



UNIVERSITAS
Miguel Hernández

**TÉCNICAS DE OPTIMIZACIÓN PARA
LA PLANIFICACIÓN Y GESTIÓN
LOGISTICO PORTUARIA**

Marta Pérez Martín

Grado en Estadística Empresarial

Tutores: Juan Carlos Gonçalves y Ana Meca
Universidad Miguel Hernández
Año académico 2021 - 2025

“No hay nada en el mundo que capacite tanto a una persona para sobrevivir a cualquier situación como la conciencia de tener una tarea que cumplir.”

Viktor E. Frankl, El hombre en busca de sentido

Dedicatoria

A mis padres, por haberme dado la oportunidad de estudiar lo que siempre quise, por la educación, el cariño y el apoyo constante. En especial a mi padre, porque estar lejos de casa no ha resultado tan sencillo como imaginaba, pero siempre le he sentido cerca; gracias por ser inspiración en mi vida.

A mis amigos de siempre por haber estado a mi lado incluso en la distancia, porque durante este tiempo me he perdido muchos momentos con ellos, pero siempre he sentido su cariño y su admiración. Y a las personas que he conocido en estos 4 años, que se han convertido en parte de mi vida y a los que siempre tendré un cariño especial a Silvia, Ana, Ángela, Héctor, Daniel, Hugo, Marina, Irene, ... pero, sobre todo, a ti Laura, por ser la mejor persona que he conocido y por la que nunca me arrepentiré de haberme venido a Elche a estudiar, me llevo una amiga para toda la vida.

Y, por último, a mi tutor del TFG y antiguo profesor, gracias por permitirme elegir libremente el tema de este trabajo, por tu implicación, tu cercanía, y por ser uno de los mejores profesores que he tenido en esta universidad.

Resumen

Este Trabajo de Fin de Grado aborda la optimización del sistema logístico del Puerto de Valencia mediante modelos de investigación operativa. Se plantean tres análisis: la optimización de rutas de barcos mediante el Problema de Rutas de Vehículos (VRP); la mejora de la planificación de actividades portuarias a través del método Técnicas de Revisión y Evaluación de Programas (PERT) y la gestión eficiente de inventarios mediante el modelo Cantidad Económica de Pedido (EOQ). El objetivo es contribuir a la eficiencia y sostenibilidad del puerto ofreciendo soluciones aplicables a otros entornos logísticos.

Abstract

This Final Degree Project addresses the optimization of the logistics system at the Port of Valencia through operations research models. Three analyses are presented: the optimization of ship routes using the Vehicle Routing Problem (VRP); the improvement of port activity planning through the Program Evaluation and Review Technique (PERT); and the efficient inventory management using the Economic Order Quantity (EOQ) model. The aim is to contribute to the efficiency and sustainability of the port by offering solutions applicable to other logistics environments.

Resum

Aquest Treball de Fi de Grau aborda l'optimització del sistema logístic del Port de València mitjançant models d'investigació operativa. Es plantegen tres anàlisis: l'optimització de rutes de vaixells mitjançant el Problema de Rutes de Vehicles (VRP); la millora de la planificació d'activitats portuàries a través del mètode Tècniques de Revisió i Avaluació de Programes (PERT), i la gestió eficient d'inventaris mitjançant el model de Quantitat Econòmica de Comanda (EOQ). L'objectiu és contribuir a l'eficiència i sostenibilitat del port oferint solucions aplicables a altres entorns logístics.

Índice

1. Introducción

- 1.1 Importancia estratégica de la optimización de rutas marítimas
- 1.2 Aplicación actual del Modelo PERT en Logística Portuaria
- 1.3 Aplicación actual del Modelo EOQ en Logística Portuaria

2. Modelos de Investigación Operativa

- 2.1 Optimización de rutas mediante el Problema de Rutas de Vehículos (VRP)
- 2.2 Optimización del tiempo en actividades portuarias mediante el Problema de Técnicas de Revisión y Evaluación de Programas (PERT)
- 2.3 Gestión de inventarios mediante el Problema Cantidad Económica de Pedido (EOQ)

3. Modelos de Investigación Operativa aplicados a la Logística en Puertos

- 3.1 Aplicación del Problema de Rutas de Vehículos (VRP)
- 3.2 Aplicación del Problema de Técnicas de Revisión y Evaluación de Programas (PERT)
- 3.3 Aplicación del Problema Cantidad Económica de Pedido (EOQ)

4. Conclusiones

5. Bibliografía

6. Anexo

1. Introducción

El sector marítimo tiene un papel fundamental en el desarrollo de la economía de nuestro país. Una de las actividades más importantes de este sector es el transporte marítimo, actividad fundamental para el comercio internacional y que representa el 90% del movimiento de mercancías a nivel global (APR, 2024). Según la Asociación de Navieros Españoles (ANAVE, 2024) se prevé que el comercio marítimo crecerá un 2% más en 2025 llegando hasta 12.800 millones de toneladas (Mt). Este aumento viene dado por la implantación de rutas estratégicas como la del mar Rojo y el canal de Panamá.

Actualmente, España, gracias a su ubicación estratégica y su extensa línea costera tiene un papel fundamental, contando con una red portuaria que facilita el comercio europeo e internacional.

En este trabajo nos centraremos en el Puerto de Valencia, un referente clave en el Mediterráneo y Europa. Esta gestionado por la Autoridad Portuaria de Valencia, bajo la marca Valenciaport y administra también los puertos de Sagunto y Gandía.

Si hablamos de tráfico, Valenciaport se posiciona en el cuarto puesto de Europa, siendo también el puerto más grande de España y la cuenca del mar Mediterráneo, con una capacidad de tráfico anual de más de 80 millones de toneladas de carga en 2024. Además, sus instalaciones abarcan más de 12 kilómetros de muelles y alrededor de 600 hectáreas, donde cuenta con terminales especializadas para el manejo de contenedores, graneles líquidos y sólidos, así como para el embarque y desembarque de pasajeros. La infraestructura portuaria se complementa con eficientes conexiones terrestres, entre ellas carreteras y vías ferroviarias que conectan con los principales núcleos productivos del país. Valenciaport representa un motor económico clave, generando alrededor de 15.000 empleos directos e indirectos, contribuyendo significativamente al desarrollo económico local y nacional.

En un contexto tan competitivo y dinámico, surge la importancia de la investigación operativa como herramienta fundamental para optimizar y gestionar eficazmente los recursos disponibles en los puertos, permitiendo tomar decisiones estratégicas basadas en modelos analíticos, lo que facilita no solo reducir costes operativos, sino también mejorar la sostenibilidad ambiental y la calidad del servicio ofrecido.

Este trabajo se estructura en torno a tres modelos principales de investigación operativa, aplicados al sistema logístico portuario del Puerto de Valencia. Cada uno de ellos aborda una parte crítica del proceso logístico, permitiendo un análisis más profundo y riguroso de las operaciones portuarias.

El Problema de Rutas de Vehículos (VRP) busca determinar el conjunto óptimo de rutas que debe seguir un conjunto de vehículos para satisfacer una demanda, minimizando el coste total del transporte, entendido como la distancia recorrida, el consumo de combustible y el tiempo de desplazamiento. Este modelo nos resulta especialmente útil en el contexto portuario para planificar rutas eficientes de carga y descarga de contenedores mejorando la asignación de recursos y reduciendo los tiempos de operación. Para una comprensión más detallada de este modelo, se recomienda consultar el artículo de *Christofides, N.* (1976).

La Técnica de Revisión y Evaluación de Programas (PERT), permite representar y analizar las tareas necesarias para completar un proyecto, optimizando la duración total de las operaciones. Su aplicación en puertos permite planificar con precisión el conjunto de actividades que se desarrollan desde la llegada de la mercancía hasta su expedición, mejorando la coordinación y la utilización de los recursos disponibles. Para profundizar en esta técnica, se sugiere revisar el trabajo de *Yenes, J. M. M.* (1967).

Por último, el modelo de Cantidad Económica de Pedido (EOQ) se centra en la gestión óptima de inventarios, determinando la cantidad ideal de pedido para minimizar el coste total asociados al almacenamiento. En el contexto portuario, este modelo es clave para la gestión de mercancías, favoreciendo al equilibrio entre disponibilidad y eficiencia económica. Para una explicación más detallada, se recomienda el artículo clásico de *Harris* (1913).

1.1 Importancia estratégica de la optimización de rutas marítimas

Desde el punto de vista económico, una ruta ineficiente puede suponer un aumento de los costes logísticos derivados del mayor consumo de combustible, aumento de los tiempos de tránsito y menor utilización de la capacidad de los buques. En este sector las navieras compiten por reducir costes operativos y mejorar los tiempos de entrega, una gestión óptima de las rutas es clave para mantener la competitividad de los puertos y las cadenas de suministro (*Christofides N*, 1976).

En términos medioambientales, el transporte marítimo es responsable del 3% de las emisiones globales de gases de efecto invernadero, posicionándolo en uno de los principales contribuyentes al cambio climático (El País, 2025). Además de CO₂, las embarcaciones emiten óxidos de azufre y nitrógeno, perjudiciales para la calidad del aire en zonas costeras y portuarias. A esto le añadimos otros problemas como la contaminación acústica, los vertidos, la introducción de especies invasoras mediante aguas de lastre y la alteración de ecosistemas marinos (*Zabrocki et al.*, 2021).

Frente a estos desafíos, la optimización de rutas representa una de las soluciones más eficientes. Mediante el Problema de Rutas de vehículos (VRP) se pueden diseñar trayectorias que minimicen las distancias recorridas y reduzcan el consumo energético. Existen además soluciones innovadoras que ya se han implementado, como el uso de velas de succión en buques o la utilización de combustibles alternativos como el metanol y el amoníaco verde, esto ha permitido ahorrar hasta un 15% de combustible (Cadena SER, 2025; *HuffPost*, 2025).

