CSE 202: Design and Analysis of Algorithms

Lecture 9

Instructor: Kamalika Chaudhuri

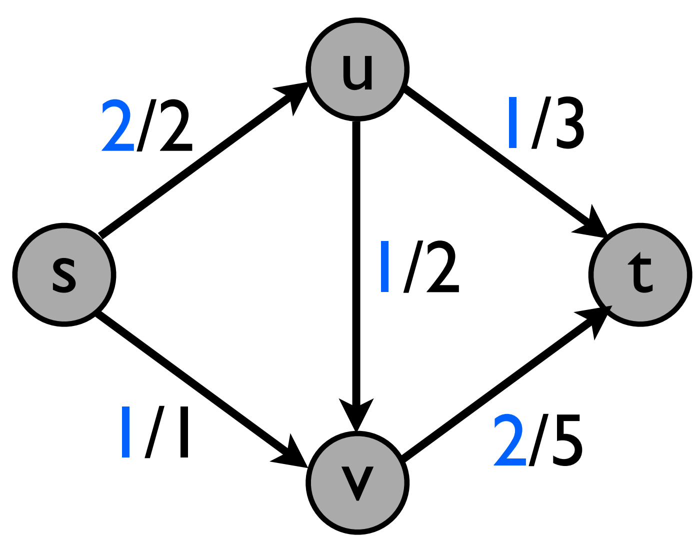
Last Class: Max Flow Problem

The Max Flow Problem: Given directed graph G=(V,E), source s, sink t, edge capacities c(e), find an s-t flow of maximum size

An s-t flow is a function f: E $\longrightarrow$ R such that:
- 0 <= f(e) <= c(e), for all edges e
- flow into node v = flow out of node v, for all nodes v except s and t,

$$\sum_{e\ into\ v} f(e) = \sum_{e\ out\ of\ v} f(e)$$

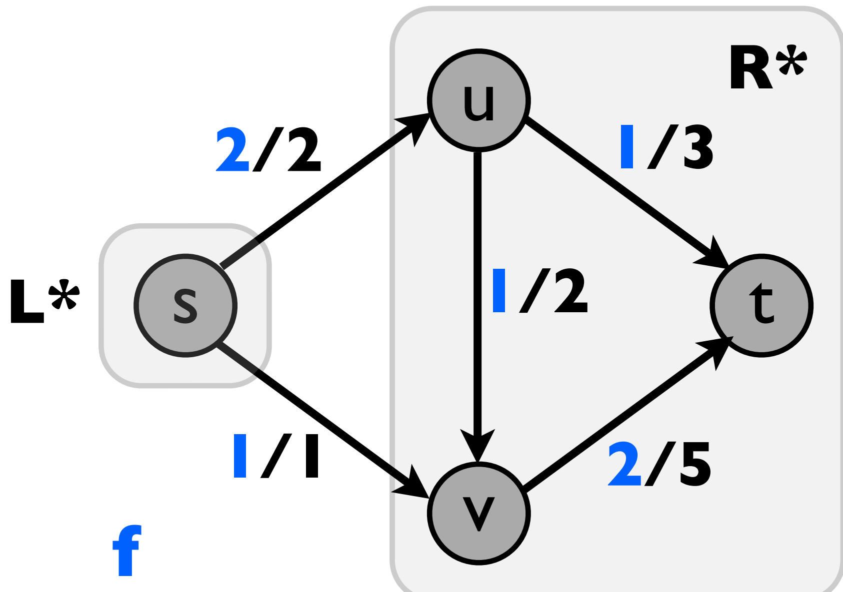
Size of flow f = Total flow out of s = total flow into t



Size of f = 3

Last Class: Facts about Flows

The Max Flow Problem: Given directed graph G=(V,E), source s, sink t, edge capacities c(e), find an s-t flow of maximum size



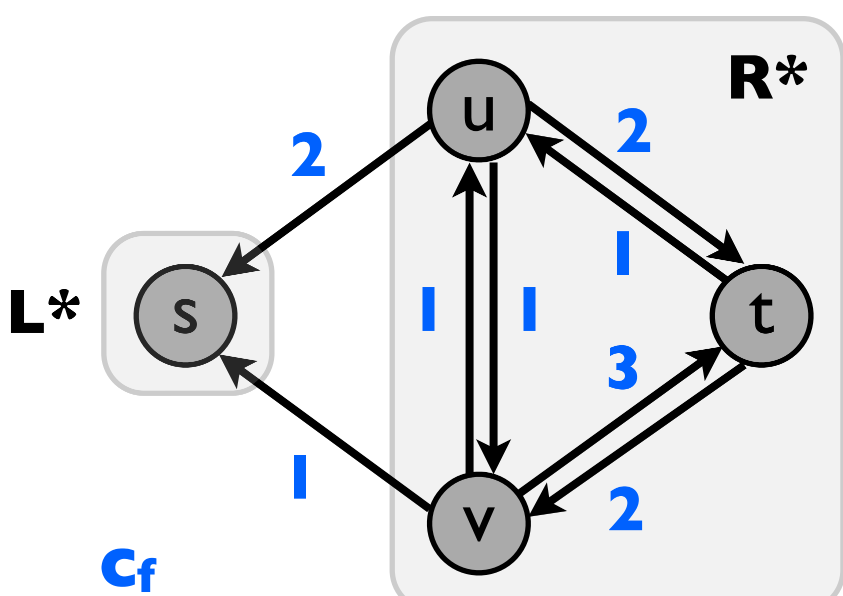
The Residual Graph: For a flow f

$G_f = (V, E_f)$ where $E_f \subseteq E \cup E^R$

For any (u,v) in E or E^R, **residual capacity:**

$c_f(u,v) = c(u,v) - f(u,v) + f(v,u)$

[ignore edges with zero c_f: don't put them in E_f]



Max Flow Min Cut Theorem:

Size(Max-Flow) = Capacity(Min-Cut)

When is f a max flow?

When t is not reachable from s in G_f

Last Class: Algorithms for Max-Flow

Recall: n = #vertices, m = #edges in G

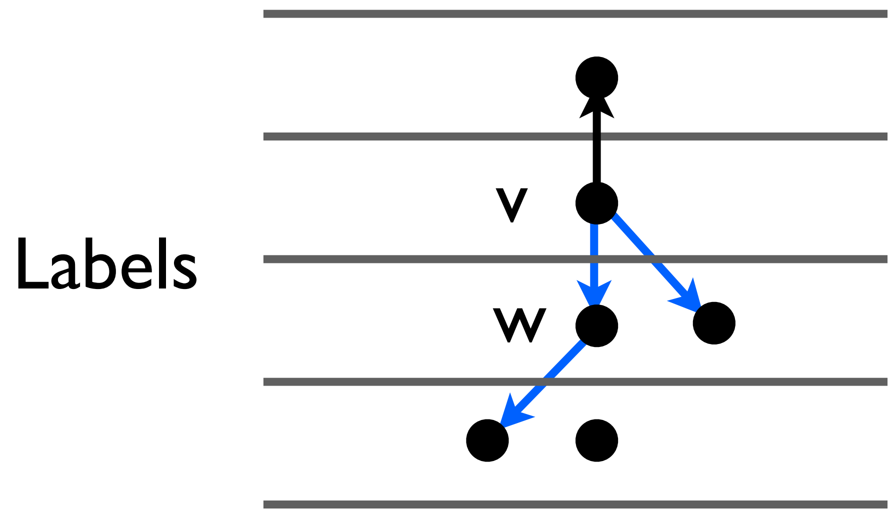
- Ford-Fulkerson: Running Time = $O(m\, C_{max})$

- Other efficient Ford-Fulkerson Style Algorithms:
  - Edmonds-Karp: Running Time = $O(nm^2)$
  - Capacity Scaling: Running Time = $O(m^2 \log C_{max})$

- Preflow-Push

Preflow-Push

Main Idea:
- Each node has a label, which is a potential
- Route flow from high to low potential



Idea: Route flow along blue edges

Preflows

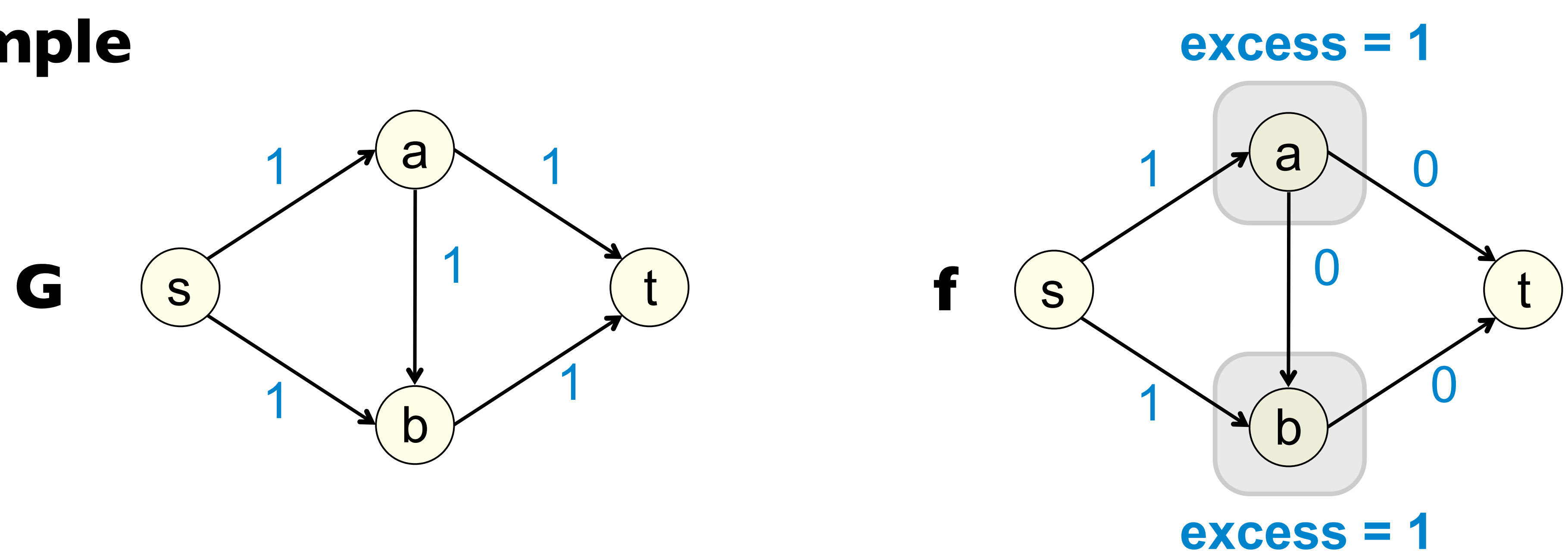
Preflow: A function f: E$\rightarrow$R is a preflow if:

1. **Capacity Constraints:** 0 <= f(e) <= c(e)
2. Instead of conservation constraints:

$$\sum_{e\ into\ v} f(e) - \sum_{e\ out\ of\ v} f(e) \geq 0$$

Excess(v) = $\displaystyle\sum_{e\ into\ v} f(e) - \sum_{e\ out\ of\ v} f(e)$

Example



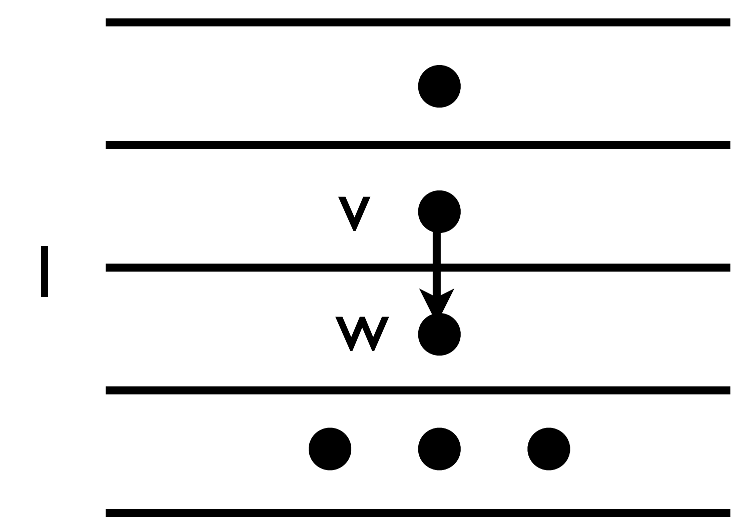
Preflow-Push: Two Operations

Preflow: A function f: E$\rightarrow$R is a preflow if:
1. **Capacity Constraints:** 0 <= f(e) <= c(e)
2. Instead of conservation constraints:

$$\sum_{e\ into\ v} f(e) - \sum_{e\ out\ of\ v} f(e) \geq 0$$

Excess(v) = $\sum_{e\ into\ v} f(e) - \sum_{e\ out\ of\ v} f(e)$

Labeling h assigns a non-negative integer label h(v) to all v in V



Push(v, w): Applies if excess(v) > 0, h(w) < h(v), (v, w) in E_f
  q = min(excess(v), c_f(v,w))
  Add q to f(v, w)

Relabel(v): Applies if excess(v) > 0, for all w s.t (v, w) in E_f, h(w) >= h(v)
  Increase h(v) by 1

Pre-Flow Push: The Algorithm

Start with labeling: $h(s) = n$, $h(t) = 0$, $h(v) = 0$, for all other v
Start with preflow f: $f(e) = c(e)$ for $e = (s, v)$, $f(e) = 0$, for all other edges e

While there is a node (other than t) with positive excess
  Pick a node v with excess(v) > 0
  If there is an edge (v, w) in E_f such that push(v, w) can be applied
    Push(v, w)
  Else
    Relabel(v)

Push(v, w): Applies if excess(v) > 0, $h(w) < h(v)$, (v, w) in E_f
  q = min(excess(v), $c_f(v,w)$)
  Add q to f(v, w)

Relabel(v): Applies if excess(v) > 0, for all w s.t (v, w) in E_f, $h(w) >= h(v)$
  Increase h(v) by 1

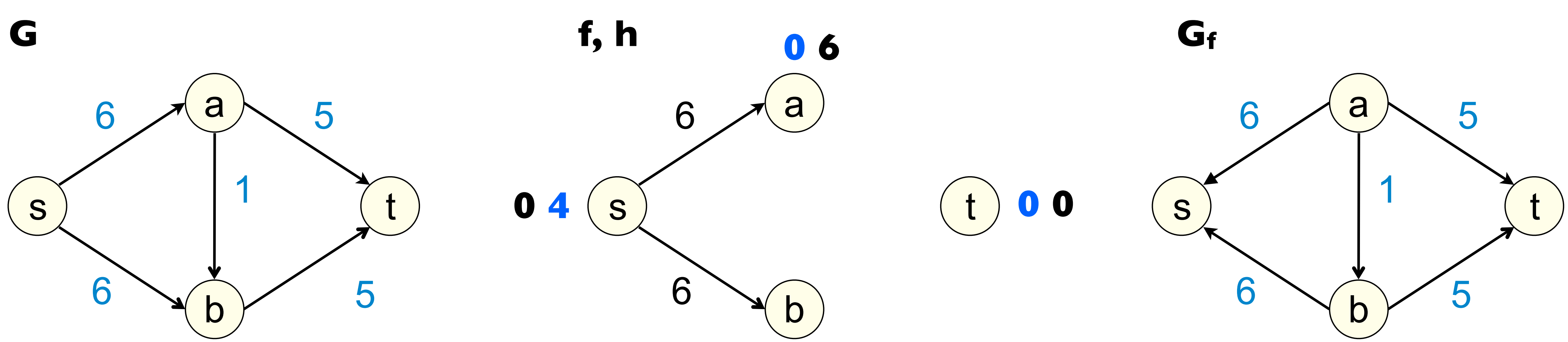
Pre-Flow Push: An Example

Start with labeling: $h(s) = n, h(t) = 0, h(v) = 0$, for other v
Start with preflow f: $f(e) = c(e)$ for $e = (s, v)$, $f(e) = 0$, ow

