Single-Source Unsplittable Flow

Input: Let $G=(V,E)$ be a digraph with edge capacities c_e for $e \in E$.
Let $s \in V$ be the source. Let (t_i, d_i) for $i \in [k]$ be k commodities, each with terminal $t_i \in V$ and demand $d_i \in \mathbb{R}^+$.

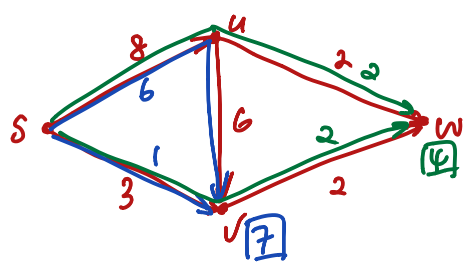
N.B. A vertex may contain multiple terminals.

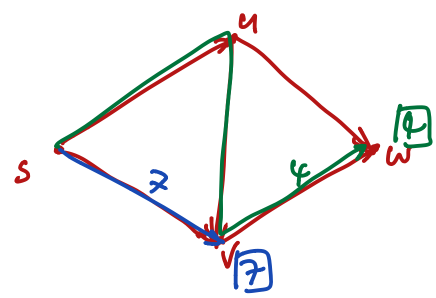
Output: A flow $f: E \to \mathbb{R}^+ \cup \{0\}$ that routes d_i units of commodity i from s to t_i, along a single path, for each $i \in [k]$.

General Unsplittable Multicommodity Flow: routes demands between terminal pairs (s_i, t_i), $i \in [k]$

Aside Surprisingly, unsplittable flow on trees / paths is also interesting.
$\downarrow$
unique path between s_i and t_i

e.g. Integrality gaps of LP relaxations





Applications

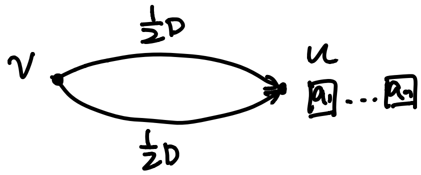
1. PARTITION Given $S=\{a_1, \dots, a_n\} \subseteq \mathbb{N}$, with $D = \sum_j a_j$, decide whether there exists $S' \subseteq S$ such that

$$\sum_{j: a_j \in S'} a_j = \frac{1}{2} D.$$

Theorem PARTITION is NP-Complete.

$V = \{u, v\}$, two parallel edges of capacity $\frac{1}{2}D$ each.

$s = v$; $t_i = u$, $d_i = a_i$, $\forall i \in [n]$



Claim The Single-source Unsplittable flow problem has a feasible solution iff PARTITION has a solution.

SSUFP $\le_p$ PARTITION $\Longrightarrow$ The single-source unsplittable flow is NP-hard.

$\Rightarrow$ approximation algorithms

2. BIN PACKING Given $S = \{a_1, \dots, a_n\} \subseteq \mathbb{N}$, $B, D \in \mathbb{N}$, decide whether there is a partition of S into $\leq B$ sets, each summing to $\leq D$.

Theorem BIN PACKING is NP-complete.

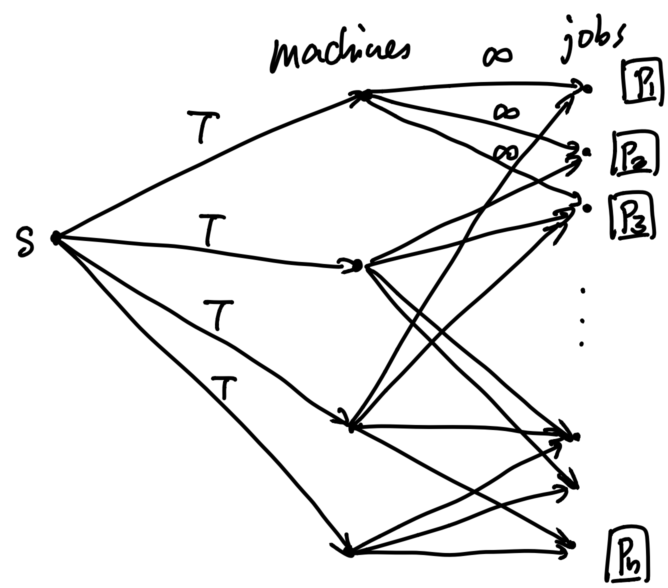
$V = \{u, v\}$, B parallel edges of capacity D each

$s = v$; $t_i = u$, $d_i = a_i$ $\forall i \in [n]$

Claim The single-source unsplittable flow problem has a feasible solution iff. BIN PACKING has a solution.

