MA4702 Programación Lineal Mixta: Teoría y Laboratorio.
Profesor: Martín Matamala.
Auxiliares: Benjamín Jauregui y Cristian Palma.
Fecha: 29 de abril de 2022.

Laboratorio 3: Tarea Individual

Sea G un digrafo $G = (V, E)$, $c\colon E \to \mathbb{R}_+$ una función de costos sobre los arcos y $s, t \in V$. Considere el problema de enviar d unidades de flujo de s hacia t en G, con costos dados por la función c y capacidades $u_e \geq 0$ para todo arco $e \in E$. Este problema se puede modelar como una problema de flujo de valor d dado por

$$
\begin{aligned}
(F) \quad \text{mín} \quad & \sum_{e\in E} c_e f_e \\
f(\delta^+(s)) - f(\delta^-(s)) &= d, \\
f(\delta^+(i)) - f(\delta^-(i)) &= 0, \qquad \forall i \in V - s - t. \\
f_e &\leq u_e, \qquad \forall e \in E \\
f_e &\geq 0, \qquad \forall e \in E.
\end{aligned}
$$

Descomposición de flujo

Para cada (s-t)-camino P en G y para cada ciclo C en G, las indicatrices de P y C, χ^P y χ^C son (s-t)-flujos con valores 1 o 0 según un arco está o no en el camino o ciclo, respectivamente. Llamemos $\mathcal{P}_{s,t}$ al conjunto de todos los (s-t)-caminos del grafo y $\mathcal{C}$ al conjunto de todos los ciclos del grafo.

El Teorema de Descomposición de Flujos, que fue visto en el curso de Algoritmos Combinatoriales, y también puede verse como una consecuencia específica del teorema de Minkowski-Weyl, dice que todo (s-t)-flujo f de valor $d \geq 0$ se descompone como combinación cónica de indicatrices de (s-t)-caminos y ciclos. Es decir para cada f de valor positivo, existen $(\lambda_P, \lambda_C \geq 0)$ tal que

$$
f = \sum_{P\in\mathcal{P}_{s,t}} \lambda_P \chi^P + \sum_{C\in\mathcal{C}_{s,t}} \lambda_C \chi^C
$$

P1.- Dado f un (s-t)-flujo, se define $\text{val}(f) = f(\delta^+(s)) - f(\delta^-(s))$ y $c(f) = \sum_{e\in E} c(e)f(e)$. Pruebe que (1) $val(f) = \sum_{p\in P_{s,t}} \lambda_P$ y (2) $c(f) \geq \sum_{P\in\mathcal{P}_{s,t}} \lambda_P c(\chi^P)$. Use esto para concluir que existe un $(s-t)$-flujo de valor d óptimo que es combinación cónica solo de indicatrices de $(s-t)$ caminos (no usa ciclos).

Master y Pricing Problem

Dado un subconjunto $X \subseteq \mathcal{P}_{s,t}$, definimos el *reduced master problem* $M(X)$ asociado a X como sigue, junto a su dual $D(X)$.

$$
\begin{aligned}
M(X) \quad \text{mín} \quad & \sum_{P\in X} \lambda_P c(P) \\
& \sum_{P\in X} \lambda_P = d, \\
- & \sum_{P\in X\colon e\in P} \lambda_P \geq -u_e, \quad \forall e \in E \\
& \lambda_P \geq 0, \quad \forall P \in X
\end{aligned}
\qquad\qquad
\begin{aligned}
D(X) \quad \text{máx} \quad & dz - \sum_{e\in E} y_e u_e \\
& z - \sum_{e\in P} y_e \leq c(P), \quad \forall P \in X \\
& y_e \geq 0, \quad \forall e \in E
\end{aligned}
$$

Sea $X \subseteq \mathcal{P}_{s,t}$ tal que $M(X)$ es factible. Sea $(\overline{\lambda}, (\overline{y}, \overline{z}))$ un par óptimo de $M(X), D(X)$.

El PRICING PROBLEM asociado consiste en **Determinar $P \in \mathcal{P}$ que maximice la cantidad** $(z - y(P)) - c(P)$.

Si esta cantidad es menor o igual que 0, entonces estamos parados en un óptimo global. Si no, el camino que maximice dicha cantidad es el que debemos agregar.

P2.- Argumente que el camino $P \in \mathcal{P}_{s,t}$ que maximiza $z - y(P) - c(P)$ representa un (s-t)-camino de largo mínimo en G con función de largo $\ell_e = c_e + y_e$ para todo arco $e \in E$.

P3.- Investigue en la librería `LightGraphs` de JULIA la función que usa Dijkstra para calcular caminos de largo mínimo en un digrafo $G = (V, E)$. Use esta función para calcular los caminos de largo minimo en las instancias que que se le dan en el archivo `dijkstra.ipynb`. Investigue también como usar la función `DiGraph` para crear un digrafo con largo en los arcos dados por ℓ, usando una matriz de $M \in \mathbb{R}_{|V|\times|V|}$ tal que $M_{i,j} = \ell((i,j))$ si $(i,j) \in E$, y si no $M_{i,j} = 0$. Debe usar la función para calcular todos los caminos mínimos desde el nodo $s = 1$ y luego recuperar el largo del camino mínimo hacia $t = |V|$.

Nota: **NO** debe hacer una implementación propia de Dijkstra, si no usar la función que tiene implementado Dijkstra en la libreria mencionada.