UNIVERSIDAD DE CONCEPCION
FACULTAD DE CIENCIAS
FISICAS Y MATEMATICAS
DEPARTAMENTO DE INGENIERIA MATEMATICA

Optimización III (525551)

Pauta Evaluación 1

P1) (30 ptos.) Determine el valor de verdad de las siguientes proposiciones. Justifique su respuesta.

a) (10 ptos.) Sea d el arreglo retornado por el algoritmo BFS con entrada un grafo (no dirigido) $G = (V, E)$ y $s \in V$. Si $\exists u, v \in V, d[v] > d[u] + 1$, entonces $\{u, v\} \notin E$.

b) (10 ptos.) Sea $G = (V, A)$ un digrafo, $w : A \to \mathbb{R}^+ \cup \{0\}$ una función de peso con valores no negativos y $s \in V$. Entonces, existen infinitos caminos de peso mínimo de s a algún vértice $v \in V$, $v \neq s$, en G si y sólo si existe un ciclo C alcanzable desde s con $w(C) = 0$.

c) (10 ptos.) PATH es NP-completo si y sólo si P=NP.

Solución:

a) (Verdadera)

Sabemos de lo visto en clase que para todo $\{u, v\} \in E$, $\delta(s, v) \leq \delta(s, u) + w(u, v)$.

Por otro lado, también sabemos que el algoritmo BFS es equivalente a usar Dijkstra con función de peso w constante igual a uno. Así, por resultados vistos en clases, al término del algoritmo BFS con entrada un grafo (no dirigido) $G = (V, E)$ y $s \in V$ se tiene para todo $u \in V$ que: $d[u] = \delta(s, u)$. De aquí, para todo $\{u, v\} \in E$ se tiene que:

$$d[v] = \delta(s, v) \leq \delta(s, u) + w(u, v) = d[u] + 1.$$

En otras palabras, $\{u, v\} \in E \implies d[v] \leq d[u] + 1$. Por lo tanto, la proposición es verdadera.

b) (Verdadera) ($\Longrightarrow$) Supongamos que hay infinitos caminos de peso mínimo de s a algún vértice v, $v \neq s$. Como el número posible de caminos entre dos vértices, sin repetición de vértices, es finito, entonces necesariamente existe un camino peso mínimo de s a v en G:

$$p_{sv} : s = v_1, v_2, \ldots, v_r,$$

y para el que existe $i < j$ con $v_i = v_j$. Así, $c : v_i, v_{i+1}, \ldots, v_j = v_i$ es un ciclo en p_{sv} que es alcanzable desde s.

Como por hipótesis los pesos son todos no negativos, entonces $w(c) \geq 0$. Sin embargo, si $w(c) > 0$, entonces podemos remover el ciclo c de p_{sv} dejando todavía un camino de s a v de peso menor que el peso de p_{sv}, lo cual es una contradicción con ser p_{sv} de peso mínimo. Por lo tanto, $w(c) = 0$.

($\Longleftarrow$) Sea $c : v_1, v_1, \dots, v_j = v_1$ un ciclo con $w(c) = 0$ alcanzable desde s, es decir, existe un camino p_{sv} en G de s a algún $v \in V(c) = \{v_1, \dots, v_j\}$, que podemos suponer sin pérdida de generalidad que es de peso mínimo, pues al ser todos los pesos, no negativos no hay ciclos negativos y así por resultado visto en clase, hay al menos un camino de peso mínimo.

Luego, si suponemos sin pérdida de generalidad que $v = v_1$ se tiene que para todo $k \in \mathbb{N}$, $p^k := p_{sv}c^k$ es un camino de s a v, donde c^k es la concatenación de k veces c. Así, $w(p^k) = w(p_{sv})w(c^k) = w(p_{sv})$, es decir p^k es camino de peso mínimo. Por lo tanto, existen infinitos caminos de peso mínimo de s a $v \in V,\ v \neq s$.

c) (Verdadera) ($\Longrightarrow$) Supongamos que PATH es NP-completo, entonces todo problema de decisión Q en NP verifica que $Q \leq_p$ PATH. Sabemos que PATH es un problema polinomial. Asi por resultado visto en clase, todo Q en NP es también P. Luego, NP $\subseteq P$ y como sabemos que P$\subseteq$ NP, entonces P=NP.