While there is a node (other than t) with positive excess
  Pick a node v with excess(v) > 0
  If there is an edge (v, w) in E_f s. t. push(v, w) applies
    Push(v, w)
  Else
    Relabel(v)

Push(v, w):
Applies if excess(v) > 0, $h(w) < h(v)$
  $q = \min(\text{excess}(v), c_f(v,w))$
  Add q to f(v, w)

Relabel(v):
Applies if excess(v) > 0 and for all w s.t (v, w) in E_f, $h(w) >= h(v)$
  Increase h(v) by 1



Labels
Excesses

Pre-Flow Push: An Example

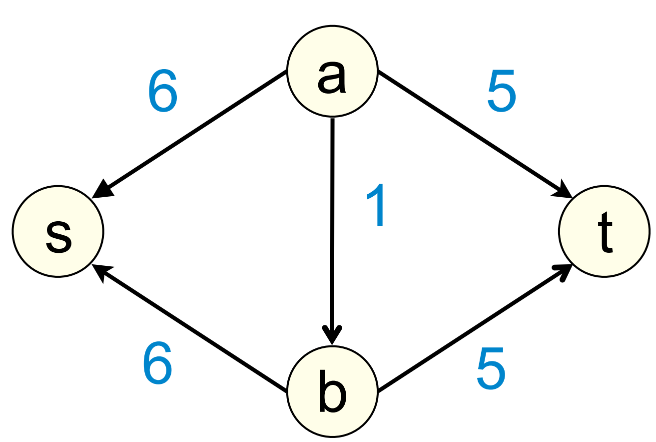
Start with labeling: h(s) = n, h(t) = 0, h(v) = 0, for other v
Start with preflow f: f(e) = c(e) for e = (s, v), f(e) = 0, ow

While there is a node (other than t) with positive excess
- Pick a node v with excess(v) > 0
- If there is an edge (v, w) in E_f s. t. push(v, w) applies
  - Push(v, w)
- Else
  - Relabel(v)

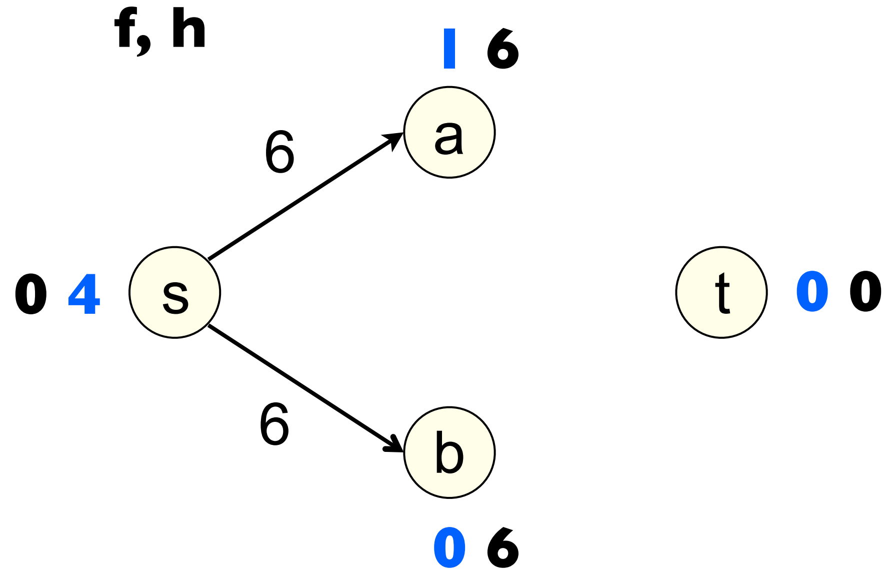
Push(v, w):
Applies if excess(v) > 0, h(w) < h(v)
- q = min(excess(v), c_f(v,w))
- Add q to f(v, w)

Relabel(v):
Applies if excess(v) > 0 and for all w s.t (v, w) in E_f, h(w) >= h(v)
- Increase h(v) by 1

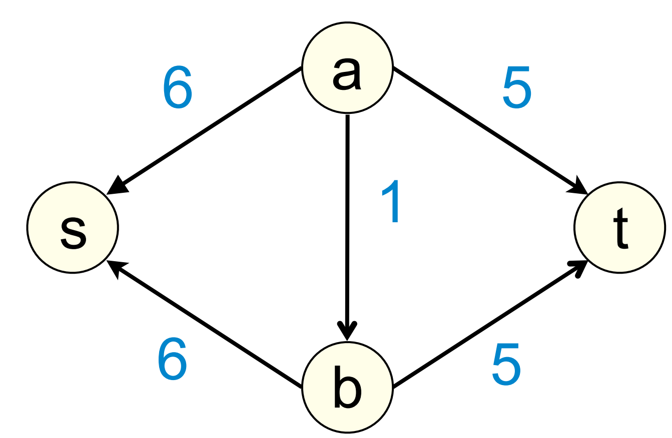
G_f (before)



f, h



G_f



Labels

Excesses

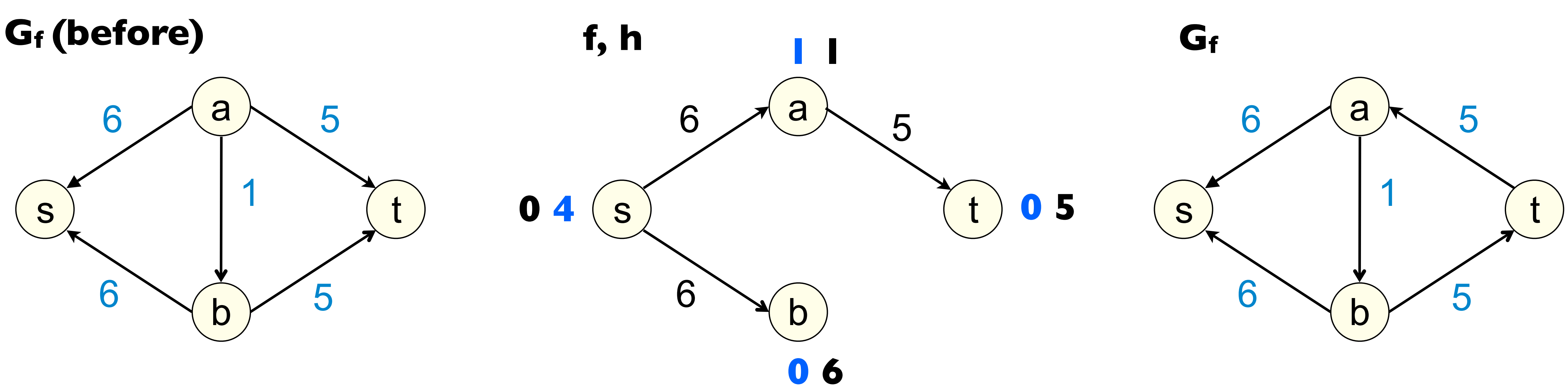
Pre-Flow Push: An Example

Start with labeling: $h(s) = n, h(t) = 0, h(v) = 0$, for other v
Start with preflow f: $f(e) = c(e)$ for $e = (s, v)$, $f(e) = 0$, ow

While there is a node (other than t) with positive excess
    Pick a node v with excess(v) > 0
    If there is an edge (v, w) in E_f s. t. push(v, w) applies
        Push(v, w)
    Else
        Relabel(v)

Push(v, w):
Applies if excess(v) > 0, $h(w) < h(v)$
    $q = \min(\text{excess}(v), c_f(v,w))$
    Add q to f(v, w)

Relabel(v):
Applies if excess(v) > 0 and for all w s.t (v, w) in E_f, $h(w) >= h(v)$
    Increase h(v) by 1



Labels
Excesses

Pre-Flow Push: An Example

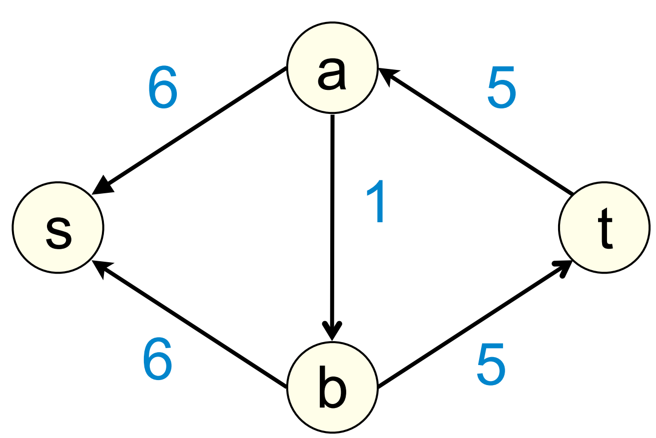
Start with labeling: h(s) = n, h(t) = 0, h(v) = 0, for other v
Start with preflow f: f(e) = c(e) for e = (s, v), f(e) = 0, ow

While there is a node (other than t) with positive excess
- Pick a node v with excess(v) > 0
- If there is an edge (v, w) in E_f s. t. push(v, w) applies
  - Push(v, w)
- Else
  - Relabel(v)

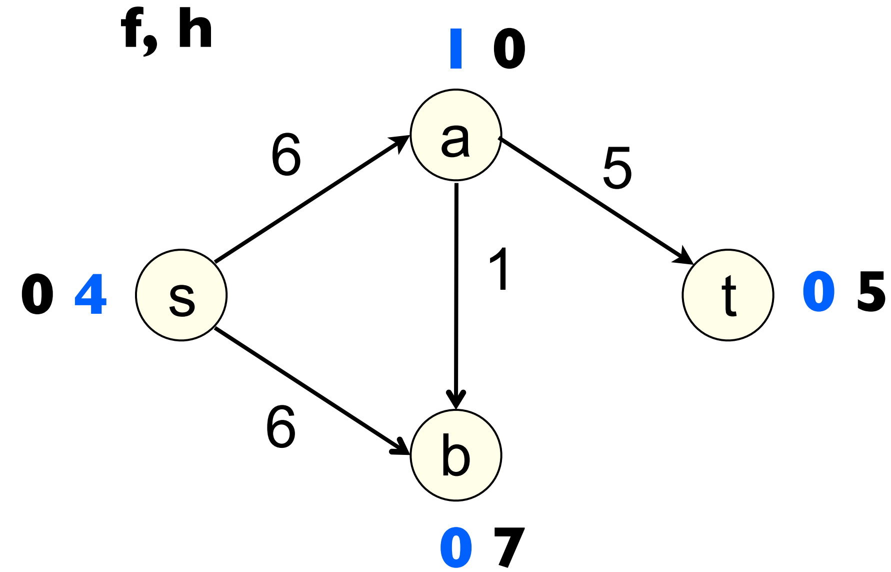
Push(v, w):
Applies if excess(v) > 0, h(w) < h(v)
- q = min(excess(v), c_f(v,w))
- Add q to f(v, w)

Relabel(v):
Applies if excess(v) > 0 and for all w s.t (v, w) in E_f, h(w) >= h(v)
- Increase h(v) by 1

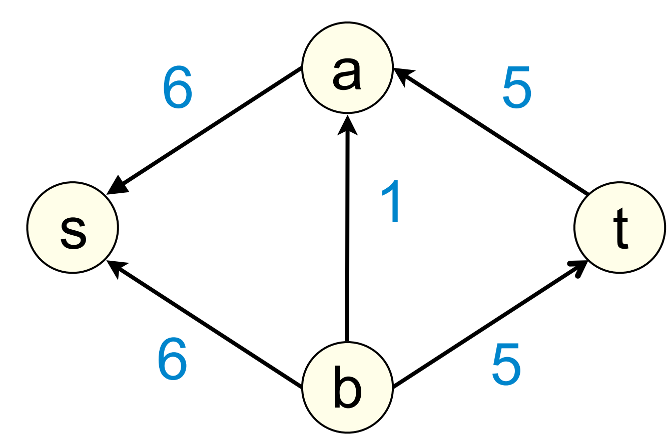
G_f (before)



f, h



G_f



Labels

Excesses

Pre-Flow Push: An Example

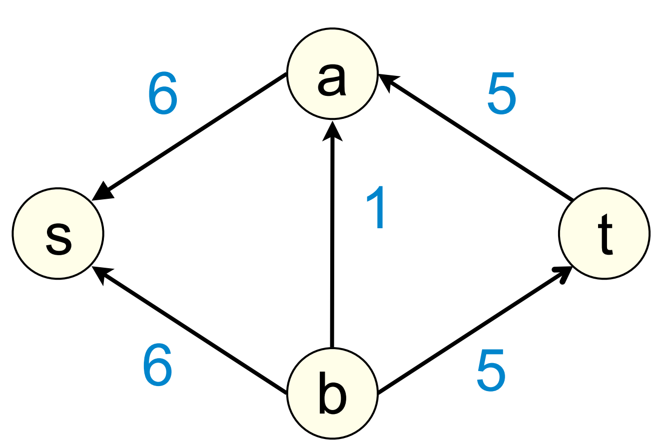
Start with labeling: $h(s) = n, h(t) = 0, h(v) = 0$, for other v
Start with preflow f: $f(e) = c(e)$ for $e = (s, v)$, $f(e) = 0$, ow

While there is a node (other than t) with positive excess
- Pick a node v with excess(v) > 0
- If there is an edge (v, w) in E_f s. t. push(v, w) applies
  - Push(v, w)
- Else
  - Relabel(v)