El Puerto de Valencia, tiene un papel protagonista en este proceso, debido, al alto volumen de actividad, su conexión con más de 1.000 puertos en todo el mundo y su ubicación estratégica en las principales rutas comerciales marítimas es imprescindible contar con una planificación eficiente y sostenible. La aplicación de problemas de investigación operativa como el VRP permitirá contribuir a un sistema portuario más responsable con el entorno y alineado con los objetivos de desarrollo sostenible de la Agenda 2030.

1.2 Aplicación actual del Modelo PERT en Logística Portuaria

En la gestión logística portuaria, la técnica PERT (Program Evaluation and Review Technique) se ha consolidado como una herramienta operativa clave para la planificación y control de proyectos debido a su capacidad de manejar la complejidad e incertidumbre de las operaciones portuarias. Este modelo PERT es especialmente útil en proyectos portuarios de gran envergadura (expansiones de terminales, introducción de nuevos sistemas logísticos, etc.) donde intervienen múltiples fases interdependientes. Entre sus beneficios concretos destaca la identificación de la ruta crítica, es decir, la secuencia de tareas que determina la duración mínima del proyecto, ayudando así a los gestores portuarios a focalizar los recursos en las actividades más sensibles a retrasos. Asimismo, PERT facilita una planificación más precisa de los tiempos de ejecución y una toma de decisiones informada, al proporcionar una visión integral de todo el proceso logístico. Esto se traduce en beneficios operativos tangibles, como la reducción de plazos y la mejora de la eficiencia en la cadena de suministro portuaria. Diversos casos reales avalan la utilidad de PERT en entornos portuarios: por ejemplo, Collier *et al.* (2018) aplicaron PERT/CPM en un puerto de contenedores automatizado, demostrando que este método permite planificar estratégicamente proyectos de expansión de capacidad y operaciones en un puerto con inversiones de miles de millones de dólares y millones de movimientos de contenedores al año, identificando los hitos críticos para minimizar perturbaciones.

1.3 Aplicación actual del Modelo EOQ en Logística Portuaria

El modelo de Cantidad Económica de Pedido (EOQ), propuesto originalmente por Harris en 1913, se mantiene como una herramienta clásica para determinar el tamaño de pedido óptimo que minimice el coste total de aprovisionamiento y almacenamiento. En el contexto de los puertos, su utilidad operativa es notable: una política de inventario basada en EOQ permite equilibrar la disponibilidad de repuestos críticos con la eficiencia económica, evitando los costes asociados tanto a faltantes de stock como a excesos de inventario. La indisponibilidad de piezas esenciales (por ejemplo, un cable de izado para una grúa) puede paralizar las operaciones portuarias durante horas o días, generando elevados costes económicos y retrasos en la cadena logística. Por ello, esta aplicación, contribuye a reducir estos tiempos de inactividad y asegurar la continuidad del servicio al menor coste posible.

2. Modelos de Investigación Operativa

2.1 Optimización de rutas mediante el Problema de Rutas de Vehículos (VRP)

El problema de Rutas de Vehículos es una cuestión fundamental en logística y transporte con aplicaciones en sectores como la distribución de mercancías, la planificación de redes de transporte y la gestión de flotas. Su objetivo es determinar un conjunto óptimo de rutas para un grupo de vehículos que deben visitar distintos puntos minimizando costes y cumpliendo con ciertos requisitos (*Tan, S.-Y., & Yeh, W.-C. 2021*).

2.1.1 Elementos del Modelo

Para empezar, debemos definir la red que conecta los distintos nodos visitados entre sí, así como los costes de desplazamiento entre ellos y las demandas de dichos nodos.

Sea la red $G = (V, E, C)$, donde V representa el conjunto de nodos visitados, incluyendo el nodo de partida y llegada que denotaremos por 0. Cada nodo tendrá una demanda asociada d_i , para todo nodo i . Las conexiones entre los nodos están representadas por el conjunto de aristas E , cada una con un costo asociado c_{ij} , que corresponde a la distancia entre los nodos i y j . En este caso, se asume que siempre existe un camino entre cada par de nodos, por lo que la red G es completa.

Disponemos de una flota de K vehículos, a los cuales se les asignará a cada uno de ellos una ruta específica. Cada vehículo debe recorrer su ruta de manera eficiente, comenzando y terminando en el nodo de partida, sin repetir visitas a un mismo nodo dentro de su itinerario y respetando la cantidad establecida C , siendo esta la capacidad máxima de cada uno de ellos.

Suponiendo que cada nodo ha de ser visitado una única vez, por un único vehículo tal que todas las rutas deben empezar y acabar en el mismo punto y las sumas de las demandas de los clientes visitados no puede exceder la capacidad del vehículo, nos interesa encontrar el conjunto de rutas de mínimo coste.

2.2.2 Variables

En primer lugar, definiremos las variables donde $y_{ik}=1$ si el vehículo k , $k \in \{1, \dots, K\}$, visita el nodo i , $i \in V \setminus \{0\}$ en caso contrario $y_{ik}=0$. Y $x_{ijk}=1$ si el vehículo k , $k \in \{1, \dots, K\}$, viaja desde el nodo i , $i \in V$ hasta el nodo j , $j \in V \setminus \{i\}$ en caso contrario $x_{ijk}=0$.

Estas variables permiten describir la asignación de nodos a cada ruta como la secuencia de desplazamientos que componen la ruta de cada barco.

2.2.3 Función Objetivo y Restricciones

El objetivo del modelo es minimizar el coste total del transporte, expresado matemáticamente de la siguiente forma:

$$\min \sum_{i \in V} \sum_{j \in V \setminus \{i\}} \sum_{k=1}^K c_{ij} \cdot x_{ijk}$$

Para garantizar que nuestro modelo sea factible, se incluyen las siguientes restricciones:

1. Asignación de visitas única.

Esta restricción garantiza que cada nodo será atendido por un único vehículo. Así, se asegura que no haya duplicidad en las visitas y que todos los nodos de demanda queden correctamente cubiertos.

$$\sum_{k=1}^K y_{ik} = 1, \quad \forall i \in V \setminus \{0\}$$

2. Todos los vehículos salen del depósito.

Se establece que todos los vehículos inician su ruta en el nodo depósito. Esta condición refleja el punto de partida común para todas las rutas y asegura que el modelo respete la disponibilidad total de vehículos.

$$\sum_{k=1}^K y_{0k} = K$$

3. Equilibrio de flujo para cada nodo.

Si un vehículo entra a un nodo, también debe salir de el, manteniendo un equilibrio en el flujo.

$$\sum_{j \in V \setminus \{i\}} x_{ijk} = \sum_{j \in V \setminus \{i\}} x_{jik} = y_{ik}, \quad \forall i \in V, k = 1, \dots, K$$

4. Límite de capacidad

Cada vehículo tiene una capacidad máxima que no puede ser superada. Esta restricción asegura que la suma de las demandas de los nodos asignados a un vehículo no exceda su capacidad de carga.

$$\sum_{i \in V \setminus \{0\}} d_i y_{ik} \leq C, \quad \forall k = 1, \dots, K$$

5. Evitar subrutas:

Se debe impedir que se formen subrutas independientes entre nodos. De esta forma, se garantiza la coherencia de las rutas dentro de la red.

$$\sum_{i \in S} \sum_{j \notin S} x_{ijk} \geq y_{hk}, \quad \forall S \subseteq V \setminus \{0\}, h \in S, k = 1, \dots, K$$

6. Restricciones de integridad

Se establece que todas las variables de decisión sean binarias.

$$y_{ik} \in \{0,1\}, \quad \forall i \in V, k = 1, \dots, K$$

$$x_{ijk} \in \{0,1\}, \quad \forall i, j \in V, k = 1, \dots, K$$

Hay que tener en cuenta que el VRP presentado es un problema de programación lineal entera. Los métodos de resolución clásicos como ramificación y acotación o planos de corte solo son aplicables en problemas pequeños. Es por ello que vamos a proponer un algoritmo constructivo para la resolución de este tipo de problemas.

2.2.4 Algoritmo de Ahorros de Clarke y Wright

El algoritmo de ahorros propuesto por Clarke y Wright es uno de los métodos heurísticos más eficaces y utilizados en la resolución de problemas de rutas de vehículos (VRP), debido a su simplicidad y rapidez. Su principal función es analizar el ahorro en distancia o coste que se obtiene al enlazar directamente dos nodos en una misma ruta, en lugar de visitarlos de forma separada desde el depósito. Este enfoque fue introducido por Clarke y Wright en 1964, donde presentaron una heurística constructiva basada en el concepto de “ahorro” para combinar rutas individuales en rutas más eficientes (Clarke, G., & Wright, J. W. 1964).

El ahorro entre dos clientes i y j se define como:

$$S_{ij} = c_{i0} + c_{0j} - c_{ij}$$

Donde c_{i0} y c_{0j} representan los costes de ida y vuelta al depósito si los nodos se atienden por separado, y c_{ij} el coste de unirlos en una misma ruta.

A continuación, se describe el procedimiento general en forma estructurada:

Fase inicial.

Se genera, para cada nodo $i \in V \setminus \{0\}$, una ruta individual del tipo $(0, i, 0)$, es decir, que parte y termina en el depósito.

Paso 1

Se calculan los ahorros $S_{ij} = c_{i0} + c_{0j} - c_{ij}$ para todo $i, j \in V \setminus \{0\}$.

Paso 2

Se ordenan los ahorros de forma descendente, dando prioridad a los pares de que clientes que ofrecen un mayor ahorro al ser unidos.

