ETIQUETAS: MD2::(03)Matchings

¿Que es un matching en G grafo?

Es un subgrafo M de G tal que todo vértice de M tiene grado 1

Un matching es un conjunto de lados {{c1::disjuntos}}

¿Que es un matching maximal?

El matching que tenga la mayor cantidad posible de lados

¿En que tipos de grafos nos centraremos en buscar matchings?

En grafos bipartitos

¿Cómo enconrtar un matching maximal en un grafo bipartito?

1. Transformar el grafo en un network con capacidades enteras
2. Encontrar un flujo entero maximal en el network
3. A partir del flujo maximal construir el matching maximal

¿Cómo transformar un grafo bipartito con partes X e Y en un network con capacidades enteras?

1. Agregar un nodo fuente s y un nodo sumidero t
2. Agregar un lado de cada nodo de X a s
3. Agregar un lado de cada nodo de Y a t
4. Agregar los lados del grafo bipartito de X a Y

¿Que capacidad tienen los lados del network que se construye a partir de un grafo bipartito?

1

¿Supongamos que tenemos un flujo maximal sobre el network que se construye a partir de un grafo bipartito $G = (V, E)$, cuales son los vertices del matching maximal?

$$\{v \in V : IN_f(v) = 1\}$$

¿Supongamos que tenemos un flujo maximal sobre el network que se construye a partir de un grafo bipartito $G = (V, E)$, cuales son los lados del matching maximal?

$$\{xy : x \in X, y \in Y : f(\overrightarrow{xy}) = 1\}$$

{{c2::El valor de f}} es igual a {{c1::la cantidad de lados del matching maximal}}

¿Cómo obtener un flujo maximal a partir de un matching maximal $M = (V, E)$?

- $f(\overrightarrow{xy}) = 1 \text{ si } xy \in E$
- $f(\overrightarrow{sx}) = 1 \text{ si } xy \in V \cap X$
- $f(\overrightarrow{yt}) = 1 \text{ si } xy \in V \cap Y$
- $f(\overrightarrow{uv}) = 0 \text{ cualquier otro caso}$

¿Cuando un matching es perfecto?

Cuando los vertices de M son todos los vertices de G

¿Cuando un matching es completo en X?

Cuando todos los vertices de X estan en M

¿Que condición es necsaria para que un matching sea perfecto?

Que $|X| = |Y|$

¿Que condición es necesaria para que un matching sea completo en X?

Que $|X| \leq |Y|$

¿Que dice el teorema de Hall?

Si G es un grafo bipartito con partes X e Y
Entonces existe un matching completo sobre $X \Leftrightarrow |S| \leq |\Gamma(S)| \;\; \forall S \subseteq X$

¿Que dice el teorema del matrimonio (König)?

Todo grafo bipartito regular tiene un matching perfecto

¿Cómo se define un coloreo propio de los lados de un grafo?

Es un coloreo de los lados donde lados que comparten un vértice no pueden tener el mismo color

¿Que es el indice cromatico de un grafo?

Es la cantidad mínima de colores necesarios para colorear los lados de un grafo

¿Cómo se denota el indice cromatico de un grafo G?

$$\chi'(G)$$

¿Cuales son las cotas del indice cromatico de un grafo G?

$$\Delta \leq \chi'(G) \leq \Delta + 1$$

¿Que dice el teorema de König para indice cromatico?

Si G es un grafo bipartito entonces $\chi'(G) = \Delta$

Si G es un grafo bipartito $\Rightarrow \exists\ H$ grafo {{c1::bipartito regular}} tal que {{c1::$G \subseteq H$}} y {{c1:: $\Delta(H) = \Delta(G)$ }}

¿Que criterios se utilizan para minimizar costos?

- Minimizar la suma de los costos
- Minimizar el costo máximo

Si A tiene {{c1::entradas no negativas}} tales que {{c1::existe un matching "de ceros"}} entonces {{c2::ese es el matching de suma de costo mínimo}}

¿Que es un matching "de ceros"?

Un matching donde los costos de los lados son todos cero

Supongamos que A es una matriz de costos $(n \times n)$ y sea $\tilde{A}$ {{c1::una matriz que se obtiene de A restandole una constante a una fila o columna}}.
Entonces {{c2::todo matching que minimize la suma de costos respecto de A también minimiza la suma de costos respecto de $\tilde{A}$ (y viceversa)}}

¿Cómo es el algoritmo para buscar un matching?

1. Buscamos matching usando Greedy
2. Agregamos la etiqueta s a las filas que no tienen matching
3. Extender el matching

¿Cómo es el paso de extender el matching si tenemos una fila etiquetada?

Miramos cuales columnas tienen 1 en esa fila y etiquetamos esas columnas

¿Cómo es el paso de extender el matching si tenemos una columna etiquetada?

Miramos cual es la fila que tiene el 1 matcheado en esa columna y etiquetamos esa fila

¿Cómo es el orden de revisión de las etiquetas?

