Proyecto de Diseño y Análisis de Algoritmos: Problema de Logística Discreta (DLP)

Salma Fonseca Curbelo C-412, José Ernesto Morales Lazo C-412

3 de enero de 2026

1. Formalización del Modelo de Logística Discreta

1.1. Estructuras de Datos de Entrada

- **Mulas (M):** Un arreglo `mulas[]` de tamaño n, donde cada posición i contiene la capacidad c_i.
  - $c_i \in \mathbb{R}^+$: Capacidad de carga de la mula m_i.
- **Artículos (A):** Un arreglo `articulos[]` de tamaño k, donde cada posición j contiene una tupla $\langle w_j, v_j \rangle$.
  - $w_j \in \mathbb{R}^+$: Peso del artículo a_j.
  - $v_j \in \mathbb{R}^+$: Valor del artículo a_j.

1.2. Variables de Decisión

Definimos la matriz binaria de asignación:

$$x_{ij} = \begin{cases} 1 & \text{si el artículo } j \text{ se asigna a la mula } i \\ 0 & \text{en otro caso} \end{cases}$$

$\forall i \in \{1, \ldots, n\}, \forall j \in \{1, \ldots, k\}$.

1.3. Modelo Matemático (Linealizado)

Para que el modelo sea procesable por un computador, linealizamos la función objetivo utilizando dos variables auxiliares: V_{max} (valor de la mula más cargada) y V_{min} (valor de la mula menos cargada).

Función Objetivo:

$$\text{Minimizar } Z = V_{max} - V_{min}$$

Sujeto a (Restricciones):

1. **Cálculo de Valores por Mula:** El valor total V_i transportado por la mula i se define como:

$$V_i = \sum_{j=1}^{k} v_j x_{ij} \quad \forall i \in \{1, \ldots, n\}$$

Y debe cumplirse que:

$$V_{min} \leq V_i \leq V_{max} \quad \forall i \in \{1, \ldots, n\}$$

2. **Asignación Única (Partición):** Cada artículo j debe asignarse exactamente a una mula.
$$\sum_{i=1}^{n} x_{ij} = 1 \quad \forall j \in \{1, \ldots, k\}$$

3. **Capacidad de Peso:** La carga de la mula i no debe exceder su capacidad individual c_i.
$$\sum_{j=1}^{k} w_j x_{ij} \leq c_i \quad \forall i \in \{1, \ldots, n\}$$

4. **Integridad de Variables:**
$$x_{ij} \in \{0, 1\}, \quad V_{max}, V_{min} \geq 0$$

1.4. Propiedades de la Salida

- **Factibilidad:** Si no existe ninguna combinación de x_{ij} que satisfaga simultáneamente la capacidad de peso de todas las mulas y la asignación de todos los artículos, el sistema debe informar que la instancia es Irresoluble.
- **Optimidad:** La solución devuelta debe garantizar que no existe otra configuración donde la brecha de riesgo ($V_{max} - V_{min}$) sea menor.
- **Formato de Salida:** Una lista de conjuntos $S = \{M_1, M_2, \ldots, M_n\}$, donde cada M_i contiene los índices j de los artículos tales que $x_{ij} = 1$.

2. Análisis de Complejidad Computacional

2.1. Demostración de que el Problema de la Partición es NP Completo

El Problema de la Partición: Dado un multiconjunto S de enteros positivos , ¿es posible dividir S en dos subconjuntos S_1 y S_2 tales que la suma de los elementos en S_1 sea igual a la suma de los elementos en S_2?

2.1.1. Demostración de que el problema de la partición es NP:

Un problema está en la clase NP si una solución propuesta (un çertificado") puede ser verificada en tiempo polinomial.

1. **Certificado:** Una lista de los elementos que pertenecen al subconjunto S_1.
2. **Verificación:** Un algoritmo simplemente suma los elementos de S_1 y suma los elementos de $S \setminus S_1$ (que sería S_2). Luego compara si las sumas son iguales.
3. **Tiempo:** La suma y la comparación son operaciones lineales. Por lo tanto, el problema de la partición es NP.

2.1.2. Demostración de que el problema de la partición es NP-Completo:

Se usará el problema de la Suma de Subconjuntos (Subset-Sum), el cual se vió en clase que es NP-Completo.

Subset-Sum: Dado un conjunto de enteros A y un entero objetivo t, ¿existe un subconjunto $A' \subseteq A$ tal que la suma de sus elementos sea t?

Reducción Subset-Sum $\leq_p$ Partición

Se debe transformar cualquier instancia de Subset-Sum en una instancia del problema de la Partición en tiempo polinomial.