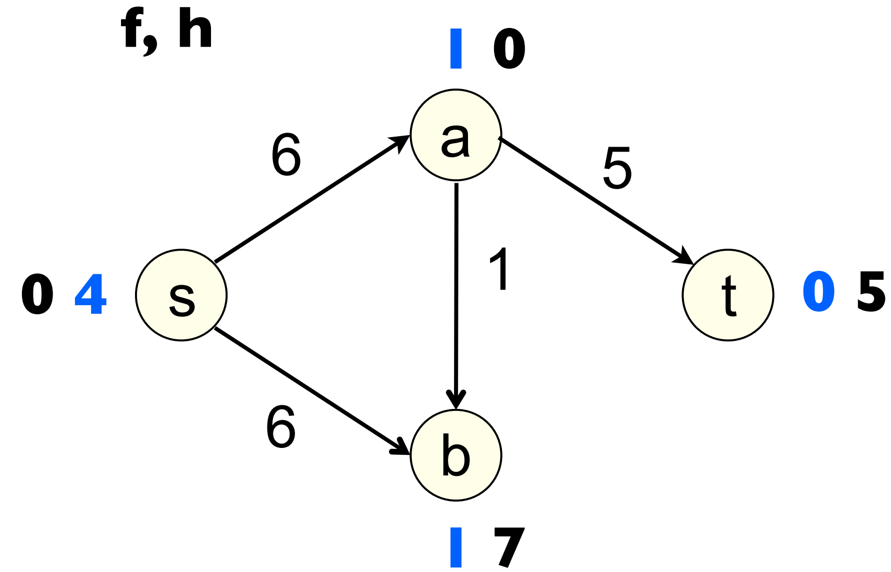
Push(v, w):
Applies if excess(v) > 0, h(w) < h(v)
- q = min(excess(v), c_f(v,w))
- Add q to f(v, w)

Relabel(v):
Applies if excess(v) > 0 and for all w s.t (v, w) in E_f, h(w) >= h(v)
- Increase h(v) by 1

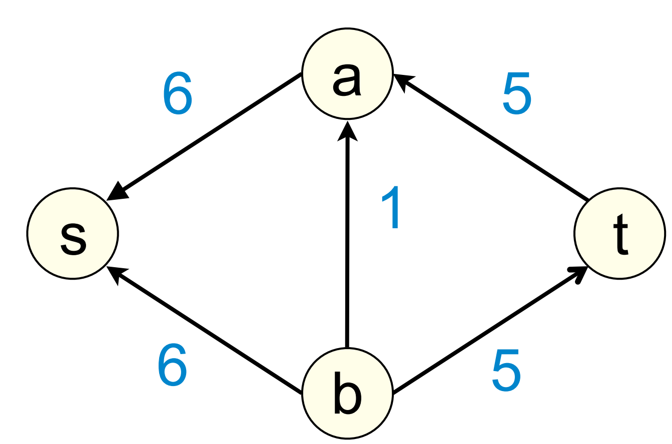
G_f (before)



f, h



G_f



Labels

Excesses

Pre-Flow Push: An Example

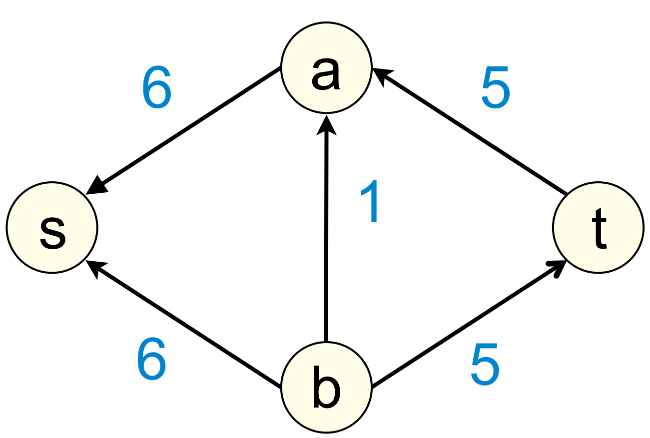
Start with labeling: $h(s) = n, h(t) = 0, h(v) = 0$, for other v
Start with preflow f: $f(e) = c(e)$ for $e = (s, v)$, $f(e) = 0$, ow

While there is a node (other than t) with positive excess
  Pick a node v with excess(v) > 0
  If there is an edge (v, w) in E_f s. t. push(v, w) applies
    Push(v, w)
  Else
    Relabel(v)

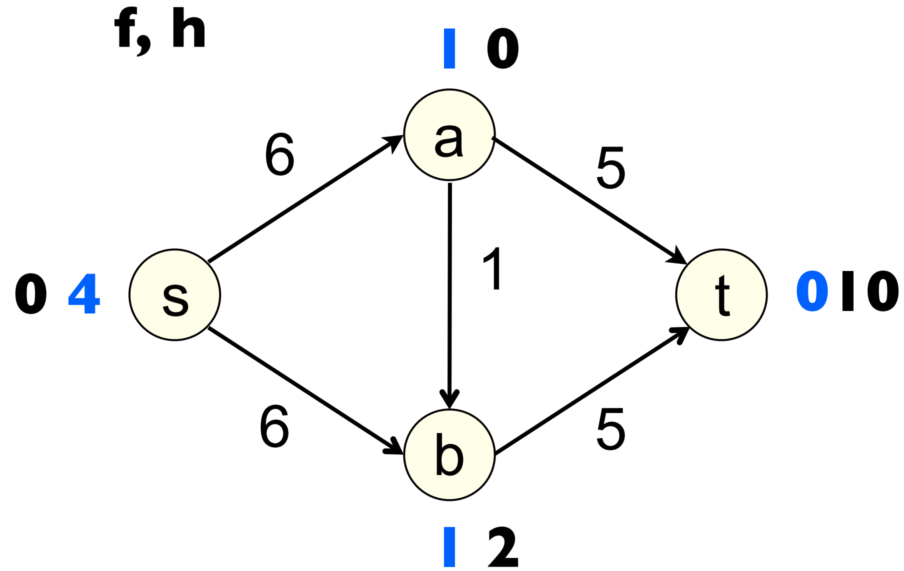
Push(v, w):
Applies if excess(v) > 0, $h(w) < h(v)$
  $q = \min(\text{excess}(v), c_f(v,w))$
  Add q to f(v, w)

Relabel(v):
Applies if excess(v) > 0 and for all w s.t (v, w) in E_f, $h(w) >= h(v)$
  Increase h(v) by 1

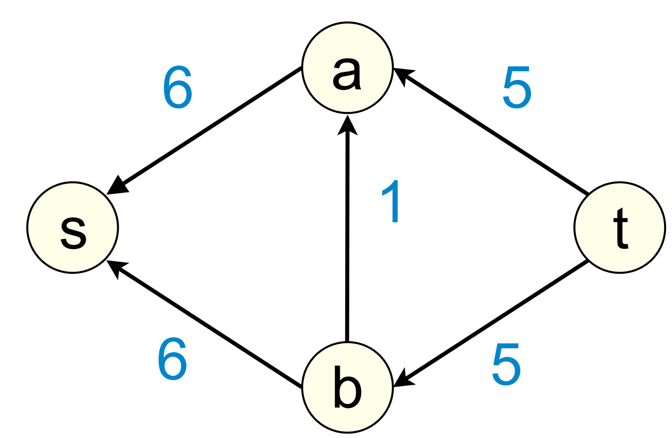
G_f (before)



f, h



G_f



Labels
Excesses

Pre-Flow Push: An Example

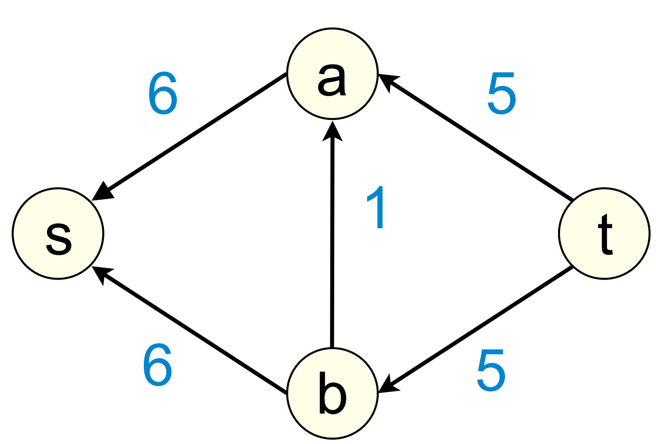
Start with labeling: h(s) = n, h(t) = 0, h(v) = 0, for other v
Start with preflow f: f(e) = c(e) for e = (s, v), f(e) = 0, ow

While there is a node (other than t) with positive excess
- Pick a node v with excess(v) > 0
- If there is an edge (v, w) in E_f s. t. push(v, w) applies
  - Push(v, w)
- Else
  - Relabel(v)

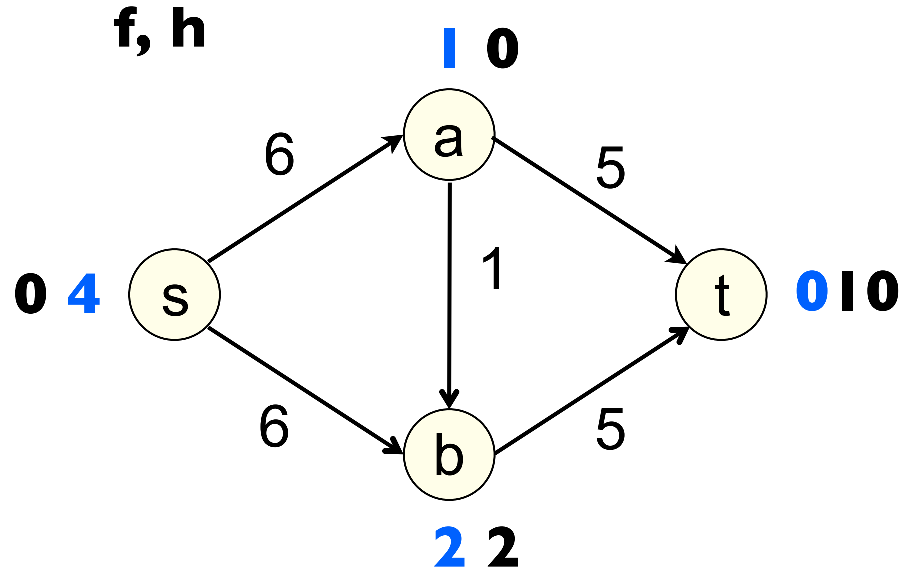
Push(v, w):
Applies if excess(v) > 0, h(w) < h(v)
- q = min(excess(v), c_f(v,w))
- Add q to f(v, w)

Relabel(v):
Applies if excess(v) > 0 and for all w s.t (v, w) in E_f, h(w) >= h(v)
- Increase h(v) by 1

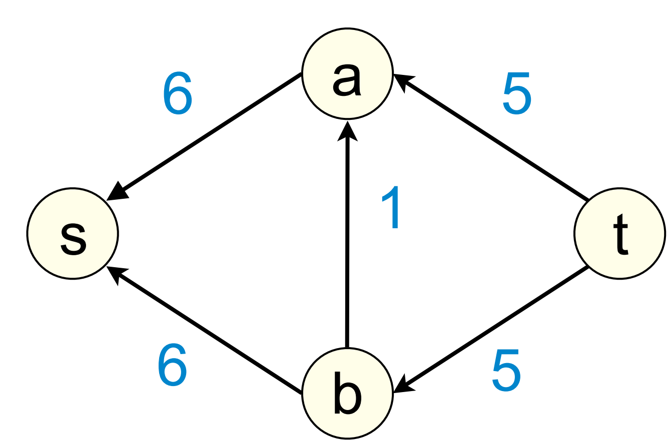
G_f (before)



f, h



G_f



Labels

Excesses

Pre-Flow Push: An Example

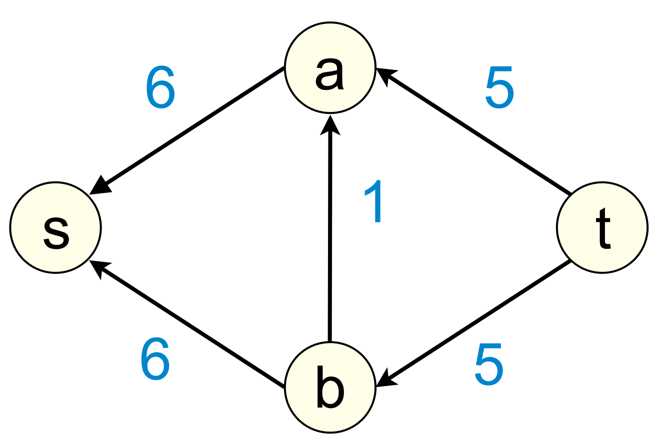
Start with labeling: $h(s) = n, h(t) = 0, h(v) = 0$, for other v
Start with preflow f: $f(e) = c(e)$ for $e = (s, v)$, $f(e) = 0$, ow

While there is a node (other than t) with positive excess
- Pick a node v with excess(v) > 0
- If there is an edge (v, w) in E_f s. t. push(v, w) applies
  - Push(v, w)
- Else
  - Relabel(v)

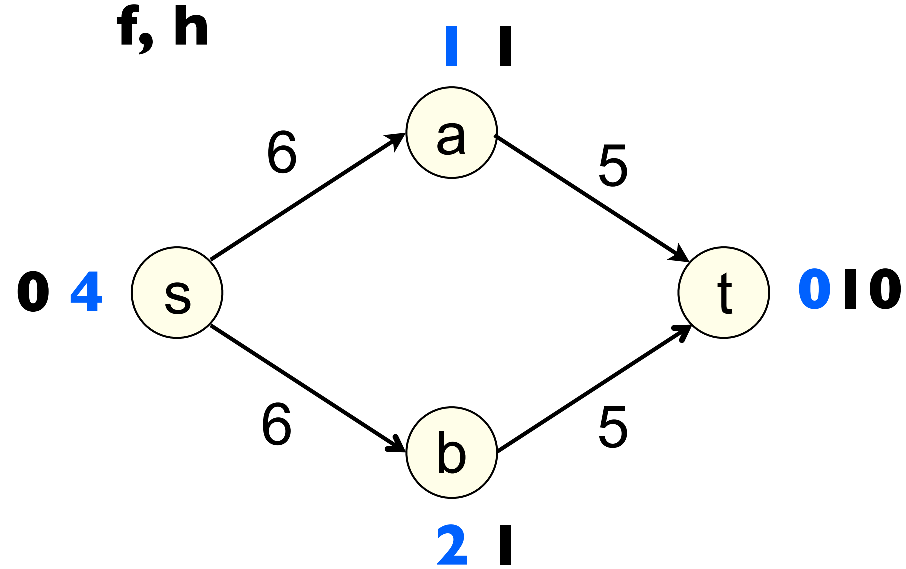
Push(v, w):
Applies if excess(v) > 0, $h(w) < h(v)$
- $q = \min(\text{excess}(v), c_f(v,w))$
- Add q to f(v, w)

Relabel(v):
Applies if excess(v) > 0 and for all w s.t (v, w) in E_f, $h(w) >= h(v)$
- Increase h(v) by 1