Paso 3

Considerar $S_{ij} = \max S_{lt}$. Si $S_{ij} > 0$ y considerando r_i y r_j , las rutas que contienen a i y j , si en r_i , i es el último nodo y en r_j , j es el primer nodo y, además, las demandas de dichos nodos no superan la capacidad del vehículo, la ruta es factible, eliminar los arcos $(i, 0)$ y $(j, 0)$ e implementar las ruta (i, j) . Eliminar los ahorros $S_{ij} = S_{ji}$.

Paso 4

Se continúa fusionando rutas mientras existan ahorros positivos y combinaciones viables.

Cabe destacar que el cálculo de los ahorros solo debe realizarse una única vez al inicio del algoritmo. Por tanto, es recomendable almacenarlos para evitar una repetición innecesaria en cada iteración del procedimiento. Por otro lado, es importante tener en cuenta que diferentes combinaciones de rutas pueden presentar ahorros idénticos, lo que nos puede llevar a obtener soluciones finales distintas.

A continuación, se presenta un ejemplo práctico de un problema de rutas de vehículos que nos permitirá ver el funcionamiento de este algoritmo.

2.2.5 Un ejemplo de rutas de vehículos

Un barco de carga debe visitar cuatro almacenes de distribución regional, (A1, A2, A3, A4), y la ruta debe empezar y terminar en el centro logístico principal, 0. Sabemos que el barco es único y tiene una capacidad de 12 toneladas. No existe ningún otro tipo de restricciones.

Por tanto, en nuestro problema tenemos un grafo $G=(V,E,C)$, tal que $V=\{0,A1,A2,A3,A4\}$ y E corresponde a los arcos (i,j) de viajar desde el nodo i hasta el nodo j . Además, $K=1$ y $C=12$.

Ahora, se nos indica que los almacenes A1, A2 y A4 tienen una demanda igual a 3, 2 y 2 toneladas respectivamente, mientras que A3 tiene una demanda de 4 toneladas.

Las distancias entre dos puntos $i, j \in V$ están representadas en la Tabla 1.1. En ella, podemos observar que se obtiene una matriz simétrica, lo que implica que la distancia es igual en ambas direcciones.

	0	A1	A2	A3	A4
0	-	10	8	12	7
A1	10	-	6	9	5
A2	8	6	-	5	4
A3	12	9	5	-	6
A4	7	5	4	6	-

Tabla 1.1. Distancias entre nodos.

Ahora que tenemos todos los datos del problema, procedemos a la resolución del problema mediante el algoritmo de ahorros. Implementamos las 4 rutas individuales desde el centro logístico a cada cliente y regreso, de la forma $(0, i, 0)$ tal que $i \in V \setminus \{0\}$.

Por tanto, el coste total inicial de estas rutas es:

$$CT = c_{01} + c_{10} + c_{02} + c_{20} + c_{03} + c_{30} + c_{04} + c_{40} = 2(10 + 8 + 12 + 7) = 74$$

A continuación, calculamos los ahorros, Paso 1, al ser la matriz de distancias simétrica, se tiene que $S_{ij} = S_{ji}$ para cualquier par $i, j \in V \setminus \{0\}$. Y aplicamos la formula:

$$S_{ij} = c_{i0} + c_{0j} - c_{ij}$$

Obtenemos los siguientes ahorros:

$$S_{A1,A2} = 10 + 8 - 6 = 12$$

$$S_{A1,A3} = 10 + 12 - 9 = 13$$

$$S_{A1,A4} = 10 + 7 - 5 = 12$$

$$S_{A2,A3} = 8 + 12 - 5 = 15$$

$$S_{A2,A4} = 8 + 7 - 4 = 11$$

$$S_{A3,A4} = 12 + 7 - 6 = 13$$

En el Paso 2, ordenamos los ahorros de forma no creciente

$$S_{A2,A3} = 15 > S_{A1,A3} = 13 = S_{A3,A4} > S_{A1,A2} = 12 = S_{A1,A4} > S_{A2,A4} = 11$$

Pasamos al Paso 3, donde comenzamos seleccionando el mayor ahorro disponible. En este caso $S_{A2,A3} = 15$. La suma de sus demandas es $2 + 4 = 6$ toneladas, esta es menor a la capacidad de 12 toneladas que posee el camión, por lo tanto, es una ruta factible. Como A2 es el último cliente en su ruta, $(0, A2, 0)$, y A3 el primero en la suya, $(0, A3, 0)$, podemos eliminar los arcos $(A2, 0)$ y $(0, A3)$ e implementar el arco $(A2, A3)$. La nueva ruta resultante es $(0, A2, A3, 0)$. Eliminamos los ahorros $S_{A2,A3}$ y $S_{A2,A4}$.

El siguiente ahorro más alto es $S_{A1,A3}$, evaluamos la posibilidad de añadir el nodo A1 después de A3, siempre que no sobrepase la capacidad máxima del barco. La suma de las demandas en la ruta sería $d_{A2} + d_{A3} + d_{A1} = 2 + 4 + 3 = 9$, cumple los requisitos. Por tanto, se añade el cliente A1 a la ruta después de A3, que da lugar a la ruta $(0, A2, A3, A1, 0)$.

El siguiente ahorro en la lista es $S_{A3,A4}$, pero no podemos aplicarlo ya que después de A3 va el cliente A1 y antes de A3 nos encontramos el cliente A2, lo que impide conectar A3 con A4 por lo que descartamos esta opción.

Analizamos el ahorro $S_{A1,A2}$. Sin embargo, ambos clientes ya están integrados en la ruta actual, por lo que no podemos considerar esta combinación.

Por último, se evalúa el ahorro $S_{A1,A4}$ que si puede utilizarse. Si añadimos el cliente A4 después de A1, la demanda acumulada de la ruta sería $2 + 4 + 3 + 2 = 11$, cumple con los requisitos de capacidad del barco. Por tanto, nuestra ruta final sería (0, A2, A3, A1, A4, 0).

No existen más fusiones posibles sin superar la capacidad del camión o sin reutilizar nodos ya asignados, por lo tanto, el algoritmo termina.

La distancia total recorrida se obtiene como:

$$C_{VRP} = c_{0,A2} + c_{A2,A3} + c_{A3,A1} + c_{A1,A4} + c_{A4,0} = 8 + 5 + 9 + 5 + 7 = 34 \text{ km}$$

Por lo tanto, la solución final del problema es:

$$C_{VRP} = 34 \text{ km}$$

En la siguiente figura puede verse representada la solución obtenida para el problema de rutas de vehículos, donde se muestran los nodos visitados, las conexiones realizadas y las distancias recorridas en cada tramo.

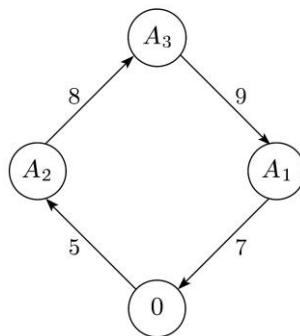


Figura 1.2: Solución del ejemplo del problema de rutas de vehículos.

2.2 Optimización del tiempo en actividades portuarias mediante el Problema de Técnicas de Revisión y Evaluación de Programas (PERT)

El método Técnicas de Revisión y Evaluación de Programas es una herramienta clave en la planificación y gestión de proyectos, la cual, permite analizar la duración de distintas actividades independientes, especialmente cuando los tiempos son inciertos. Es útil en contextos logísticos, como el portuario, donde permite optimizar la secuencia de operaciones para reducir tiempos y mejorar la eficiencia del sistema (Kerzner, 2017).

El objetivo principal es calcular, entre otras cosas, la duración mínima del proyecto, es decir, el menor tiempo necesario para completarlo respetando la precedencia entre actividades.

2.2.1 Elementos del modelo

El modelo PERT se basa en representar un proyecto mediante una red dirigida acíclica, en la que se reflejan las relaciones de precedencia entre actividades. Esta red se define como un grafo $G = (X, N)$, donde X es el conjunto de nodos y N es el conjunto de arcos (actividades) del proyecto. El grafo cuenta con un único nodo inicial y un nodo final. Además, $x^0 \in R_+^n$ es el vector de duraciones planificadas, tal que $x_i^0 \geq 0$ para cada actividad $i \in N$. Cada arco i se identifica con (x_{i1}, x_{i2}) donde $x_{i1} \in X$ es el nodo inicial y $x_{i2} \in X$ es el nodo final. Para cada nodo $x \in X$ se define como $\text{Pred}(x) = \{i \in N: x_{i2} = x\}$ el conjunto de actividades inmediatamente predecesoras a nodo x y $\text{Suc}(x) = \{i \in N: x_{i1} = x\}$ como el conjunto de actividades inmediatamente sucesoras al nodo x . Algunas actividades pueden tener duración cero y representar dependencias lógicas sin actividad real.

