

Optimization of the Waste Collection and Recycling Process

**Ulises Díez Santaolalla Ignacio Felices Vera Teresa Franco Corzo**

**Sofía Negueruela Avellaneda**

## Optimización y Simulación

## 3º Grado en Ingeniería Matemática e Inteligencia Artificial

Content Table

[1. Introduction 3](#_Toc186901395)

[2. Problem Statement and Visualization 3](#_Toc186901396)

[3. Methodology 5](#_Toc186901397)

[3.1 First Phase: Funtional solution 6](#_Toc186901398)

[3.2 Second Phase – Realistic solucion 6](#_Toc186901399)

[4. Bibliografía 12](#_Toc186901400)

# **Introduction**

The goal of this project is to provide an optimal solution for an everyday problem: waste collection. This is a crucial step towards building a sustainable and eco-friendly urban environment. Waste management refers to the activities and actions required to handle waste from its generation to its final disposal. This includes collection and transportation to the landfill site.

However, as global populations continue to grow and waste generation increases, the efficient management of garbage becomes more important than ever. By streamlining collection systems, improving sorting techniques, and enhancing recycling processes, communities can reduce their environmental impact, conserve natural resources, and contribute to a circular economy.

In the following pages we have described how to manage the routes a truck could use to collect as much litter as possible. We want to reduce how long it takes to empty all trash containers and the number of trucks needed for it. This approach not only benefits the planet but also creates economic opportunities and promotes healthier living conditions for residents. If we reduce the time, we are reducing carbon dioxide emissions; and by using less trucks the price goes down, making it more affordable.

With the solution proposed below, we want to transform waste management into a more effective and responsible practice by optimizing the truck routes. Such a small change can make a difference for the planet and our pockets.

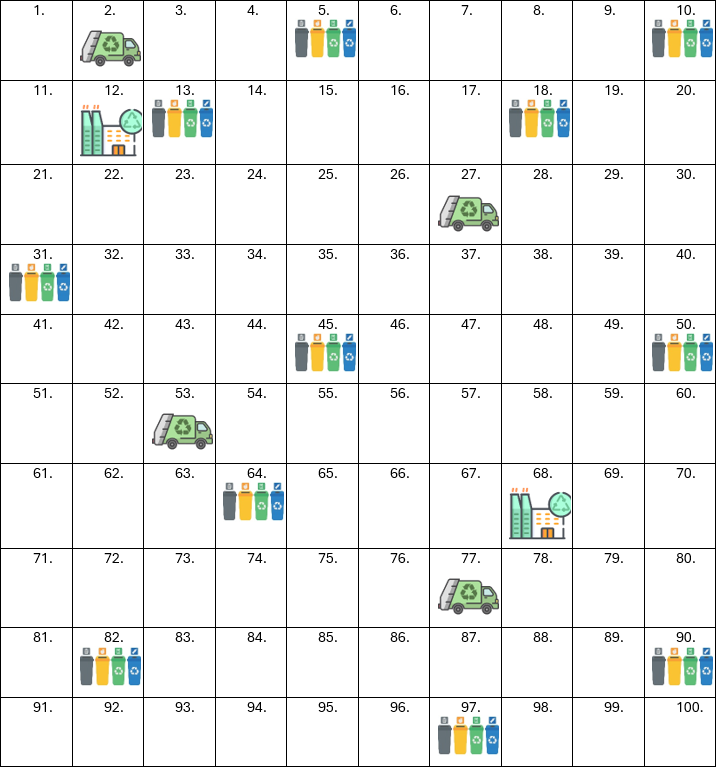
# **Problem Statement and Visualization**

Waste collection companies must manage routes with varying distances in their daily operations, which presents a significant challenge in terms of efficiency and cost. To illustrate the real-world relevance of this issue, consider the example of managing an average of 19,600 daily routes to serve 20 million residential customers and 2 million commercial clients. The company operates a fleet of 26,000 collection trucks, with an approximate operational cost of $120,000 per vehicle. Given this high cost, it is crucial that each route is managed efficiently and cost-effectively.

In this context, route optimization must focus on reducing costs while ensuring the complete collection of waste within the estimated time. In our specific context, the goal is to optimize the number of recycling trucks within a network and their distribution, based on a system that measures the distance between network nodes using Manhattan distance, to allow coverage of the greatest number of containers. To simulate traffic and the conditions of a street, we have included the time it takes to pass through all containers considering that maybe the time spent on different streets varies.



🡪Below we have presented the visualization of the initial problem from which we will start. This problem begins with 1 depot from which the trucks depart, 1 recycling truck, and 2 garbage dumps with varying amounts of waste. All of this is set within a 3x3 grid. The truck, once they have collected the waste, returns full to the depot. It’s important to note that the problem parameters will be modified to ensure that the trucks always collect all the waste. This first version of the problem does not include all the constraints we have talked about.



