singlecenter-71-n50	371.41	462.87	49.05	0.018	24.63	9.16
	singlecenter-72-n50	508.78	<b>482.3</b>	51.75	0.015	-5.21	
	singlecenter-73-n50	348.40	391.5	49.19	0.059	12.37	
	singlecenter-74-n50	490.51	494.51	54.27	0.031	0.82	
	singlecenter-75-n50	654.93	765	48.27	0.036	16.81	
	singlecenter-76-n50	531.08	662.2	51.98	0.024	24.69	
	singlecenter-77-n50	580.74	<b>565.2</b>	46.03	0.037	-2.68	
	singlecenter-78-n50	544.06	574.14	52.39	0.0312	5.53	
	singlecenter-79-n50	547.33	597.1	54.45	0.033	9.09	
singlecenter-80-n50	674.09	711.8	49.94	0.022	5.59		
75	singlecenter-81-n75	754.23	838.2	180.67	0.12	11.13	9.10
	singlecenter-82-n75	577.26	656.7	157.08	0.06	13.76	
	singlecenter-83-n75	653.47	<b>632.4</b>	177.86	0.081	-3.22	
	singlecenter-84-n75	603.91	673.3	199.62	0.071	11.49	
	singlecenter-85-n75	648.03	725.6	176.39	0.064	11.97	
	singlecenter-86-n75	758.90	835	182.23	0.08	10.03	
	singlecenter-87-n75	637.05	740	166.19	0.05	16.16	
	singlecenter-88-n75	867.97	925.4	172.53	0.066	6.62	
	singlecenter-89-n75	630.04	664.9	176.12	0.051	5.53	

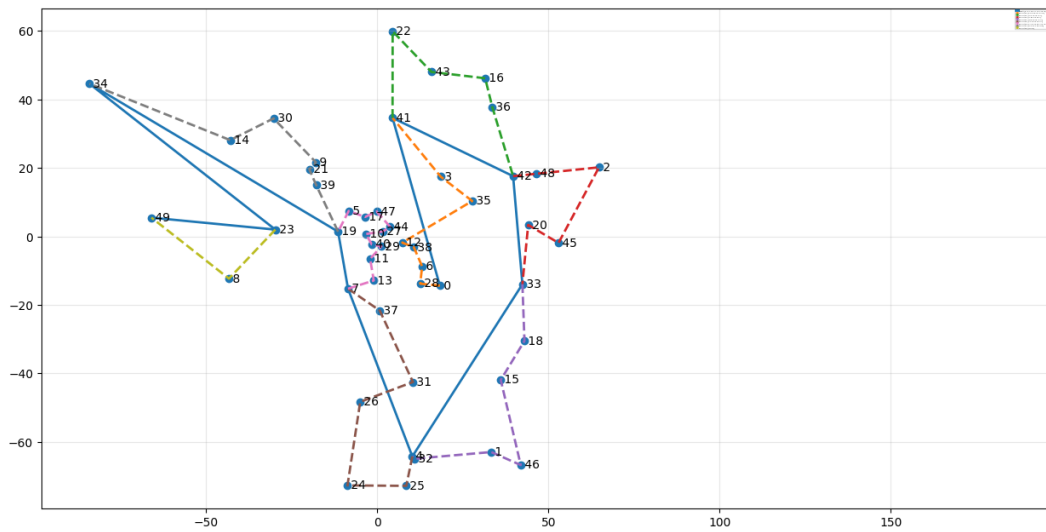
	singlecenter-90-n75	725.57	779.9	158.56	0.056	7.49	
100	singlecenter-91-n100	808.85	834	418.33	0.15	3.11	4.16
	singlecenter-92-n100	877.96	902.2	402.72	0.149	2.76	
	singlecenter-93-n100	925.36	<b>810.89</b>	414.00	0.153	-12.37	
	singlecenter-94-n100	883.31	1034.38	403.27	0.09	17.10	
	singlecenter-95-n100	739.57	809.6	403.47	0.098	9.47	
	singlecenter-96-n100	829.52	838.2	444.20	0.1	1.05	
	singlecenter-97-n100	858.51	899.1	398.27	0.13	4.73	
	singlecenter-98-n100	1064.99	1096.64	435.34	0.24	2.97	
	singlecenter-99-n100	635.74	722.4	413.05	0.12	13.63	
	singlecenter-100-n100	730.85	<b>724.4</b>	404.84	0.14	-0.88	
175	singlecenter-101-n175	999.13	1094.7	612.59	0.52	9.57	9.72
	singlecenter-102-n175	943.57	1104.3	627.45	0.48	17.03	
	singlecenter-103-n175	961.80	1112.1	603.67	0.43	15.63	
	singlecenter-104-n175	1049.49	1094.9	618.20	0.7	4.33	
	singlecenter-105-n175	1038.95	1159.1	610.06	0.61	11.56	
	singlecenter-106-n175	1164.87	1271.4	641.94	0.29	9.15	
	singlecenter-107-n175	1122.50	1175	614.59	0.66	4.68	
	singlecenter-108-n175	960.42	1024.3	631.55	0.62	6.65	
	singlecenter-109-n175	1037.13	1078.7	611.58	0.79	4.01	
	singlecenter-110-n175	1001.09	1147.2	623.17	0.41	14.60	
250	singlecenter-111-n250	1057.63	1163.1	848.37	1.19	9.97	7.51
	singlecenter-112-n250	1102.81	<b>1091</b>	689.28	1.53	-1.07	
	singlecenter-113-n250	1067.19	1204	893.30	1.16	12.82	
	singlecenter-114-n250	1197.76	1291.2	768.00	1.33	7.80	
	singlecenter-115-n250	1269.42	1421	674.92	0.97	11.94	
	singlecenter-116-n250	1101.80	1260.2	829.27	1.18	14.38	
	singlecenter-117-n250	1446.97	1490.1	734.19	1.4	2.98	
	singlecenter-118-n250	1267.52	1328.9	764.77	0.95	4.84	
	singlecenter-119-n250	1215.42	1325.7	713.00	1.29	9.07	
	singlecenter-120-n250	1124.71	1151.3	625.19	1.08	2.36	
	Tiempo medio de ejecución (s)	402.5431	0.399304			7.93	