SSUFP $\leq_p$ BIN PACKING

3. Scheduling Problem n jobs, m parallel machines
job j can only be scheduled on machines in $M(j) \subseteq [m]$ & has processing time p_j



Claim The single-source unsplittable flow problem has a feasible solution iff. there is a feasible schedule of makespan $\leq T$.
time difference between start and end.

Theorem (Lenstra et al.; Shmoys, Tardos)
If there exists a fractional solution to the single-source unsplittable flow problem, then there exists a feasible schedule of makespan $T + \max_j p_j$.

Assumes that the fractional solution is an extreme point of the corresponding polyhedron

Congestion Problem

What is the smallest $\alpha \geq 1$ such that if we multiply all capacities by α, an unsplittable flow satisfying all demands exist?

Terminology

Definition A (fractional) flow is a function $f: E \to \mathbb{R}^+ \cup \{0\}$ such that the net inflow at any $v \in V \setminus \{s\}$ is nonnegative and at most the sum of the demands at v, i.e.

$$0 \leq \sum_{e \in \delta^-(v)} f(e) - \sum_{e \in \delta^+(v)} f(e) \leq \sum_{\substack{i \in [k] \\ t_i = v}} d_i \qquad \forall v \in V \setminus \{s\}.$$

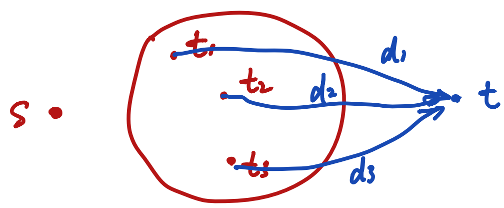
Definition We say that a flow f is feasible if $f(e) \leq c_e$ for all $e \in E$.

Definition We say that a flow f satisfies $I \subseteq [k]$ of commodities if the net inflow at any $v \in V$ equals the sum of the demands at v that belongs to I, i.e.

$$\sum_{e \in \delta^-(v)} f(e) - \sum_{e \in \delta^+(v)} f(e) = \sum_{\substack{i \in I \\ t_i = v}} d_i, \qquad \forall v \in V.$$

Definition We say that a flow f is unsplittable if for each commodity $i \in [k]$ is routed along a single path from s to t_i.

Theme Transform any (fractional) flow into an unsplittable flow, while increasing the capacities by at most the maximum demand.



$\boxed{d = \sum_j d_j}$

Theorem (Ford-Fulkerson) Let $G = (V, E)$ with edge capacities c_e for $e \in E$. Let $s \in V$. Let (t, d) be the single demand. Then there exists a feasible, fractional flow in G if and only if the cut condition is satisfied.

(Cut Condition) For any $S \subseteq V$ with $t \in S$, $s \notin S$, the total capacity of the edges entering S is at least d, i.e.

$$\sum_{e \in \delta^-(S)} c_e \geq d \qquad \forall S \subseteq V,\ t \in S,\ s \notin S.$$

Remark. There is a polynomial time algorithm for deciding feasibility where P. $c_e \in \mathbb{N}$ for $e \in E$. (eg. Dinic, Edmonds-Karp).

No Bottleneck Assumption (NBA)

Let d_{max} be the maximum demand over all commodities, i.e. $d_{max} = \max_{i \in [k]} d_i$.
Let c_{min} be the minimum edge capacity over all edges $e \in E$.

(NBA) $$d_{max} \leq c_{min}$$

Remark The Dinitz-Garg-Goemans algorithm transforms any (fractional) flow into an unsplittable flow, while increasing the capacities by $\leq d_{max}$.

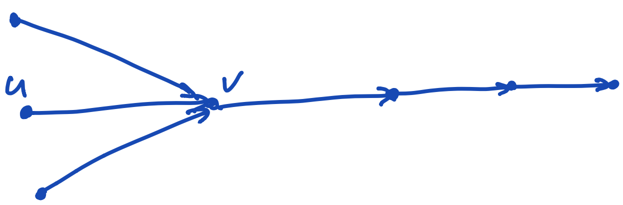
$$c'_e \leq c_e + d_{max} \leq c_e + c_{min} \leq c_e + c_e = 2c_e$$

$\Longrightarrow$ Congestion ≤ 2

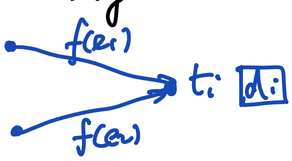
This is the best possible!

