Michael Forbes
miforbes@illinois.edu
2019-10-17.4 → 2019-10-22.1
2019-10-22.2 ←
CS473

# CS473 Algorithms : Lecture 16

logistics: - pset 6 due W10

last time: - max capacity augmenting paths ⟹ efficient flow algo

today: - applications of flow algo

=

summary:

Ford Fulkerson $O(mC)$, $C = \sum_e c(e)$ [pseudo poly]

max capacity augmenting path $O(m^2 \lg^2 C)$ [Edmonds Karp] [weakly poly]

shortest augmenting path $O(nm^2)$ [Edmonds Karp, Dinitz] [strongly poly]

$O(nm)$ [Orlin 13]

Prop: $c: E \to \mathbb{Z}_{\geq 0}$ integral capacities
$\Rightarrow |f^*| = \max_f |f| \in \mathbb{Z}$
exists $f^*: E \to \mathbb{Z}_{\geq 0}$ [some max flow is integral] [other max flow may not be] [our algo find integral max flow ⟹ algo proof]

Prop: $c: E \to \mathbb{R}_{\geq 0}$
given any $f^*: E \to \mathbb{R}_{\geq 0}$ can find min cut $V = S \sqcup T$, $S \ni s$, $T \ni t$ in $O(m)$ time [we saw sketch, more on pset 6 optional]

=

Q: what other problems can we solve?

idea: reduce other problems to max flow / min cut.

ex: does graph $G=(V,E)$ have property $P$ ⟼ is max flow in $G'=(V',E')$ at least $k$?

def: A multi source/sink network is a directed graph $G=(V,E)$ with sources $S \subseteq V$, sinks $T \subseteq V$ and $S \cap T = \emptyset$, capacities $c: E \to \mathbb{R}_{\geq 0}$

A flow is $f: E \to \mathbb{R}_{\geq 0}$ st
- capacity constraints: $\forall e \in E$ $0 \leq f(e) \leq c(e)$
- conservation constraints: $\forall v \notin S \cup T$ $f(v) = 0$, $f(v) = f^{out}(v) - f^{in}(v)$
- value $= \sum_{s \in S} f(s)$

A cut is partition $V = S' \sqcup T'$, $S' \supseteq S$, $T' \supseteq T$ w/ capacity $c(S',T') = \sum_{u \in S', v \in T'} c(u,v)$

Q: compute max flow? min cut? [could clearly redo the entire theory]

idea: reduce to single source networks

reduction: given $G=(V,E)$ capacities $c: E \to \mathbb{R}_{\geq 0}$
sources $s_1, \ldots, s_k \in V$ sinks $t_1, \ldots, t_\ell \in V$

construct $G' = (V', E')$ capacities $c': E' \to \mathbb{R}_{\geq 0}$

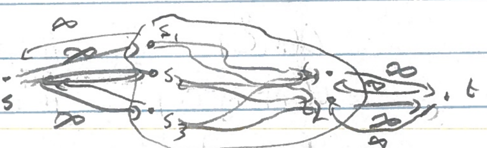
$V' = V \cup \{s, t\}$

$E' = E \cup \{(s,s_i),(s_i,s)\}_i \cup \{(t_j,t),(t,t_j)\}_j$

Michael Forbes
miforbes@illinois.edu
2019-10-22_2 ⇆ 2019-10-22_1
2019-10-22_3
cs473

$$c'(e') = \begin{cases} c(e') & \text{if } e' \in E \\ \infty & e' \in \{(s,s_i),(s_i,s)\}_i \cup \{(t_j,t),(t,t_j)\}_j \end{cases}$$

[or] $\hookrightarrow = 1 + \sum_{e\in E} c(e)$   [could change algo to explicitly handle $\infty$]



[get max flow value and flow itself]

lem: flow $f$ in $G$ $\mapsto$ flow $f'$ in $G'$ w/
- $|f'| = |f|$
- $f'$ constructable in $O(m)$ time