**Tabla 3**

Fuente: *Elaboración propia*

Los resultados obtenidos para las instancias *single-center*, reflejados en la **Tabla 3**, muestran un comportamiento igualmente robusto del método propuesto. En este tipo de configuraciones espaciales, caracterizadas por una fuerte concentración de clientes alrededor de un mismo punto, la heurística mantiene una desviación relativa moderada respecto a IGH, incluso alcanzando soluciones competitivas o superiores en algunos casos concretos. Más allá de la calidad de

las soluciones, vuelve a observarse una diferencia muy significativa en los tiempos de ejecución, donde la heurística desarrollada obtiene resultados prácticamente instantáneos frente a los elevados tiempos requeridos por los métodos comparados. Este comportamiento confirma la capacidad del algoritmo para adaptarse a diferentes estructuras espaciales sin perder eficiencia computacional. A continuación, en la **Figura 10**, mostramos un ejemplo visual referido a estas instancias.



**Figura 10: Singlecenter-71-n50**

Fuente: *Elaboración propia*

INSTANCIAS DOUBLE-CENTER

Ahora vamos a obtener los resultados y comparación de las instancias *double-center* mencionadas:

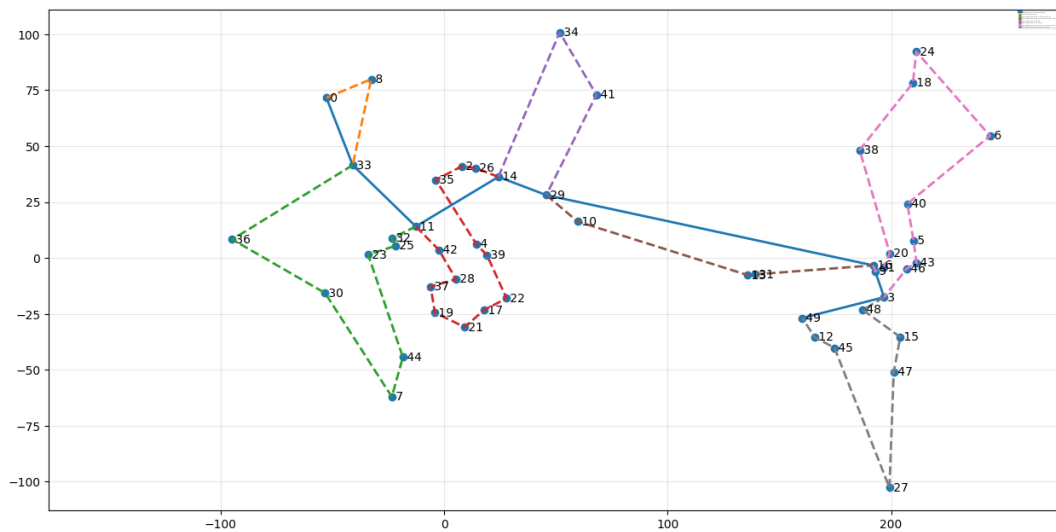
n	ID INSTANCIA	IGH	VFR	Tiempo IGH (s)	Tiempo VFR (s)	RPD	$\overline{RPD}$
50	doublecenter-71-n50	764.22	865.6	44.64	0.019	13.27	11.61
	doublecenter-72-n50	694.90	700.1	50.45	0.019	0.75	
	doublecenter-73-n50	634.86	678.3	53.11	0.0156	6.84	
	doublecenter-74-n50	530.83	694.3	49.44	0.025	30.80	
	doublecenter-75-n50	575.03	704.57	48.17	0.03	22.53	
	doublecenter-76-n50	741.76	822.2	52.83	0.02	10.84	
	doublecenter-77-n50	641.29	646	51.62	0.021	0.73	
	doublecenter-78-n50	631.36	688.78	48.31	0.016	9.10	
doublecenter-79-n50	659.33	743.1	54.30	0.0167	12.71		

	doublecenter-80-n50	719.35	780.7	49.89	0.018	8.53	
75	doublecenter-81-n75	659.14	692	171.39	0.078	4.99	11.03
	doublecenter-82-n75	773.80	848.2	174.58	0.054	9.62	
	doublecenter-83-n75	689.61	845.9	174.22	0.04	22.66	
	doublecenter-84-n75	923.25	1010.2	171.25	0.072	9.42	
