



---

**Universidad de Valladolid**

FACULTAD DE CIENCIAS

TRABAJO FIN DE GRADO

GRADO EN ESTADÍSTICA

# Métodos escalables de resolución para el Dial-a-Ride Problem (DARP)

Autor:

**Sergio García Prado**





---

**Universidad de Valladolid**

FACULTAD DE CIENCIAS

TRABAJO FIN DE GRADO

GRADO EN ESTADÍSTICA

# Métodos escalables de resolución para el Dial-a-Ride Problem (DARP)

Autor:

**Sergio García Prado**

Tutor:

**Jesús Saez Aguado**



## **Abstract**

[TODO]

## **Resumen**

[TODO]

Este trabajo puede ser consultado a través del siguiente enlace:  
<https://github.com/garciparedes/tfg-pickup-and-delivery>



# Agradecimientos

[TODO]





# Prefacio

[TODO]



# Índice general

<b>Resumen</b>	<b>1</b>
<b>Agradecimientos</b>	<b>3</b>
<b>Prefacio</b>	<b>5</b>
<b>1. Introducción</b>	<b>9</b>
1.1. Introducción . . . . .	9
1.2. Objetivos . . . . .	9
1.3. Metodología . . . . .	9
1.4. Problema de recogidas y envíos . . . . .	9
1.5. Aplicaciones . . . . .	9
1.6. Conclusiones . . . . .	9
<b>2. Formulación del Problema</b>	<b>11</b>
2.1. Introducción . . . . .	11
2.2. Notación . . . . .	16
2.3. Formulación básica . . . . .	16
2.4. Restricciones Adicionales . . . . .	17
2.5. Funciones Objetivo . . . . .	17
2.6. Tiempo Real . . . . .	17
2.7. Conclusiones . . . . .	17
<b>3. Métodos de Resolución Exactos</b>	<b>19</b>
3.1. Introducción . . . . .	19
3.2. Conclusiones . . . . .	19
<b>4. Heurísticas</b>	<b>21</b>
4.1. Introducción . . . . .	21
4.2. Greedy . . . . .	21
4.3. Metropolis Hastings . . . . .	22
4.4. Conclusiones . . . . .	22

<b>5. Metaheurísticas</b>	<b>23</b>
5.1. Introducción . . . . .	23
5.2. GRASP . . . . .	23
5.3. Simulated Anneling . . . . .	23
5.4. Tabu Search . . . . .	23
5.5. Ant Colony . . . . .	23
5.6. Variable Neighborhood Search . . . . .	23
5.7. Large Neighborhood Search . . . . .	23
5.8. Conclusiones . . . . .	24
<b>6. Implementación</b>	<b>25</b>
6.1. Introducción . . . . .	25
6.2. Conclusiones . . . . .	25
<b>7. Resultados</b>	<b>27</b>
7.1. Introducción . . . . .	27
7.2. Conclusiones . . . . .	27
<b>8. Conclusiones Generales y Próximos pasos</b>	<b>29</b>
8.1. Introducción . . . . .	29
8.2. Conclusiones . . . . .	29
<b>Bibliografía</b>	<b>29</b>

# Capítulo 1

## Introducción

### 1.1. Introducción

[TODO]

### 1.2. Objetivos

[TODO]

### 1.3. Metodología

[TODO]

### 1.4. Problema de recogidas y envíos

[TODO]

### 1.5. Aplicaciones

[TODO]

### 1.6. Conclusiones

[TODO]



# Capítulo 2

## Formulación del Problema

### 2.1. Introducción

El problema de recogidas y entregas (*Pickup and Delivery*), o *PDP* en modo abreviado representa una de las modelizaciones más interesantes en el ámbito de los problemas de *optimización combinatoria*. Esto se debe a la gran cantidad de situaciones del mundo real que pueden ser representadas siguiendo dicho esquema. Sin embargo, antes de profundizar en los aspectos más detallados que caracterizan el problema de recogidas y entregas, es necesario describir el contexto del mismo, así como la clase a la cual pertenece. Una vez se haya completado dicha tarea, se estará en condiciones necesarias para poder describir tanto la versión básica como las extensiones más interesantes, tanto desde el punto de vista de los aspectos matemáticos, como desde la cantidad de situaciones reales que permiten resolver.

