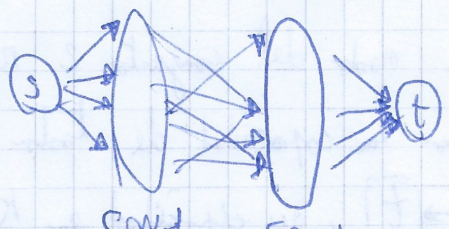
MICELI, JUAN PABLO
LU: 424/19

ALGO 3 2° PARCIAL

HOJA N° 1
FECHA 18/06/2021

1) Resolveremos este problema de ~~matching~~ flujo usando matching.

Tendremos un modelo como el siguiente:



Como se puede ver la red dirige su flujo estrictamente de izquierda a derecha y contamos con la fuente s, el sumidero t y 2 conjuntos que forman un digrafo bipartito. En el conjunto 1 pondremos todas las aristas de G. En el caso en que en A $(v_i \to v_j)$ se agrega la misma con esa dirección al conjunto. En casos en que $(v_i \leftrightarrow v_j) \in E$, se agrega $(v_i \to v_j)$ y la ~~misma~~ misma representa ambas conexiones. En el conjunto 2 tendremos 1 nodo por cada vertice de G. ~~[illegible]~~ Si corre 1 unidad de flujo desde s hasta t, pasando por $\boxed{v_i \to v_j} \in$ conj 1 y por $\boxed{v_j} \in$ conj 2 eso quiere decir que en el digrafo de salida D existe $v_i \to v_j$, por lo tanto esa arista aporta 1 al grado de salida de v_i. Habiendo visto esta interpretación, veamos ~~[illegible]~~ que aristas agregamos a la red y que capacidades les ponemos.

- Como ~~el conj 1 indica~~ la salida del conj 1 indica la dirección que se le da a una arista, deberemos limitar el flujo que le entra a cada nodo en 1, para que solo se pueda elegir una ~~or~~ orientación para cada arista. ∴ $\forall (s \to v), v \in$ conj 1, cap $(s \to v) = 1$.
- Como las aristas de A ya están fijadas entonces van a aportar 1 al grado de salida que les corresponda. Es decir, $\underbrace{\boxed{v_i \to v_j}}_{\in \text{conj } 1} \longrightarrow \underbrace{\boxed{v_i}}_{\in \text{conj } 2}$ con capacidad 1 $\forall\ v_i \to v_j \in A$.

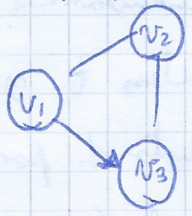
NOTA

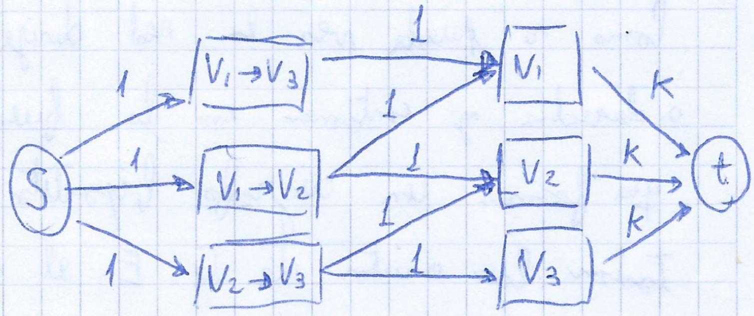
Además las aristas $\in E$ deben tener la oportunidad de usar cualquiera de las 2 direcciones disponibles. Es por eso que $\forall\ v_i \leftrightarrow v_j \in E$, existe

$$\underset{\in \text{conj } 1}{\boxed{v_i \to v_j}} \longrightarrow \underset{\in \text{conj } 2}{\boxed{v_i}} \quad y \quad \underset{\in \text{conj } 1}{\boxed{v_i \to v_j}} \longrightarrow \underset{\in \text{conj } 2}{\boxed{v_j}}$$ ambas con capacidad 1.

Por último, como queremos que cada nodo del conjunto 2 represente su grado de salida en D entonces limitamos la capacidad de todas las aristas resultantes en K. Es decir, $\forall\ \underset{\in \text{conj } 2}{\boxed{v_i}} \to \boxed{t}$, su capacidad es K.

