MA3705. Algoritmos Combinatoriales 2020.
Profesor: José Soto
Escriba(s): Yeniffer Muñoz,Manuel Torres.
Fecha: 14 de Diciembre de 2020
(https://youtu.be/VR52joB2Nhs)

Cátedra 23

Resumen de la clase

En esta clase se enuncia y demuestra el teorema de descomposición de flujo, que será de utilidad para entender mejor la estructura de un flujo y los algoritmos relacionados. Se presentan algoritmos de Edmonds-Karp que usa el camino de mayor capacidad y el que usa el camino más corto, junto con lemas y teoremas relacionados a su correctitud y compeljidad.

Descomposición de Flujo

Teorema 1 (Descomposición de flujo). *Sea $f \geq 0$ un $s-t$ flujo en G. Sean $\mathcal{P}_{s,t}$ el conjunto de los $s-t$ caminos, $\mathcal{P}_{t,s}$ el conjunto de los $t-s$ caminos y $\mathcal{C}$ el conjunto de ciclos. Entonces existen constantes $\lambda \geq 0$ tal que:*

$$\begin{aligned} valor(f) = 0 &\Rightarrow f = \sum_{C\in\mathcal{C}} \lambda_C \chi^C \\ valor(f) > 0 &\Rightarrow f = \sum_{P\in\mathcal{P}_{s,t}} \lambda_P \chi^P + \sum_{C\in\mathcal{C}} {}_C \chi^C \\ valor(f) < 0 &\Rightarrow f = \sum_{P\in\mathcal{P}_{t,s}} \lambda_P \chi^P + \sum_{C\in\mathcal{C}} {}_C \chi^C \end{aligned}$$

El número total de coeficientes no nulos es a lo más $|E|$.

Demostración. Realizando inducción sobre $|E|$:

1. Borrar arcos con flujo 0. Para así disminuir el número de arcos cada vez que sea posible, ya que si f es flujo en el grafo original, también lo es en el grafo sin estos arcos.
2. Si existe algún ciclo C. Sea $\lambda_C = \min_{e\in C} f(e) = f(e^*)$ entonces $f' = f - \lambda_C \chi^C$ es $s-t$ flujo no negativo en $G - e$ de igual valor, donde además

   $$\text{valor}(f') = \text{valor}(f) - \lambda_C \text{valor}(\chi^C) = \text{valor}(f)$$

   Luego los arcos e^* encontrados en cada ciclo, ahora tendrán valor nulo ($f'(e^*) = 0$), por lo que se podrán eliminar también. Así, se podrán ir guardando inductivamente las aristas de forma que sean de ciclos y como eventualmente se acabarán los ciclos se formará la descomposición deseada.

Suponiendo que lo que queda es acíclico, si valor$(f) > 0$, sea $X = \{v : v \text{ es alcanzable desde } s \text{ con arcos}\}$. Si $t \notin S$, entonces:

$$0 < \text{valor}(f) = f^{\text{out}}(X) = f(\delta^(X)) - f(\delta^-(X)) \leq 0$$

lo que es contradicción, por lo tanto $t \in S$. Por otro lado sea $t \in X$ y existe $s - t$ camino P

$$\lambda_p = \min_{e \in P} f(e) = f(e^*) > 0, e^* \in P$$

donde $e^* \in P$, además se puede eliminar e^* de E, y $f' = f - \lambda_p \chi^P$, luego f es combinación cónica de $s - t$ caminos.

Por otro lado, si f < 0, la demostración es análoga al primer caso. □

Corolario 1. *Sea f un $s - t$ flujo factible en N con valor valor$(f) > 0$. El teorema de descomposición de flujo garantiza que*

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

con todos los $\lambda \geq 0$ y a lo más m términos no nulos. Luego existe un $s - t$ camino en N con $\min_{e \in P} f_e \geq \lambda_P \leq \dfrac{valor(f)}{m}$ donde en la descomposición hay $\leq m$ caminos.

Edmonds-Karp 1: Caminos de mayor capacidad

Algorithm 1: Algoritmo (Edmonds Karp, 1972).

Entrada: Red $N = (G, u, s, t)$ con capacidades enteras
$f : E \to \mathbb{R}$, $f \leftarrow 0$
while Exista camino aumentante P en N_+^f **repeat**
  Elegir P de mayor capacidad residual
  $f \leftarrow f + \chi^P cap^f(P)$
  Recalcular N_+^f;
until;
return f

Observación. Para encontrar el camino P de mayor capacidad residual se puede usar una modificación del algoritmo de Dijkstra.

Teorema 2. *EK1 termina en $O(m \log OPT)$ iteraciones.*

Demostración. Sea OPT el valor del flujo óptimo. Sea v_k el valor del flujo f^k de la $k-$-ésima iteración (i.e. $v_k = valor(f^k)$).

1. Existe flujo en N^{f^k} de valor: $OPT - v_k$
   Pues como ya se demostró en la clase anterior, si existen dos flujos en el grafo con valores distintos,

existirá un flujo en la red con valor esta diferencia, y en este caso nos referimos al flujo óptimo y al de la $k-$ésima iteración los que son diferentes.

2. Luego, existe camino aumentante de valor: $\frac{OPT-v_k}{m}$
   Esto ya que dada la existencia del flujo, debe existir un camino aumentante que se compare con el valor que posee, y por lo mencionado en el corolario importante, sabemos que existen a los más m caminos al descomponer un flujo.

