Michael A. Forbes
miforbes@illinois.edu
2024-02-08.1

CS473 Algorithms: Lecture 8 (2024-02-08)

logistics: - pset 2 due F17
- pset 3 out F17

last lecture: flows
- def
- residual graph
- augmenting path
- integrality
- FF-algo
  - complexity

today: flows

reading: KT 7.2

[pensive]

[[review last lecture]]

def - capacitated graph is simple [no isolated vertices, parallel edges, loops] directed graph $G=(V,E)$ w/ edge capacities $(c_e)_{e\in E}$ over $\mathbb{N}_{>0}$ [pos ints]. for s [source], t [sink], an (s,t)-flow over G is $f=(f_e)_{e\in E}$ over $\mathbb{R}_{\geq 0}$ [non-neg reals] [continuous flow] ($s\neq t$, >0)

st - capacity constraints: $0\leq f_e \leq c_e$ [no overflow]

- conservation constraints: $f^{in}(v)=\sum_{e:\,\bullet\to v} f_e$ [in flow]

$f^{out}(v)=\sum_{e:\,v\to\bullet} f_e$ [out flow]

$f(v)=f^{out}(v)-f^{in}(v)$ [net flow]

$\forall v\in V\setminus\{s,t\}$ $f(v)=0$ [conservation] [only produce/consume at s,t]

value of flow is $|f|=f(s)$ [net flow at source]

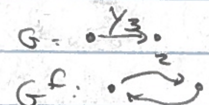
max (s,t)-flow problem is to compute $\arg\max_{f\ (s,t)\text{-flow}} |f|$ [state] [drop s,t when clear]

Q - compute max flow?

idea - $f_e=0\ \forall e\in E$ is valid flow

repeatedly push more flow along $s\rightsquigarrow t$ path, || allowing backwards flow

def - flow f over capacitated $G=(V,E)$. residual graph G^f is capacitated graph $G^f=(V^f,E^f)$ w/ (directed, capacitated, G^f not simple)
- $V^f=V$
- $E^f=\{e:\ f_e<c_e\}$ ← forward edge
  $\cup\{\overleftarrow{e}:\ 0<f_e\}$ ← backwards edge (↖ reverse orientation)
- residual capacities: $e\in E$, if $e\in E^f$ $(c^f)_e=c_e-f_e>0$ [unused capacity]
  $\overleftarrow{e}\in E^f$ $(c^f)_{\overleftarrow{e}}=f_e>0$ [flow to reverse]

eg -


def - flow f over capacitated G, residual G^f. An augmenting path p is simple [no repeat vertices] $s\rightsquigarrow t$ path in G^f. The value is $|p|=\min_{e\in p}(c^f)_e$ [bottleneck]

Augmenting f by p yields flow $f+p$, where for $e\in E$, $(f+p)_e=\begin{cases} f_e+|p| & e\in p \text{ [forward]}\\ f_e-|p| & \overleftarrow{e}\in p \text{ [backward]}\\ f_e & \text{else}\end{cases}$

prop [last lecture] $f+p$ is valid flow, w/ value $|f+p|=|f|+|p|>|f|$

$|p|=\min_{e\in p}(c^f)_e$, >0 [by construction]

idea - repeatedly augment flow

Michael A. Forbes
miforbes@illinois.edu
2024-02-08.2

algo (Ford Fulkerson)

- $f_e \leftarrow 0$, $e \in E$ [valid starting flow]
- init G^f
- while exists augmenting path p in G^f
  - $f \leftarrow f + p$
  - $G^f \leftarrow G^{f+p}$
- return f

prop: capacitated G w/ [integral] capacities $\Rightarrow$ during FF all flows integral [induction] [integers closed under ops]

prop: [any] flow f in G has value $\le \sum_{e \in E} c_e =: C$ [sum of zero flow]

cor: FF takes $\sim O(C)$ iterations [augment increases value by 1] [can't exceed C]

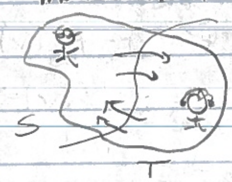
adj list — $\sim O(mC)$ time [O(m) time per iteration]