Veamos un ejemplo:

G: 

$\Longrightarrow$ N: 

c) La existencia del digrafo D está atada a conseguir un flujo máximo igual a m, eso nos indicará que pudimos orientar todas las aristas sin sobrepasar el grado de salida en ningún nodo.

Si este flujo se consigue podemos armar D colocando todos los nodos de G y agregando las aristas $v_i \to v_j$ sii hay flujo pasando por $\boxed{v_i \to v_j} \longrightarrow \boxed{v_i}$. Caso contrario, por hipótesis de que $\exists$ D hay flujo entre $\boxed{v_i \to v_j} \longrightarrow \boxed{v_j}$ y agregamos $v_j \to v_i$ a D.

d) La complejidad de E&K es $O(\#\text{aristas}^2 \cdot \#\text{vertices})$.

$\#$ aristas $\simeq 2m + n \simeq m + n$

$\Longrightarrow$ E&K queda en $O\left((m+n)^2 (m+n)\right) = O\left((m+n)^3\right)$

$\#$ nodos $= m + n$

Igualmente vimos que si $F_{max} \ll m \cdot n \Rightarrow$ E&K es $O(F \cdot \#\text{aristas})$. Vimos que $F \leq m$. Por lo tanto, podemos conseguir una complejidad final de $O(m \cdot (m+n))$.

MICELI, JUAN PABLO | ALGO 3 | HOJA N°: 3
LU: 424/19 | 2° PARCIAL | FECHA 18/06/2021

2) a) El certificado es el mismo para los 3 problemas.

CERTIFICADO: ~~Pedimos~~ Una lista de nodos que represente el camino de S a t.

El certificado es trivialmente polinomial.

Para verificar el certificado, en los 3 problemas debemos chequear que la lista de nodos sea un camino de S a t en G y que su peso sea menor o igual a P. Luego, si estamos verificando K-MISS-MIN-PATH debemos chequear que el largo de la lista sea de $|V|-k$. Por contrario, verificamos que la lista tenga largo K.

Todas las acciones previas son trivialmente polinomiales.

$\Rightarrow$ Los 3 problemas $\in NP$.

---

b) Como K no es parte de la entrada, entonces se lo considera una constante. Entonces, puedo conseguir todos los caminos de k nodos en G en $O\left(\binom{K}{n}\right) \in O(n^k)$ (polinomial en n) y puedo verificar también en tiempo polinomial si alguno de esos caminos cumple lo pedido.

---

c) QvQ LEN-MIN-PATH (LMP) es NPC.

1) LMP es NP? Si, por punto a).

2) $TSP \leq_p LMP$?

2) vale $\Leftrightarrow$ $TSP(x) = Si \Leftrightarrow LMP(f(x)) = Si$ y f es polinomial

Dada $\langle G, S, t, P \rangle$ entrada de TSP, podemos tomar $\langle G, S, t, |V|-1, P \rangle$ entrada de LMP. La transformación es constante $O(1)$.

NOTA

$\Rightarrow$ TSP(x) = Si

QvQ: LMP(f(x)) = Si

Como TSP es Si entonces existe camino en G de s a t con peso $\leq$ P que pasa por todos los vertices. Por ende, este camino tiene largo |V| - 1 así que se va a cumplir que LMP es Si ya que era en el K pasado.

$\Leftarrow$ LMP(f(x)) = Si

QvQ: TSP(x) = Si

Sabemos que existe camino en G de s a t con peso $\leq$ P que ~~pasa por todos los vertices~~ tiene largo |V| - 1. Al no haber ciclos, el camino debe estar pasando por |V| nodos de G. Por lo tanto se pasa por todos los nodos y TSP es Si.

---

d) QvQ KMMP es NP-COMPLETO.

1) KMMP es NP? Si por punto a.

2) TSP $\leq_p$ KMMP?

2) vale $\Leftrightarrow$ TSP(G, s, t, P) = Si $\Leftrightarrow$ KMMP(f(G, s, t, P)) = Si , f polinomial

Dado < G, s, t, P > entrada de TSP, podemos tomar < G $\cup$ $\overline{K_{(K-1)}}$ s, t, P > entrada de KMMP. Esta transformación NO agrega aristas entre G y $\overline{K_{(K-1)}}$, solo agrega los (K-1) nodos. La misma es polinomial.

