Análisis de Algoritmos
Tarea 2

17 de junio de 2022

Ejercicios

- Peso: 10 puntos. Considera el problema de la mochila fraccional. Entrada: una capacidad de mochila C y una secuencia de n parejas $item_i = (v,w_i)$, de tal forma que el elemento iésimo tiene valor total v_i, y su peso total es w_i. Salida: un vector $q_1, \ldots, q_n$, con entradas en $[0, 1]$, que indican las fracciones del total de los respectivos elementos, de tal forma que:

- La suma de los pesos de las fracciones de los elementos es compatible con la capacidad de la mochila ($\sum_{i=1}^{n} q_i \cdot w_i \leq C$), y
- La suma de las fracciones de los valores de los elementos es la máxima posible.

1. Enuncia y demuestra un lema de intercambio, que te permita transformar toda solución no óptima A en una solución de mejor valor A'.

   Respuesta:

   Supongamos sin pérdida de la generalidad que las parejas (v_i, w_i) están ordenadas de tal forma que

   $$\frac{v_1}{w_1} > \frac{v_2}{w_2} > \ldots > \frac{v_i}{w_i}$$

   Lema 1. Sea $A = \langle q_1, \ldots, q_n \rangle$ una solución no óptima al problema y sean i, j tales que $i < j$, $q_i < 1$, $q_j > 0$. Entonces existe otra solución $A' = \langle q_1', \ldots, q_n' \rangle$ tal que $V(A') > V(A)$.

   Demostración.

   Sea $A = \langle q_1, \ldots, q_n \rangle$ una solución no óptima al problema y sean i, j tales que $i < j$, $q_i < 1$, $q_j > 0$.

   Procedemos por casos:

   - <u>*Caso 1:*</u> Si $w_j q_j \leq w_i(1 - q_i)$, entonces

     $$A' = \langle q_1, \ldots, q_n \rangle$$

     donde:

     $$q_k' = q_k \forall k \neq i, j$$
     $$q_i' = q_i + \left(\frac{w_j}{w_i} \right) q_j$$
     $$q_j' = 0$$

     Básicamente, estamos pasando al término i todo el peso ocupado por el elemento j, de tal manera que q_j' queda en cero.

     Ahora debemos mostrar que todos los elementos de A' son valores en $[0, 1]$, entonces es claro que como $q'_k = q_k \forall k \neq i,j$, entonces $q'_k \in [0,1] \forall k \neq i,j$, además como $q'_i = q_i + \left(\frac{w_j}{w_i}\right) q_j$ y $q_i, q_j \in [0,1], w_i, w_j > 0$, entonces $q'_i \geq 0$. Además:

$$\begin{aligned}
w_j q_j &\leq w_i(1-q_i) \\
\left(q_i + \frac{1}{w_i}\right) w_j q_j &\leq \left(q_i + \frac{1}{w_i}\right) w_i(1-q_i) \\
q_i + \frac{w_j}{w_i} q_j &\leq q_i + \frac{w_i(1-q_i)}{w_i} \\
q_i + \frac{w_j}{w_i} q_j &\leq q_i + (1-q_i) \\
q_i + \frac{w_j}{w_i} q_j &\leq 1 \\
q'_i &\leq 1
\end{aligned}$$

Por lo que $q'_i \in [0,1]$, y como $q'_j = 0$, entonces efectivamente todos los elementos de A' son valores en $[0,1]$.

Ahora debemos mostrar que:

$$\sum_{t=1}^{n} w_t q'_t \leq C$$

entonces:

$$\begin{aligned}
\sum_{t=1}^{n} w_t q'_t &= \left(\sum_{t \neq i,j} w_t q'_t\right) + w_i q'_i + w_j q'_j \\
&= \left(\sum_{t \neq i,j} w_t q_t\right) + w_i \left(q_i + \left(\frac{w_j}{w_i}\right) q_j\right) + w_j \cdot 0 \\
&= \left(\sum_{t \neq i,j} w_t q_t\right) + w_i \left(q_i + \left(\frac{w_j}{w_i}\right) q_j\right) \\
&= \left(\sum_{t \neq i,j} w_t q_t\right) + w_i q_i + \left(w_i \left(\frac{w_j}{w_i}\right) q_j\right) \\
&= \left(\sum_{t \neq i,j} w_t q_t\right) + w_i q_i + w_j q_j \\
&= \sum_{t=1}^{n} w_t q_t \leq C
\end{aligned}$$