En cuanto a la organización del capítulo, en este apartado se describe de manera detallada el contexto del problema desde el punto de vista de la clase de problemas matemáticos a la que pertenecen (apartados 2.1.1 a 2.1.4), las características generales que presenta (apartado 2.1.5) y algunas de las situaciones de gran relevancia para nuestra sociedad que están siendo resueltas siguiendo la modelización *PDP* (apartado 2.1.6).

[TODO: continuar descripción del resto de apartados del capítulo]

A continuación se procede a describir la clase de problemas matemáticos a la cual pertenece el *problema de recogidas y entregas*. Dicha descripción se llevará a cabo de fuera hacia dentro, esto es desde la categoría de problemas más amplia hasta la más concreta, pasando por una breve contextualización así como ejemplificación de problemas similares.

#### 2.1.1. Problemas de Optimización

La clase de *problemas de optimización* representa una de las áreas de investigación más interesante en la actualidad, ya que muchas de las innovaciones obtenidas en dicho campo permiten resolver problemas aplicables al mundo real de manera práctica que antes únicamente podían ser resueltos

teóricamente. En concreto, los problemas de optimización son aquellos que se basan en la minimización (o maximización) de una determinada función objetivo (posiblemente vectorial, lo cual define modelos multiobjetivo) de manera que se satisfaga un conjunto de restricciones previamente fijadas sobre un conjunto de variables de decisión que afectan mutuamente a la satisfacibilidad de las restricciones y el valor de la función objetivo.

Dichas variables de decisión pueden ser tanto categóricas como numéricas (discretas o continuas), lo cual genera una gran cantidad de subproblemas diferentes (Nótese que las variables categóricas con  $k$  niveles diferentes pueden ser representadas de manera sencilla a partir de  $k - 1$  variables binarias). De la misma manera, tanto el valor de función objetivo como las restricciones pueden tener una naturaleza muy diferente: estas pueden estar formadas por funciones lineales de las variables de decisión, como por complicadas funciones no lineales que complican el proceso de obtención del valor óptimo del problema. De manera matemática, la ecuación 2.1 define la formulación de optimización, donde tanto las funciones  $f_i(\cdot)$  como  $g_k(\cdot)$  son funciones arbitrarias que proyectan el vector de variables de decisión  $n$ -dimensional  $\mathbf{x}$  en un espacio unidimensional (generando un valor escalar).

$$\begin{aligned} \underset{\mathbf{x} \in \mathbb{R}^n}{\text{Minimizar}} \quad & f_i(\mathbf{x}), \quad \forall i \in \{1, \dots, I\} \\ \text{sujeto a} \quad & g_k(\mathbf{x}) \leq 0, \quad \forall k \in \{1, \dots, K\} \end{aligned} \tag{2.1}$$

### **Ecuación 2.1:** Formulación del modelo de Optimización General

Muchos de los problemas que resolvemos a diario en nuestra vida cotidiana son en cierta medida *problemas de optimización*, desde qué elementos decidimos añadir a nuestra mochila cada día (basados en restricciones de capacidad, funciones objetivo de utilidad y variables de decisión binarias) hasta la detección del rostro por nuestros teléfonos móviles para aplicar un filtro de la manera más realista posible en una videollamada (basados en restricciones de forma, funciones objetivo multidimensionales y millones de variables de decisión numéricas).

#### **2.1.2. Problemas de Optimización Lineal**

Una de las categorías de problemas de optimización más ampliamente estudiados por su relativa simplicidad (ya se han desarrollado métodos capaces de obtener soluciones óptimas en un número reducido de pasos) y su gran capacidad de modelización ante muchas situaciones del mundo real son los *problemas de optimización lineal*. Dichos problemas se caracterizan por estar compuestos por variables de decisión compuestas por transformaciones lineales respecto de la función objetivo. Esto quiere decir que tanto el valor de la función objetivo como las posibles restricciones escritas en forma de desigualdades están compuestas por sumas de las variables de decisión multiplicadas por determinados pesos.