Algorithm (Dinitz-Garg-Goemans)

Definition We say that an edge $e \in E$ is singular if v and all vertices reachable from v have out-degree at most 1, i.e. the vertices reachable from v form a dipath.



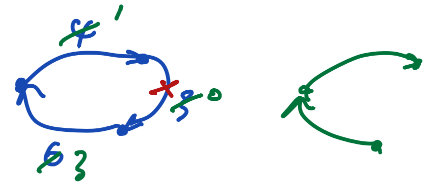
Definition We say that a terminal t_i is regular if $d_i > f(e)$ for all $e \in \delta^-(v)$. Otherwise, we say that t_i is irregular.



Remark If a flow satisfies t_i, then $\sum_{e \in \delta^-(v)} f(e) \geq d_i$. This implies that the in-degree of t_i is at least 2.

Let $f: E \to \mathbb{R}^+ \cup \{0\}$ be a feasible fractional flow that satisfies all demands.
Throughout the algorithm, we remove $e \in E$ whenever $f(e)$ vanishes.

1. While there exists a cycle C in the digraph G:



for all $e \in E(C)$:

$$f(e) := f(e) - \min_{e' \in E(C)} f(e')$$

N.B. Doing so does not change the inflow of any $v \in V(C)$. Hence the new flow remains feasible and still satisfies all demands.

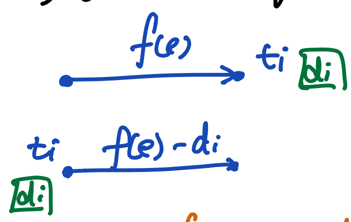
N.B. After this step, G is acyclic.

2. While there exist $i \in [k]$ and $e = (u, t_i) \in \delta^-(t_i)$ such that $f(e) \geq d_i$:

$t_i := u$

$f(e) := f(e) - d_i$

If $t_i = s$: remove commodity i



N.B. The new flow remains feasible and still satisfies all demands.

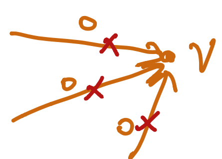
Claim 1 At the end of step 2, for all $v \in V$, if v contains a terminal, then v has at least 2 incoming edges.

Proof. After step 2, $f(e) < d_i$ for all $i \in [k]$ and $e \in \delta^-(t_i)$. Since f satisfies all demands, then $\sum_{e \in \delta^-(t_i)} f(e) = d_i$. Hence, there exist at least 2 incoming edges. □

Claim 2 At the end of each iteration of Step 3, f satisfies all demands. Furthermore, for all $v \in V$, if v contains an irregular terminal, then v also contains a regular terminal.

Proof. Deferred. We present the claim as the induction hypothesis. □

N.B. Since f satisfies all demands, and since zero-flow edges are automatically removed, then any $v \in V$ with out-degree 0 contains a terminal.



Corollary At the end of each iteration, for all $v \in V$, if v contains a terminal, then v has at least 2 incoming edges.

3. Repeat until all terminals reach s:

(a) (Find an alternating cycle)

Let $v_0 \in V$.

Follow outgoing edges as long as possible from v. Until we reach a vertex with out-degree 0. (forward path)

N.B. This process terminates at $v \in V$ with out-degree 0 that contains a terminal since G is acyclic.

Claim & Corollary $\Rightarrow$ v has at least two incoming edges.

Let $e \in \delta(v)$ be such that e in not in the previous forward path.

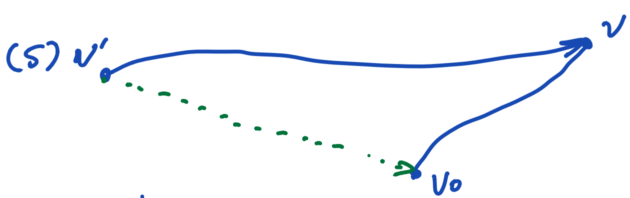
Follow singular incoming edges as long as possible from e. (backward path)

N.B. This process terminates since G is acyclic.

Let $e' = (v', u)$ be the last edge on the backward path.

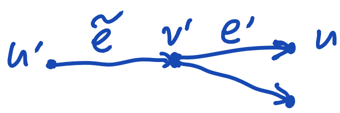
Claim 3 v' has at least 2 outgoing edges.

