

Proyecto Final de Diseño y Análisis de Algoritmos

Problema: “El Comerciante Holandés”

Claudia Hernández Pérez Joel Aparicio Tamayo

9 de diciembre de 2025

Resumen

Este documento presenta el análisis completo del problema “*El Comerciante Holandés*”.

1. Definición del problema

La prestigiosa Compañía Holandesa de las Indias Orientales, en su afán por dominar el comercio mundial, se enfrenta a un desafío monumental. Un capitán experimentado, al mando de una de sus valiosas flotas, debe emprender una expedición comercial que partirá de Ámsterdam y, tras recorrer los puertos más lucrativos del Viejo y Nuevo Continente, deberá regresar a su puerto de origen.

Los inversores de la Compañía han proporcionado un capital inicial considerable y han establecido un plazo máximo para la duración de la expedición. El capitán tiene la libertad de elegir qué puertos visitar y en qué orden, con la única condición de no visitar el mismo puerto dos veces en el mismo viaje (por cuestiones de acuerdos comerciales y evitar saturación del mercado).

En cada puerto, el capitán encontrará una lista de mercancías disponibles, con sus respectivos precios de compra y venta (que pueden variar significativamente). El capitán puede vender las mercancías que lleva a bordo y comprar nuevas. Sin embargo, debe ser astuto:

- La capacidad de carga de su barco es limitada, por lo que no puede llevar más de lo que su bodega permite.

- No es necesario vender todas las mercancías al llegar a un puerto; el capitán puede decidir retener parte de su cargamento si cree que podrá venderlo a un precio más alto en un puerto posterior.
- Debe asegurarse de que, después de cada operación de compra, le quede suficiente dinero para cubrir los salarios de la tripulación, los impuestos portuarios y las posibles reparaciones del barco hasta el siguiente destino.
- El tiempo es oro; la duración total del viaje, incluyendo el tiempo de navegación entre puertos, no debe exceder el plazo fijado por los inversores.

El objetivo del capitán es claro: planificar la ruta y las transacciones en cada puerto de tal manera que, al regresar a Ámsterdam, el capital final de la expedición sea el máximo posible, superando con creces la inversión inicial.

2. Formalización del Problema

Antes de abordar cualquier solución algorítmica, es fundamental establecer una representación matemática precisa del problema. Esta sección presenta nuestra formalización del Comerciante Holandés, definiendo las estructuras de datos de entrada, las restricciones que deben cumplirse y la función objetivo a optimizar. Este modelo elimina las ambigüedades de la descripción narrativa original.

2.1. Definiciones Matemáticas

Sea:

- $G = (V, E)$: grafo completo de n puertos, donde $V = \{0, 1, \dots, n - 1\}$ (0 representa Ámsterdam, punto de inicio y fin).
- d_{ij} : tiempo de navegación entre puertos i y j . Se asume:
 - $d_{ii} = 0, \quad \forall i$
 - $d_{ij} = d_{ji}, \quad \forall i, j$
 - $d_{ij} \leq d_{ik} + d_{kj}, \quad \forall i, k, j$
- T_{max} : tiempo máximo total permitido para la expedición.
- C_{max} : capacidad máxima de carga del barco.

- K_0 : capital inicial proporcionado por los inversores.
- K_{min} : capital mínimo que debe mantenerse tras cada transacción (para gastos operativos).

En cada puerto $i \in V$:

- $M_i = \{(w_{ik}, p_{ik}^{compra}, p_{ik}^{venta})\}_{k=1}^{m_i}$: conjunto de m_i mercancías disponibles, donde:
 - $w_{ik} \in \mathbb{R}$: peso de la mercancía k en puerto i
 - $p_{ik}^{compra} \in \mathbb{R}$: precio de compra de la mercancía k en puerto i
 - $p_{ik}^{venta} \in \mathbb{R}$: precio de venta de la mercancía k en puerto i
- **Importante:** No hay garantía de que $p^{venta} > p^{compra}$. Puede haber mercancías con pérdida potencial, lo que obliga al capitán a tomar decisiones estratégicas.