# **Methodology**

In this optimization problem, as explained before, we have considered minimizing both the number of trucks and the distance traveled by each of them. However, we have observed that reducing the distance does not always decrease the total cost, as some routes, although longer, may be completed in less time. Our goal is to optimize the waste management process.

Minimizing the number of trucks also presents limitations. Although fewer trucks could be used, this might result in less efficiency than before. In the long run, it could be more cost-effective to employ a larger number of trucks, which would significantly reduce the total time of the process. In this sense, it is a trade-off between labor costs and the additional use of trucks, which could ultimately be more beneficial.

Therefore, we have determined that time is the key variable to minimize, as it directly impacts costs such as wages, fuel, and other expenses related to the actual duration of the process, which are not directly addressed in our optimization problem. For this reason, we have decided that the best strategy to optimize this problem is to minimize the route time for each truck, so that is our goal.

It must be clarified that for the first version of the problem we considered that the time between cells was constant, therefore we just needed to minimize the number of steps the truck made in the matrix. Later, as the problem evolved, we created a matrix of time that varied between cells and added it to our



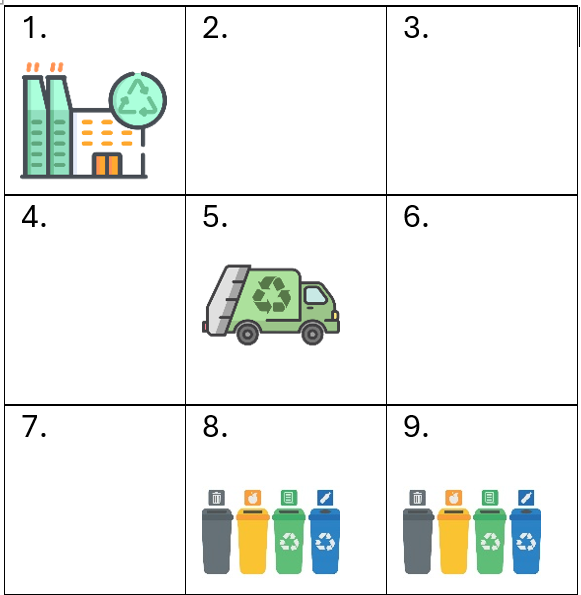
The difficulty of the problem makes us divide the project in two phases: functionality and realism. We want for sure a real solution, but we are giving priority to finding an optimal one first. We will first find the shortest route for one truck, assuming that minimizing distance and time is equivalent, although we have discussed before that it is not always true.

After achieving an optimal solution, we will start incrementing the difficulties of the problem step by step, only moving forward once we have reached an optimal solution for each new upgrade. We want the final version of the problem to address the time it takes to travel from one node to another in the grid, simulating the streets, with some being longer to travel. This time is predetermined. Initially, it will be constant, but in an expanded version of the problem, it will vary, allowing for optimization of the different possible routes.

In the expansion, the dumps will be added, ensuring that once the truck goes to the dump, it empties its load and returns empty to the depot to rest until the next day. The time between nodes will also be addressed,

# **First Phase – Functional solution**

To start with, the problem will consist of a 3x3 grid with only one truck, one deposit, and three litter dumps. We included initially a constant representing the time it takes to move from one cell to another. However, we noticed soon that this was useless, because minimizing the time of the route is the same as minimizing the number of cells the truck passes, so we formulated the solution of the simplified version. The initial grid is presented below.



* **Sets**

A set is a well-defined collection of distinct objects or elements. Sets are used to group related items, such as locations, vehicles, tasks, or constraints, which form the basis for defining variables and restrictions. In our problem, we will use several sets to structure the key elements of the formulation.

* *i*: Rows of the grid {1, ..., N\_FIL}
* *j*: Columns of the grid {1, ..., N\_COL}

Nodes (*i*, *j*) represent all relevant locations for the problem, including depots and dumps.

* *k*: Trucks in service responsible for collecting the waste. {1, ..., NC}
* **Parameters**

A parameter is a fixed numerical value that represents known or constant data in the model. Unlike decision variables, parameters are not subject to optimization; instead, they provide the essential inputs and structure necessary to define the objective function and constraints of the problem.

Although we initially considered including service time at each stop as a parameter, we assume that, for our case, the service time per stop will be uniform. This assumption is based on the use of identical containers and processes at each regular stop. Consequently, it is unnecessary to include service time as a parameter, as it would not contribute to optimizing the problem.

In our problem, we will include the following parameters:

* Regular Stops – *P(i, j)*

This binary matrix indicates which cells in our grid contain stops where trucks must collect waste. The positions of the stops are preselected before solving the problem.