1. **Construcción:** Sea una instancia de Subset-Sum con conjunto A y objetivo t. Se calcula la suma total de A, denotada como $\Sigma = \Sigma_{i=1}^{n} a_i$. Se construye un nuevo conjunto S para el problema partición agregando dos elementos adicionales especiales, J y K, al conjunto A.
   - $J = 2\Sigma - t$
   - $K = \Sigma + t$
   - $S = A \cup \{J, K\}$

2. **Equivalencia:** La suma total de los elementos en el nuevo conjunto es:

   $$\mathrm{Sum}(S) = \Sigma + (2\Sigma - t) + (\Sigma + t) = 4\Sigma$$

   Para que exista una partición válida en S, se debe dividir S en dos subconjuntos, cada uno sumando exactamente la mitad del total, es decir, 2Σ.

   Cómo se puede llegar a 2Σ:

   - Los elementos J y K no pueden estar en el mismo subconjunto porque $J+K = 3\Sigma$, lo cual excede el objetivo 2Σ.
   - Por lo tanto, J debe estar en un subconjunto (S_1) y K en el otro (S_2).
   - Para que S_1 sume 2Σ, debe contener a J más algunos elementos de A (A_{sub}) tal que:

     $$J + \Sigma A_{sub} = 2\Sigma$$
     $$(2\Sigma - t) + \Sigma A_{sub} = 2\Sigma$$
     $$\Sigma A_{sub} = t$$

Conclusión: Existe una partición válida en S si y solo si existe un subconjunto en A que sume exactamente t. Como la transformación es polinomial, el problema de la Partición es NP-Completo.

2.2. Demostración de que el Problema de la DLP es NP Completo

2.2.1. Demostración de que DLP es NP:

1. **Certificado:** Una asignación propuesta de cada artículo a una mula específica.

2. **Verificación:** Un algoritmo verificador calcula la suma de pesos de cada una de las m mulas y comprueba si excede el límite de peso C. Luego calcula la suma de valores V_j para cada mula y compara para cada par de mulas V_a y V_b y verifica si $|V_a - V_b| \leq K$.

3. **Tiempo:** Las sumas son operaciones lineales $O(n)$ y como hay $m(m-1)/2$ pares de mulas y $m \leq n$, la compración es $O(n^2)$, por lo que se puede verificar en tiempo polinomial.

Como la verificación se realiza en tiempo polinomial, DLP es NP.

2.2.2. Demostración que DLP es NP-difícil

Se usará el problema de la Partición, el cual se demostró anteriormente que es NP-Completo.

Reducción Partición $\leq_p$ DLP

Se tomará una instancia arbitraria de Partición con el conjunto S. Sea $W_{total} = \Sigma_{i=1}^{n} x_i$. Se construye una instancia de DLP con los siguientes parámetros específicos:

1. **Artículos:** Para cada número $x_i \in S$, se crea un artículo i.

2. **Pesos y Valores:** Se define tanto el peso como el valor igual al número original: $w_i = x_i, v_i = x_i$.

3. **Número de Mulas (m):** Se fija $m = 2$.

4. **Diferencia permitida (K):** Se fija $K = 0$.

5. **Capacidad (C):** Se fija $C = W_{total}/2$. (Si W_{total} es impar, no hay solución para Partición ni para este DLP, así que se asume par).

Demostración de Equivalencia:

Si Partición tiene solución: Existen S_1 y S_2 tales que $\Sigma S_1 = \Sigma S_2 = W_{total}/2$. Se asignan los artículos correspondientes a S_1 a la Mula 1 y los de S_2 a la Mula 2. Verificación de Peso: La Mula 1 lleva peso $W_{total}/2$, que es $= C$. La Mula 2 igual. (Cumple). Verificación de Balance: El valor en la Mula 1 es $W_{total}/2$. El valor en la Mula 2 es $W_{total}/2$. La diferencia es $|W_{total}/2 - W_{total}/2| = 0$. Como $0 \leq K$ (donde $K = 0$), cumple la condición. Por tanto, DLP devuelve SÍ.

Si DLP devuelve SÍ: Significa que existe una distribución en 2 mulas tal que la diferencia de valores es ≤ 0. Como el valor absoluto no puede ser negativo, la diferencia debe ser exactamente 0.

$$\Sigma_{i \in m1} v_i = \Sigma_{i \in m2} v_i$$

Dado que se definió $v_i = x_i$, esto implica:

$$\Sigma_{i \in m1} x_i = \Sigma_{i \in m2} x_i$$

Esto constituye una partición válida del conjunto original en dos sumas iguales.

Conclusión: Se ha demostrado que el problema Partición es un caso particular del problema DLP (específicamente cuando $m = 2, w_i = v_i$ y $K = 0$). Dado que el caso particular es NP-Completo, el caso general es al menos igual de difícil.

Por lo tanto, el Problema de Logística Discreta (DLP) es NP-Completo.