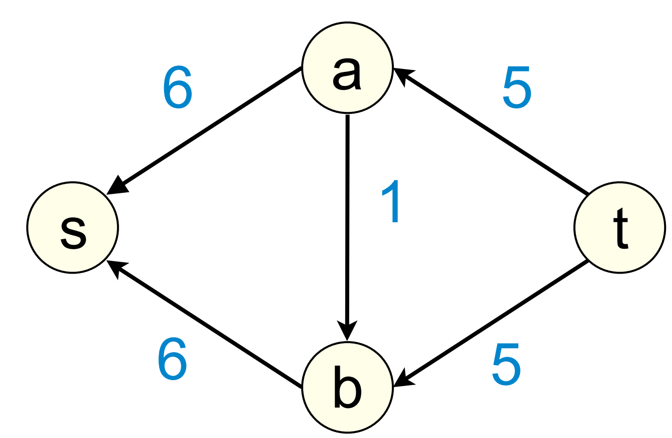
G_f (before)



f, h



G_f



Labels

Excesses

Pre-Flow Push: An Example

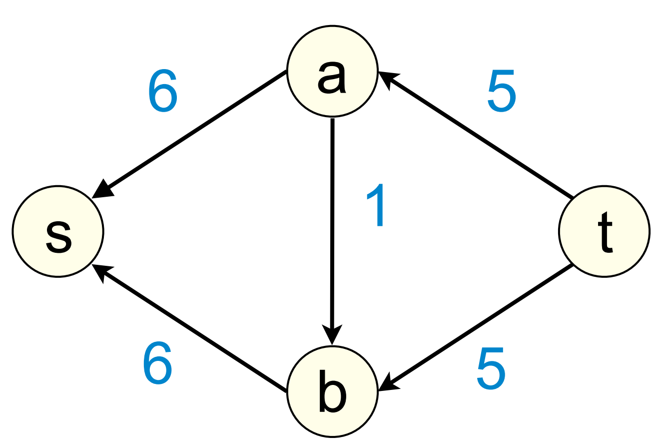
Start with labeling:h(s) = n,h(t) = 0,h(v) = 0, for other v
Start with preflow f: f(e) = c(e) for e = (s, v), f(e) = 0, ow

While there is a node (other than t) with positive excess
    Pick a node v with excess(v) > 0
    If there is an edge (v, w) in E_f s. t. push(v, w) applies
        Push(v, w)
    Else
        Relabel(v)

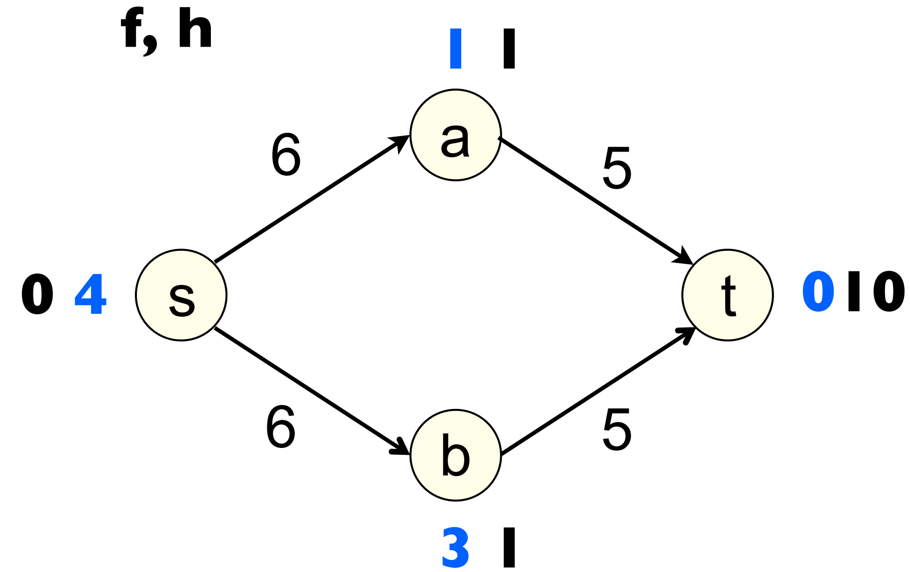
Push(v, w):
Applies if excess(v) > 0, h(w) < h(v)
    q = min(excess(v), c_f(v,w))
    Add q to f(v, w)

Relabel(v):
Applies if excess(v) > 0 and for all w s.t (v, w) in E_f, h(w) >= h(v)
    Increase h(v) by 1

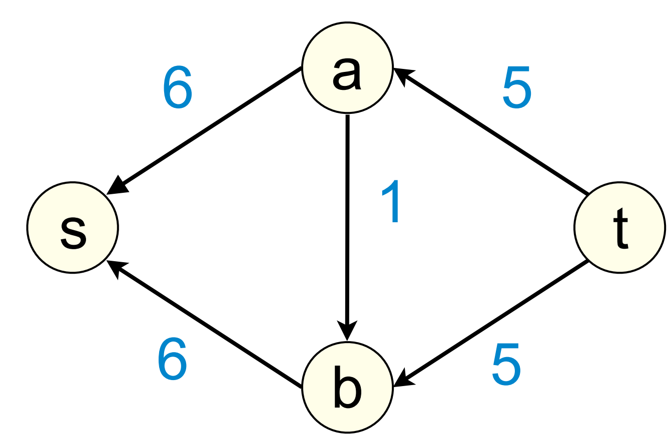
G_f (before)



f, h



G_f



Labels

Excesses

Pre-Flow Push: An Example

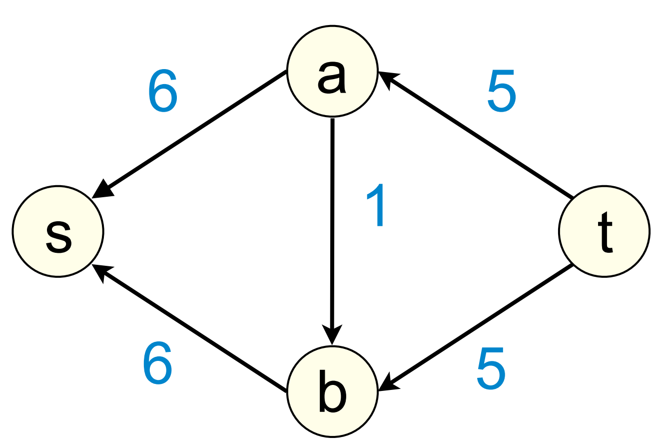
Start with labeling: h(s) = n, h(t) = 0, h(v) = 0, for other v
Start with preflow f: f(e) = c(e) for e = (s, v), f(e) = 0, ow

While there is a node (other than t) with positive excess
- Pick a node v with excess(v) > 0
- If there is an edge (v, w) in E_f s. t. push(v, w) applies
  - Push(v, w)
- Else
  - Relabel(v)

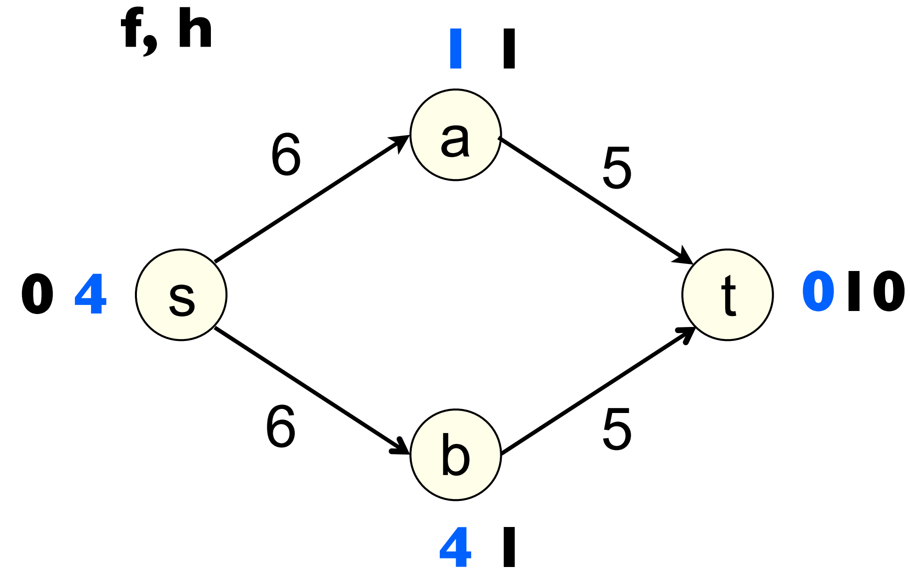
Push(v, w):
Applies if excess(v) > 0, h(w) < h(v)
- q = min(excess(v), c_f(v,w))
- Add q to f(v, w)

Relabel(v):
Applies if excess(v) > 0 and for all w s.t (v, w) in E_f, h(w) >= h(v)
- Increase h(v) by 1

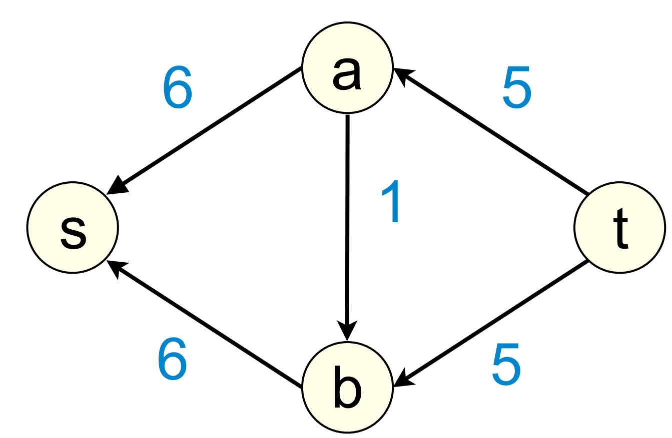
G_f (before)



f, h



G_f



Labels

Excesses

Pre-Flow Push: An Example

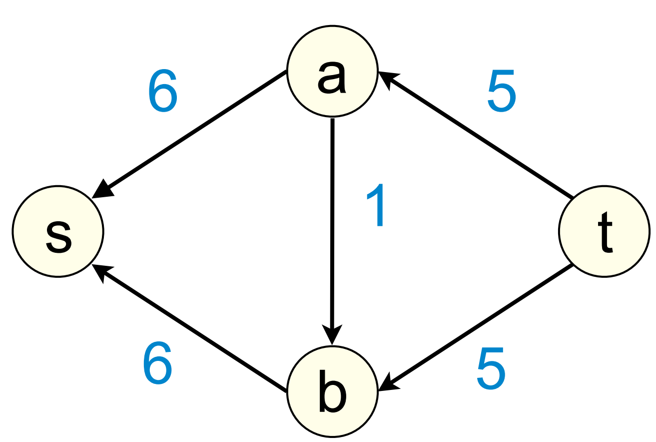
Start with labeling: $h(s) = n, h(t) = 0, h(v) = 0$, for other v
Start with preflow f: $f(e) = c(e)$ for $e = (s, v)$, $f(e) = 0$, ow

While there is a node (other than t) with positive excess
- Pick a node v with excess(v) > 0
- If there is an edge (v, w) in E_f s. t. push(v, w) applies
  - Push(v, w)
- Else
  - Relabel(v)

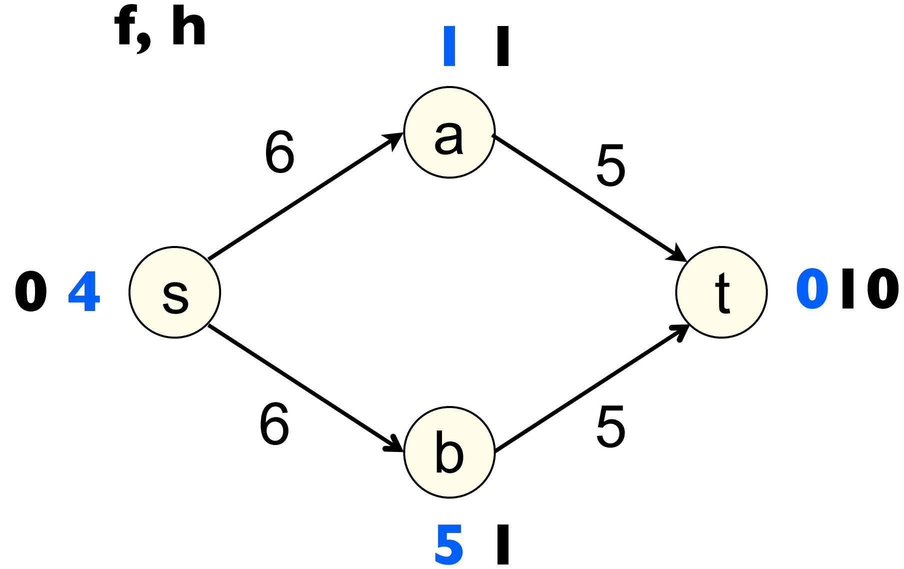
Push(v, w):
Applies if excess(v) > 0, $h(w) < h(v)$
- $q = \min(\text{excess}(v), c_f(v,w))$
- Add q to f(v, w)

Relabel(v):
Applies if excess(v) > 0 and for all w s.t (v, w) in E_f, $h(w) >= h(v)$
- Increase h(v) by 1

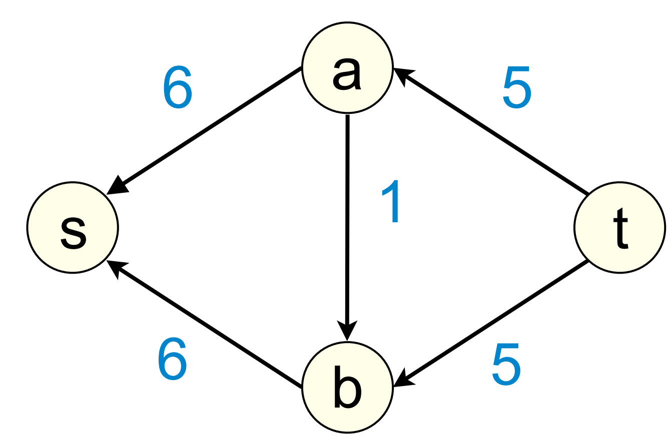
G_f (before)



f, h



G_f



Labels

Excesses

Pre-Flow Push: An Example

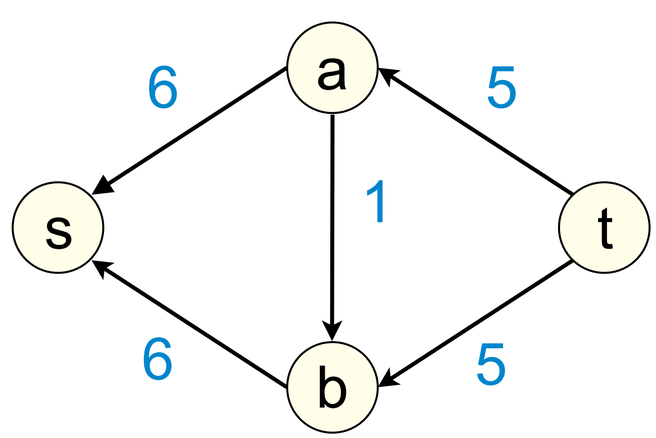
Start with labeling:h(s) = n,h(t) = 0,h(v) = 0, for other v
Start with preflow f: f(e) = c(e) for e = (s, v), f(e) = 0, ow

While there is a node (other than t) with positive excess
- Pick a node v with excess(v) > 0
- If there is an edge (v, w) in E_f s. t. push(v, w) applies
  - Push(v, w)
- Else
  - Relabel(v)