Para cada actividad $i \in N$, se definen (Gonçalves-Dosantos et al., 2020):

El tiempo más temprano de una actividad i , $D_i^E(G, x^0)$, se define como, el tiempo mínimo requerido para completar todas las actividades predecesoras inmediatas de $x_{i,1}$, es decir, el tiempo más temprano en el que la actividad i puede comenzar, teniendo en cuenta la estructura del grafo:

$$D_i^E(G, x^0) = \max_{j \in \text{Pred}(x_{i,1})} \{D_j^E(G, x^0) + x_j^0\}$$

El tiempo más tardío de finalización, $D_i^L(G, x^0)$, es el último momento en el que la actividad puede terminar sin retrasar el proyecto:

$$D_i^L(G, x^0) = \max_{j \in N} \{D_j^L(G, x^0) + x_j^0\} \text{ si } \text{Suc}(x_{i,2}) = \emptyset$$

$$D_i^L(G, x^0) = \min_{j \in \text{Suc}(x_{i,2})} \{D_j^L(G, x^0) - x_i^0\} \text{ en otro caso}$$

Con estos valores, se obtiene la duración mínima del proyecto como:

$$D(G, x^0) = \max_{i \in N} \{D_i^E(G, x^0) + x_i^0\}$$

La holgura total de una actividad, denotada $S_i(G, x^0)$, mide cuánto puede retrasarse una actividad sin afectar la duración total del proyecto:

$$S_i(G, x^0) = D_i^L(G, x^0) - D_i^E(G, x^0) - x_i^0$$

Si la holgura de una actividad es igual a 0, entonces esta actividad es crítica, es decir, cualquier perturbación en su duración modifica la duración del proyecto. También se pueden definir otros dos tipos de holgura:

- La holgura libre de una actividad i , $FS_i(G, x^0)$, es el tiempo máximo que puede retrasarse una actividad sin afectar el inicio temprano de sus sucesoras:

$$FS_i(G, x^0) = \min_{j \in \text{Suc}(x_{i,2})} \{D_j^E(G, x^0)\} - D_i^E(G, x^0) - x_i^0$$

- La holgura independiente de una actividad i , $IS_i(G, x^0)$, representa el retraso máximo permitido en la duración de una actividad sin alterar los tiempos de otras actividades:

$$IS_i(G, x^0) = \max \left\{ \min_{j \in \text{Suc}(x_{i,2})} (D_j^E(G, x^0)) \right\} - D_i^L(G, x^0) - x_i^0$$

Dada la holgura de una actividad, definimos el tiempo más tardío de inicio como el último momento en el que una actividad i puede comenzar sin retrasar el proyecto:

$$D_i^{EL}(G, x^0) = D_i^E(G, x^0) + S_i(G, x^0)$$

Y el tiempo más temprano de finalización como el tiempo más temprano en el que una actividad i puede terminar si se inicia lo antes posible:

$$D_i^{LE}(G, x^0) = D_i^L(G, x^0) - S_i(G, x^0)$$

2.2.2 Un Ejemplo del Modelo PERT

Para mostrar la aplicación práctica de los conceptos previamente expuestos se presentará un ejemplo práctico del modelo PERT donde se plantea un evento en una universidad donde el proceso completo se compone de varias actividades con relaciones de dependencia entre ellas. El objetivo es calcular la duración mínima necesaria para llevar a cabo todo el proyecto e identificar la ruta crítica.

La siguiente tabla recoge las siete actividades que componen el proyecto, indicando para cada una su descripción, los precedentes inmediatos y la duración estimada.

Actividad	Descripción	Precedentes	Duración (días)
A	Definir la fecha del evento	—	2
B	Reservar sala	A	3
C	Confirmar ponentes	A	5
D	Diseñar y lanzar campaña de difusión	B, C	4
E	Inscripción de asistentes	D	5
F	Preparar material	D	2
H	Realizar el evento	E, F	1

En la Figura 2.1 se muestra la red PERT correspondiente, en la que se representan visualmente las dependencias entre actividades y se facilitan los cálculos necesarios para localizar la ruta crítica y la duración total del proyecto.

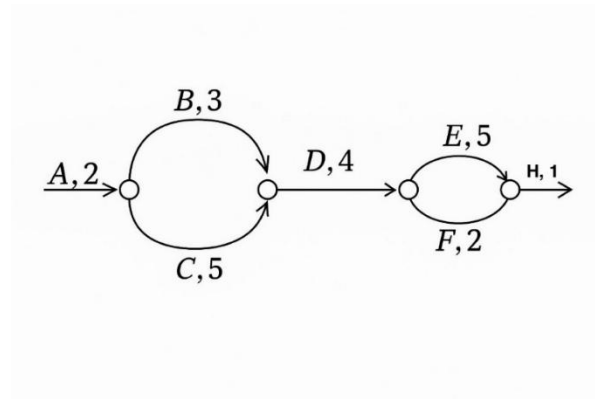


Figura 2.1: Diagrama PERT.

Empezamos calculando el tiempo más temprano

$$D_A^E(G, x^0) = 0$$

$$D_B^E(G, x^0) = \max\{D_A^E(G, x^0) + X_A^0\} = 2$$

$$D_C^E(G, x^0) = \max\{D_A^E(G, x^0) + X_A^0\} = 0 + 2 = 2$$

$$D_D^E(G, x^0) = \max\{D_A^E(G, x^0) + X_A^0, D_B^E(G, x^0) + X_B^0, D_C^E(G, x^0) + X_C^0\} = \max\{2, 5, 7\} = 7$$

$$D_E^E(G, x^0) = \max\{D_A^E(G, x^0) + X_A^0, D_B^E(G, x^0) + X_B^0, D_C^E(G, x^0) + X_C^0, D_D^E(G, x^0) + X_D^0\} \\ = \max\{2, 5, 7, 11\} = 11$$

$$D_F^E(G, x^0) = \max\{D_A^E(G, x^0) + X_A^0, D_B^E(G, x^0) + X_B^0, D_C^E(G, x^0) + X_C^0, D_D^E(G, x^0) + X_D^0\} \\ = \max\{2, 5, 7, 11\} = 11$$

$$D_H^E(G, x^0) = \max\{D_A^E(G, x^0) + X_A^0, D_B^E(G, x^0) + X_B^0, D_C^E(G, x^0) + X_C^0, D_D^E(G, x^0) + X_D^0, \\ + X_D^0, D_E^E(G, x^0) + X_E^0, D_F^E(G, x^0) + X_F^0\} = \max\{2, 5, 7, 11, 16, 13\} = 16$$

Calculamos el tiempo más tardío de finalización

$$D_H^L(G, x^0) = \max\{D_A^E(G, x^0) + X_A^0, D_B^E(G, x^0) + X_B^0, D_C^E(G, x^0) + X_C^0, D_D^E(G, x^0) + X_D^0, \\ + X_D^0, D_E^E(G, x^0) + X_E^0, D_F^E(G, x^0) + X_F^0, D_H^E(G, x^0) + X_H^0\} = \max\{0 + 2, 2 \\ + 3, 2 + 5, 7 + 4, 11 + 5, 11 + 2, 16 + 1\} = \max\{2, 5, 7, 11, 16, 13, 17\} = 17$$

$$D_F^L(G, x^0) = \min\{D_H^L(G, x^0) - X_H^0\} = \min\{17 - 1\} = 16$$

$$D_E^L(G, x^0) = \min\{D_H^L(G, x^0) - X_H^0\} = \min\{17 - 1\} = 16$$

$$D_D^L(G, x^0) = \min\{D_E^L(G, x^0) - X_E^0, D_H^L(G, x^0) - X_H^0, D_F^L(G, x^0) - X_F^0\}$$

$$= \min\{17 - 1, 16 - 5, 16 - 2\} = \min\{16, 11, 14\} = 11$$

$$D_B^L(G, x^0) = \min\{D_D^L(G, x^0) - X_D^0, D_E^L(G, x^0) - X_E^0, D_H^L(G, x^0) - X_H^0, D_F^L(G, x^0) - X_F^0\}$$

$$= \min\{11 - 4, 17 - 1, 16 - 5, 16 - 2\} = \min\{7, 16, 11, 14\} = 7$$

$$D_C^L(G, x^0) = \min\{D_D^L(G, x^0) - X_D^0, D_E^L(G, x^0) - X_E^0, D_H^L(G, x^0) - X_H^0, D_F^L(G, x^0) - X_F^0\}$$

$$= \min\{11 - 4, 17 - 1, 16 - 5, 16 - 2\} = \min\{7, 16, 11, 14\} = 7$$

$$D_A^L(G, x^0) = \min\{D_B^L(G, x^0) - X_B^0, D_C^L(G, x^0) - X_C^0, D_D^L(G, x^0) - X_D^0, D_E^L(G, x^0) - X_E^0,$$

$$D_H^L(G, x^0) - X_H^0, D_F^L(G, x^0) - X_F^0\}$$

$$= \min\{7 - 3, 7 - 5, 11 - 4, 17 - 1, 16 - 5, 16 - 2\}$$

$$= \min\{4, 2, 7, 16, 11, 14\} = 2$$

El cálculo de holguras es:

$$S_A(G, X^0) = D_A^L(G, X^0) - D_A^E(G, X^0) - X_A^0 = 2 - 0 - 2 = 0$$

$$S_B(G, X^0) = D_B^L(G, X^0) - D_B^E(G, X^0) - X_B^0 = 7 - 2 - 3 = 2$$

$$S_C(G, X^0) = D_C^L(G, X^0) - D_C^E(G, X^0) - X_C^0 = 7 - 2 - 5 = 0$$

$$S_D(G, X^0) = D_D^L(G, X^0) - D_D^E(G, X^0) - X_D^0 = 11 - 7 - 4 = 0$$

$$S_E(G, X^0) = D_E^L(G, X^0) - D_E^E(G, X^0) - X_E^0 = 16 - 11 - 5 = 0$$

$$S_F(G, X^0) = D_F^L(G, X^0) - D_F^E(G, X^0) - X_F^0 = 16 - 11 - 2 = 3$$

$$S_H(G, X^0) = D_H^L(G, X^0) - D_H^E(G, X^0) - X_H^0 = 17 - 16 - 1 = 0$$

Cuando la holgura es igual a 0, significa que la actividad pertenece al camino crítico del proyecto. En este caso las actividades críticas son A, C, D, E, H. En cambio, cuando la holgura es mayor que 0, la actividad puede retrasarse hasta ese número de unidades de tiempo sin afectar al proyecto.