	doublecenter-85-n75	1002.96	1027	177.95	0.058	2.40	
	doublecenter-86-n75	685.47	769.8	182.37	0.05	12.30	
	doublecenter-87-n75	717.12	779.4	177.77	0.04	8.68	
	doublecenter-88-n75	896.37	938.4	175.89	0.05	4.69	
	doublecenter-89-n75	792.01	1031.8	183.89	0.05	30.28	
	doublecenter-90-n75	823.12	866.7	175.45	0.04	5.29	
100	doublecenter-91-n100	881.74	<b>880.2</b>	406.25	0.07	-0.17	5.16
	doublecenter-92-n100	818.97	872.3	396.31	0.1	6.51	
	doublecenter-93-n100	775.35	840.7	395.36	0.11	8.43	
	doublecenter-94-n100	891.90	1000.7	381.52	0.2	12.20	
	doublecenter-95-n100	964.18	1002.9	408.75	0.16	4.02	
	doublecenter-96-n100	899.67	950.4	420.84	0.09	5.64	
	doublecenter-97-n100	1097.93	1158	405.59	0.13	5.47	
	doublecenter-98-n100	945.01	1006.1	430.36	0.09	6.47	
	doublecenter-99-n100	874.77	924.36	411.02	0.19	5.67	
	doublecenter-100-n100	993.44	<b>967.7</b>	395.62	0.13	-2.59	
175	doublecenter-101-n175	1151.35	<b>1091</b>	629.25	0.57	-5.24	6.03
	doublecenter-102-n175	1224.98	1270.7	638.36	0.85	3.73	
	doublecenter-103-n175	1213.88	1430.1	615.80	0.7	17.81	
	doublecenter-104-n175	1222.34	1350.4	640.80	0.5	10.48	
	doublecenter-105-n175	1323.37	1387.8	631.70	0.34	4.87	
	doublecenter-106-n175	1213.38	1297.1	606.11	0.72	6.90	
	doublecenter-107-n175	1109.58	1161.8	628.59	0.67	4.71	
	doublecenter-108-n175	1291.88	1373	601.36	0.82	6.28	
	doublecenter-109-n175	1209.71	1240.1	632.70	0.58	2.51	
	doublecenter-110-n175	1256.01	1359.6	634.36	0.53	8.25	
250	doublecenter-111-n250	1532.53	1551.5	818.14	1.2	1.24	5.83
	doublecenter-112-n250	1450.97	1492.2	717.44	1.34	2.84	
	doublecenter-113-n250	1310.39	1546.2	672.25	1.22	18.00	
	doublecenter-114-n250	1347.42	1421.2	820.56	0.65	5.48	
	doublecenter-115-n250	1479.77	1589.6	737.72	2.06	7.42	
	doublecenter-116-n250	1412.86	1528.9	1159.80	1.42	8.21	
	doublecenter-117-n250	1489.27	1532.85	815.56	0.88	2.93	
	doublecenter-118-n250	1467.79	1609.8	728.86	1.59	9.68	
	doublecenter-119-n250	1538.89	1594.14	856.02	1.31	3.59	
	doublecenter-120-n250	1442.11	<b>1426.7</b>	804.34	1.36	-1.07	
	Tiempo medio de ejecución (s)	414.1774971	0.426246			7.93	