* Depot – *D(i, j)*  
  This binary matrix indicates which cells in our grid contain depots from which the trucks depart. (The positions of the depots are preselected before solving the problem.)
* Number of Rows – *N\_FIL*
* Number of Columns – *N\_COL*
* Truck Capacity – *Ck*  
  The amount of waste a truck can carry, measured in kilograms, is predetermined.
* Amount of Waste at a Regular Stop – *CP(i, j)*  
  A matrix is required to indicate the amount of waste (in kilograms) at each stop.
* **Variables**
* **Optimal Function**
* **Constraints**

After obtaining an optimal solution, we tried with a 5x5 grid and three litter dumps.



# **Second Phase – Realistic solution**

Sets:

#set más secuenciación de máquinas…? O igual recolectar el orden de casillas que hace, el último n debe ser el depósito.

En qué orden van recorriendo las casillas…

Hacer xijk sea no binaria y marque el orden

#como recorro la matriz, que recorro uno más en i y en j.





Parámetros:

Un **parámetro** es un valor numérico fijo que representa datos conocidos o constantes en el modelo. A diferencia de las variables de decisión, los parámetros no están sujetos a optimización; en su lugar, proporcionan las entradas esenciales y la estructura necesaria para definir la función objetivo y las restricciones del problema.

En nuestro problema, incluiremos los siguientes parámetros:

### Suposición sobre el Tiempo de Servicio

Aunque inicialmente consideramos incluir el tiempo de servicio en cada parada como un parámetro, asumimos que, para nuestro caso, el tiempo por parada será uniforme. Esta suposición se basa en el uso de contenedores y procesos idénticos en cada parada regular. En consecuencia, no es necesario incluir el tiempo de servicio como un parámetro, ya que no contribuiría a optimizar el problema.

1. **Paradas regulares P (i, j):**

Esta matriz binaria nos indica en que celdas de nuestra malla se encuentran las paradas en las que los camiones deben recoger basura. (Seleccionamos las posiciones de las paradas previamente a solucionar el problema).

1. **Depósito D (i, j):**

Esta matriz binaria nos indica en que celdas de nuestra malla se encuentran los depósitos desde los que parten los camiones. Seleccionamos las posiciones de las paradas previamente a solucionar el problema).

1. **Número de camiones - NC**
2. **Número de filas - N\_FIL**
3. **Número de columnas - N\_COL**
4. **Número de depósitos totales - ND (¿hace falta?)**
5. **Tiempo entre nodos – T:**

El tiempo que se tarda en ir de un nodo a otro en la malla viene determinado. En un principio será constante y en la ampliación del problema variará, de forma que se puedan optimizar las distintas rutas posibles.

### Capacidad de un camión - Ck:

### La cantidad de basura que puede cargar un camión vendrá determinada y puede variar, medida en kilos.

### Cantidad de basura en una parada regular- CP (i, j):

### Necesitamos de una matriz que nos indique cuanta basura hay en cada parada (en kilos).

Variables:

Las variables de decisión son los valores desconocidos que el modelo de optimización debe determinar. Representan las decisiones a tomar para alcanzar el objetivo, como qué rutas tomar, qué vehículos utilizar o cuánto recoger en cada parada.

En nuestro problema, utilizaremos las siguientes variables de decisión:

1. **Recorrido de un camión - x𝒊𝒋𝒌:**

Como lo que queremos es optimizar el tiempo, debemos obtener una matriz que nos indique que nodos atraviesa un camión. Para ello generaremos una matriz binaria que nos indique porqué nodos atraviesa. Esto es 0 si no pasa por ahí o 1 si pasa.

### Cantidad de nodos que atraviesa un camión - y𝒌:

### Como la función objetivo que minimizamos es el tiempo (en principio constante) pro los nodos que recorre un camión necesitamos esta información, ya que con la anterior variable no se define toda la información.

### Cantidad de basura de un camión - b𝒌:

Para ir asegurándonos de que el camión se va llenando y no de más, necesitamos esta variable, que se verá modificada cada vez que el camión pase por una parada regular y en el futuro al descargar en un vertedero.

Que no pueda pasar dos veces por la parada, aunque esté de vuelta hacia el depósito.