Para finalizar, calcularemos la duración mínima del proyecto:

$$D(G, x^0) = \max_{i \in N} \{D_i^E(G, x^0) + x_i^0\} = \max\{0 + 2, 2 + 3, 2 + 5, 7 + 4, 11 + 5, 11 + 2, 16 + 1\}$$

$$= 17$$

Recogemos los datos en la siguiente tabla:

Actividad	Tiempo más temprano	Tiempo más tardío	Holgura
A	0	2	0
B	2	7	2
C	2	7	0
D	7	11	0
E	11	16	0
F	11	16	3
H	16	17	0
Duración mínima del proyecto	17		

En la Figura 2.2 se destaca en color rojo el camino crítico, el cual está compuesto por aquellas actividades críticas que determinan la duración mínima del proyecto. Cualquier retraso en estas tareas afectaría directamente al plazo total de ejecución.

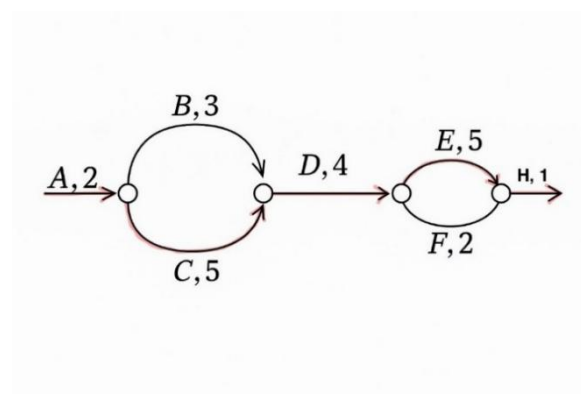


Figura 2.2: Diagrama camino crítico.

2.3 Gestión de inventarios mediante Problema de Cantidad Económica de Pedido (EOQ)

El modelo de Cantidad Económica de Pedido (EOQ) es una herramienta clásica en la gestión de inventarios que permite determinar la cantidad óptima de pedido para minimizar los costes asociados al almacenamiento y la reposición de productos. Su aplicación resulta de gran utilidad en logística portuaria, donde una buena planificación del inventario contribuye a mejorar la eficiencia operativa, reducir costes y evitar acumulaciones innecesarias o escasez de productos (Christopher, M. 2016).

2.3.1 Elementos del Modelo

Los elementos que definen el modelo son λ es la demanda anual del producto, es constante y conocida, k es el coste fijo asociado a la preparación de cada pedido, c es el coste unitario del producto, I es la tasa de almacenamiento por euro invertido, $h = I \cdot c$ es el coste de almacenamiento por unidad y por año, Q cantidad pedida en cada reposición y, por último, $Z(Q)$ es la función de coste total asociada a una política de inventario de tamaño Q .

El objetivo del modelo es determinar el tamaño de pedido óptimo Q^* que consiga minimizar el coste total anual. La función objetivo se define como:

$$Z(Q) = c \cdot \lambda + \frac{k \cdot \lambda}{Q} + \frac{h \cdot Q}{2}$$

La cantidad óptima de pedido Q^* se obtiene resolviendo el problema de minimización de $Z(Q)$. Derivando respecto a Q e igualando a cero se obtiene:

$$Q^* = \sqrt{\frac{2Kd}{h}}$$

El número óptimo de pedidos anuales N^* , refleja la frecuencia con la que debe realizarse el pedido de tamaño Q^* para satisfacer la demanda anual λ con coste mínimo. Se obtiene directamente de:

$$N^* = \frac{\lambda}{Q^*} = \sqrt{\frac{\lambda h}{2K}}$$

La duración óptima del ciclo de reposición, denotada como t^* , indica el intervalo de tiempo que debe transcurrir entre dos pedidos consecutivos de tamaño Q^* . Se define como el cociente entre el tamaño de pedido y la demanda anual, es decir:

$$t^* = \frac{Q^*}{\lambda} = \sqrt{\frac{2K}{\lambda h}}$$

2.3.2 Ejemplo del Modelo EOQ

A continuación, se muestra un caso práctico que ejemplifica la implementación del modelo EOQ descrito anteriormente. Una empresa necesita almacenar su producto para satisfacer la demanda constante de sus clientes durante el año. El objetivo es determinar cuántos paquetes debe pedir cada vez que realice un pedido para minimizar los costes de inventarios.

La demanda anual del producto es de $\lambda = 5800$ paquetes, el coste fijo de preparación es de $k = 50$ euros, el coste de adquisición de cada paquete de folios es de $c = 2$ euros. El coste de almacenamiento anual es el 35% del valor del producto almacenado, es decir, $I = 0,35$, por lo tanto, $h = I \cdot c = 0,35 \cdot 2 = 0,7$ euros por paquete y por año.

Aplicando la formula del modelo EOQ tenemos:

- La cantidad económica de pedido:

$$Q^* = \sqrt{\frac{2k\lambda}{h}} = \sqrt{\frac{2 \cdot 50 \cdot 5800}{0,7}} \approx 910,26 \text{ paquetes}$$

- El número óptimo de pedidos al año:

$$N = \frac{\lambda}{Q^*} = \frac{5800}{910,26} \approx 6,37 \Rightarrow \text{entre 6 y 7 pedidos al año}$$

- Duración de los ciclos de reposición:

$$T = \frac{Q^*}{\lambda} = \frac{910,26}{5800} \approx 0,157 \text{ años} \Rightarrow \text{aproximadamente cada 1,9 meses}$$

3. Modelos de Investigación Operativa aplicados a la Logística en Puertos

3.1 Aplicación del Problema de Rutas de Vehículos (VRP)

En el contexto de este trabajo, el VRP se aplicará a la optimización de rutas marítimas de tres barcos que proceden del Puerto de Valencia y donde operan hacia distintos puertos europeos. La finalidad de este estudio es aplicar técnicas de optimización para reducir las distancias recorridas y mejorar la eficiencia en el transporte. Una mejor planificación de las rutas puede significar reducir costes operativos, minimizar el consumo de combustible y disminuir el impacto ambiental. Además, contribuye a reforzar la competitividad del Puerto de Valencia, uno de los principales centros logísticos de la región mediterránea.

Para realizar este estudio, se cuenta con datos reales relativos a las distancias, en kilómetros, entre los puertos. Estas distancias se han recogido en una matriz, en la que cada fila y columna representan un puerto, y la intersección indica la distancia estimada entre dichos puntos. Con esto, disponemos de la información necesaria para modelar el grafo $G = (V, E, C)$, siendo V el conjunto de puertos formado por Algeciras, Atenas, Barcelona, Bilbao, Casablanca, Copenhague, Cádiz, Estambul, Estocolmo, Gdansk, Hamburgo, Helsinki, Lisboa, Londres, Marsella, Nápoles, Oslo, Rotterdam, San Petersburgo, incluido el Puerto de Valencia (nodo de salida y llegada) y el resto de los puertos europeos, E es el conjunto de aristas que representan las conexiones marítimas posibles entre puertos y, por último, C es la matriz de distancias, Tabla 3.2 que se localiza en el Anexo, asociada a cada arista, en este caso, se asume que es simétrica.

La flota de ValenciaPort dispone de 3 buques idénticos, con una capacidad de 28.000 TEU que parten del depósito 0 (Puerto de Valencia) y deben atender los 19 puertos europeos.

El problema se modela con un Vehicle Routing Problem con número máximo de rutas $K = 3$, mediante el algoritmo de ahorros presentado en la sección 2.2.4 *Clarke & Wright* (1964) y cuyo código podemos encontrar en el Anexo (Código 5.1).



Imagen 3.1 Rutas Iniciales.

Antes de aplicar técnicas de optimización se parte del itinerario inicial, compuesto por tres recorridos independientes (Imagen 3.1). La siguiente Tabla 3.3 resume las distancias totales asociadas a los recorridos iniciales, obtenidas a partir de la matriz de distancias de la Tabla 3.2:

Barco	Secuencia actual	Distancia total
Azul	Valencia → Barcelona → Nápoles → Lisboa → Londres → Róterdam → Hamburgo → Oslo → Copenhague → Estocolmo → Helsinki → San Petersburgo → Valencia	11 479,93
Rojo	Valencia → Barcelona → Marsella → Atenas → Estambul → Algeciras → Bilbao → Róterdam → Hamburgo → Valencia	10 918,55
Verde	Valencia → Cádiz → Casablanca → Lisboa → Bilbao → Londres → Róterdam → Gdansk → Valencia	8 588,38
Total		30 986,86

Tabla 3.3 Tabla de Distancias Asociadas.

Este valor de referencia de 30.986,86 representa el coste operativo actual, en términos de Kilómetros.

Tras ejecutar la función `clarke_wright_max_routes()` con los parámetros descritos, se obtiene la Tabla 3.4, donde se recogen las nuevas rutas, las cargas consolidadas y el coste total:

Ruta	Secuencia obtenida	Carga	Coste
1	Valencia → Marsella → Hamburgo → Atenas → Helsinki → Lisboa → Valencia	24 500	6 153,9
2	Valencia → Cádiz → Bilbao → Estocolmo → Oslo → Barcelona → Gdansk → Algeciras → Estambul → Nápoles → Valencia	22 500	12 378,2
3	Valencia → Casablanca → San Petersburgo → Copenhague → Róterdam → Londres → Valencia	27 000	9 370,8
Total		74 000	27 902,9

Tabla 3.4 Resultados computacionales.

La solución heurística respeta la capacidad de los tres buques y reduce en **9,95 %** el coste frente a la planificación original (30 986,86).

$$\Delta\text{Coste} = \frac{30\,986,86 - 27\,902,9}{27\,902,9} \times 100 \approx 9,95\%$$

En síntesis, la aplicación del VRP al contexto portuario demuestra que una gestión científica de las rutas puede traducirse en ventajas económicas y ambientales tangibles, reforzando la competitividad del Puerto de Valencia y sentando las bases para futuras mejoras en la cadena logística marítima.