**Tabla 4**

Fuente: *Elaboración propia*

En el caso de las instancias double-center, donde los clientes se agrupan en dos regiones diferenciadas del espacio, el problema presenta una mayor complejidad estructural debido a la aparición de trayectos más largos entre clusters. A pesar de ello, la heurística propuesta mantiene, como se muestra en la **Tabla 4**, un comportamiento estable y consistente, generando soluciones razonables en todos los casos analizados. Si bien las desviaciones respecto a IGH pueden aumentar ligeramente en determinadas instancias, los resultados continúan situándose en rangos aceptables considerando el carácter constructivo y no iterativo del método. Una vez más, el aspecto más relevante es la rapidez del procedimiento, que permite obtener soluciones en tiempos muy reducidos incluso en instancias de gran tamaño, lo que pone de manifiesto la escalabilidad del enfoque desarrollado. En la **Figura 11** podemos ver un ejemplo de ruta camión-dron en instancias *double-center*.



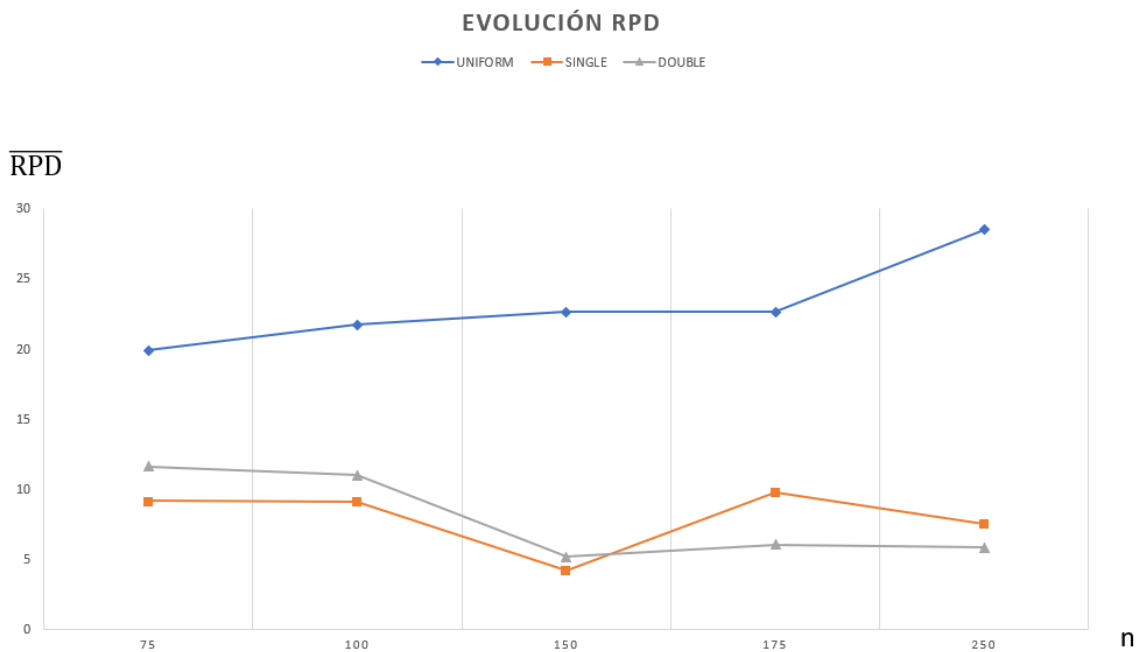
**Figura 11:** doublecenter-75-n50

Fuente: *Elaboración propia*

## ANÁLISIS DE EVOLUCIÓN DEL RELATIVE PERCENTAGE DEVIATION

En la **Figura 12**, situada al final del apartado, se muestra la evolución del RPD en función del tamaño de las instancias para los tres tipos considerados (*uniform*, *single-center* y *double-center*). En el caso de las instancias *uniform*, se observa un ligero incremento del RPD a medida que aumenta el tamaño del problema, lo que indica que la calidad relativa de la solución inicial tiende a empeorar de forma progresiva en este tipo de distribuciones.