Variable que deje pasar si se ha recogido. Lo que he recogido. Modificar la D🡪variable

Tengo que contabilizar cuantos camiones pasan por una parada

Función Objetivo

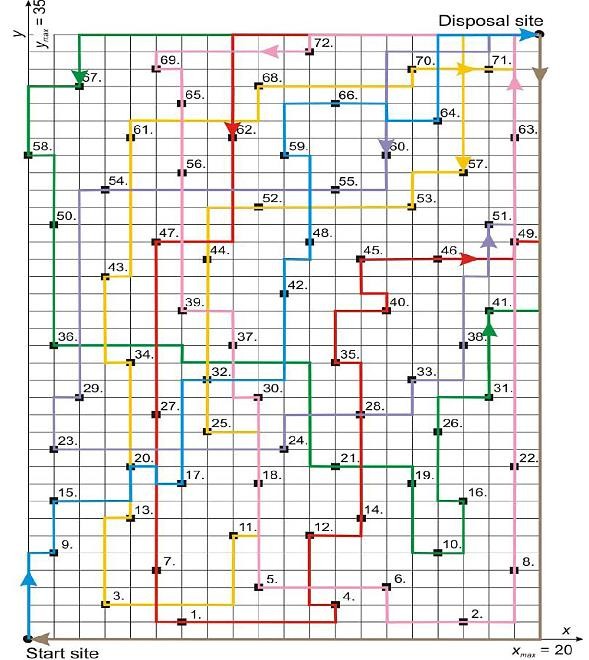
La función objetivo es una expresión matemática que describe el objetivo principal del problema de optimización. En nuestro caso, el objetivo es minimizar el tiempo total de recorrido de todos los camiones en el conjunto KKK. Este tiempo total se calcula como la suma del tiempo asociado a todos los arcos utilizados en las rutas de los camiones.

La función objetivo se define como:

min ∑ T ∙ y𝑘

k

Minimizar el tiempo total permite optimizar factores como salarios, combustible y costos operativos, que están directamente relacionados con la duración del proceso. Por ello, esta formulación asegura que el modelo sea tanto eficiente como económicamente viable.



Restricciones:

Las restricciones son las condiciones necesarias que debe cumplir cualquier solución del modelo para ser válida. Estas limitan las posibles soluciones del problema y garantizan que las operaciones de recolección de basura respeten las condiciones del sistema.

1. Cada parada debe ser atendida por exactamente un vehículo:

Esta restricción asegura que cada parada de recolección sea visitada por un solo vehículo, evitando duplicidades o paradas desatendidas.

1. Cada ruta debe comenzar desde el depósito: (¿?)🡪 convertir a parámetro, decirle donde está
2. La basura recolectada debe caber en el camión
3. Restricción para actualizar la carga del camión si visita una parada

* Si 𝑥𝑖,𝑗,𝑘\*Pi,j = 1: El camión recoge basura, y la carga en el nodo i,j que es la parada. Se actualiza sumando la carga de los nodos anteriores (bk) y la basura recolectada.
* Si 𝑥𝑖,𝑗,𝑘\*Pi,j = 0: El camión no recoge basura en i,j, por lo que la carga no cambia.
* M grande??

1. Ultimo nodo de cada ruta debe ser el depósito (¿¿del que proviene entiendo??, dudas es como la segunda)
2. Se recoge toda la basura (¿aunque ya es implícito en el problema?)🡪 sobra

# Bibliografía

* CONTENUR, 2024. Madrid estrena sistema de recogida de carga lateral con contenedores CONTENUR. Disponible en: <https://www.contenur.com/noticias/madrid-estrena-sistema-de-recogida-de-carga-lateral-con-contenedores-contenur#:~:text=Los%20nuevos%20contenedores%20para%20las,que%20hay%20en%20estos%20momentos>
* Diario Madrid, 2024. Madrid incorporará 36 nuevos camiones eco para la recogida de residuo orgánico. Disponible en: <https://diario.madrid.es/blog/notas-de-prensa/madrid-incorporara-36-nuevos-camiones-eco-para-la-recogida-de-residuo-organico/>
* Ayuntamiento de Madrid, 2024. Recogida de residuos mediante el sistema de carga lateral. Disponible en: <https://www.madrid.es/portales/munimadrid/es/Inicio/Medio-ambiente/Recogida-de-residuos/Recogida-de-residuos-mediante-el-sistema-de-carga-lateral/?vgnextfmt=default&vgnextoid=91b6a48e8cf1c510VgnVCM1000001d4a900aRCRD&vgnextchannel=f81379ed268fe410VgnVCM1000000b205a0aRCRD>
* Koushki, P.A., 2012. Benefits from GIS-Based Modelling for Municipal Solid Waste Management. [online] ResearchGate. Disponible en: <https://www.researchgate.net/publication/221914795_Benefits_from_GIS_Based_Modelling_for_Municipal_Solid_Waste_Management>
* Erol, R. and Ferhatosmanoglu, H., 2016. Solving the problem of vehicle routing by evolutionary algorithm. ResearchGate. Disponible en: <https://www.researchgate.net/publication/293907531_Solving_the_problem_of_vehicle_routing_by_evolutionary_algorithm>