3.2 Aplicación del Problema de Técnicas de Revisión y Evaluación de Programas (PERT)

En este trabajo utilizaremos el modelo PERT para optimizar la planificación de las distintas actividades que forman parte del proceso logístico en el Puerto de Valencia. La finalidad de esta aplicación es identificar la duración mínima del proceso logístico y detectar aquellas actividades críticas cuyo retraso afectaría directamente al cumplimiento de los plazos.

Esto resulta especialmente útil en un contexto portuario, donde la gestión eficiente del tiempo no solo mejora el rendimiento operativo, sino que también contribuye a reducir los costes y reforzar la fiabilidad de la cadena logística.

En la siguiente tabla recoge las diferentes actividades que se realizan en el puerto, así como los tiempos necesarios para completar cada una de ellas.

	Actividad	Predecesoras	Duración (días)
A	Preparación y embalaje de la mercancía	—	1
B	Carga de las mercancías en los camiones	A	5
C	Transporte al puerto de origen	B	2
D	Gestión de inventarios	A	1
E	Arribo y control en aduana del puerto	C, D	2
F	Recopilación de la documentación	D	4
G	Inspección automática (flujo habitual)	F	1
H	Inspección automática (vía directa)	E, F	2
I	Descarga y almacenamiento en el puerto	G, H	1
J	Carga de la mercancía en el buque	I, F	2
K	Salida del buque hacia destino	J	3
L	Confirmación de entrega en destino	K	1

Tabla 4.1 Datos iniciales.

Actividad	Descripción	Duración	ES	EF	LS	LF	Holgura	¿Ruta crítica?
A	Preparación y embalaje	1	0	1	0	1	0	✓
B	Carga en camiones	5	1	6	1	6	0	✓
C	Transporte al puerto	2	6	8	6	8	0	✓
D	Gestión de inventarios	1	1	2	5	6	4	—
E	Control aduanero	2	8	10	8	10	0	✓
F	Documentación	4	2	6	6	10	4	—
G	Inspección habitual	1	6	7	11	12	5	—
H	Inspección vía directa	2	10	12	10	12	0	✓
I	Descarga y almacenaje	1	12	13	12	13	0	✓
J	Carga al buque	2	13	15	13	15	0	✓
K	Zarpe del buque	3	15	18	15	18	0	✓
L	Confirmación de entrega	1	18	19	18	19	0	✓

Tabla 4.2 Resultados Computacionales.

En este ejercicio se ha implementado el algoritmo PERT en R mediante el paquete ProjectManagement, definiendo la matriz de precedencias y las duraciones de las doce actividades involucradas (A–L). Obtenemos los siguientes resultados recogidos en la siguiente Tabla:

Finalmente, se determina que la duración mínima del proyector es de 19 días. El camino crítico, corresponde a la cadena de actividades $A \rightarrow B \rightarrow C \rightarrow E \rightarrow H \rightarrow I \rightarrow J \rightarrow K \rightarrow L$, cuyas holguras son nulas y, por tanto, condicionan directamente la fecha de finalización.

3.3 Aplicación del Problema Cantidad Económica de Pedido (EOQ)

En el contexto logístico portuario, una gestión adecuada de inventarios es esencial para asegurar el flujo continuo de las operaciones. Los puertos, al manejar una gran cantidad de mercancías requieren numerosos equipos de apoyo, por lo que es imprescindible disponer de un inventario de repuestos que permita atender con rapidez cualquier necesidad operativa o de mantenimiento.

No tener disponibles artículos como un cable de izado o una polea para una grúa, puede paralizar durante horas o días las operaciones de carga y descarga, generando elevados costes económicos y retrasos en la cadena logística.

El modelo EOQ proporciona una forma sencilla de ajustar estos inventarios, minimizando los costes de adquisición y almacenamiento, como el de riesgo de rotura de stock en componentes esenciales para el funcionamiento del puerto.

En el Puerto de Valencia se gestiona un inventario de cables de izado, necesarios para el mantenimiento de las grúas. Estos cables deben reemplazarse periódicamente para garantizar la seguridad y la eficiencia de las operaciones.

El tamaño de pedido resultante tiene dos implicaciones operativas, en primer lugar, genera la actividad D (Gestión de inventarios) del PERT, cuyo inicio coincide con la llegada del lote, y en segundo lugar, establece el volumen base de mercancía que los buques deberán trasladar alimentando el problema VRP.

La demanda anual de cables es de $\lambda = 1250$, el coste fijo por pedido $k = 2150$ euros, el coste de adquisición de cada cable es de $c = 1500$ euros. La tasa de coste de almacenamiento anual es el 20% del valor del producto almacenado, es decir, $I = 0.20$, por lo tanto, el coste de almacenamiento anual por cable es: $h = I \cdot c = 0,20 \cdot 1500 = 300$ euros.

Aplicando el modelo EOQ obtenemos:

- Cantidad económica de pedido:

$$Q^* = \sqrt{\frac{2k\lambda}{h}} = \sqrt{\frac{2 \cdot 2150 \cdot 1250}{300}} = \sqrt{\frac{5\,375\,000}{300}} = \sqrt{17\,916,67} \approx 133,85 \text{ cables}$$

- Número óptimo de pedidos al año:

$$N = \frac{\lambda}{Q^*} = \frac{1250}{133,85} \approx 9,34 \text{ pedidos al año}$$

- Duración optima del ciclo de reposición:

$$T = \frac{Q^*}{\lambda} = \frac{133,85}{1250} \approx 1,3 \text{ ciclos por año}$$

EL modelo EOQ determina que el puerto debe realizar pedidos de aproximadamente 134 cables en cada ciclo, con una frecuencia aproximada de 1,3 meses, lo que corresponde a 10 pedidos al año, para conseguir mantener su inventario de repuestos en niveles óptimos.

4. Conclusiones

La aplicación de los tres modelos de investigación operativa, Problema de Rutas de Vehículos (VRP), Técnica de Revisión y Evaluación de Programas (PERT) y Cantidad Económica de Pedido (EOQ), ha demostrado ser válida para abordar problemas de logística portuaria en un entorno real como el de ValenciaPort.

En primer lugar, la heurística de Clarke & Wright aplicada al VRP permitió consolidar la carga de los tres buques estudiados y reducir en torno a un diez por ciento la distancia total navegada, lo que se traduce en un ahorro de costes y beneficios medioambientales.

En segundo lugar, el análisis PERT reveló que el proyecto completo puede ejecutarse en diecinueve días siempre que se preste especial atención a la secuencia crítica $A \rightarrow B \rightarrow C \rightarrow E \rightarrow H \rightarrow I \rightarrow J \rightarrow K \rightarrow L$, de modo que cualquier retraso en esas nueve tareas repercutiría directamente en la fecha de finalización.

Por último, el modelo EOQ determinó que un pedido de unas 134 unidades, con diez pedidos anuales y un ciclo de unas cinco semanas, minimiza el coste conjunto de aprovisionamiento y almacenamiento para los cables de izado, evitando tanto roturas de stock como acumulaciones innecesarias.

5. Bibliografía

- **ANAVE (Asociación de Navieros Españoles).** "El comercio marítimo global alcanzará 12.600 Mt en 2024".
- **APR.** (2024, marzo 13). *El impacto del combustible ecológico en el transporte marítimo internacional*. <https://www.apr.es/el-impacto-del-combustible-ecologico-en-el-transporte-maritimo-internacional/>
- **Bergantiños, G.** (s.f.). *Project Planning*. Universidade de Vigo.
- **Cadena SER.** (2025, abril 8). *La empresa cántabra Bound4blue instala las velas de succión más grandes del mundo en un buque que ahorrará un 15% de combustible*. <https://cadenaser.com/cantabria/2025/04/08/la-empresa-cantabra-bound4blue-instala-las-velas-de-succion-mas-grandes-del-mundo-en-un-buque-que-ahorrara-un-15-de-combustible-radio-santander/>
- **Carpente, M. L.** (2014). *Modelos interactivos de la IO. Juegos PERT*. Universidade da Coruña.
- **Christopher, M.** (2016). *Logistics and supply chain management: logistics & supply chain management*. Pearson UK.
- **Christofides, N.** (1976). *The vehicle routing problem*. *Revue française d'automatique, informatique, recherche opérationnelle. Recherche opérationnelle*, 10(V1), 55–70.
- **Clarke, G., & Wright, J. W.** (1964). *Scheduling of Vehicles from a Central Depot to a Number of Delivery Points*. *Operations Research*, 12(4), 568–581.
- **El País.** (2025, abril 11). *La Organización Marítima Internacional aprueba un sistema para gravar las emisiones del sector*. <https://elpais.com/clima-y-medio-ambiente/2025-04-11>
- **Gonçalves-Dosantos, J. C., García-Jurado, I., & Costa, J.** (2020). *ProjectManagement: An R package for managing projects*. *The R Journal*, 12(1), 419–434.
- **Harris, F. W.** (1913). *How many parts to make at once [Cuántas piezas fabricar de una vez]*. *The Magazine of Management*, 10(2), 135–136.
- **Hillier, F. S., & Lieberman, G. J.** (2010). *Introducción a la investigación de operaciones* (9.^a ed.). McGraw-Hill/Interamericana Editores.

- **HuffPost.** (2025, febrero 10). *El plan de dos puertos españoles para transformar el sector marítimo con amoníaco verde atrae el dinero de Europa.*
<https://www.huffingtonpost.es/economia/el-plan-dos-puertos-espanoles-transformar-sector-maritimo-amoniac-verde-atrae-dinero-europabr.html>

- **Kerzner, H.** (2017). *Project Management: A Systems Approach to Planning, Scheduling, and Controlling* (12th ed.). Wiley.