Por el contrario, en las instancias *single-center* y *double-center*, el RPD se mantiene moderadamente estable a lo largo de los distintos tamaños analizados, con pequeñas oscilaciones, pero sin una tendencia claramente creciente. Esto sugiere que la heurística propuesta es capaz de mantener un comportamiento consistente cuando existe cierta estructura en la distribución de los nodos, independientemente del tamaño de la instancia.



**Figura 12:** Evolución RPD

**Fuente:** Elaboración propia

## 5.2 COMPARACIÓN CON PROBLEMA TSP CLÁSICO

Comparando le heurística objeto de estudio con el problema TSP clásico podemos observar los siguientes resultados. La fórmula del RPD en este caso será:

$$RPD(\%) = \frac{Z_{TSP} - Z_{VFR}}{Z_{VFR}} \times 100$$

De esta forma, un RPD igual al 100% indicará el doble de eficiencia en el resultado.

n	ID INSTANCIA	TSP	VFR	Tiempo TSP (s)	Tiempo VFR (s)	RPD
50	71-n50	569.75	315.15	0.0122	0.0176	80.79
	72-n50	615.12	329.51	0.0199	0.024	86.68
	73-n50	590.13	338.6	0.0418	0.044	74.29
	74-n50	615.59	388	0.02	0.02	58.66
	75-n50	613.08	348.91	0.024	0.026	75.71
	76-n50	611.97	345.6	0.022	0.023	77.07
	77-n50	618.61	384	0.017	0.029	61.10
	78-n50	577.75	321	0.025	0.027	79.98
	79-n50	573.92	351.06	0.0265	0.0286	63.48
	80-n50	564	326	0.03	0.033	73.01
75	81-n75	715.97	422	0.042	0.045	69.66
	82-n75	691.11	386	0.0297	0.04	79.04
	83-n75	670.66	357.08	0.056	0.061	87.82
	84-n75	724.97	391.26	0.0603	0.066	85.29
	85-n75	731.02	412.15	0.033	0.0558	77.37
	86-n75	692.86	379.8	0.0347	0.066	82.43
	87-n75	689.29	363.11	0.0702	0.077	89.83
	88-n75	731.22	422	0.049	0.067	73.27
	89-n75	648.7	378.23	0.09	0.11	71.51
	90-n75	689.47	374	0.051	0.0521	84.35
100	91-n100	814.36	421	0.1656	0.1713	93.43
	92-n100	763.03	396.6	0.066	0.099	92.39
	93-n100	770.71	427.11	0.1167	0.121	80.45
	94-n100	780.2	434.2	0.098	0.133	79.69
	95-n100	822.26	413.95	0.1578	0.1681	98.64
	96-n100	826.3	471.6	0.12	0.1486	75.21
	97-n100	803.15	445.4	0.12	0.16	80.32
	98-n100	775.61	393.9	0.07	0.09	96.91
	99-n100	837.4	446.9	0.079	0.1	87.38
	100-n100	820.12	455	0.19	0.201	80.25