3. Luego como $v_{k+1} \geq v_k + \frac{OPT-v_k}{m}$ entonces :
   $OPT - v^{k+1} \leq (OPT - v_k) - \frac{OPT-v_k}{m} = (OPT - v_k)(1 - \frac{1}{m})$
   Esto quiere decir que lo que falta en la etapa $k+1$ para llegar a OPT es lo que faltaba en la etapa k multiplicado por el factor $(1 - \frac{1}{m}$, así iterativamente, se tendría un decrecimiento geométrico de lo que falta para llegar a OPT, y se podría comparar con el valor del comienzo v_0.

4. Por lo tanto $OPT - v_k \leq OPT - v_k(1 - \frac{1}{m})^k \leq OPT \cdot e^{-\frac{k}{m}}$
   Luego, lo lógico es preguntarte luego de cuántas iteraciones el lado derecho de esta desigualdad es menor a 1. Es fácil notar que con $k = mln(OPT)$ se obtiene esto y así luego de $O(mon(OPT))$ iteraciones $v_k = OPT$ demostrándose el teorema.

□

Observación. EK1 es un algoritmo débilmente polinomial, es lo que ya se demostró..

Observación. No se sabe si es fuertemente polinomial (abierto, ya que podría no depender de OPT).

Edmonds-Karp 2: Caminos de mayor capacidad

Algorithm 2: Algoritmo (Edmonds Karp, 1972).

Entrada: Red $N = (G, u, s, t)$ con capacidades enteras
$f : E \to \mathbb{R}$, $f \leftarrow 0$
while Exista camino aumentante P en N_+^f **repeat**
  Elegir P de menor número de arcos
  $f \leftarrow f + \chi^P cap^f(P)$
  Recalcular N_+^f;
until;
return f

Teorema 3. *Edmonds Karp es fuertemente polinomial.*

Capas: BFS en N_+^f

Sea G_i grafo residual (N_+^f) en la i-ésima iteración.El algoritmo lo que hace es construir capas $L_0^i, L_1^i, ..., L_r^i$ basadas en BFS(pues buscamos el camino más corto) considerando: $dist_i(s, v) = k \Leftrightarrow v \in L$

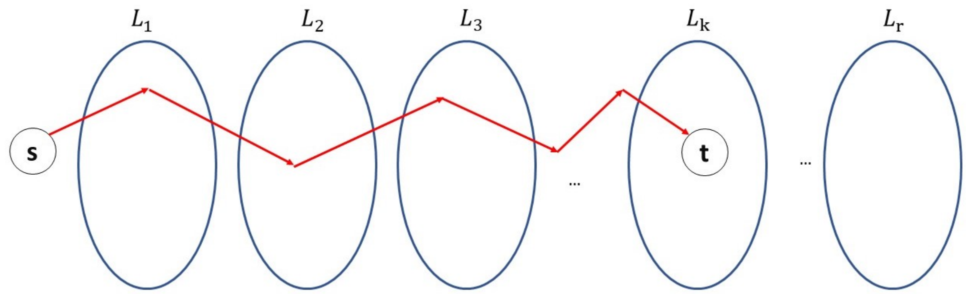

Figura 1: Ilustración de capas que genera BFS en N_+^f

De esta forma el camino P elegido respeta las capas (va de una a la siguiente). Notemos que después de iterar en P, en el flujo original algunos arcos aumentan su flujo y otros lo disminuyen esto depende de si el arco estaba en P o si el arco que estaba en P era el reverso de algún arco del grafo original, esto permite deducir el siguiente lema.

Lema 1. *Arcos que salen:* $E(G_i)\backslash E(G_{i+1}) \subset P$

Es decir los únicos arcos que salen del grafo residual son arcos que estaban en el camino P, pues puede ocurrir que al estar en P al iterar llegue a su capacidad si era directo, y luego desaparezca del grafo reverso i.e. se quede con capacidad 0, o si era el reverso de otro arco en el grafo original, lo que se aumentó por el camino aumentante hizo que el arco original disminuyera su capacidad hasta llegar a cero desapareciendo de igual forma, en cualquier caso solo serán arcos de P.

Lema 2. *Arcos que entran* $E(G_{i+1})\backslash E(Gi) \subset$ *reversos de* P

Similarmente los únicos arcos que entran al grafo residual son los arcos reversos de P, pues para que un arco entre se debe crear capacidad en alguna parte y si se agrega un arco que no estaba en P, al iterar recién existirá su parte reversa con capacidad residual positiva. En resumen a través de este algoritmo se puede tener un mayor control del grafo, pues este no cambia demasiado de una iteración a la siguiente.

Próxima Clase

Consecuencias

Lema 3. $\forall v, i$: $dist_i(s, v) \leq dist_{i+1}(s, v)$

Lema 4. $\forall e \in E \cup \overleftarrow{E}$. *e sale de* $E(G_i)$ *para a lo más* $n/2$ *iteraciones* i *distintas.*

Complejidad de Edomds-Karp 2

1. En la iteración i, al menos 1 arco sale de $E(G_i)$.
2. Como cada arco sale a lo más $O(n)$ veces, el número total de iteraciones es:
3. Cada iteración toma:

Teorema 4. *Edmonds-Karp tiene complejidad*

Resumen Algorítimico para flujos y cortes

Observación. Suponemos $n = O(m)$

1. Dado f es $s - t$ flujo factible máximo, se puede encontrar $s - t$ corte mínimo en tiempo $O(m)$.
2. Algoritmos para datos enteros

ALgoritmo	Complejidad
Ford-Fulkerson (1956)	$O(mOPT)$
Edmonds-Karp I (1972)	$O(mlogOPT)$
Edmonds-Karp II (1972)	$O(nm2)$
Goldberg-Tarjan (1988)	$O(n^2m), O(n^3), O(nmlog(n))$
Orlin+King Rao Tarjan (2013)	$O(nm)$

Con muchas mejoras en el caso de capacidades pequeñas.