En la ecuación 2.2 se muestra a modo de ejemplo la formulación de un problema de optimización lineal (del cual se hablará posteriormente). Como se puede apreciar, en este caso las funciones arbitrarias definidas en la formulación general han sido sustituidas por transformaciones lineales respecto de las variables de decisión. Para la resolución de problemas de este tipo se han desarrollado una gran cantidad de métodos, entre los que destaca un algoritmo altamente eficiente el cual se conoce como *Simplex* [Klee and Minty, 1970]. Este algoritmo se basa en pivotaje entre soluciones de manera que tras cada iteración se llegue a una solución igual o mejor. Una de las mayores ventajas de la formulación de un problema como lineal es que algoritmos como *Simplex* proporcionan garantías de optimalidad al alcanzar el valor óptimo al terminar completamente su ejecución.

$$\begin{aligned}
&\text{Minimizar} && \sum_{i=1}^N \sum_{j=1}^M c_{ij} x_{ij} \\
&\text{sujeto a} && \sum_{i=1}^N x_{ij} = s_i, \quad \forall i \in \{1, \dots, N\} \\
&&& \sum_{j=1}^M x_{ij} = d_j, \quad \forall j \in \{1, \dots, M\} \\
&&& x_{ij} \geq 0, \quad \forall i \in \{1, \dots, N\}, \forall j \in \{1, \dots, M\}
\end{aligned} \tag{2.2}$$

**Ecuación 2.2:** Formulación de un modelo de *Optimización Lineal*. En concreto, el *Problema de Transporte*

A pesar de que el algoritmo *Simplex* siempre proporcione resultados óptimos, en algunas ocasiones no se posee la capacidad suficiente de cálculo para llegar a la mejor solución. Por lo tanto, se han desarrollado una gran cantidad de métodos conocidos como *heurísticos* (de los cuáles se hablará más en detalle en el capítulo 4 y capítulo 5) que a pesar de proporcionar unos buenos resultados, no ofrecen ninguna garantía de optimalidad. Por contra, también existen otros métodos exactos que son capaces de llegar al valor óptimo utilizando menor cantidad de recursos (algunos de los cuales se describen en el capítulo 3).

Antes de describir algunos de los ejemplos y aplicaciones reales más destacadas basadas en *problemas de optimización lineal*, es necesario describir unos de los problemas más populares de esta categoría, el cual se conoce como *Problema de Transporte* y se caracteriza por permitir representar de manera matemática la tarea sobre cómo distribuir un conjunto de recursos procedentes de  $N$  puntos de origen hasta  $M$  puntos de destino, donde cada trayecto tiene un coste diferente, y cada origen y destino unas capacidades de oferta y demanda. La formulación sobre dicho problema se corresponde con la utilizada a modo de ejemplo en la ecuación 2.2.

Sin embargo, los problemas de optimización lineal permiten representar una amplia cantidad de situaciones de nuestra vida diaria. Entre ellos se encuentran los *problemas de mezclas* (donde se pretende generar un compuesto con unas ciertas características a partir de la combinación de otros tratando de reducir los costes) aunque los problemas de optimización lineal también son de gran

utilidad en el ámbito de la economía y los estudios de mercado, permitiendo representar de una manera relativamente sencilla el comportamiento de los clientes ante cambios de precio u otras variables más elaboradas.

### 2.1.3. Problemas de Optimización Combinatoria

Uno de los factores más relevantes cuando se trata de *problemas de optimización lineal* se refiere a la naturaleza de las variables de decisión, en que (como se ha comentado anteriormente) estas pueden ser de una naturaleza continua o discreta. En este sentido, los problemas pueden clasificarse en 4 categorías: problemas de optimización continua puros (donde existen métodos muy eficientes que permiten ser resueltos en un tiempo razonable), problemas de *optimización discretos*, problemas de optimización binaria (o de *optimización combinatoria*) y problema de optimización mixtos (donde mezclan ambos tipos de variables).

A pesar de que la exigencia de que las variables de decisión sean discretas en un primer momento puede parecer muy sencilla en un primer momento (por ser muy intuitiva a nivel conceptual), esta complica mucho la labor de optimización del problema. En modo abstracto, esto se debe a que si el espacio de soluciones de un problema de optimización lineal era visto como un *prima* en un espacio  $n$ -dimensional (donde  $n$  es el número de variables de decisión) y los puntos de interés se refieren a los vértices, que presentarán el máximo/mínimo valor óptimo, en el caso de las variables discretas el *prima* se transforma en una estructura “pixelada” lo cual incrementa de manera exponencial el número de vértices (y por tanto de puntos de evaluación) del problema en la búsqueda del valor óptimo.