175	101-n175	1044.23	545.34	0.43	0.43	91.48
	102-n175	1033.72	521.99	0.53	0.54	98.03
	103-n175	989.31	516.62	0.41	0.506	91.50
	104-n175	1039.47	538.3	0.47	0.49	93.10
	105-n175	1084.66	549.35	0.16	0.3022	97.44
	106-n175	1025.57	536.2	0.52	0.664	91.27
	107-n175	1031.03	524	0.62	0.664	96.76
	108-n175	1043.31	540	0.41	0.51	93.21
	109-n175	1035.17	566	0.599	0.6371	82.89
	110-n175	1036.53	537.4	0.377	0.44	92.88
250	111-n250	1209.37	629	1.38	1.543	92.27
	112-n250	1232.7	623.9	0.839	0.9541	97.58
	113-n250	1222.47	663.8	0.96	1.12	84.16
	114-n250	1236.86	650.47	0.916	0.9952	90.15
	115-n250	1255.04	650	0.88	1.1	93.08
	116-n250	1219.1	634.4	1.033	1.16	92.17
	117-n250	1255.8	676.1	1.04	1.18	85.74
	118-n250	1181.2	599.8	1.01	1.12	96.93
	119-n250	1294.7	682.29	1.034	1.216	89.76
	120-n250	1241.5	622.04	0.75	1.06	99.59
	Tiempo medio de misión	873.2014	468.9224			84.92
	Tiempo medio de ejecución (s)	0.326108	0.378714			

**Tabla 5**

**Fuente:** *Elaboración propia*

La comparación presentada en la **Tabla 5** permite analizar el impacto del uso del dron frente a un escenario en el que únicamente se emplea el camión para realizar las entregas. Como puede observarse, la solución generada por la heurística híbrida produce reducciones significativas en el tiempo total de misión en prácticamente todas las instancias analizadas. Este resultado confirma el potencial de los sistemas cooperativos camión–dron para mejorar la eficiencia del reparto de última milla. Además, en la comparación de tiempos, podemos observar que la heurística propuesta es capaz de capturar este beneficio mediante un procedimiento computacional extremadamente ligero, lo que refuerza su utilidad como herramienta práctica para generar rápidamente soluciones iniciales de alta calidad en problemas de gran escala.

## 6 CONCLUSIÓN

En este Trabajo de Fin de Grado se ha abordado el problema de reparto de última milla mediante un sistema híbrido camión–dron, un ámbito de investigación de creciente interés debido a su elevada complejidad operativa y a su potencial impacto en la eficiencia de los sistemas logísticos modernos. En particular, el estudio se ha centrado en el desarrollo de una heurística constructiva inicial para el problema Truck–Drone Team Logistics (TDTL), orientada a la generación rápida de soluciones factibles y de calidad para instancias de tamaño medio y grande.

A partir de una revisión detallada del estado del arte, se ha puesto de manifiesto que, si bien existen métodos exactos y metaheurísticos capaces de alcanzar soluciones de alta calidad, su aplicabilidad práctica se ve limitada por elevados tiempos de cómputo o por una fuerte dependencia de la solución inicial. Esta observación ha servido como principal motivación para el diseño de la heurística propuesta, cuyo objetivo no es competir directamente con métodos de optimización intensiva, sino proporcionar un punto de partida eficiente y estructuralmente informado para procesos de mejora posteriores.

La heurística desarrollada combina una ruta base obtenida mediante un TSP clásico con un procedimiento iterativo de reasignación de clientes al dron basado en criterios geométricos y restricciones de autonomía. Este enfoque permite capturar de forma sencilla el impacto temporal de cada cliente sobre el tiempo total de misión, manteniendo al mismo tiempo una complejidad computacional polinómica y un comportamiento altamente escalable. Los resultados obtenidos confirman que el método es capaz de generar soluciones coherentes en tiempos de ejecución muy reducidos, incluso para instancias de gran tamaño donde otros enfoques no resultan viables.

