14. Dado un digrafo completo y pesado D el *problema de viajante de comercio* (TSP por sus siglas en inglés: *traveling salesman problem*) consiste en encontrar un ciclo que recorra todos los vértices de D y tenga costo mínimo. Queremos resolver el caso particular de TSP en el cual $|V(D)| = 2n$ y sabemos en qué orden deben recorrerse los nodos "pares". Es decir, además de D, el input contiene una secuencia $w_2, w_4, \ldots, w_{2n}$ de vértices; el output debe ser un ciclo $v_1, v_2, \ldots, v_{2n}$ tal que $v_{2i} = w_{2i}$ para todo $i = 1, \ldots, n$.

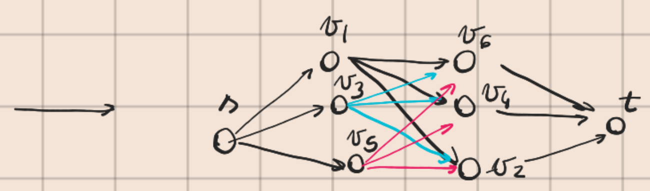
   a) Modelar el TSP como un problema de matching bipartito de peso mínimo en grafo G.

   b) Dar una interpretación a cada matching de G como representante de un ciclo de D.

   c) Demostrar que el modelo es correcto.

   d) Determinar la complejidad de resolver el modelo resultante con el algoritmo del Ejercicio 13.

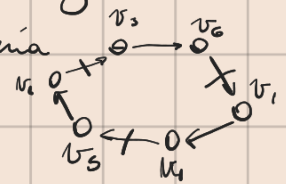
Supongamos que $|V(D)| = 6$ y sabemos que debemos recorrer los pares en este orden: v_4, v_6, v_2



Si lo modelamos de esta forma se convierte en un problema de matching de costo mínimo donde si $v_i \longrightarrow w_j$ entonces es un camino del ciclo, no obstante no sabemos cuanto costaría ir de w_j a un v_i

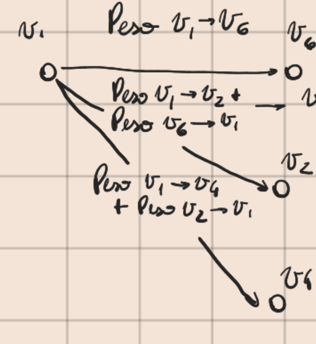
En este modelo si el matching es $(v_3, v_6), (v_1, v_4), (v_5, v_2)$

$\Rightarrow$ el ciclo sería



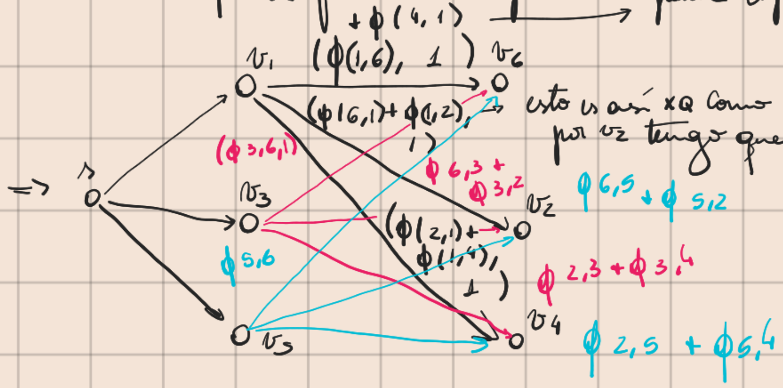
Hay un problema: no estaríamos considerando los costos de ir de v_6 a v_1, de v_4 a v_5, de v_2 a v_3

$\Rightarrow$ Podríamos sumar costos. Como la secuencia de pares tiene un orden que ya conocemos podríamos al costo de moverme de v_i a w_j sumarle el costo de salir de w_{j-1}



$\longrightarrow$ vale considerarlo así xQ D es completo

⇒ Sea ϕ la función de peso → para el primero tengo que sumar salir del último



esto es así xq como los pares están ordenados, para pasar por v_2 tengo que haber salido de v_6

De esta forma, un matching M va a devolver el ciclo que buscamos ahora sí considerando de donde podemos salir

Modelo Red N /

ordenados en el orden de la secuencia de pares

$$V(N) = V(D) \cup \{s, t\}$$

$$E(N) = \{s \to v_i \,/\, i \equiv 1 \,(2)\} \cup \{v_i \to t \,/\, i \equiv 0 \,(2)\} \cup \{v_i \to v_j \,/\, i \equiv 1 (2), j \equiv 0 (2)\}$$

$$(q(e), c(e)) = \begin{cases} (0, 1) & \text{si } e = s \to v_i \\ (q(i,j) + q(i,j-2), 1) & \text{si } e = v_i \to w_j,\ j > 2 \\ (q(i,j) + q(i,2n), 1) & \text{si } e = v_i \to w_j \text{ si } j = 2 \\ (0, 1) & \text{si } e = v_j \to t \end{cases}$$

La interpretación del matching va a ser la misma que dimos antes. Una arista $v_i \to w_j$ perteneciente al matching ⇒ que $w_{j-2} \to v_i \to w_j$ es parte del ciclo

ⓒ $\exists$ solución $\longleftrightarrow$ $\exists$ flujo factible

→) Si $\exists$ solución $\longrightarrow$ $\exists$ un ciclo que recorre todos los vértices con costo mínimo $Q = v_1, v_2, \ldots, v_{2n}$

⇒ puedo definir un flujo factible en mi red ya que $\forall e / e = v_i \to v_j / i \equiv 1 (2), j \equiv 0 (2)$ le pongo flujo 1 y a toda arista $s \to v_i$ y $v_j \to t$ también

→ Cumple que $0 \leq f(e) \leq c(e)$ y la ley de conservación de flujo ya que $f(s \to v_i) = 1 = f(v_i \to v_j) = f(v_j \to t)$

y por como definimos capacidades $\exists !$ arista $e / N^+(v) = e$ y otra ! arista $e' / N^-(v) = e'$

←) $\exists$ flujo factible → podemos maximizarlo con el algoritmo del 12 → obtenemos matching de costo mínimo → $\exists$ solución

(d) $n = 2n$

$m = n^2 + 2n$

$F = n$

Complejidad $O(n^2 (n^2 + n \log n)) = O(n^4 + n^2 \log n)$