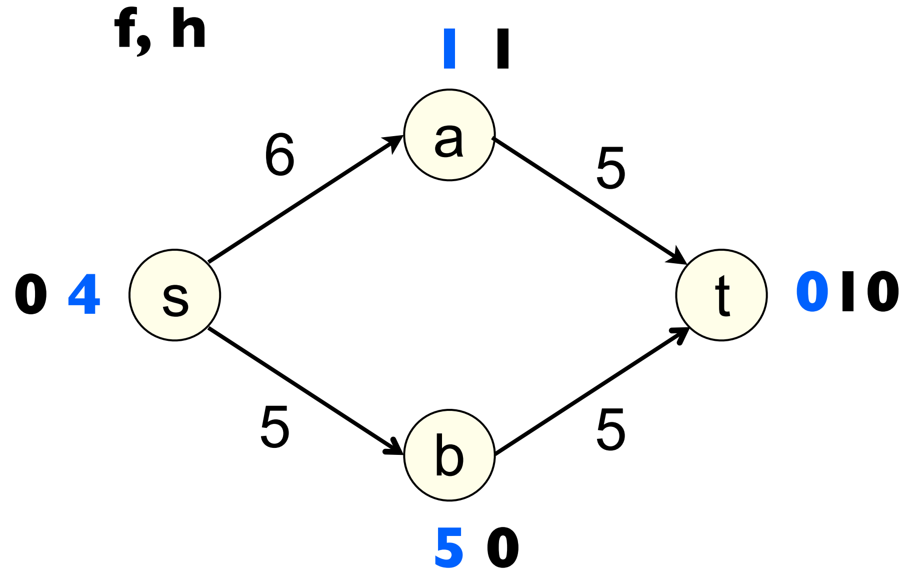
Push(v, w):
Applies if excess(v) > 0, h(w) < h(v)
- q = min(excess(v), c_f(v,w))
- Add q to f(v, w)

Relabel(v):
Applies if excess(v) > 0 and for all w s.t (v, w) in E_f, h(w) >= h(v)
- Increase h(v) by 1

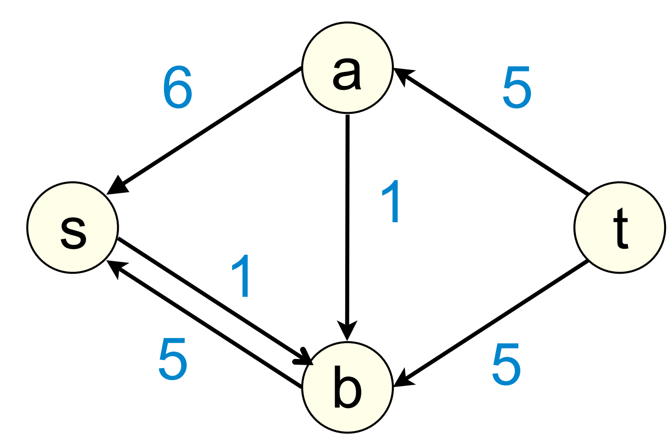
G_f (before)



f, h



G_f



Labels

Excesses

Pre-Flow Push: An Example

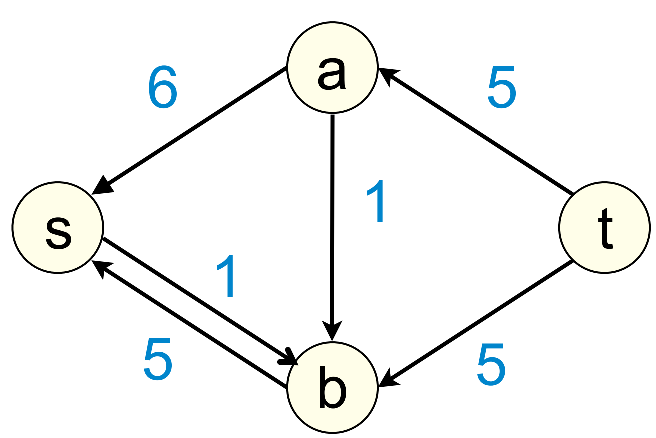
Start with labeling: $h(s) = n, h(t) = 0, h(v) = 0$, for other v
Start with preflow f: $f(e) = c(e)$ for $e = (s, v)$, $f(e) = 0$, ow

While there is a node (other than t) with positive excess
  Pick a node v with excess(v) > 0
  If there is an edge (v, w) in E_f s. t. push(v, w) applies
    Push(v, w)
  Else
    Relabel(v)

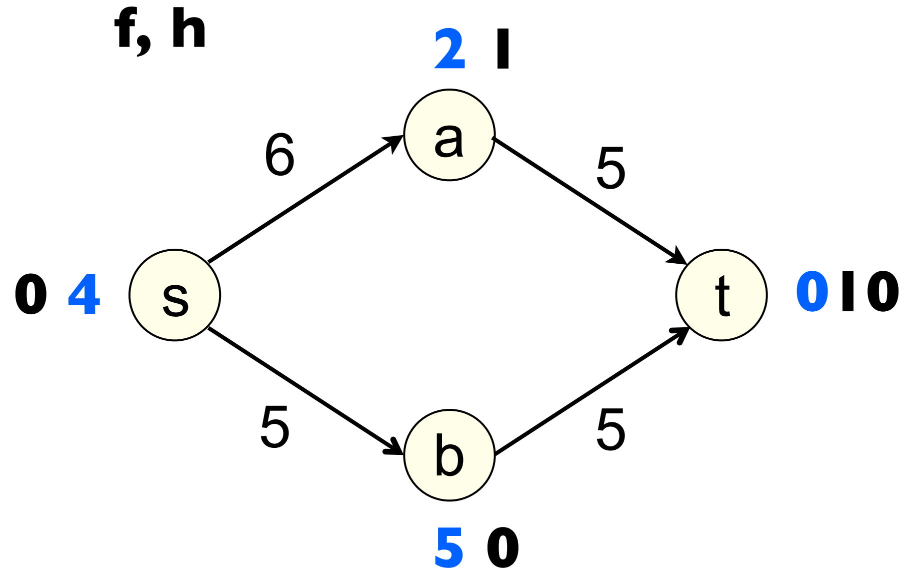
Push(v, w):
Applies if excess(v) > 0, h(w) < h(v)
  q = min(excess(v), c_f(v,w))
  Add q to f(v, w)

Relabel(v):
Applies if excess(v) > 0 and for all w s.t (v, w) in E_f, h(w) >= h(v)
  Increase h(v) by 1

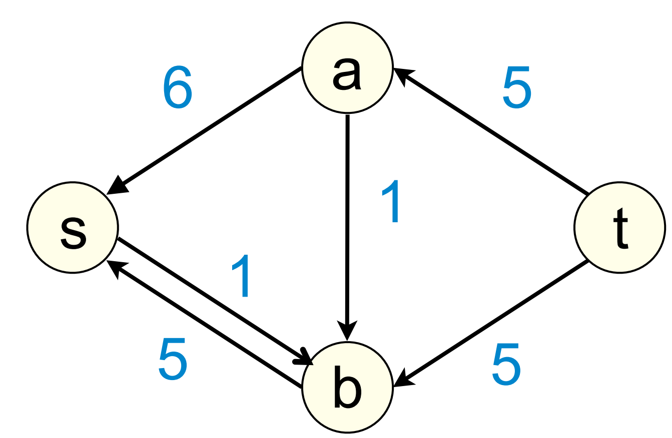
G_f (before)



f, h



G_f



Labels

Excesses

Pre-Flow Push: An Example

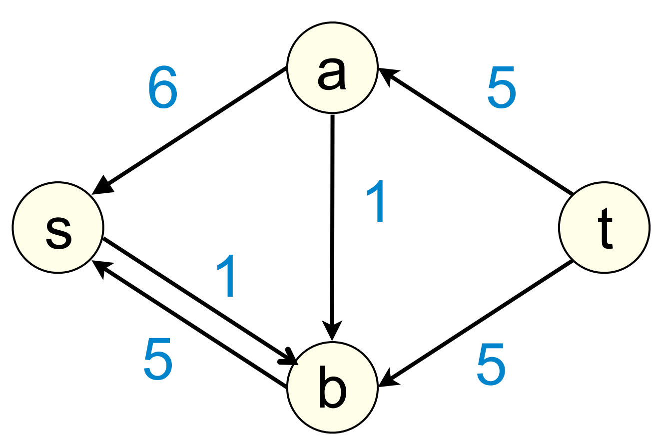
Start with labeling: $h(s) = n, h(t) = 0, h(v) = 0$, for other v
Start with preflow f: $f(e) = c(e)$ for $e = (s, v)$, $f(e) = 0$, ow

While there is a node (other than t) with positive excess
- Pick a node v with excess(v) > 0
- If there is an edge (v, w) in E_f s. t. push(v, w) applies
  - Push(v, w)
- Else
  - Relabel(v)

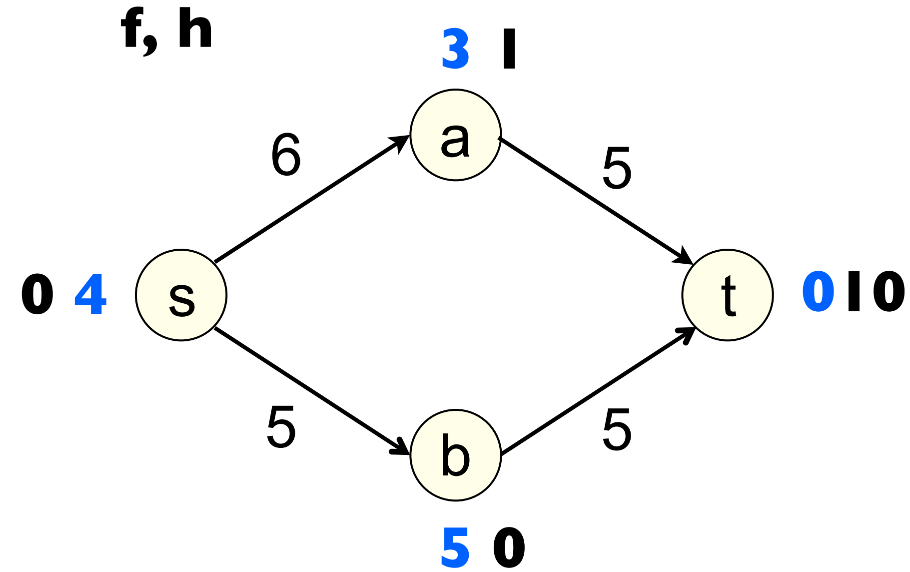
Push(v, w):
Applies if excess(v) > 0, $h(w) < h(v)$
- $q = \min(\text{excess}(v), c_f(v,w))$
- Add q to f(v, w)

Relabel(v):
Applies if excess(v) > 0 and for all w s.t (v, w) in E_f, $h(w) >= h(v)$
- Increase h(v) by 1

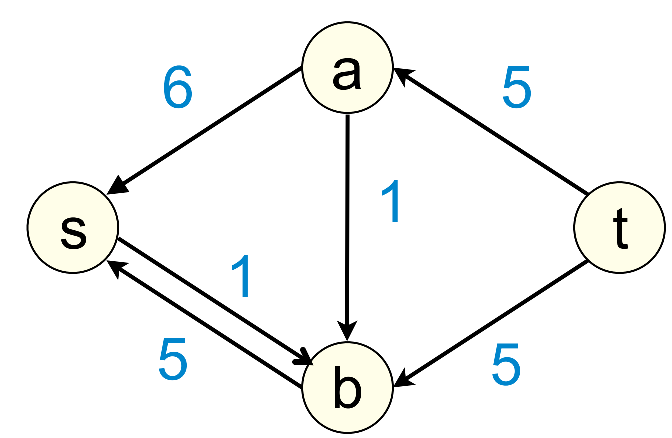
G_f (before)



f, h



G_f



Labels

Excesses

Pre-Flow Push: An Example

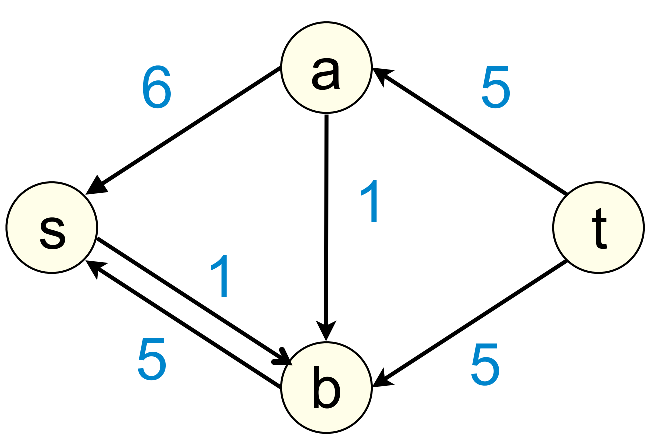
Start with labeling: $h(s) = n, h(t) = 0, h(v) = 0$, for other v
Start with preflow f: $f(e) = c(e)$ for $e = (s, v)$, $f(e) = 0$, ow

While there is a node (other than t) with positive excess
  Pick a node v with $excess(v) > 0$
  If there is an edge (v, w) in E_f s. t. push(v, w) applies
    Push(v, w)
  Else
    Relabel(v)

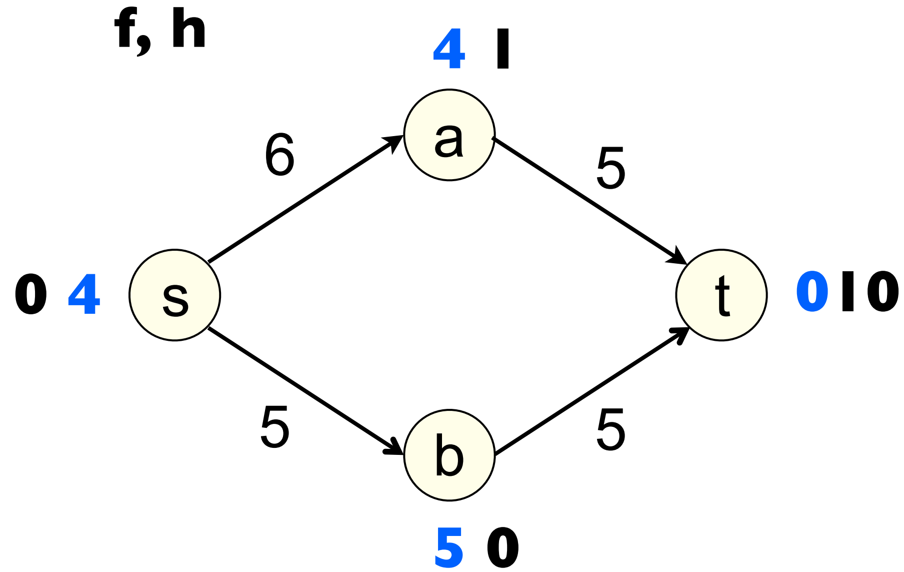
Push(v, w):
Applies if $excess(v) > 0$, $h(w) < h(v)$
  $q = \min(excess(v), c_f(v,w))$
  Add q to f(v, w)