El análisis comparativo realizado muestra que, aunque las soluciones generadas por la heurística propuesta presentan valores de la función objetivo superiores a los alcanzados por metaheurísticas avanzadas, la drástica reducción en los tiempos de cómputo justifica plenamente su uso como solución inicial. En este

sentido, el método desarrollado cumple con el objetivo principal del trabajo: ofrecer una herramienta rápida, robusta y fácilmente integrable dentro de esquemas híbridos heurístico–metaheurísticos, mejorando la eficiencia global del proceso de optimización.

Como principal limitación del estudio, cabe señalar que la heurística no incorpora mecanismos de mejora local ni procedimientos de refinamiento posteriores, lo que restringe la calidad final de las soluciones obtenidas. No obstante, esta característica responde deliberadamente al planteamiento del trabajo y abre interesantes líneas de investigación futura. Entre ellas destacan la integración del método propuesto como fase inicial de metaheurísticas existentes — resultando especialmente interesante su combinación con la metaheurística IGH empleada en la comparación de resultados, iniciando dicho algoritmo a partir de la solución generada en este TFG—, la extensión a escenarios con múltiples drones o restricciones más realistas, así como la adaptación del criterio de priorización de clientes para considerar factores adicionales como ventanas temporales o variaciones en la velocidad de los vehículos.

En conclusión, este Trabajo de Fin de Grado contribuye al estudio del problema TDTL mediante el diseño de una heurística constructiva inicial eficiente y escalable, demostrando que es posible obtener soluciones de buena calidad en tiempos de cómputo muy reducidos. Los resultados alcanzados refuerzan la importancia de las soluciones iniciales en problemas de reparto híbrido camión–dron y sientan una base sólida para el desarrollo de métodos de optimización más avanzados en aplicaciones logísticas reales.

## 7 REFERENCES

- 36KrEnglish, 2024. *Meituan's drone delivery debuts in Dubai, eyeing a future in last-mile logistics*. [Online] Available at: <https://kr-asia.com/meituans-drone-delivery-debuts-in-dubai-eyeing-a-future-in-last-mile-logistics>.
- Agatz, N., Bouman, P. & Schmidt, M., 2018. Optimization Approaches for the Traveling Salesman. *Transportation Science*, p.18.
- Dorling, K., Heinrichs, J., Messier, G.G. & Magierowski, S., 2017. Vehicle Routing Problems for Drone Delivery. *IEEE*, p.16.
- González-R, P.L. et al., 2020. Truck-drone team logistics: A heuristic approach to multi-drop. *ELSEVIER*, p.24.
- Ha, Q.M., Deville, Y., Dung Pham, Q. & Hoang Ha, M., 2017. On the Min-cost Traveling Salesman Problem with. p.57.
- Izquierdo, U. & Mayo, A., 2021. *BUSINESS INSIDER*. [Online] Available at: <https://www.businessinsider.es/retail/conductores-reparto-amazon-pie-guerra-objetivos-entrega-910595?>
- Khattar, M., 2023. Bringing Order to Chaos: Last-Mile Delivery Costs and How to Reduce Them. *Locus*, p.4.
- Murray, C.C. & Chu, A.G., 2015. The flying sidekick traveling salesman problem: Optimization. *ELSEVIER*, p.24.
- Poikonen, S..W.X..G.B., 2017. The vehicle routing problem with drones: extended models and connections. *Networks*, p.34.
- Schaumann, S.K. et al., 2025. The flying sidekick traveling salesman problem with multiple drops: An effective heuristic approach. *ELSEVIER*, p.19.
- Taihelong, Z.e.a., 2025. *An End-to-End Deep Reinforcement Learning Approach for Solving the Traveling Salesman Problem with Drones*. [Online] Available at: <https://chatpaper.com/es/chatpaper/paper/207238>.
- Windras Mara, S.T., Pratama Rifai, A. & Sopha Maya, B., 2022. An adaptive large neighborhood search heuristic for the flying sidekick traveling salesman problem with multiple drops. *ELSEVIER*, p.20.