2.2. Estado del Sistema

El estado al llegar a un puerto i se define como una tupla:

$$S_i = (K_i, L_i, R_i, t_i)$$

donde:

- $K_i \in \mathbb{R}$: capital disponible al llegar al puerto i .
- $L_i \subseteq \{m = (i', k, w, p^{compra}) : i', k \in \mathbb{N}; w, p^{compra} \in \mathbb{R}\}$: conjunto de mercancías a bordo, donde cada mercancía está identificada por:
 - i' : puerto donde fue comprada
 - k : índice de la mercancía en ese puerto
 - w : peso de la mercancía
 - p^{compra} : precio pagado por la mercancía
- $R_i \subseteq V$: conjunto de puertos ya visitados (ruta parcial).
- $t_i \in \mathbb{R}^+$: tiempo acumulado hasta llegar al puerto i .

2.3. Operaciones

Al llegar a un puerto i , el capitán realiza secuencialmente:

1. **Venta parcial:** Para cada mercancía $g = (i_g, k_g, w_g, p_g^{compra}) \in L_i$, el capitán decide si venderla en el puerto actual i obteniendo p_{i,k_g}^{venta} (precio de venta de esa mercancía en puerto i). Sea $M'_i \subseteq L_i$ el subconjunto de mercancías vendidas:

$$K_i \leftarrow K_i + \sum_{g \in M'_i} p_{i,k_g}^{venta}$$

Las mercancías vendidas se eliminan de la carga: $L_i \leftarrow L_i \setminus M'_i$.

2. **Compra selectiva:** Considerando las mercancías disponibles en M_i , el capitán selecciona un subconjunto $C_i \subseteq M_i$ para comprar, sujeto a:

- **Restricción financiera:**

$$K_i - \sum_{(w, p^{compra}, p^{venta}) \in C_i} p^{compra} \geq K_{min}$$

- **Restricción de capacidad:**

$$\sum_{g \in L_i} w_g + \sum_{(w, p^{compra}, p^{venta}) \in C_i} w \leq C_{max}$$

Las mercancías compradas se añaden a bordo:

$$\forall (w, p^{compra}, p^{venta}) \in C_i : \quad L_i \leftarrow (i, k, w, p^{compra})$$

Luego se actualiza el capital:

$$K_i \leftarrow K_i - \sum_{(w, p^{compra}, p^{venta}) \in C_i} p^{compra}$$

3. **Selección de próximo destino:** Sea $j \in V \setminus (R_i \cup \{i\})$ un candidato a próximo puerto, j es seleccionable si:

$$t_i + d_{ij} + d_{j0} \leq T_{max}$$

Si ningún j cumple esta condición, el capitán debe regresar a Ámsterdam ($j = 0$). Luego de seleccionar el próximo destino, se actualiza el recorrido actual:

$$R_i \leftarrow R_i \cup \{j\}$$

2.4. Función Objetivo

$$\max K_f ,$$

donde K_f es el capital tras vender cualquier carga restante en Ámsterdam (a precios de Ámsterdam).

sujeto a:

1. **Ruta simple:** Sea $\pi = (v_0, v_1, \dots, v_m, v_{m+1})$ la secuencia de puertos visitados, donde $v_0 = v_{m+1} = 0$ (Ámsterdam). Entonces:

$$\forall i \in \{1, \dots, m\} : v_i \in V \setminus \{0\} \quad \wedge \quad \forall i \neq j \in \{1, \dots, m\} : v_i \neq v_j$$

Es decir, ningún puerto no-inicial se repite en la ruta.

2. **Tiempo límite:** $t_f \leq T_{max}$, siendo t_f el tiempo de llegada a Ámsterdam al finalizar el viaje.
3. **Capital suficiente para operatividad:** $\forall i, K_i \geq K_{min}$.
4. **Capacidad:** $\forall i, \sum_{g \in L_i} w_g \leq C_{max}$.