Las dificultades descritas en el párrafo anterior provocan que la clase de problemas de optimización de variables discretas, entre los que se encuentra la de los problemas de optimización combinatoria (a la cuál a su vez pertenecen los de rutas, de los que se hablará posteriormente), hacen que el espacio de soluciones se extienda de tal manera que para casos relativamente pequeños, este sea inabordable. A pesar de existir métodos exactos (de los cuales se hablará en el capítulo 3), la mayor parte de la literatura sobre este tema se ha dedicado a la investigación de métodos que a pesar de no ofrecer garantías de optimalidad, proporcionan un buen acuerdo entre eficiencia computacional y calidad de las soluciones obtenidas (los cuales se detallan en los capítulos 4 y 5).

$$\begin{aligned}
 &\text{Maximizar} && \sum_{i=1}^N u_i x_i \\
 &\text{sujeto a} && \sum_{i=1}^N w_i x_i \leq W, \quad \forall i \in \{1, \dots, N\} \\
 &&& x_i \in \{0, 1\}, \quad \forall i \in \{1, \dots, N\}
 \end{aligned} \tag{2.3}$$

**Ecuación 2.3:** Formulación de un modelo de *Optimización Combinatoria*. En concreto, el *Problema de la Mochila*

Los problemas de *optimización combinatoria* se caracterizan porque las variables de decisión del modelo son de carácter binario. Dicha razón permite representar situaciones muy interesantes y diversas del mundo real, lo cual ha hecho que los campos de aplicaciones de este tipo de problemas sean desde la generación de rutas de vehículos con garantías de conectividad, hasta problemas de decisión relacionados con la toma o no de una determinada acción. Dentro de los problemas de decisión, existe uno que destaca sobre el resto por su relativa simplicidad teórica, pero a la vez extremada practicidad, el cual se conoce como *problema de la mochila*. A pesar de tener muchas vertientes y extensiones, la idea básica de este es la de maximizar el valor en el proceso de selección de un subconjunto de elementos, dadas unas limitaciones de capacidad y un valor determinado para cada producto. La “dificultad” de este modelo reside en que los elementos no pueden ser escogidos parcialmente, por lo que surge un problema de combinatoria entre los elementos que añadir o no al subconjunto que forma la solución. A modo de ejemplo, en la ecuación 2.4 se muestra la versión más básica del *problema de la mochila*, donde tal y como se puede apreciar, las variables de decisión tan solo pueden tomar valores  $\{0, 1\}$  lo cual determina la presencia o no del elemento  $i$ -ésimo en el subconjunto solución.

Tal y como se ha remarcado a lo largo de todo el apartado, es especialmente importante remarcar que la mayor dificultad surgida al añadir las restricciones en el soporte de las variables de decisión genera un mayor espacio de puntos de evaluación, lo cual complica extremadamente los métodos de generación de soluciones. Por contra, esta dificultad amplía en gran medida la capacidad de representación de los modelos, lo cual es algo beneficioso ya que permite aprovechar en mayor medida recursos que de otra forma serían imposibles de aprovechar de la misma manera.

#### 2.1.4. Optimización de de Rutas

[TODO]

$$\begin{aligned}
& \text{Minimizar} && \sum_{i=0}^N \sum_{j=1, j \neq i}^N c_{ij} x_{ij} \\
& \text{sujeto a} && \sum_{i=0, i \neq j}^N x_{ij} = 1, && \forall i \in \{1, \dots, N\} \\
& && \sum_{j=0, j \neq i}^N x_{ij} = 1, && \forall j \in \{1, \dots, N\} \\
& && u_i - u_j + N x_{ij} \leq N - 1, && 1 \leq i \neq j \leq N \\
& && x_{ij} \in \{0, 1\}, && \forall i \in \{1, \dots, N\}, \forall j \in \{1, \dots, N\}
\end{aligned} \tag{2.4}$$

**Ecuación 2.4:** Formulación de un modelo de *Optimización de Rutas*. En concreto, el *Problema del viajero*

#### 2.1.5. Problemas de Recogidas y Entregas

[TODO]

### 2.1.6. Aplicaciones Reales

[TODO]

## 2.2. Notación

[TODO]

- $V_i$ : [TODO]
- $A_l$ : [TODO]
- $K_k$ : [TODO]

[TODO]