[complexity]

Q: correctness? [different problem]

Q: why is the internet slow? [bottleneck]

eg: [video chat] [cut limits flow] [me → you video not helped by T→S edges]



[formalized]

cuts — def: G capacitated graph $G=(V,E)$, $s \ne t \in V$. An *(s,t)-cut* C is a partition $V = S \sqcup T$ w/ $s \in S$, $t \in T$. The *capacity* is $|C| = \sum_{e: u \to v,\ u \in S,\ v \in T} c_e$ [ignore T→S edges]

the *Min (s,t) cut problem* is to compute $\min_{C\ (s,t)\text{-cut}} |C|$ [discrete] [arg]

prop: ↳ computable by brute force [reduce understood task to understood either]

Q: flows vs cuts?

def: f flow in G, $S \subseteq V$. the *flow through S* is $f^{out}(S) = \sum_{e: u \to v,\ u \in S,\ v \in V \setminus S} f_e$ [out flow]

$f^{in}(S) = \sum_{e: u \to v,\ u \in V \setminus S,\ v \in S} f_e$ [in flow]

$f(S) = f^{out}(S) - f^{in}(S)$ [net flow]

no loops — prop: $f(v) = f(\{v\})$ [generalizes]

prop: $f(S) = \sum_{v \in S} f(v)$ [set flow] [vertex flows]

pf: $\sum_{v \in S} f(v) = \sum_v \left(\sum_{e: v \to \cdot} f_e - \sum_{e: \cdot \to v} f_e\right)$ [out] [in]

$$= \underbrace{\sum_{e: u \to v,\ u \in S,\ v \notin S} (1) f_e}_{f^{out}(S)} + \underbrace{\sum_{e: u \to v,\ u \notin S,\ v \in S} (-1) f_e}_{-f^{in}(S)} + \underbrace{\sum_{e: u \to v,\ u \in S,\ v \in S} (1+(-1)) f_e}_{=0} + \underbrace{\sum_{e: u \to v,\ u \notin S,\ v \notin S} (0) f_e}_{=0} = f(S)$$

[self flow] ∎

cor: $f(t) = -f(s) = -|f|$ [sink net flow] [source net flow] [s produces, t consumes]

pf: $f(V) \overset{\text{[prop]}}{=} \sum_{v \in V} f(v) \overset{\text{[conservation]}}{=} f(s) + f(t)$

$f(V) \overset{\text{[defn]}}{=} f^{out}(V) - f^{in}(V)$, $f^{out}(V) = 0$ [no edges], $f^{in}(V) = 0$ [no edges]

Michael A. Forbes
mforbes@illinois.edu
2024-02-08 J

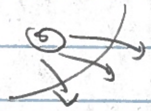
[[key lem]]

lem: G capacitated. f flow. C cut, then $|f| \le |C|$ [[omit s, t]]   Thanks ⇒.

pf: C is $V = S \sqcup T$, $s \in S$, $t \in T$

$|f| \overset{[def]}{=} f(s) \overset{[prop]}{=} f(S) \overset{[def]}{=} f^{out}(S) - f^{in}(S) \overset{[nonneg]}{\le} f^{out}(S) \le \sum_{e: S \to V} f_e \le |C|$ [$f_e \le c_e$] ∎

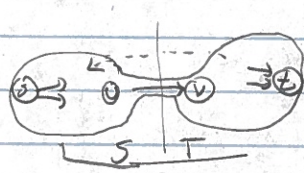
eg:



$\{s\} = S$, T

$|f| \le |C(S,T)| = \sum_{e: s \to \bullet} c_e \le \sum_{e \in E} c_e$ — ub on any $|f|$ used to prove FF terminates; better ub

eg:



[[no small cuts at source]]
[[bottleneck elsewhere]]
[[omit back edges]]

$|C(S,T)| = c_{uv}$ [[clear]]

thm: $\max_{f\ (s,t)\text{-flow}} |f| = \min_{C\ (s,t)\text{-cut}} |C|$ [[related problems]]

Q: FF correctness?

must implicitly provide proof a flow attains max [[how to do this?]]