Ahora solo nos falta mostrar que A' es mejor solución que A, entonces se tiene que:

$$\begin{aligned}
V(A') - V(A) &= (v_1 q_1' + \cdots + v_n q_n') - (v_1 q_1 + \cdots + v_n q_n) \\
&= v_i q_i' + v_j q_j' - v_i q_i - v_j q_j \\
&= v_i \left(q_i + \left(\frac{w_j}{w_i} \right) q_j \right) + v_j \cdot 0 - v_i q_i - v_j q_j \\
&= v_i \left(q_i + \left(\frac{w_j}{w_i} \right) q_j \right) - v_i q_i - v_j q_j \\
&= v_i q_i + v_i \left(\frac{w_j}{w_i} \right) q_j - v_i q_i - v_j q_j \\
&= v_i \left(q_i + \left(\frac{w_j}{w_i} \right) q_j - q_i \right) - v_j q_j \\
&= v_i \left(\frac{w_j}{w_i} \right) q_j - v_j q_j \\
&= \left(\frac{v_i}{w_i} \right) w_j q_j - v_j q_j
\end{aligned}$$

Por otro lado ya que $i < j$:

$$\begin{aligned}
\frac{v_i}{w_i} &> \frac{v_j}{w_j} \\
\frac{v_i}{w_i} w_j q_j &> \frac{v_j}{w_j} w_j q_j \\
\frac{v_i}{w_i} w_j q_j &> v_j q_j \\
\frac{v_i}{w_i} w_j q_j - v_j q_j &> v_j q_j - v_j q_j \\
\frac{v_i}{w_i} w_j q_j - v_j q_j &> 0
\end{aligned}$$

Por lo tanto:

$$\begin{aligned}
V(A') - V(A) &> 0 \\
V(A') &> V(A)
\end{aligned}$$

Por lo tanto A' es una solución de mejor valor que A.

- _Caso 2:_ Si $w_j q_j > w_i(1 - q_i)$, entonces

$$A' = \langle q_1, \ldots, q_n \rangle$$

donde:

$$\begin{aligned}
q_k' &= q_k \forall k \neq i, j \\
q_i' &= 1 \\
q_j' &= q_j + \frac{w_i}{w_j}(1 - q_i)
\end{aligned}$$

Básicamente, estamos pasando al término i algo del peso ocupado por el elemento j, de tal manera que q_j' resulta en una menor fracción y q_i' queda en 1.

Se verifica de manera análoga al caso anterior que todos los elementos de A' son valores en $[0, 1]$, que $\sum_{t=1}^{n} w_t q_t' \leq C$ y que $V(A') > V(A)$.

Por lo tanto como en ambos casos llegamos a una solución A' mejor que A, el lema es verdadero. ∎

2. Demuestra con el lema anterior, que la solución es única (usando la hipótesis de que los valores unitarios no se repiten).

   Demostración.

   Sea $A = \langle q_1, \ldots, q_n \rangle$ una solución no óptima al problema y sean i, j tales que $i < j$, $q_i < 1, q_j > 0$. Entonces usando el lema demostrado anteriormente (que ya toma como hipótesis que los valores unitarios no se repiten) se tiene que existe otra solución $X = \langle x_1, \ldots x_n \rangle$ tal que $V(X) > V(A)$. Ahora supongamos que existe otra solución $Y = \langle y_1, \ldots y_n \rangle$ tal que $V(Y) = V(X) > V(A)$.
   Entonces se tienen dos casos:

   - _Caso 1:_ Si $w_j q_j \leq w_i(1 - q_i)$, entonces por el lema se tiene que:

$$\begin{aligned} x_k &= q_k \forall k \neq i, j \\ x_i &= q_i + \left(\frac{w_j}{w_i}\right) q_j \\ x_j &= 0 \\ y_k &= q_k \forall k \neq i, j \\ y_i &= q_i + \left(\frac{w_j}{w_i}\right) q_j \\ y_j &= 0 \end{aligned}$$

     por lo que:

$$\begin{aligned} x_k &= y_k \forall k \neq i, j \\ x_i &= y_i \\ x_j &= y_j \end{aligned}$$

     es decir,

$$x_t = y_t \forall t \in \{1, n\}$$

     por lo que en este caso:

$$X = Y \qquad \text{(ambas soluciones son la misma)}$$

   - _Caso 2:_ Si $w_j q_j > w_i(1 - q_i)$.