- **Lorenzo Freire, S.** (2015). *Cooperación en modelos de inventario: Modelo EOQ clásico.* Universidad de Vigo.

- **Tan, S.-Y., & Yeh, W.-C.** (2021). *The Vehicle Routing Problem: State-of-the-Art Classification and Review.* Applied Sciences, 11(21), 10295.

- **United States. Bureau of Naval Weapons. Special Projects Office.** (1958). *Program Evaluation Research Task (PERT): Summary Report. Phase 2.* Special Projects Office, Bureau of Naval Weapons, Department of the Navy.

- **Zabrocki, L., Leroutier, M., & Bind, M.-A.** (2021). *Estimating the Local Air Pollution Impacts of Maritime Traffic: A Principled Approach for Observational Data.* arXiv.
<https://arxiv.org/abs/2105.03996>

6. Anexo

Código 5.1 Problema de Rutas de Vehículo.

```
clarke_wright_max_routes <- function(demand, distance_matrix,
vehicle_capacity, max_routes) {

  # demand: vector de longitud n-1 (clientes). Se asume que el
  depósito es el índice 1 en la matriz.

  # distance_matrix: matriz cuadrada (n x n) con n =
  length(demand) + 1

  # vehicle_capacity: capacidad de cada vehículo (escalar)

  # max_routes: número máximo de rutas que queremos (si es
  factible)

  n <- length(demand) + 1  # número de nodos total (1
  depósito + n-1 clientes)

  # 1) Calcular matriz de “ahorros” (savings)

  savings <- matrix(-Inf, n, n)

  for (i in 2:n) {
    for (j in 2:n) {
      if (i != j) {
        savings[i, j] <- distance_matrix[i, 1] +
        distance_matrix[1, j] - distance_matrix[i, j]
      }
    }
  }

  # 2) Inicializar rutas individuales (depo -> i -> depo) para
  cada cliente i = 2..n

  routes <- list()

  for (i in 2:n) {
```

```

    routes[[i - 1]] <- list(
        route = c(1, i, 1),
        load  = demand[i - 1],
        cost  = 2 * distance_matrix[1, i]
    )
}

# 3) Fase 1: combinar rutas mientras haya ahorro > 0 y rutas
> max_routes

repeat {
    if (length(routes) <= max_routes) break # ya tenemos como
    máximo max_routes rutas

    max_saving <- max(savings)

    if (max_saving <= 0) break # no quedan pares con ahorro
    positivo

    idx_pair <- which(savings == max_saving, arr.ind =
    TRUE)[1, ]

    i <- idx_pair[1]
    j <- idx_pair[2]

    # Marcar este par como "usado"

    savings[i, j] <- -Inf
    savings[j, i] <- -Inf

    # 3.a) Encontrar en qué rutas salen j (en segundo lugar) y
    terminan i (penúltimo)

    ri <- NULL; rj <- NULL

    for (k in seq_along(routes)) {
        r <- routes[[k]]$route

        # si termina en i (antes de regresar a 1)

```

```

    if (length(r) >= 3 && r[length(r) - 1] == i) ri <- k

    # si arranca en j (después de 1)

    if (length(r) >= 3 && r[2] == j) rj <- k

  }

  # 3.b) Si ambos existen y no son la misma ruta, intento
  fusionar

  if (!is.null(ri) && !is.null(rj) && ri != rj) {

    total_load <- routes[[ri]]$load + routes[[rj]]$load

    if (total_load <= vehicle_capacity) {

      # Construir la ruta fusionada:

      new_route <- c(routes[[ri]]$route[-
length(routes[[ri]]$route)],
routes[[rj]]$route[2:length(routes[[rj]]$route)])

      # Calcular nuevo costo

      new_cost <- 0

      for (k2 in seq_len(length(new_route) - 1)) {

        new_cost <- new_cost +
distance_matrix[new_route[k2], new_route[k2 + 1]]

      }

      # Reemplazar ri con la ruta fusionada; eliminar rj

      routes[[ri]] <- list(route = new_route, load =
total_load, cost = new_cost)

      routes[[rj]] <- NULL

      routes <- routes[!sapply(routes, is.null)]

    }

  }
}

```



```

}

# 4) Fase 2: si todavía hay más rutas que max_routes, forzar
fusiones "con penalización mínima"

while (length(routes) > max_routes) {

  best_penalty <- Inf

  best_pair <- NULL      # (ri, rj)

  best_insert <- NULL    # posición donde insertamos (para
reconstruir luego)

  # Recorro cada par de rutas distintas (ri, rj) y trato de
encajar un extremo con otro:

  for (ri in seq_along(routes)) {
    for (rj in seq_along(routes)) {
      if (ri == rj) next

      seq_i <- routes[[ri]]$route
      seq_j <- routes[[rj]]$route

      load_i <- routes[[ri]]$load
      load_j <- routes[[rj]]$load

      # Vehículo es el mismo para todas, pero verificamos
carga total:

      if (load_i + load_j > vehicle_capacity) next

      # Sólo consideramos unir el "penúltimo de ri" con
"segundo de rj"

      # Esto equivale a i = seq_i[length(seq_i)-1], j =
seq_j[2]

      a <- seq_i[length(seq_i) - 1]

      b <- seq_j[2]

      d_ab <- distance_matrix[a, b]

```

```

        d_a1 <- distance_matrix[a, 1]    # a → depo (cierre
actual)

        d_1b <- distance_matrix[1, b]    # depo → b (apertura
actual)

        # Si fusionamos rutas (a -> b) en lugar de (a->1->b),
la penalización es:

        penalty <- (d_a1 + d_1b) - d_ab

        # Guardar la fusión que minimice penalty (aunque
penalty sea negativo).

        if (penalty < best_penalty) {

            best_penalty <- penalty

            best_pair    <- c(ri, rj)

            best_insert <- list(a = a, b = b)

        }

    }

}

# Si no encontramos ningún par factible (best_pair sigue
NULL), salgo del bucle

if (is.null(best_pair)) break

# 4.b) Realizo la fusión del par (ri, rj)

ri <- best_pair[1]

rj <- best_pair[2]

seq_i <- routes[[ri]]$route

seq_j <- routes[[rj]]$route

# Construir la nueva ruta: quitamos el último "1" de seq_i
y el primer "1" de seq_j

new_route <- c(seq_i[-length(seq_i)], seq_j[-1])

```

```

# Recomputar carga y costo

new_load <- routes[[ri]]$load + routes[[rj]]$load

new_cost <- 0

for (k2 in seq_len(length(new_route) - 1)) {
  new_cost <- new_cost + distance_matrix[new_route[k2],
new_route[k2 + 1]]
}

# Reemplazo ri y elimino rj

routes[[ri]] <- list(route = new_route, load = new_load,
cost = new_cost)

routes[[rj]] <- NULL

routes <- routes[!sapply(routes, is.null)]
}

# 5) Mostrar rutas finales (restamos 1 para que depósito sea
"0" en pantalla.)

total_cost <- 0

for (k in seq_along(routes)) {
  cat(sprintf(
    "Ruta %d: %s | Carga: %d | Costo: %.1f\n",
    k,
    paste(routes[[k]]$route - 1, collapse = " → "),
    routes[[k]]$load,
    routes[[k]]$cost
  ))
  total_cost <- total_cost + routes[[k]]$cost
}