$\Rightarrow$ TSP(x) = Si

QvQ: KMMP(f(x)) = Si

Sabemos que existe un camino en G de s a t con peso $\leq$ P y con largo |V| - 1. Ese mismo camino existe en f(x), la diferencia es que allí |V| aumenta en K-1, pero a que el largo del camino sigue igual. Entonces, ahora el largo del camino es |V| - K. Luego, KMMP(x) es Si.

$\Leftarrow$ KMMP($f(x)$) = Si

QvQ: TSP(x) = Si

Sabemos que existe un camino de largo $|V|-K$ en G que va de s a t y que tiene peso $\leqslant P$. Además sabemos que el camino pasa por todos los vertices de G ya que en $f(G)$ hay $K-1$ vertices que seguro no pertenecen al camino. Por lo tanto TSP(X) = Si.

---

3) a) Vimos en 2 que LMP es NPC y sabemos que TSP es NPC. ~~Entonces~~ Sabemos que un problema Π es NPC si:

1) Π es NP

2) $\forall \Gamma \in NP$, $\Gamma \leqslant_p \Pi$

~~Además sabemos~~
~~Si tomamos Π = TSP y Γ = LMP, como ambos son NPC $\Rightarrow$ NP podemos decir:~~

Como ambos problemas son NPC (implica que son NP) entonces todo problema NP se reduce a TSP y LMP. En particular podemos tomar LMP y TSP respectivamente y tendremos:

LMP $\leqslant_p$ TSP y TSP $\leqslant_p$ LMP.

b) Como vimos en el 2) $\exists$ TSP $\leqslant_p$ KMMP. Entonces si A resuelve KMMP polinomialmente, podemos aplicar la reducción antes vista a la entrada de TSP en orden polinomial y luego aplicar A.

c) Queremos que $\forall \Pi \in P$, KLMP($f(x)$) = Si $\Leftrightarrow$ $\Pi(x)$ = Si, f polinomial

Podemos tomar la siguiente función:

$$f(x) = \begin{cases} \langle K_{k+1}, V_0, V_1, K \rangle, & V_0 \text{ y } V_1 \text{ nodos cualesquiera de } K_K \text{ y todas las aristas de peso } 1. \quad \Pi(x) = SÍ \\ \langle \overline{K_2}, V_0, V_1, 1 \rangle & \Pi(x) = NO \end{cases}$$

Como $\Pi(x)$ es polinomial la podemos aplicar y si $\Pi(x)$ es SÍ entonces daremos un grafo completo de $k+1$ nodos ya que siempre habrá caminos de tamaño K y si sus pesos son todos 1 $\Rightarrow$ el peso del camino será siempre K.

En el caso en que $\Pi(x) = NO$, pasamos como G $\overline{K_2}$ ya que en ese grafo nunca encontraremos un camino entre s y t.

---

4) El alfabeto del problema será una lista ordenada de vertices ~~que forman un~~ ~~tour~~

Las soluciones parciales serán los caminos en G que parten de S, tienen longitud $< K$, peso $< P$ y no llegan a t.

La solución válida será el camino en G que parte de S, llega a t y tiene longitud $= K$ y peso menor o igual a P.

La función de extensión para una solución parcial con un camino C será:

- Si $|C| = K-1$: Debemos tomar t. Si t no es un vecino entonces la solución es inválida. También lo será si el peso de $C+t > P$.
- Si $|C| < K-1$: Tomamos un nodo vecino de C que continúe el camino, que $\notin C$, que no sea t y que $C + v$ tenga peso $< P$. De no existir un nodo v que cumpla con todo esto, entonces es una solución inválida.

MICELI, JUAN PABLO | ALGO 3 | HOJA N° 7
LU: 424/19 | 2° PARCIAL | FECHA 18/06/2021

Una posible poda por optimalidad sería tratar como inválidos a todos los caminos que tengan largo $< K-1$ y $\forall v \in N(t)$, $v \in C$. Ya que sabemos que no tendremos manera de llegar a t porque está bloqueado por toda su vecindad.