Proof Case 1 $v' = s$. Suppose for contradiction that e' is the only outgoing edge of v'. Since e' is singular, then G is a dipath.



Contradiction!

Case 2 There exists an sv'-path P such that $f(e) > 0$ for all $e \in E(P)$. Hence e' has an incoming edge $\tilde{e} = (u', v')$. By the maximality of the backward path, $\tilde{e}$ is not singular. Hence, v' has at least 2 outgoing edges.



$\square$

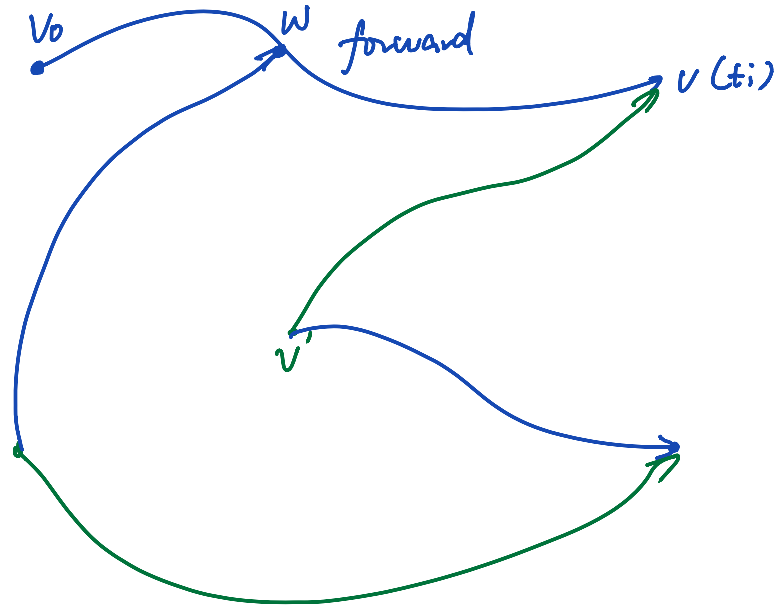
Let e'' be an outgoing edge of v' not in the proceeding backward path.

Repeat the above procedure from e'' to construct forward and backward paths, until we reach $w \in V$ belonging to a previous forward or backward path.

N.B. This process terminates by the pigeonhole principle.

N.B. The paths form a cycle in the underlying undirected graph.

If the two adjacent edges to w on the constructed paths have the same direction, we combine the two paths into one. (alternating cycle)



(b) Let C be the alternating cycle.

Let $\varepsilon_1 = \{f(e) : e \in E(C),\ e \text{ on a forward path}\}$

$\varepsilon_2 = \{d_i - f(e) : i \in [k],\ e = (u, t_i) \in E(C),\ e \text{ on a backward path}\}$

Let $\varepsilon = \min\{\varepsilon_1, \varepsilon_2\}$.

N.B. $\varepsilon > 0$ by definition.

For $e \in E(C)$, e on the forward path:

$f(e) := f(e) - \varepsilon$

For $e \in E(C)$, e on the backward path:

$f(e) := f(e) + \varepsilon$.

(c) (Move terminals)

If (e is singular and $f(e) = d_i$) or (e is not singular and $f(e) \geq d_i$):

$t_i := u$

$f(e) := f(e) - d_i$

This is the end of step 3.

Claim 2 At the end of each iteration of step 3, f satisfies all demands. Furthermore, for all $v \in V$, if v contains an irregular terminal, then v also contains a regular terminal.

Proof. Clearly, f satisfies all demands at the end of each iteration. Let $v \in V$ be such that v contains an irregular terminal, say t_i, $i \in [k]$. Then there exists $e = (u, v) \in \delta^-(v)$ such that $f(e) > d_i$. (if $f(e) = d_i$, then t_i should have been moved in (c).)

Since f cannot be augmented from below d_i to above d_i, then t_i was moved along $e'=(v,w)\in\delta^+(v)$ in a previous iteration, say iteration j.

$$u \xrightarrow[\boxed{d_i}]{f(e)>d_i} v(t_i) \xrightarrow[e']{\times} w \quad (\text{iteration } j)$$

Then $f(e)>d_i$ at the end of iteration j. Since t_i was not moved to u along (u,v) during iteration j, then e is singular. Hence $e'=(v,w)$ is also singular, and $e'=(v,w)$ is the only outgoing edge of v.
Since t_i was moved to v in iteration j, then (v,w) vanishes after moving t_i. Since $f(e)>d_i$, then there exists $t_{i'}$, $i'\neq i$ contained in v.
Note that the out-degree of v becomes 0 after moving the first irregular terminal to v, and the out-degree of v remains 0. Hence there exists at most one irregular terminal at v. Hence, $t_{i'}$ is regular. $\square$.