($\Longleftarrow$) Por otro lado, supongamos que P=NP. Como PATH está en P, entonces PATH es también NP. Sea Q un problema de decisión en NP. Por hipótesis, Q es también P. Luego existe un algoritmo A_Q que resuelve Q.

Mostremos que $Q \leq_p$ PATH. En efecto, fijemos $x, y \in I_{PATH}$ una instancia afirmativa y negativa de PATH respectivamente, es decir PATH$(x) = s$ y PATH$(y) = n$ (basta tomar un grafo que no tenga un camino entre dos vértices dados y otro que sí). Luego, definamos la función $f : I_Q \to I_{PATH}$ tal que $\forall w \in I_Q :\ f(w) = x$ si $Q(w) = s$ y $f(w) = y$ en caso contrario. Luego, se tiene por definición que:

$$\forall w \in I_Q :\ PATH(f(w)) = s \Longleftrightarrow Q(w) = s.$$

Por otro lado, para calcular f basta con correr el algoritmo A_Q con entrada w y proceder según si $Q(w) = s$ o $Q(w) = n$ lo que puede ser determinado en tiempo polinomial por hipótesis. Así, f es una reducción polinomial.

En resumen, PATH es NP-hard y como es también NP, entonces es NP-completo.

P2) (30 ptos.) Sea $G = (V, A)$ un grafo dirigido y $w : A \to \mathbb{R}$ una función de peso. Se definen los siguientes problemas:

- *CICLO PESO POSITIVO (CPP)*: Dado (G, w) ¿Existe un ciclo C en G con $w(C) > 0$?
- *CICLO PESO CERO (CPC)*: Dado (G, w) ¿Existe un ciclo C en G con$w(C) = 0$?

a) (15 ptos.) Pruebe que el problema CPP es polinomial.

b) (15 ptos.) Pruebe que el problema CPC es NP-completo.

Solución:

a) Notar que para todo ciclo C en G, $w(C) > 0 \Longleftrightarrow -w(C) < 0$. Así, determinar si G con w tiene un ciclo de peso positivo es equivalente a determinar si G tiene un ciclo de peso negativo con respecto a $-w$.

Sabemos que el algoritmo de Bellman Ford detecta los ciclos de peso negativo alcanzables desde un vértice inicial $s \in V$ en tiempo polinomial. Así, podemos usar Bellmand-Ford con entrada $(G, -w, s)$ para determinar la existencia de un ciclo positivo con respecto a w alcanzables desde s. Luego, una forma de determinar la existencia de un ciclo positivo (cualquiera) de G con respecto a w es hacer lo anterior para todo vértice inicial $v \in V$, lo cual es obviamente polinomial.

b) Para mostrar que CPC es NP-completo, debemos probar que es NP y NP-Hard. Lo primero es fácil de ver usando un algoritmo polinomial que con entrada G y w y un certificado correspondiente a una secuencia de vértices candidata a ciclo de peso cero. Esto último puede ser chequeado directamente en tiempo polinomial.

Para mostrar que CPC es NP-Hard construiremos una reducción polinomial $f : I_{HAMPATH} \to I_{CPC}$ definida por:

$$\forall (G, w) \in I_{HAMPATH},\ f(G) = (G', w'),$$

donde HAMPATH es el problema de Camino Hamiltoniano en un grafo dirigido, $G' = (V', A')$ con $V' = V \cup \{x\},\ x \notin V$ y $A' = A \cup \{(u, x), (x, u) : u \in V\}$. Además, $\forall (u, v) \in A'$:

$$w'(u, v) = \begin{cases} 1 & \text{si } (u, v) \in A, \\ 0 & \text{si } (u, v) = (u, x), \\ 1 - |V| & \text{si } (u, v) = (x, v). \end{cases}$$

De esta manera, f puede ser calculada polinomialmente y existe un camino Hamiltoniano en G si y sólo si existe un ciclo C en G' con $w'(C) = 0$ (ejercicio). Por lo tanto, como HAMPTH es NP-Hard, por resultado visto en clase, CPC es también NP-Hard y así es NP-Completo.