$\longleftarrow$
- $|f| = |f'|$
- $f$ constructible in $O(m)$ time

pf: $f \mapsto f'$: $f: E \to \mathbb{R}_{\ge 0}$ want $f': E' \to \mathbb{R}_{\ge 0}$

$$f'(e') = \begin{cases} f(e') & e' \in E \\ f^{out}(s_i) & e' = (s,s_i) \\ f^{in}(s_i) & (s_i,s) \\ f^{in}(t_j) & (t_j,t) \\ f^{out}(t_j) & (t,t_j) \end{cases}$$

capacity constraints: $e' \in E \Rightarrow 0 \le f'(e') \le c(e') = c'(e')$

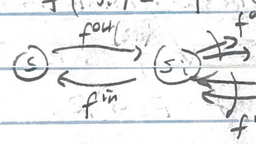
$e' = (s,s_i) \Rightarrow 0 \le f^{out}(s_i) \le \sum_{e\in E} c(e) \le \infty$

$e' = (s_i,s) \Rightarrow 0 \le f^{in}(s_i) \le \sum_{e\in E} c(e) \le \infty$

same for $(t_j,t)$, $(t,t_j)$

conservation: $v \notin \{s,t\} \cup \{s_i\} \cup \{t_j\}$, $f'(v) = f(v) = 0$

$v = s_i$   $f'(s_i) = f^{out}(s_i) - f^{in}(s_i)$



$+ f^{in}(s_i) - f^{out}(s_i) = 0$

$v = t_j$   same

value: $f'(s) = (f')^{out}(s) - (f')^{in}(s) = \sum_i \big(\underbrace{\underbrace{f(s\to s_i)}_{f^{out}(s_i)} - \underbrace{f'(s_i \to s)}_{f^{in}(s_i)}}_{= f(s_i)}\big) = |f|$

$O(m)$ time: clear

$f' \to f$: have $f': E' \to \mathbb{R}_{\ge 0}$

want $f: E \to \mathbb{R}_{\ge 0}$

define $f(e) = f'(e)$ as $E \subseteq E'$

~~efficiency~~: clear

capacity: clear, as $E \subseteq E'$

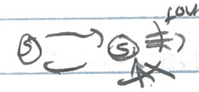
conservation: clear, as $V \setminus (S \cup T) \subseteq V' \setminus (S \cup T \cup \{s,t\})$

value: $f'(s) + \sum_i \underbrace{f'(s_i)}_{=0} = \sum_i \big(f'(s \to s_i) - \cancel{f'(s_i \to s)}\big)$

$f^{out}(s_i)$

$+ \sum_i \big(\underbrace{f^{out}(s_i) + f'(s_i \to s)}_{(f')^{out}(s_i)}\big)$

$- \big(\underbrace{f^{in}(s_i) + f'(s \to s_i)}_{(f')^{in}(s_i)}\big)$



$f^{in}(s_i)$

$= \sum \underbrace{f^{out}(s_i) - f^{in}(s_i)}_{f(s_i)} = |f|$

Michael Forbes
miforbes@illinois.edu
2019-10-22.2 →
2019-10-22.4 ← 2019-10-22.3
CS473

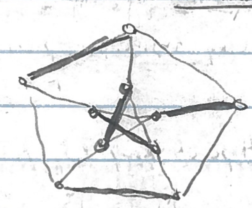
**rmk:** – may *not* have "$f'(s \to s_i) = f^{out}(s_i)$" "intended $f'$"
– $f \mapsto f'$ is injective map, *not* surjective

**Cor.** multi source/sink max flow in $G$ in $O(mn)$ time

**Pf.** construct single source network $G'$ – $O(n)$ vertices – $O(m+n)$ edges
compute max flow $(f')^*$ in $G'$ ← $O(mn)$ time via Orlin
compute corresponding flow $f^*$ in $G$ ← $O(m)$
↳ *is* max flow

**rmk:** can also show correspondence for min cut. [do not get exact correspondence in cut values as vertex/edges are different, and have $\infty$-edges, but suffices for *min* cut]

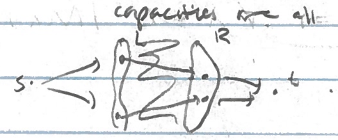
**def.** $G=(V,E)$ *undirected*. A matching $M \subseteq E$ is so $\forall v \in V$ $v$ is incident to *at most one* edge in $M$. $M$ is *perfect* if all $v$ incident to edge in $M$.

ex  ← perfect matching
[find partners amongst friends]

**Q:** given $G$ compute maximum matching? if $G$ bipartite? $=(L \sqcup R,\ E \subseteq L \times R)$ [we assume bipartition is *given* to us, but can also find it efficiently] [using even/odd distances from vertex $v$]