     La demostración de este caso es análoga a la del primero, al final se llega a que $X = Y$.

   Por lo tanto como en ambos casos llegamos a que $X = Y$, entonces la solución es única. ∎

3. Enuncia y demuestra un lema que caracterice la estructura de la solución única (que es la única a la que ya no se le puede aplicar un intercambio que la mejore).

   Respuesta:

   Supongamos sin pérdida de la generalidad que las parejas (v_i, w_i) están ordenadas de tal forma que

$$\frac{v_1}{w_1} > \frac{v_2}{w_2} > \ldots > \frac{v_i}{w_i}$$

   Lema 2. La solución única al problema es aquella de la forma $S = \langle q_1, \ldots, q_i, \ldots, q_n \rangle$ donde $\forall j < i(q_j = 1)$, $\forall k > i(q_k = 0)$ y q_i es el mayor valor posible entre 0 y 1 que se puede tomar del item i.

   Demostración.

Sea A una solución no óptima del problema, entonces podemos aplicar el lema de intercambio mostrado anteriormente tantas veces hasta que tengamos una solución de la forma $S = \langle 1, \ldots, 1, q_i, 0, \ldots, 0 \rangle$, donde q_i es el mayor valor posible entre 0 y 1 que se puede tomar del item i, además es una solución a la cual no podremos aplicar más el lema de intercambio puesto que este requiere que haya mas de un elemento en la solución que no sea 0 ni 1. Además por el segundo lema podemos afirmar que esta solución es única. Por lo tanto $S = \langle q_1, \ldots, q_i, \ldots, q_n \rangle$ donde $\forall j < i(q_j = 1)$ y $\forall k > i(q_k = 0)$ es la única solución al problema. ■

4. Propón un algoritmo que, apoyándose en el lema anterior, dada una instancia del problema, construya la solución óptima, en tiempo $O(n \log n)$.

Algoritmo 1 Algoritmo iterativo que resuelve el problema de la mochila fraccional

```
mochila-fraccional (listaDeTuplas[(v,w)|v es el valor, w es el peso], C)

Crear un arreglo s[] de tamaño n con todas las entradas 0.

Ordenar la lista de tuplas de mayor a menor valor unitario.

peso = 0
for i = 1 to n
   //listaDeTuplas.get(i).getPeso() devuelve el peso del item i.
   if peso + listaDeTuplas.get(i).getPeso() <= C then
      s[i] = 1
      peso = peso + listaDeTuplas.get(i).getPeso()
   else
      s[i] = (C - peso) / listaDeTuplas.get(i).getPeso()
      peso = C
      break
return s
```

- Demuestra que el algoritmo es correcto.

  Demostración.

  Nuestro algoritmo va tomando el peso total de cada item si es que aún cabe en la mochila (es decir tomamos 1), así hasta que tomamos solo una fracción del peso de un item i que ya no cabe completo en la mochila, y los demás elementos (desde el $i + 1$ en adelante) no los tomamos, es decir tomamos 0 de tales items. Además por como está definido nuestro algoritmo, nos aseguramos de tomar la mayor fracción posible del peso de ese item i. Por lo tanto la salida de nuestro algoritmo será un arreglo de la forma `s = [1,1,...,`q_i`,0,..,0]`, entonces gracias al lema 2 que a su vez se apoya del lema 1, y a que nuestro algoritmo se encarga de ordenar el arreglo de parejas de mayor a menor valor unitario desde el inicio, entonces nuestro algoritmo devuelve la solución óptima al problema y por lo tanto es correcto.

- Demuestra que su complejidad es la requerida.

  Demostración.

  Nuestro algoritmo crea un arreglo de tamaño n lo cual le toma tiempo $O(n)$, además realiza un ciclo for desde 1 hasta n, que también le toma tiempo $O(n)$, sin embargo en el peor de los casos, el arreglo `parejas` no estará ordenado como se desea, entonces el algoritmo ordenará el arreglo de mayor a menor valor unitario que le tomará un tiempo no menor a $O(n \log n)$. Por lo tanto esa es la complejidad de nuestro algoritmo.