Relabel(v):
Applies if $excess(v) > 0$ and for all w s.t (v, w) in E_f, $h(w) >= h(v)$
  Increase h(v) by 1

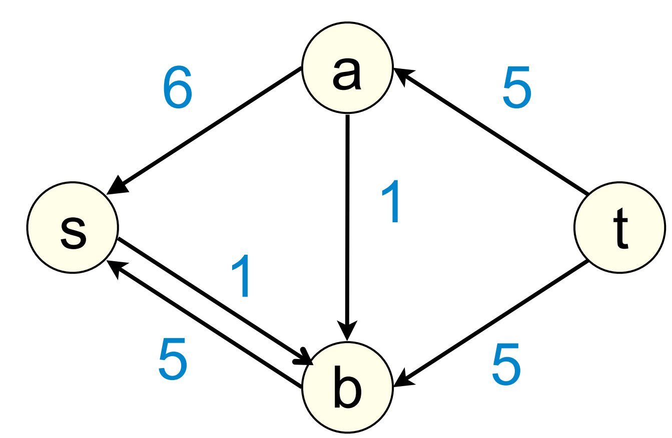
G_f (before)



f, h



G_f



Labels

Excesses

Pre-Flow Push: An Example

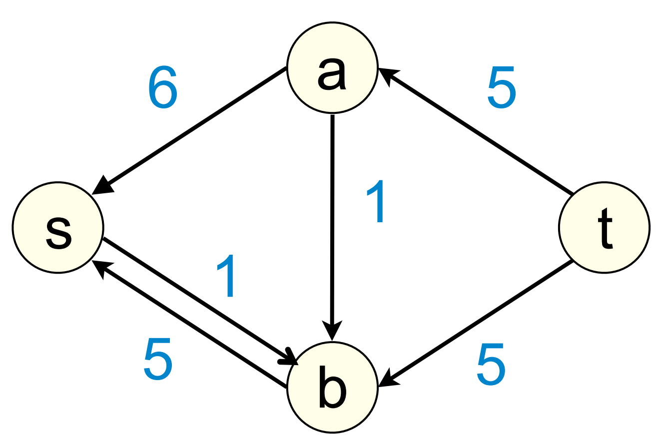
Start with labeling: $h(s) = n, h(t) = 0, h(v) = 0$, for other v
Start with preflow f: $f(e) = c(e)$ for $e = (s, v)$, $f(e) = 0$, ow

While there is a node (other than t) with positive excess
  Pick a node v with excess(v) > 0
  If there is an edge (v, w) in E_f s. t. push(v, w) applies
    Push(v, w)
  Else
    Relabel(v)

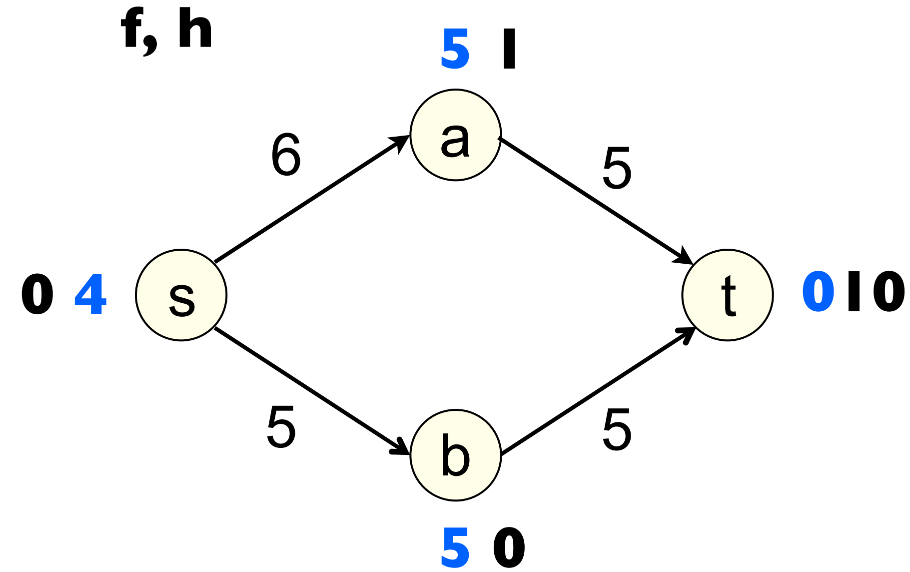
Push(v, w):
Applies if excess(v) > 0, h(w) < h(v)
  q = min(excess(v), c_f(v,w))
  Add q to f(v, w)

Relabel(v):
Applies if excess(v) > 0 and for all
w s.t (v, w) in E_f, h(w) >= h(v)
  Increase h(v) by 1

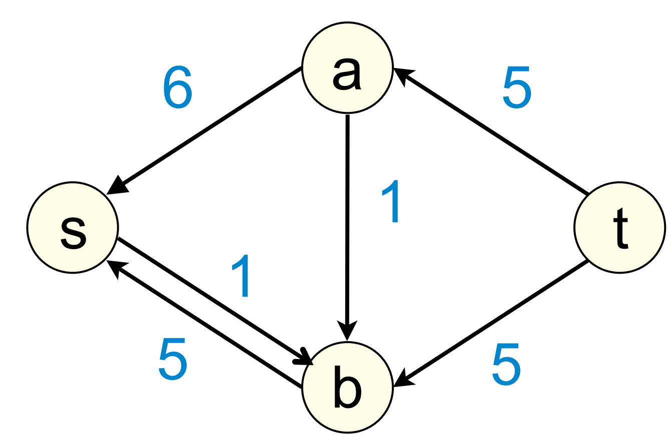
G_f (before)



f, h



G_f



Labels

Excesses

Pre-Flow Push: An Example

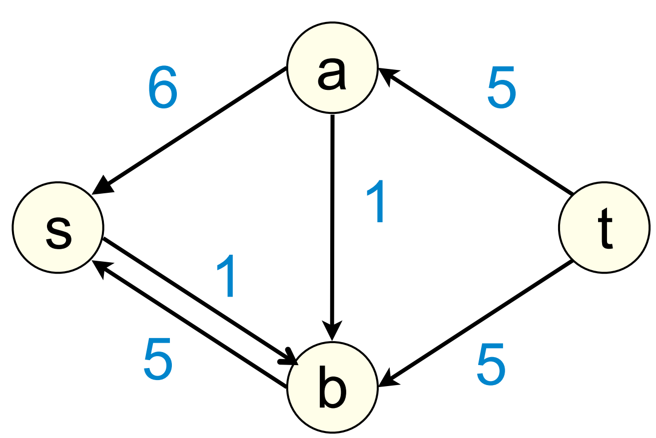
Start with labeling: h(s) = n, h(t) = 0, h(v) = 0, for other v
Start with preflow f: f(e) = c(e) for e = (s, v), f(e) = 0, ow

While there is a node (other than t) with positive excess
- Pick a node v with excess(v) > 0
- If there is an edge (v, w) in E_f s. t. push(v, w) applies
  - Push(v, w)
- Else
  - Relabel(v)

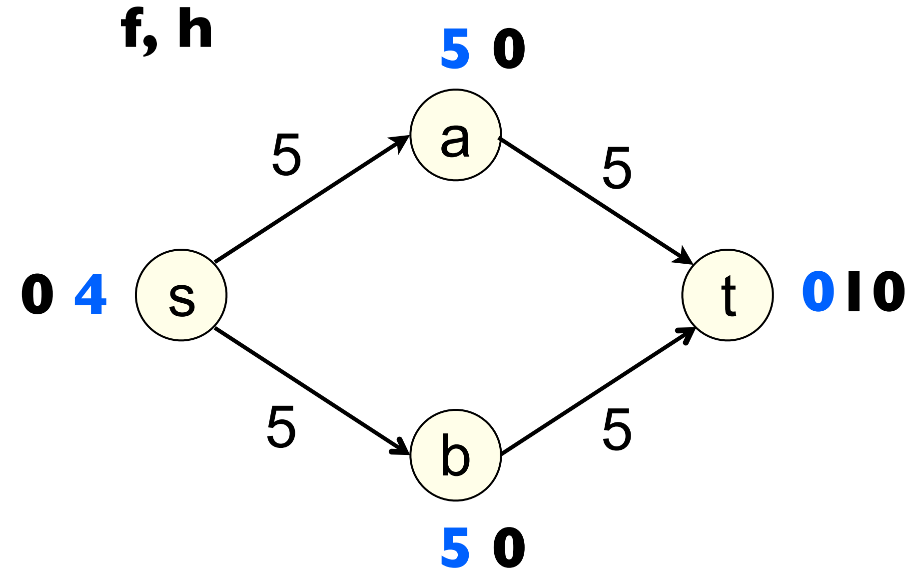
Push(v, w):
Applies if excess(v) > 0, h(w) < h(v)
- q = min(excess(v), c_f(v,w))
- Add q to f(v, w)

Relabel(v):
Applies if excess(v) > 0 and for all w s.t (v, w) in E_f, h(w) >= h(v)
- Increase h(v) by 1

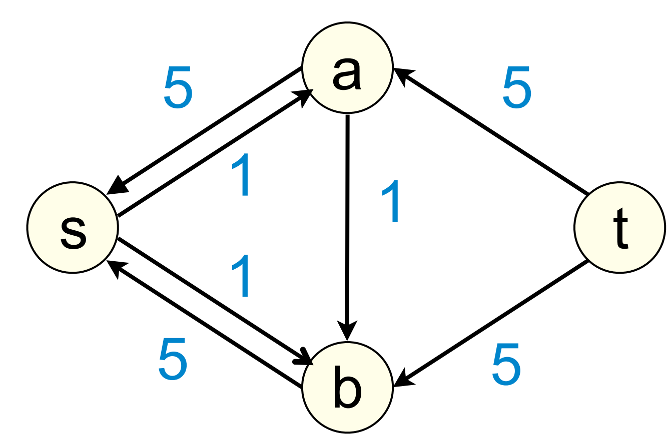
G_f (before)



f, h



G_f



Labels

Excesses

Pre-Flow Push

- Algorithm
- Correctness
- Running Time Analysis

Correctness: Proof Outline

Three Steps:

- Compatibility: Show that the preflow f and the labeling h maintained by the algorithm always obeys a **compatibility** property

- If a flow f is compatible with some labeling, then f is a max-flow

- Preflow-push outputs a flow on termination

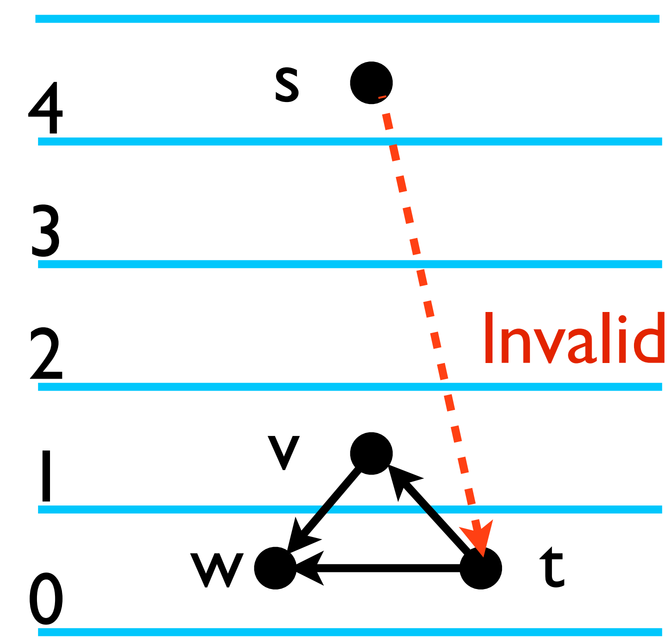
Correctness: Compatible Pre-Flows

Preflow: A function f: E $\rightarrow$ R is a preflow if:

1. **Capacity Constraints:** 0 <= f(e) <= c(e)
2. Instead of conservation constraints:

$$\sum_{e\ into\ v} f(e) - \sum_{e\ out\ of\ v} f(e) \geq 0$$

Excess(v) = $\sum_{e\ into\ v} f(e) - \sum_{e\ out\ of\ v} f(e)$



Preflow f and labeling h are compatible if:

1. h(s) = n, h(t) = 0
2. For all edges (v, w) in the residual graph G_f, h(v) <= h(w) + 1

PreFlow Push: Correctness

Start with labeling: $h(s) = n, h(t) = 0, h(v) = 0$, for other v
Start with preflow f: $f(e) = c(e)$ for $e = (s, v)$, $f(e) = 0$, ow

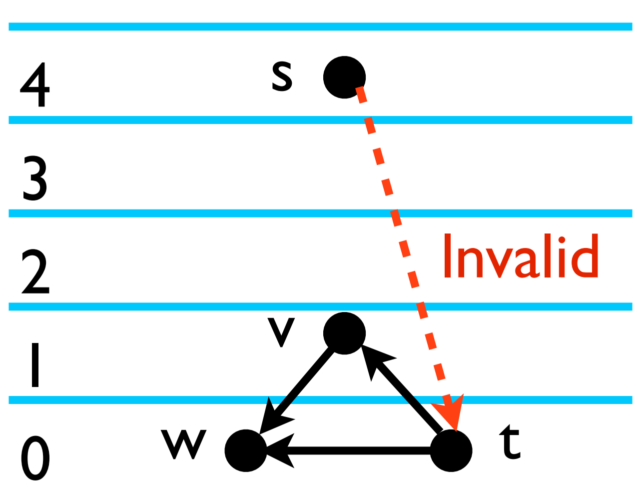
While there is a node (other than t) with positive excess
  Pick a node v with $excess(v) > 0$
  If there is an edge (v, w) in E_f s. t. push(v, w) applies
    Push(v, w)
  Else
    Relabel(v)

Push(v, w):
Applies if $excess(v) > 0$, $h(w) < h(v)$
  $q = \min(excess(v), c_f(v,w))$
  Add q to f(v, w)

Relabel(v):
Applies if $excess(v) > 0$ and for all w s.t (v, w) in E_f, $h(w) >= h(v)$
  Increase h(v) by 1

Preflow f and labeling l are **compatible** if:
1. $h(s) = n$, $h(t) = 0$
2. For all edges (v, w) in G_f, $h(v) <= h(w) + 1$

Invariant: Preflow f and labeling h are always compatible over the Preflow-Push algorithm



PreFlow Push: Correctness

Start with labeling: $h(s) = n, h(t) = 0, h(v) = 0$, for other v
Start with preflow f: $f(e) = c(e)$ for $e = (s, v)$, $f(e) = 0$, ow

While there is a node (other than t) with positive excess
    Pick a node v with excess(v) > 0
    If there is an edge (v, w) in E_f s. t. push(v, w) applies
        Push(v, w)
    Else
        Relabel(v)

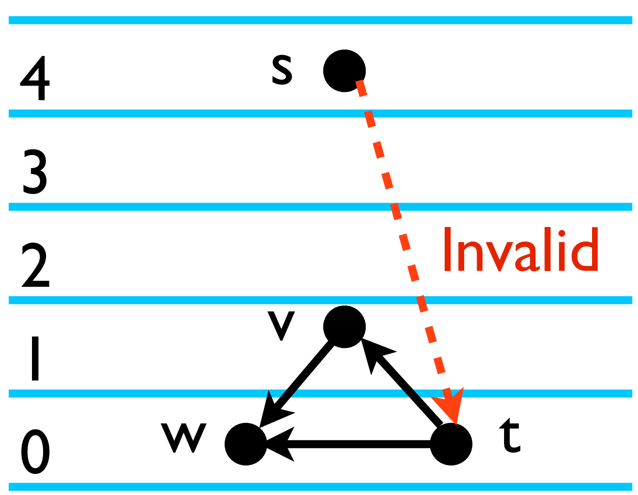
Push(v, w):
Applies if excess(v) > 0, $h(w) < h(v)$
    $q = \min(\text{excess}(v), c_f(v,w))$
    Add q to f(v, w)

Relabel(v):
Applies if excess(v) > 0 and for all w s.t (v, w) in E_f, $h(w) >= h(v)$
    Increase h(v) by 1

Preflow f and labeling l are **compatible** if:
1. $h(s) = n$, $h(t) = 0$
2. For all edges (v, w) in G_f, $h(v) <= h(w) + 1$

Invariant: Preflow f and labeling h are always compatible over the Preflow-Push algorithm

Proof: By induction.