**idea.** reduce to max flow

**construction.** given $G=(L \sqcup R, E)$ *undirected* $|L|=|R|=n$
define $G'=(V', E')$ *directed*
$V' = L \cup R \cup \{s,t\}$ $E' = \{(u,v) : (u,v) \in E\}$ (directed / undirected) $\cup \{(s,u) : u \in L\} \cup \{(v,t) : v \in R\}$
capacities are all 1

ex 

**Prop.** size $k$ matching in $G$ $\mapsto$ flow value $k$ in $G'$

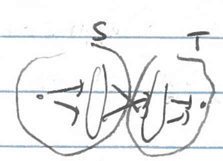
**Pf.** $M = \{(u_1,v_1), \ldots, (u_k,v_k)\}$

$$f(e) = \begin{cases} 1 & e \in M \\ 1 & e=(s,u_i) \\ 1 & e=(v_j,t) \\ 0 & \text{else} \end{cases}$$

capacity: clear
conservation: $(s) \to (u_i) \to (v_i) \to (t)$ [each $v_i$ has $\le 1$ edge] ... net flow zero
value $|f| = |M|$

**Prop.** integral flow value $k$ in $G'$ $\mapsto$ size $k$ matching ← $O(m)$ time

**Pf.** $f' : E' \to \mathbb{Z}_{\geq 0}$
define $M = \{e \in E : f'(e) = 1\}$ ← $O(m)$ time

**Clm.** $|M| = k$ 

**pf.** cut $S = \{s\} \cup L$ $T = R \cup \{t\}$ $f'(S,T) = \underline{f'(S \to T)} - \overbrace{f'(T \to S)}^{=0}$ [no edges $T \to S$]
$k = |f'|$
\# edges $L \to R$ w/ flow 1
↳ capacities 1, flow integral ∎

Michael Forbes
mforbes@illinois.edu
2019-10-22-4 ← 2019-10-22-3
→ 2019-10-24-1
cs473

$\to (v_i) \rightrightarrows$

Clm: $M$ matching

Pf: $v_i \in L$ $(f^*)^{out}(v_i) = \{\#\text{ edges in } M \text{ incident to } v_i\}$

$(f^*)^{in}(v_i) \le c(s, v_i) = 1$

$v_j \in R$ same. ☐

Cor: max matching in $G \overset{O(m)\text{ time}}{\geq}$ max flow in $G'$

Pf. $\le$

$\ge$ ← integrality of max flow

Cor: max matching in bipartite graphs in $O(mn)$ time

Pf: $G = (L \sqcup R, E) \longrightarrow G' = (V', E')$
- $O(n)$ vertices
- $O(m+n)$ edges
- $C = \sum_e \underbrace{c(e)}_{=1} \le O(m+n)$
- $|f^*| \le n = |L| = |R|$

solve max flow, recover matching in $O(m)$ time

Ford Fulkerson $O(m|f^*|) \le O(mn)$
- integrality of max flow was crucial. ☐

rmk: - Ford Fulkerson is better here than max capacity augmenting path ← always 1 here!

[ despite pseudopolynomial time ]

- best known algo
  - $O(m\sqrt{n})$ Hopcroft-Karp 73, Karzanov 73
  - $O(m^{10/7} \operatorname{polylog} m)$ Madry 13

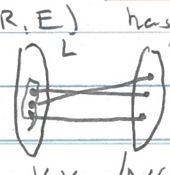
- maximum matchings in general graphs
  - "cannot" be reduced to max flow
  - augmenting paths still used
  - best known $O(m\sqrt{n})$ [Micali Vazirani 80]

[ per groups ]

- 3-tuple hypergraph matching is NP-hard

Lem: $G = (L \sqcup R, E)$ has perfect matching $\Rightarrow \forall X \subseteq L \ |N(X)| \ge |X|$ (set of neighbors)

Pf:



Thm [Hall]: if $\forall X \ |N(X)| \ge |X| \Rightarrow$ perfect matching

Pf: create flow network $G' \Rightarrow$ min cut in $G'$ size $\ge n$

$=$ max flow

$=$ max matching

[ instantiates, max flow = mincut ]

logistics: - pset 6 due W10

today: - applications of max flow algs
- multi-source/sink
- maximum matching

next time: - more advanced applications