4. For each $i\in[k]$, define the path to route commodity i to be the reverse path to the one given by moving t_i.

$\square$

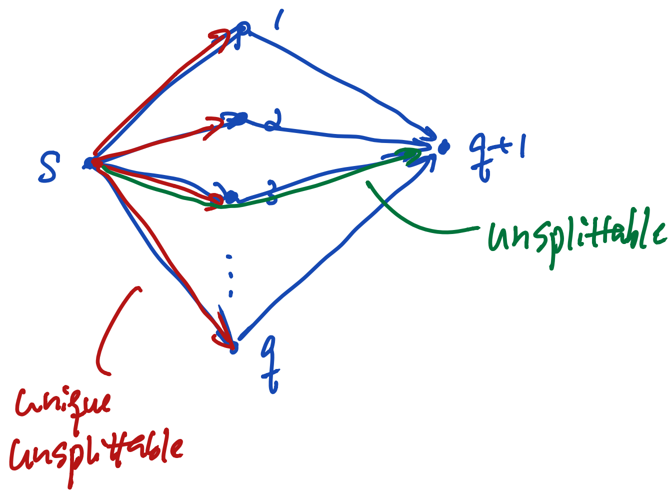
Correctness

<u>Theorem</u> The algorithm finds an unsplittable flow for each commodity $i\in[k]$. Furthermore, the total flow on $e\in E$ exceeds the initial flow on e (hence c_e) by at most d_{max}.

<u>Proof</u> Clearly, the flow from s to t_i is unsplittable for all $i\in[k]$.
Note that $f(e)$ increases only if e is on a backward path, hence only if e is singular.
Hence $f(e)$ does not exceed the initial flow on e.
Since at most one terminal t_i is ever moved along a singular edge.
Since we never add edges, a singular edge cannot become non-singular.
Hence $f(e)$ is at most the initial flow on e plus d_{max}. $\square$

Tightness

Let $q \in \mathbb{N}$. Let $V = \{0, \ldots, q+1\}$ and $E = \{(0,i) : i \in [q]\} \cup \{(i, q+1) : i \in [q]\}$.
Let $G = (V, E)$ with unit capacity. Let $s = 0$. Let $(t_i, d_i) = (i, 1 - \frac{1}{q})$ for all $i \in [q]$, and $(t_{q+1}, d_{q+1}) = (q+1, 1)$.



$c_e' = 1 + 1 - \frac{1}{q} \qquad c_e = 1 \implies c_e' - c_e = 1 - \frac{1}{q}$.

Running Time

Let $n = |V|$, $m = |E|$.
Since at least one edge is removed at each iteration, then the number of iterations is $\leq m$. Finding an alternating cycle takes $O(n)$ time. Since each terminal moves at most n steps to s, then the running time for moving terminals is $O(kn)$. Computing ε_a takes $O(k)$ time in each iteration. Hence, the total running time is $O(nm + km)$.

binary heap, updated each time a terminal is moved
$O(k \log k)$
total: $O(nm + kn\log k)$.

What if the cut condition is not satisfied?

Binary search to determine the smallest $\alpha \geq 1$ such that multiplying all capacities by α satisfies the cut condition. Let f be a feasible fractional flow. Then

$$f'(e) \leq \alpha c_e + d_{max} \leq (\alpha + 1) c_e \leq 2\alpha c_e.$$

$\implies$ 2-approximation

$\boxed{\text{Open Problem}}$

Weighted Single-Sink Unsplittable Flow Problem

Let $G=(V,E)$ be a graph (directed or undirected). Each edge $e\in E$ has a cost $c(e)$ and a capacity $u(e)$. Each terminal $t_i \in V$, $i\in[n]$ has a demand d_i. The problem wants to route d_i units of flow from t_i to the root t along a single path in the capacitated network. Suppose there is a flow whose cost is $C=\sum_{e\in E} f(e)c(e)$ where $f(e)$ is the total flow on edge e. (It is easy to think of an s-t flow where s is a new node with an edge to each terminal t_i of capacity d_i.)

Conjecture If there is such a flow, then there is an unsplittable flow of cost $\le C$ in the instance where the capacity of each edge is $u(e)+d_{max}$.