PreFlow Push: Correctness

Start with labeling: $h(s) = n, h(t) = 0, h(v) = 0$, for other v
Start with preflow f: $f(e) = c(e)$ for $e = (s, v)$, $f(e) = 0$, ow

While there is a node (other than t) with positive excess
  Pick a node v with excess(v) > 0
  If there is an edge (v, w) in E_f s. t. push(v, w) applies
    Push(v, w)
  Else
    Relabel(v)

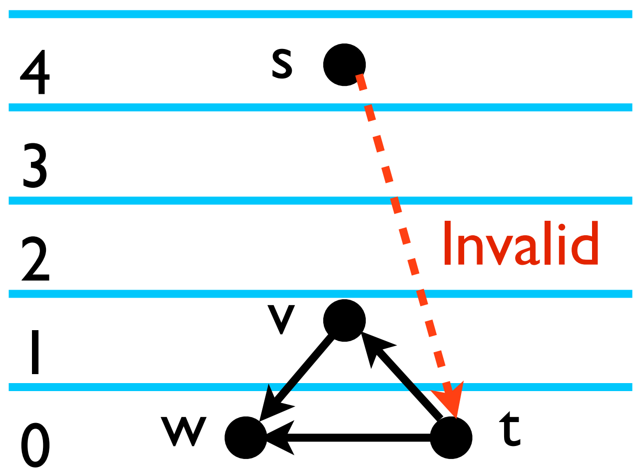
Push(v, w):
Applies if excess(v) > 0, h(w) < h(v)
  q = min(excess(v), c_f(v,w))
  Add q to f(v, w)

Relabel(v):
Applies if excess(v) > 0 and for all w s.t (v, w) in E_f, h(w) >= h(v)
  Increase h(v) by 1

Preflow f and labeling l are **compatible** if:
  1. $h(s) = n$, $h(t) = 0$
  2. For all edges (v, w) in G_f, $h(v) \le h(w) + 1$

Invariant: Preflow f and labeling h are always compatible over the Preflow-Push algorithm

Proof: By induction. Initially, compatible, as G_f has no (s, v) edges



PreFlow Push: Correctness

Start with labeling:h(s) = n,h(t) = 0,h(v) = 0, for other v
Start with preflow f: f(e) = c(e) for e = (s, v), f(e) = 0, ow

While there is a node (other than t) with positive excess
    Pick a node v with excess(v) > 0
    If there is an edge (v, w) in E_f s. t. push(v, w) applies
        Push(v, w)
    Else
        Relabel(v)

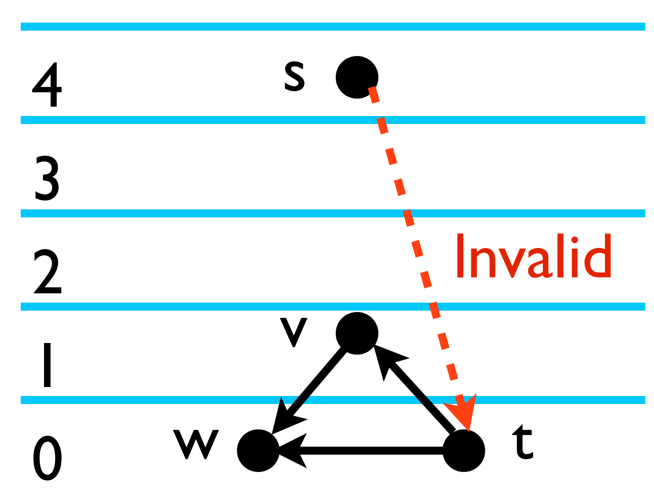
Push(v, w):
Applies if excess(v) > 0, h(w) < h(v)
    q = min(excess(v), c_f(v,w))
    Add q to f(v, w)

Relabel(v):
Applies if excess(v) > 0 and for all w s.t (v, w) in E_f, h(w) >= h(v)
    Increase h(v) by 1

Preflow f and labeling l are **compatible** if:
1. h(s) = n, h(t) = 0
2. For all edges (v, w) in G_f, h(v) <= h(w) + 1

Invariant: Preflow f and labeling h are always compatible over the Preflow-Push algorithm

Proof: By induction. Initially, compatible, as G_f has no (s, v) edges
Suppose f and h are compatible at time t. At time t+1:



PreFlow Push: Correctness

Start with labeling: $h(s) = n, h(t) = 0, h(v) = 0$, for other v
Start with preflow f: $f(e) = c(e)$ for $e = (s, v)$, $f(e) = 0$, ow

While there is a node (other than t) with positive excess
  Pick a node v with excess(v) > 0
  If there is an edge (v, w) in E_f s. t. push(v, w) applies
    Push(v, w)
  Else
    Relabel(v)

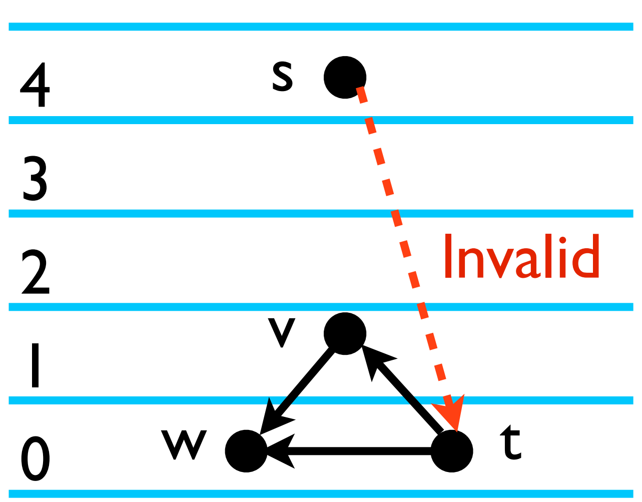
Push(v, w):
Applies if excess(v) > 0, $h(w) < h(v)$
  $q = \min(\text{excess}(v), c_f(v,w))$
  Add q to f(v, w)

Relabel(v):
Applies if excess(v) > 0 and for all w s.t (v, w) in E_f, $h(w) >= h(v)$
  Increase h(v) by 1

Preflow f and labeling l are **compatible** if:
  1. $h(s) = n, h(t) = 0$
  2. For all edges (v, w) in G_f, $h(v) <= h(w) + 1$

Invariant: Preflow f and labeling h are always compatible over the Preflow-Push algorithm

Proof: By induction. Initially, compatible, as G_f has no (s, v) edges
Suppose f and h are compatible at time t. At time t+1:
  - **Relabel:** Labels increase only if no downward edges in G_f



PreFlow Push: Correctness

Start with labeling: $h(s) = n, h(t) = 0, h(v) = 0$, for other v
Start with preflow f: $f(e) = c(e)$ for $e = (s, v)$, $f(e) = 0$, ow

While there is a node (other than t) with positive excess
  Pick a node v with excess(v) > 0
  If there is an edge (v, w) in E_f s. t. push(v, w) applies
    Push(v, w)
  Else
    Relabel(v)

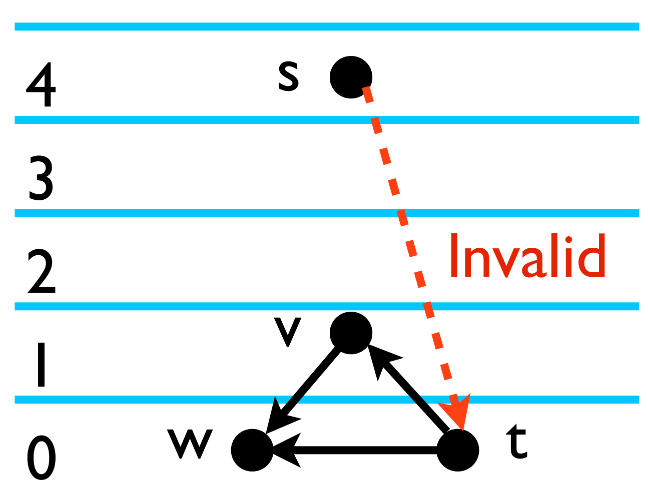
Push(v, w):
Applies if excess(v) > 0, $h(w) < h(v)$
  q = min(excess(v), $c_f(v,w)$)
  Add q to f(v, w)

Relabel(v):
Applies if excess(v) > 0 and for all w s.t (v, w) in E_f, $h(w) >= h(v)$
  Increase h(v) by 1

Preflow f and labeling l are **compatible** if:
1. $h(s) = n$, $h(t) = 0$
2. For all edges (v, w) in G_f, $h(v) <= h(w) + 1$

Invariant: Preflow f and labeling h are always compatible over the Preflow-Push algorithm

Proof: By induction. Initially, compatible, as G_f has no (s, v) edges
Suppose f and h are compatible at time t. At time t+1:
- **Relabel:** Labels increase only if no downward edges in G_f
- **Push:** Edges in G_f may be reversed.



PreFlow Push: Correctness

Start with labeling:h(s) = n,h(t) = 0,h(v) = 0, for other v
Start with preflow f: f(e) = c(e) for e = (s, v), f(e) = 0, ow

While there is a node (other than t) with positive excess
    Pick a node v with excess(v) > 0
    If there is an edge (v, w) in E_f s. t. push(v, w) applies
        Push(v, w)
    Else
        Relabel(v)

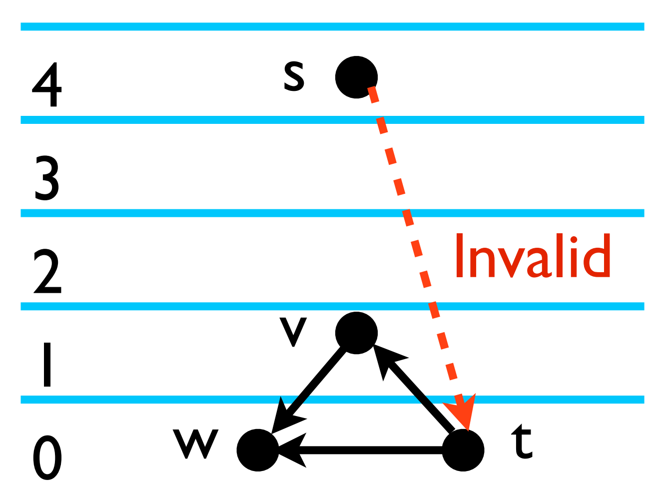
Push(v, w):
Applies if excess(v) > 0, h(w) < h(v)
    q = min(excess(v), c_f(v,w))
    Add q to f(v, w)

Relabel(v):
Applies if excess(v) > 0 and for all w s.t (v, w) in E_f, h(w) >= h(v)
    Increase h(v) by 1

Preflow f and labeling l are **compatible** if:
1. h(s) = n, h(t) = 0
2. For all edges (v, w) in G_f, h(v) <= h(w) + 1

Invariant: Preflow f and labeling h are always compatible over the Preflow-Push algorithm

Proof: By induction. Initially, compatible, as G_f has no (s, v) edges
Suppose f and h are compatible at time t. At time t+1:
- **Relabel:** Labels increase only if no downward edges in G_f
- **Push:** Edges in G_f may be reversed. If so, as we push from high to low h a downwards edge will become an upwards edge



Correctness: Proof Outline

Three Steps:

- Compatibility: Show that the preflow f and the labeling h maintained by the algorithm always obeys a **compatibility** property

- If a flow f is compatible with some labeling, then f is a max-flow

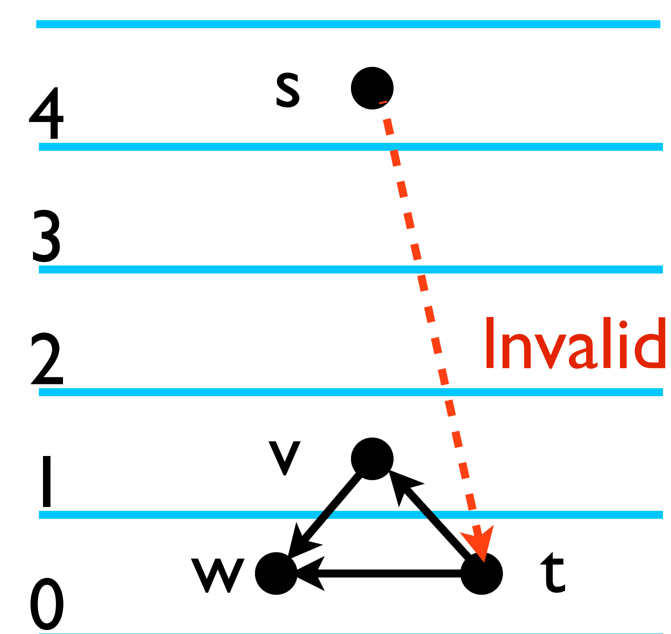
- Preflow-push outputs a flow on termination

Properties of Compatible PreFlows

Preflow f and labeling h are compatible if:

1. h(s) = n, h(t) = 0
2. For all edges (v, w) in the residual graph G_f, h(v) <= h(w) + 1

Property 1: If preflow f and labeling h are compatible, then there is no s-t path in G_f