- Izquierda a derecha
- Arriba a abajo

¿En que filas se coloca la etiqueta s?

En las filas que no tienen matching luego de aplicar el algoritmo greedy para buscar matching

Se etiquetan {{c2::todos los 1}} al revisar una {{c1::fila etiquetada}}

Se etiqueta {{c2::solo el 1 matcheado}} al revisar una {{c1::columna etiquetada}}

¿Cual es el S que se utiliza para demostrar el teorema de Hall?

S es un subconjunto de todas las filas que quedaron etiquetadas

¿Que truco hay en el algoritmo de minimizar máximo costo para evitar hacer tantas iteraciones en la busqueda binaria?

Si el minimo valor de una fila o columna es k, entonces no puede haber un matching con costo maximo menor a k

¿Que algoritmo se utiliza cuando las tareas de los trabajadores son paralelas?

Gross (Minimizar máximo costo)

¿Que algoritmo se utiliza cuando las tareas de los trabajadores son secuenciales/dependientes?

Hungaro (Minimizar suma de costos)

¿Cuales son los pasos del algoritmo de minimizar máximo costo? (Gross)

1. Hacer una lista de los valores del matching
2. Hacer busqueda binaria sobre la lista
    - En el paso del elemento i de la lista, se construye el matching de $1s$ de costo máximo i y se verifica si existe matching completo

¿Cómo se construye el matching de $1s$ de costo máximo i?

En los indices del matching que son menores a i se coloca 1 y en los indices mayores a i se coloca 0

¿Cuales son los pasos del algoritmo de minimizar suma de costos? (Hungaro)

1. Restar a cada fila el mínimo de esa fila
2. Restar a cada columna el mínimo de esa columna
3. Si existe un matching de ceros, terminar
    1. Si no, realizar paso extra

¿Que se cumple cuando no existe un matching de ceros en el algoritmo Hungaro?

Existe un S tal que $|S| > |\Gamma(S)|$

¿Que se cumple cuando no existe un matching en el algoritmo de Gross para ese i?

Existe un S tal que $|S| > |\Gamma(S)|$

¿Que hay que realizar en el paso extra del algoritmo Hungaro?

1. Calcular $m =$mínimo de las entradas de la matriz que esten en las filas de S y las columnas del complemento de $\Gamma(S)$
2. Restar m de las filas de S

3. Sumar m a las columnas de $\Gamma(S)$
4. Continuar buscando matching perfecto de ceros en la nueva matriz

¿Que filas tachamos en el paso extra del algoritmo Hungaro?

Las filas no etiquetadas (No estan en S)

¿Que columnas tachamos en el paso extra del algoritmo Hungaro?

Las columnas etiquetadas (Estan en $\Gamma(S)$)

¿De que se calcula el mínimo en el paso extra del algoritmo Hungaro? (graficamente)

De las entradas no tachadas

¿En donde se resta en el paso extra del algoritmo Hungaro? (graficamente)

En lo que no esta tachado

¿En donde se suma en el paso extra del algoritmo Hungaro? (graficamente)

En lo que esta tachado doble (Forma una "intersección" de 2 rayas $+$)

¿Que pasa con los 0 que estaban matcheados en el paso extra del algoritmo Hungaro?

Se quedan matcheados y siguen siendo 0

¿Al cambiar la matriz vamos a poder extender el matching en un lado en el paso extra del algoritmo Hungaro?

No necesariamente

¿Cual es la complejidad del algoritmo Hungaro como lo implementamos para resolver los ejercicios?

$O(n^4)$

Luego de un cambio de matriz en el algoritmo Hungaro, o crece {{c1::el matching}} o {{c1::crece el S}}

¿Que complejidad tiene el algoritmo Hungaro si lo implementamos de forma eficiente?

$O(n^3)$

¿Cuales son los pasos para obtener el matching que minimiza la suma entre los matching que minimizan el máximo costo?

1. Usar Gross para obtener el mínimo máximo costo k
2. Modificar la matriz original colocando ∞ en las entradas que sean mayores a k
3. Correr Hungaro sobre la matriz modificada

¿Cuales son los pasos para obtener el matching que minimiza el máximo costo entre los matching que minimizan la suma de costos?

1. Correr Hungaro para obtener la minima suma de costos k
2. Correr Gross para obtener el mínimo máximo costo j
3. Hacer busqueda binaria desde j hasta el máximo costo de la matriz original
    1. En el paso del elemento i de la lista, se modifica la matriz original colocando ∞ en las entradas que sean mayores a i
    2. Se corre Hungaro sobre la matriz modificada y se verifica que la suma de costo mínimo del matching modificado sea igual a k

¿Cómo modificar la matriz del matching para maximizar el mínimo o maximizar la suma de costos?

Reemplazar la entrada a_{ij} por $\text{Max valor matriz} - a_{ij}$

¿Cómo demostrar que existe un matching perfecto con ciertos parámetros?

Demostrar que es imposible que exista un S tal que $|S| > |\Gamma(S)|$