3. Análisis de Complejidad Computacional

El problema es presentado originalmente como un problema de optimización, cuyo objetivo es hallar una ruta y una estrategia de transacciones que maximicen el capital final K_f . Para demostrar su clasificación en la jerarquía de complejidad, se define su versión de decisión asociada:

Dada la instancia del problema descrita en la Sección 2 y un umbral de capital $K_{objetivo} \in \mathbb{R}$, ¿existe una secuencia de puertos y transacciones tales que el tiempo total no exceda T_{max} y el capital al regresar a Ámsterdam sea al menos $K_{objetivo}$?

A continuación, se demuestra que este problema es **NP-Completo**.

3.1. Pertenencia a la Clase NP

Sea un certificado de solución una secuencia de puertos $\pi = (v_0, v_1, \dots, v_m, v_{m+1})$ y una secuencia de conjuntos de transacciones (compras C_i y ventas M'_i) para cada puerto visitado. La verificación del certificado consiste en:

1. **Verificación de ruta:** Comprobar que π es una ruta simple (excepto por v_0 y v_{m+1}) y que el tiempo total de navegación cumple $\sum d_{v_i, v_{i+1}} \leq T_{max}$. Esta operación se realiza en $O(m)$ mediante un vector de presencia para los nodos.
2. **Verificación de carga:** En cada puerto v_i , se calcula el peso total sumando las mercancías retenidas en la bodega L_i después de vender y las nuevas adquisiciones $C_i \subseteq M_i$, verificando que $\sum_{g \in L_i} w_g + \sum_{C_i} w \leq C_{max}$. La complejidad es $O(m \cdot \max |L_i|)$, lo cual es polinomial respecto a la cantidad de mercancías.
3. **Verificación financiera:** Simular el flujo de caja K_i sumando ingresos por ventas M'_i y restando costos de compra C_i . Se verifica que en todo momento $K_i \geq K_{min}$ y que al regresar a Ámsterdam el capital final $K_f \geq K_{objetivo}$.

Como todas las operaciones de verificación se realizan en tiempo polinomial respecto al tamaño de la entrada, el problema **pertenece a NP**.

3.2. NP-Compleitud mediante Reducción de TSP

Para demostrar la NP-Compleitud, se reduce una instancia del problema de decisión de **TSP** (NP-Completo) a una instancia del problema.

Definición de TSP: Dado un grafo $G_{TSP} = (V', E', \omega)$ y una distancia límite L , ¿existe un tour que visite todos los nodos de V' tal que la distancia recorrida sea a lo sumo L ?

$\omega(e_{ij})$: distancia entre nodos i y j .

3.2.1. Construcción de la reducción:

Dada una instancia de **TSP**, se construye una instancia del problema de la siguiente forma:

- **Grafo y Tiempos:** $V = V'$, $d_{ij} = \omega(E'_{ij})$, y se fija $T_{max} = L$.
- **Incentivo a la Visita:** En cada puerto $i \in V \setminus \{0\}$, se define una única mercancía $m_i \in M_i$ con $w = 0$ y $p_{compra} = 0$. Su precio de venta es 0 en todo puerto, excepto en Ámsterdam (v_0), donde $p_{venta} = 1$.
- **Condiciones Iniciales:** $K_0 = 0$, $K_{min} = 0$, $C_{max} = \infty$.
- **Meta de Ganancia:** Se define la meta de capital final como $K_{objetivo} = |V| - 1$.

3.2.2. Demostración de equivalencia:

- (\Rightarrow) Si **TSP** tiene un tour de longitud de a lo sumo L , el capitán puede seguir dicho tour visitando cada puerto una vez, recoger las $|V| - 1$ mercancías y volver a Ámsterdam dentro del tiempo T_{max} . Al venderlas, obtiene $K_f = |V| - 1$, satisfaciendo el problema.
- (\Leftarrow) Si el problema tiene solución con capital de al menos K , el capitán necesariamente visitó todos los puertos para recolectar el capital (dada la ausencia de otros beneficios). Como lo hizo en un tiempo menor de T_{max} , el recorrido representa un tour válido para **TSP**.

La reducción es ejecutable en tiempo polinomial y dado que **TSP** se reduce al problema, concluimos que es **NP-Completo**.