[[just show]]
thm: $\max |f| \le \min |C|$

cor: if have f, C w/ $|f| = |C|$ then
- f max flow [[C bounds/attests flows]]
- C min cut [[f bounds/attests cuts]]

idea: show FF terminates w/ ↗

prop: capacitated G, $s, t \in V$, f flow, $S = \{v : s \rightsquigarrow v \text{ in } G^f\}$ [[reachable]], equiv:

(1) G^f has no $s \rightsquigarrow t$ path
(2) $C = (S, T = V \setminus S)$ is (s,t) cut w/ $|C| = |f|$ 2 ⇒ 3 explain.
(3) f is max flow

pf: $\neg(1) \Rightarrow \neg(3)$: [[contrapositive]] G^f has $s \rightsquigarrow t$ path p ⇒ [[as before]] $f + p$ is flow w/ value $|f+p| = |f| + |p| > |f|$ ($|p| > 0$) ⇒ f not max flow

(2) ⇒ (3): above

(1) ⇒ (2): clm: C is (s,t)-cut

pf: $s \rightsquigarrow s$ path length 0 in G^f ⇒ $s \in S$

$s \not\rightsquigarrow t$ in G^f ⇒ $t \notin S$ ⇒ $t \in T$ ∎

clm: $s \rightsquigarrow v$ in G^f, $v \xrightarrow{e} u$ in G, if $f_e < c_e$ then $s \rightsquigarrow u$ in G^f

pf: ⇒ $e \in E^f$ [[forward edges]]

⇒ $s \rightsquigarrow v \xrightarrow{e} u$ in G^f ∎

clm: $s \rightsquigarrow v$ in G^f, $v \xleftarrow{e} u$ in G, if $f_e > 0$ then $s \rightsquigarrow u$ in G^f

pf: ⇒ $-e \in E^f$ [[backwards edges]]

⇒ $s \rightsquigarrow v \xrightarrow{-e} u$ in G^f ∎

Michael A. Forbes
miforbes@illinois.edu
2024-02-08.4

cor: edge $v \xrightarrow{e} u$, $v \in S$, $u \in T$, $e \in E$ ⟹ $f_e = c_e$

pf: $s \to v$ in G^f, [e.dcut] $s \not\to u$ in G^f

[clm: no forward edge]

⟹ $f_e \neq c_e$ ⟹ $f_e = c_e$ □

cor: $f^{out}(S) = |C|$

pf. $f^{out}(S) = \sum_{e: v \xrightarrow{e} u,\ S \to T} f_e = |C(S,T)|$, $f_e = c_e$ □

cor: edge $e \in E$, $v \xleftarrow{e} u$, $v \in S$, $u \in T$ ⟹ $f_e = 0$

pf: — ″ — [backwards edges]

cor: $f^{in}(S) = 0$

pf: — ″ —

cor: $f(S) = f^{out}(S) - f^{in}(S) = |C|$ □

[forwards]
[backwards]

eg: (s) → (v) —e→ (u) ←e— ... (t)

cor: FF terminates w/ max flow, if integral capacities

pf: FF terminates w/ G^f having no $s \to t$ path $\xRightarrow{\text{[prop]}}$ f is max flow □

cor: given max flow can compute min cut in $O(m)$ time

pf: take $S = \{v : s \to v \text{ in } G^f\}$ ← $O(n+m)$ time via DFS

take $T = V \setminus S$

$\xRightarrow{\text{[prop]}}$ $|C| = |f|$ ⟹ C min cut

[~~claimed last time~~]

thm: integral capacities, then [some] max flow is integral

pf: FF produces – integral flow – max flow

2022 miforbes.tim

thm [max flow min cut]: capacited G w/ [integral] capacities, max $|f|$ = min $|C|$

thm: [later] — ″ — [real] — ″ —

rmk: naive FF can fail to terminate ↗

Q: are we done?

today: flows
- review last lecture
- cuts
- max flow ≤ min cut
- — ″ — = — ″ — iff G^f no $s \to t$ path
- — ″ — ⟹ FF correct
- — ″ — = — ″ — for [integral] capacities

reading: KT 7.2

next lecture: flows

Logistics:
- pset 2 due F 1/7
- pset 3 out F 1/7