```

```
cat(sprintf("Costo total: %.1f\n", total_cost))

}

# Demandas de 19 clientes

demand <- c(

  4000, # Barcelona

  1500, # Nápoles

  5000, # Lisboa

  4500, # Londres

  11000, # Róterdam

  8000, # Hamburgo

  2000, # Oslo

  2500, # Copenhagen

  3000, # Estocolmo

  2500, # Helsinki

  2000, # San Petersburgo

  3000, # Marsella

  6000, # Atenas

  1000, # Estambul

  5000, # Algeciras

  1500, # Bilbao

  1000, # Cádiz

  7000, # Casablanca

  3500 # Gdansk

)
```

```
# Matriz de distancias entre nodos (nodo 1 es el depósito)

distances <- c(

0.00,2665.27,881.14,1069.41,1499.99,2620.44,87,3063.77,3142.71
,2766.35,2327.62,3477.14,476,1917.33,1219.20,1828.29,2983.77,2
013.42,3692.80,581.20,2665.27,0.00,1874.70,2356.06,3002.86,363
5.16,2698.57,561,2409,1860,2000,2500,2844.62,3015.93,2212.76,2
890.08,2900,3061.68,2300,2084.07,881.14,1874.70,0.00,481.36,12
18.73,1760.46,914.44,2234.40,2280.24,1891.85,1615.38,2603.27,1
004.41,1141.23,338.06,1015.38,2143.53,1186.98,2816.50,299.94,1
069.41,2356.06,481.36,0.00,1407.00,2241.82,1053.19,2715.76,276
1.60,2204.86,1627.19,3084.63,718.18,931.56,819.42,1496.74,2624
.89,1198.58,3297.86,488.21,1499.99,3002.86,1218.73,1407.00,0.0
0,2958.03,1265.34,3401.36,3480.30,3103.94,2665.21,3814.73,1600
.35,2254.92,1556.79,2165.88,3321.36,2351.01,4030.39,918.79,262
0.44,3635.16,1760.46,2241.82,2958.03,0.00,2653.74,3994.86,520,
423,330,1100,2764.87,1000,2098.52,2775.84,480,700,800,2039.24,
87,2698.57,914.44,1053.19,1265.34,2653.74,0.00,3097.07,3176.01
,2799.65,2360.92,3510.44,335.01,1950.63,1252.50,1861.59,3017.0
7,2046.72,3726.10,614.50,3063.77,561,2234.40,2715.76,3401.36,3
994.86,3097.07,0.00,4514.64,1700,3849.78,2300,3238.81,2800,257
2.46,3249.78,4377.93,2200,5050.90,2482.57,3142.71,2409,2280.24
,2761.60,3480.30,520,3176.01,4514.64,0.00,554,950,400,3284.65,
1600,2618.30,2500,500,1300,400,2561.51,2766.35,1860,1891.85,22
04.86,3103.94,423,2799.65,1700,554,0.00,577.67,900,2896.26,127
3.30,2229.91,2907.23,1200,1006.28,900,2185.15,2327.62,2000,161
5.38,1627.19,2665.21,330,2360.92,2000,950,577.67,0.00,1000,234
5.37,695.63,1953.44,2000,1000,428.61,1300,1746.42,3477.14,2500
,2603.27,3084.63,3814.73,1100,3510.44,2300,400,900,1000,0.00,3
607.68,1500,2941.33,2700,400,3790.25,300,2895.94,476,2844.62,1
004.41,718.18,1600.35,2764.87,335.01,3238.81,3284.65,2896.26,2
345.37,3607.68,0.00,1649.74,1342.47,2007.64,3147.94,1916.76,38
20.91,760.55,1917.33,3015.93,1141.23,931.56,2254.92,1000,1950.
63,2800,1600,1273.30,695.63,1500,1649.74,0.00,1479.29,2156.61,
1200,267.02,2500,1336.13,1219.20,2212.76,338.06,819.42,1556.79
,2098.52,1252.50,2572.46,2618.30,2229.91,1953.44,2941.33,1342.
47,1479.29,0.00,1353.44,2481.59,1525.04,3154.56,638.00,1828.29
,2890.08,1015.38,1496.74,2165.88,2775.84,1861.59,3249.78,2500,
2907.23,2000,2700,2007.64,2156.61,1353.44,0.00,2300,1600,3000,
1247.09,2983.77,2900,2143.53,2624.89,3321.36,480,3017.07,2700,
500,1200,1000,400,3147.94,1200,2481.59,2300,0.00,1000,1200,240
2.57,2013.42,3061.68,1186.98,1198.58,2351.01,700,2046.72,2200,
1300,1006.28,428.61,1600,1916.76,267.02,1525.04,1600,1000,0.00
,1500,1432.22,3692.80,2300,2816.50,3297.86,4030.39,800,3726.10
,5050.90,400,900,1300,300,3820.91,2500,3154.56,3000,1200,1500,
0.00,3111.60,581.20,2084.07,299.94,488.21,918.79,2039.24,614.5
```

```
0,2482.57,2561.51,2185.15,1746.42,2895.94,760.55,1336.13,638.0  
0,1247.09,2402.57,1432.22,3111.60,0.00)
```

```
# Crear matriz 20x20 a partir del vector plano
```

```
distance_matrix <- matrix(distances, nrow = 20, ncol = 20,  
byrow = TRUE)
```

```
clarke_wright_max_routes(demand, distance_matrix, 28000,3)
```

Código 4.3 Técnicas de de Revisión y Evaluación de Programas (PERT).

```
# Cargar la librería

library("ProjectManagement")

# Definición de la matriz de precedencias entre actividades
portuarias:

prec1and2<-matrix(0,nrow=12,ncol=12)

# Establecemos las precedencias entre actividades.

# Cada fila i y columna j con valor 1 indica que la actividad
i debe terminar

# antes de que comience la actividad j.

prec1and2[1, 2]<-1 # A → B
prec1and2[2, 3]<-1 # B → C
prec1and2[1, 4]<-1 # A → D
prec1and2[3, 5]<-1 # C → E
prec1and2[4, 5]<-1 # D → E
prec1and2[4, 6]<-1 # D → F
prec1and2[6, 7]<-1 # F → G
prec1and2[5, 8]<-1 # E → H
prec1and2[6, 8]<-1 # F → H
prec1and2[7, 9]<-1 # G → I
prec1and2[8, 9]<-1 # H → I
prec1and2[9,10]<-1 # I → J
prec1and2[6,10]<-1 # F → J
prec1and2[10,11]<-1 # J → K
prec1and2[11,12]<-1 # K → L
```

```
# Visualización del grafo de precedencias entre actividades.  
dag.plot(prec1and2)  
  
# Definición de las duraciones de cada actividad portuaria (en  
días).  
duration<-c(1,5,2,1,2,4,1,2,1,2,3,1)  
schedule.pert(duration,prec1and2)
```


Tabla 3.2 Matriz de Distancias

	Algeciras	Atenas	Barcelona	Bilbao	Casablanca	Copenhague	Cádiz	Estambul	Estocolmo	Gdansk	Hamburgo	Helsinki	Lisboa	Londres	Marsella	Nápoles	Oslo	Rotterdam	San Petersburgo	Valencia
Algeciras	0.0	2665.27	881.14	1069.41	1499.99	2620.44	87.0	3063.77	3142.71	2766.35	2327.62	3477.14	476.0	1917.33	1219.2	1828.29	2983.77	2013.42	3692.8	581.2
Atenas	2665.27	0.0	1874.7	2356.06	3002.86	3635.16	2698.57	561.0	2409.0	1860.0	2000.0	2500.0	2844.62	3015.93	2212.76	2890.08	2900.0	3061.68	2300.0	2084.07
Barcelona	881.14	1874.7	0.0	481.36	1218.73	1760.46	914.44	2234.4	2280.24	1891.85	1615.38	2603.27	1004.41	1141.23	338.06	1015.38	2143.53	1186.98	2816.5	299.94
Bilbao	1069.41	2356.06	481.36	0.0	1407.0	2241.82	1053.19	2715.76	2761.6	2204.86	1627.19	3084.63	718.18	931.56	819.42	1496.74	2624.89	1198.58	3297.86	488.21
Casablanca	1499.99	3002.86	1218.73	1407.0	0.0	2958.03	1265.34	3401.36	3480.3	3103.94	2665.21	3814.73	1600.35	2254.92	1556.79	2165.88	3321.36	2351.01	4030.39	918.79
Copenhague	2620.44	3635.16	1760.46	2241.82	2958.03	0.0	2653.74	3994.86	520.0	423.0	330.0	1100.0	2764.87	1000.0	2098.52	2775.84	480.0	700.0	800.0	2039.24
Cádiz	87.0	2698.57	914.44	1053.19	1265.34	2653.74	0.0	3097.07	3176.01	2799.65	2360.92	3510.44	335.01	1950.63	1252.5	1861.59	3017.07	2046.72	3726.1	614.5
Estambul	3063.77	561.0	2234.4	2715.76	3401.36	3994.86	3097.07	0.0	4514.64	1700.0	3849.78	2300.0	3238.81	2800.0	2572.46	3249.78	4377.93	2200.0	5050.9	2482.57
Estocolmo	3142.71	2409.0	2280.24	2761.6	3480.3	520.0	3176.01	4514.64	0.0	554.0	950.0	400.0	3284.65	1600.0	2618.3	2500.0	500.0	1300.0	400.0	2561.51
Gdansk	2766.35	1860.0	1891.85	2204.86	3103.94	423.0	2799.65	1700.0	554.0	0.0	577.67	900.0	2896.26	1273.3	2229.91	2907.23	1200.0	1006.28	900.0	2185.15
Hamburgo	2327.62	2000.0	1615.38	1627.19	2665.21	330.0	2360.92	2000.0	950.0	577.67	0.0	1000.0	2345.37	695.63	1953.44	2000.0	1000.0	428.61	1300.0	1746.42
Helsinki	3477.14	2500.0	2603.27	3084.63	3814.73	1100.0	3510.44	2300.0	400.0	900.0	1000.0	0.0	3607.68	1500.0	2941.33	2700.0	400.0	3790.25	300.0	2895.94
Lisboa	476.0	2844.62	1004.41	718.18	1600.35	2764.87	335.01	3238.81	3284.65	2896.26	2345.37	3607.68	0.0	1649.74	1342.47	2007.64	3147.94	1916.76	3820.91	760.55
Londres	1917.33	3015.93	1141.23	931.56	2254.92	1000.0	1950.63	2800.0	1600.0	1273.3	695.63	1500.0	1649.74	0.0	1479.29	2156.61	1200.0	267.02	2500.0	1336.13
Marsella	1219.2	2212.76	338.06	819.42	1556.79	2098.52	1252.5	2572.46	2618.3	2229.91	1953.44	2941.33	1342.47	1479.29	0.0	1353.44	2481.59	1525.04	3154.56	638.0
Nápoles	1828.29	2890.08	1015.38	1496.74	2165.88	2775.84	1861.59	3249.78	2500.0	2907.23	2000.0	2700.0	2007.64	2156.61	1353.44	0.0	2300.0	1600.0	3000.0	1247.09
Oslo	2983.77	2900.0	2143.53	2624.89	3321.36	480.0	3017.07	2700.0	500.0	1200.0	1000.0	400.0	3147.94	1200.0	2481.59	2300.0	0.0	1000.0	1200.0	2402.57
Rotterdam	2013.42	3061.68	1186.98	1198.58	2351.01	700.0	2046.72	2200.0	1300.0	1006.28	428.61	1600.0	1916.76	267.02	1525.04	1600.0	1000.0	0.0	1500.0	1432.22
San Petersburgo	3692.8	2300.0	2816.5	3297.86	4030.39	800.0	3726.1	5050.9	400.0	900.0	1300.0	300.0	3820.91	2500.0	3154.56	3000.0	1200.0	1500.0	0.0	3111.6
Valencia	581.2	2084.07	299.94	488.21	918.79	2039.24	614.5	2482.57	2561.51	2185.15	1746.42	2895.94	760.55	1336.13	638.0	1247.09	2402.57	1432.22	3111.6	0.0