## 2.3. Formulación básica

Modelización basada en [Parragh et al., 2008]. [TODO]

$$\begin{aligned}
& \text{Minimizar} && \sum_{k \in K} \sum_{(i,j) \in A} c_{ij}^k x_{ij}^k \\
& \text{sujeto a} && \sum_{k \in K} \sum_{j: (i,j) \in A} x_{0j}^k = 1, && \forall i \in P \cup D \\
& && \sum_{j: (0,j) \in A} x_{i,n+\bar{n}+1}^k = 1, && \forall k \in K \\
& && \sum_{i: (i,n+\bar{n}+1) \in A} x_{ij}^k = 1, && \forall k \in K \\
& && \sum_{i: (i,j) \in A} x_{ij}^k - \sum_{j: (j,i) \in A} x_{ij}^k = 0, && \forall j \in P \cup D, \forall k \in K \\
& && x_{ij}^k = 1, \implies B_j^k \geq B_i^k + d_i + t_{ij}^k && \forall (i,j) \in A, \forall k \in K \\
& && x_{ij}^k = 1, \implies Q_j^k = Q_i^k + q_j && \forall (i,j) \in A, \forall k \in K \\
& && \max\{0, q_i\} \leq Q_i^k && \forall i \in V, \forall k \in K \\
& && Q_i^k \leq \min\{C^k, C^k + q_i\} && \forall i \in V, \forall k \in K \\
& && x_j \in \{0, 1\}, && \forall j \in \{1, \dots, n\}
\end{aligned} \tag{2.5}$$

**Ecuación 2.5:** [TODO]

[TODO]

## **2.4. Restricciones Adicionales**

[TODO]

### **2.4.1. Ventanas Temporales**

[TODO]

### **2.4.2. Duración de viaje**

[TODO]

### **2.4.3. Duración de ruta**

[TODO]

## **2.5. Funciones Objetivo**

[TODO]

### **2.5.1. Pickup and Delivery**

[TODO]

### **2.5.2. Dial a Ride**

[TODO]

### **2.5.3. Taxi Sharing**

[TODO]

## **2.6. Tiempo Real**

[TODO]

## **2.7. Conclusiones**

[TODO]



# Capítulo 3

## Métodos de Resolución Exactos

### 3.1. Introducción

[TODO]

[TODO: Definir Secciones.]

### 3.2. Conclusiones

[TODO]





# Capítulo 4

## Heurísticas

### 4.1. Introducción

[TODO]

### 4.2. Greedy

[TODO]

---

**Algorithm 1:** [TODO]

---

**Result:**  $E'$

```
1  $S \leftarrow \emptyset$ ;  
2 while  $A \neq \emptyset$  do  
3    $o \leftarrow \text{best}(A)$ ;  
4    $S \leftarrow S \cup \{o\}$ ;  
5    $A \leftarrow A \cap \{o\}$ ;  
6 end
```

---

[TODO]

#### 4.2.1. Criterios de Selección

[TODO]

#### 4.2.2. Randomized Greedy

[TODO]

### 4.3. Metropolis Hastings

[TODO]

### 4.4. Conclusiones

[TODO]

# Capítulo 5

## Metaheurísticas

### 5.1. Introducción

[TODO]

### 5.2. GRASP

[TODO]

### 5.3. Simulated Annealing

[TODO]

### 5.4. Tabu Search

[TODO]

### 5.5. Ant Colony

[TODO]

### 5.6. Variable Neighborhood Search

[TODO]

### 5.7. Large Neighborhood Search

[TODO]

## 5.8. Conclusiones

[TODO]

# Capítulo 6

## Implementación

### 6.1. Introducción

[TODO]

[TODO: Definir Secciones.]

### 6.2. Conclusiones

[TODO]



# Capítulo 7

## Resultados

### 7.1. Introducción

[TODO]

[TODO: Definir Secciones.]

### 7.2. Conclusiones

[TODO]





# Capítulo 8

## Conclusiones Generales y Próximos pasos

### 8.1. Introducción

[TODO]

[TODO: Definir Secciones.]

### 8.2. Conclusiones

[TODO]



# Bibliografía

- [Klee and Minty, 1970] Klee, V. and Minty, G. J. (1970). How good is the simplex algorithm. Technical report, WASHINGTON UNIV SEATTLE DEPT OF MATHEMATICS.
- [Parragh et al., 2008] Parragh, S. N., Doerner, K. F., and Hartl, R. F. (2008). A survey on pickup and delivery problems. *Journal für Betriebswirtschaft*, 58(1):21–51.