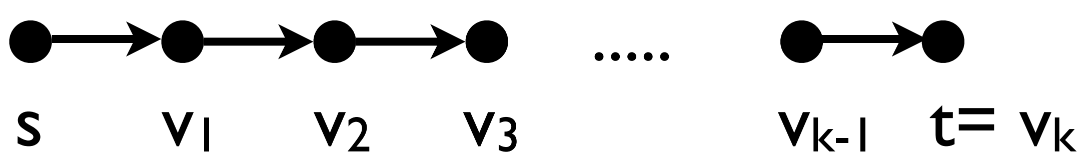
Properties of Compatible PreFlows

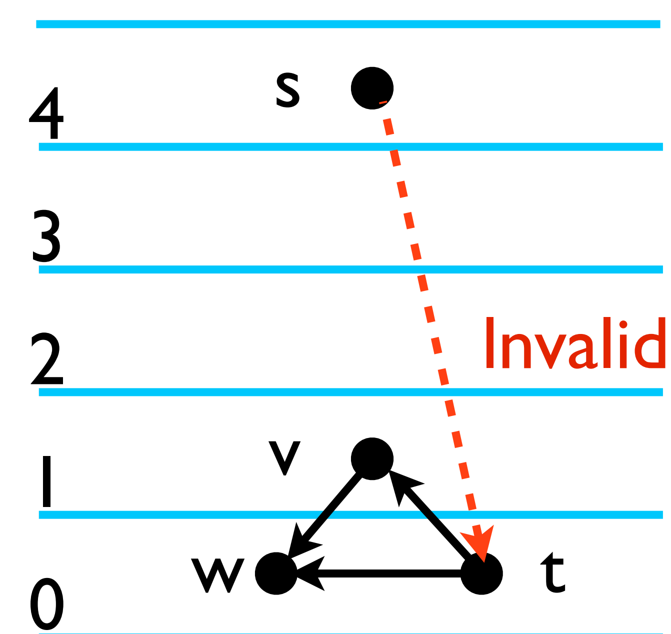
Preflow f and labeling h are compatible if:

1. $h(s) = n$, $h(t) = 0$
2. For all edges (v, w) in the residual graph G_f, $h(v) <= h(w) + 1$

Property 1: If preflow f and labeling h are compatible, then there is no s-t path in G_f

Proof: Suppose there is an s-t path in G_f



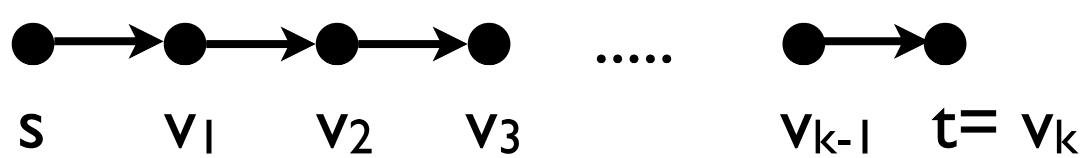


Properties of Compatible PreFlows

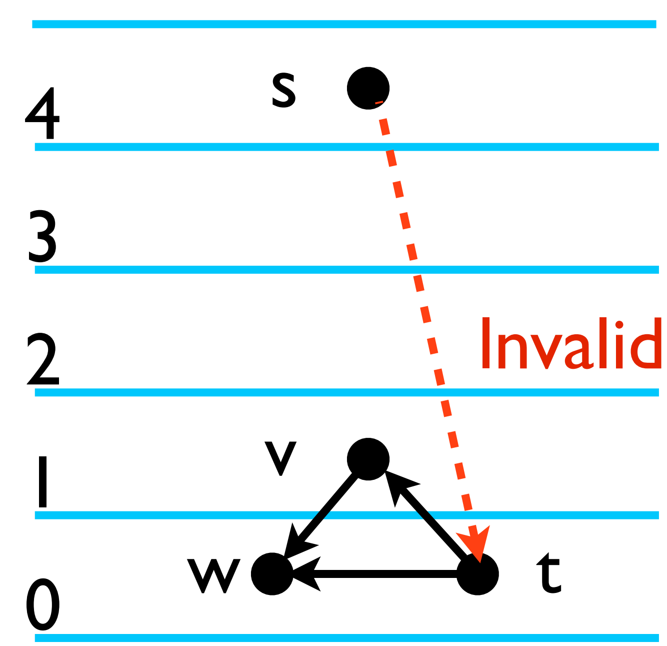
Preflow f and labeling h are compatible if:
1. $h(s) = n, h(t) = 0$
2. For all edges (v, w) in the residual graph G_f, $h(v) <= h(w) + 1$

Property 1: If preflow f and labeling h are compatible, then there is no s-t path in G_f

Proof: Suppose there is an s-t path in G_f



Due to compatibility,
$h(v_1) >= h(s) - 1 = n - 1$



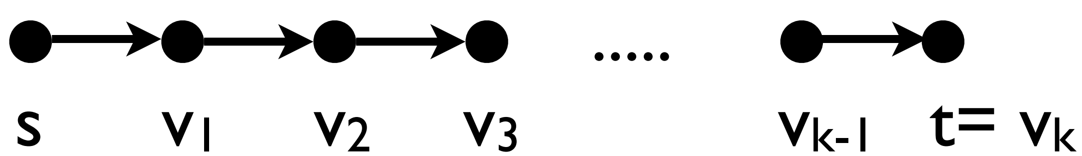
Properties of Compatible PreFlows

Preflow f and labeling h are compatible if:

1. $h(s) = n, h(t) = 0$
2. For all edges (v, w) in the residual graph G_f, $h(v) <= h(w) + 1$

Property 1: If preflow f and labeling h are compatible, then there is no s-t path in G_f

Proof: Suppose there is an s-t path in G_f

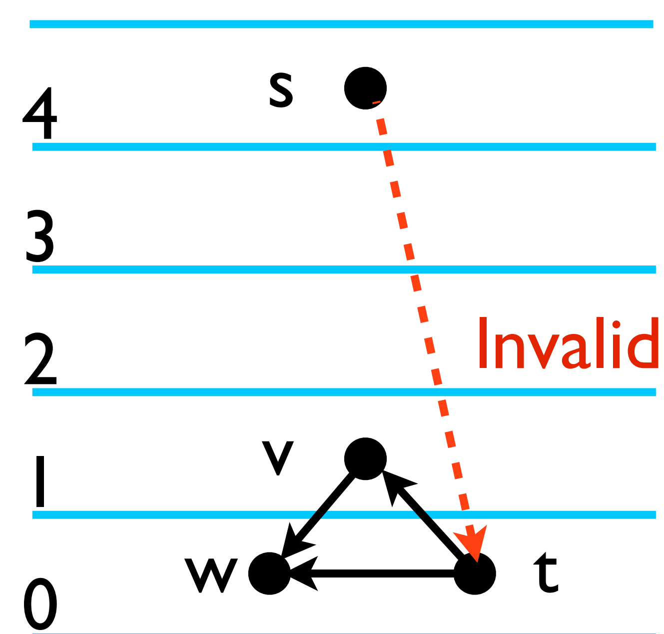


Due to compatibility,

$h(v_1) >= h(s) - 1 = n - 1$

$h(v_2) >= h(v_1) - 1 >= n - 2$

...



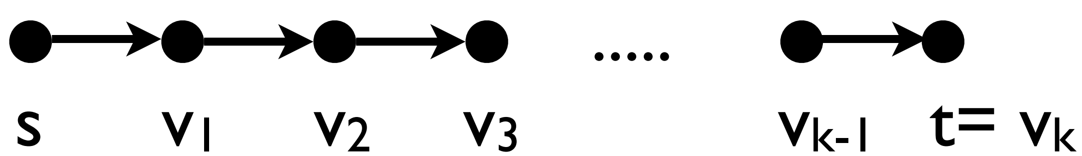
Properties of Compatible PreFlows

Preflow f and labeling h are compatible if:

1. $h(s) = n$, $h(t) = 0$
2. For all edges (v, w) in the residual graph G_f, $h(v) \leq h(w) + 1$

Property 1: If preflow f and labeling h are compatible, then there is no s-t path in G_f

Proof: Suppose there is an s-t path in G_f



Due to compatibility,

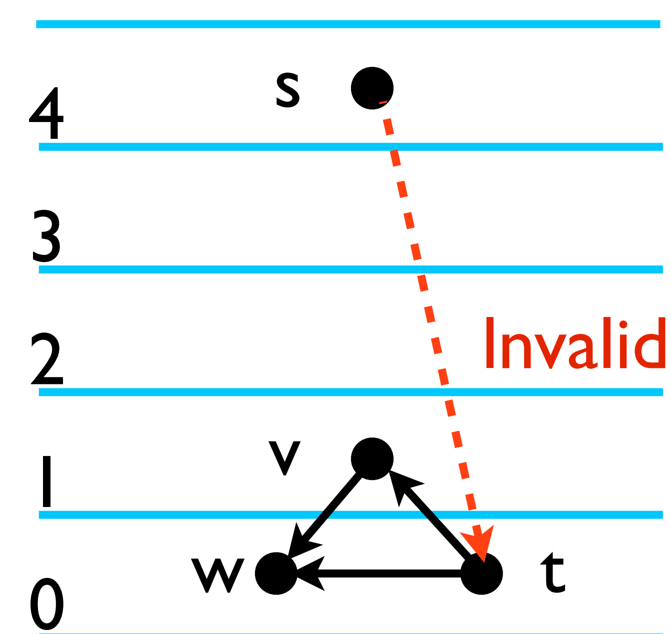
$h(v_1) >= h(s) - 1 = n - 1$

$h(v_2) >= h(v_1) - 1 >= n - 2$

...

$h(t) = h(v_k) - 1 >= n - k > 0$ (as $k < n$)

Contradiction!



Properties of Compatible PreFlows

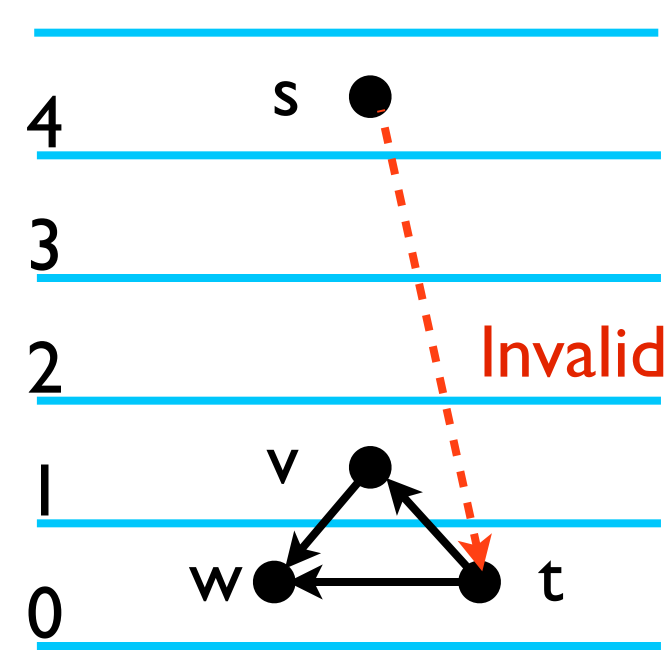
Preflow f and labeling h are compatible if:

1. h(s) = n, h(t) = 0
2. For all edges (v, w) in the residual graph G_f, h(v) <= h(w) + 1

Property 1: If preflow f and labeling h are compatible, then there is no s-t path in G_f

Property 2: If **flow** f and labeling h are compatible, then f is a max flow

Proof: From Property 1 and properties of max flow



Correctness: Proof Outline

Three Steps:

- Compatibility: Show that the preflow f and the labeling h maintained by the algorithm always obeys a **compatibility** property

- If a flow f is compatible with some labeling, then f is a max-flow

- Preflow-push outputs a flow on termination

PreFlow Push: Correctness

Start with labeling: $h(s) = n, h(t) = 0, h(v) = 0$, for other v
Start with preflow f: $f(e) = c(e)$ for $e = (s, v)$, $f(e) = 0$, ow

While there is a node (other than t) with positive excess
    Pick a node v with excess(v) > 0
    If there is an edge (v, w) in E_f s. t. push(v, w) applies
        Push(v, w)
    Else
        Relabel(v)

Push(v, w):
Applies if excess(v) > 0, $h(w) < h(v)$
    $q = \min(\text{excess}(v), c_f(v,w))$
    Add q to f(v, w)

Relabel(v):
Applies if excess(v) > 0 and for all w s.t (v, w) in E_f, $h(w) >= h(v)$
    Increase h(v) by 1

Invariant: Preflow f and labeling h are always compatible over the Preflow-Push algorithm

Preflow f and labeling h are compatible if:
1. $h(s) = n$, $h(t) = 0$
2. For all edges (v, w) in G_f, $h(v) <= h(w) + 1$

Property 1: If preflow f and labeling h are compatible, then there is no s-t path in G_f

Property 2: If **flow** f and labeling h are compatible, then f is a max flow

PreFlow Push: Correctness

Start with labeling: $h(s) = n, h(t) = 0, h(v) = 0$, for other v
Start with preflow f: $f(e) = c(e)$ for $e = (s, v)$, $f(e) = 0$, ow

While there is a node (other than t) with positive excess
    Pick a node v with excess(v) > 0
    If there is an edge (v, w) in E_f s. t. push(v, w) applies
        Push(v, w)
    Else
        Relabel(v)

Push(v, w):
Applies if excess(v) > 0, $h(w) < h(v)$
    $q = \min(\text{excess}(v), c_f(v,w))$
    Add q to f(v, w)

Relabel(v):
Applies if excess(v) > 0 and for all w s.t (v, w) in E_f, $h(w) >= h(v)$
    Increase h(v) by 1

Invariant: Preflow f and labeling h are always compatible over the Preflow-Push algorithm

Fact: When Preflow-push stops, f is a **flow**

Preflow f and labeling h are compatible if:
1. $h(s) = n, h(t) = 0$
2. For all edges (v, w) in G_f, $h(v) <= h(w) + 1$

Property 1: If preflow f and labeling h are compatible, then there is no s-t path in G_f

Property 2: If **flow** f and labeling h are compatible, then f is a max flow

PreFlow Push: Correctness

Start with labeling: $h(s) = n, h(t) = 0, h(v) = 0$, for other v
Start with preflow f: $f(e) = c(e)$ for $e = (s, v)$, $f(e) = 0$, ow

While there is a node (other than t) with positive excess
    Pick a node v with excess(v) > 0
    If there is an edge (v, w) in E_f s. t. push(v, w) applies
        Push(v, w)
    Else
        Relabel(v)

Push(v, w):
Applies if excess(v) > 0, $h(w) < h(v)$
    $q = \min(\text{excess}(v), c_f(v,w))$
    Add q to f(v, w)

Relabel(v):
Applies if excess(v) > 0 and for all w s.t (v, w) in E_f, $h(w) >= h(v)$
    Increase h(v) by 1

Invariant: Preflow f and labeling h are always compatible over the Preflow-Push algorithm

Fact: When Preflow-push stops, f is a **flow**

Proof: Why does Preflow-push stop?

Preflow f and labeling h are compatible if:
1. $h(s) = n, h(t) = 0$
2. For all edges (v, w) in G_f, $h(v) <= h(w) + 1$

Property 1: If preflow f and labeling h are compatible, then there is no s-t path in G_f

Property 2: If **flow** f and labeling h are compatible, then f is a max flow

PreFlow Push: Correctness

Start with labeling: $h(s) = n, h(t) = 0, h(v) = 0$, for other v
Start with preflow f: $f(e) = c(e)$ for $e = (s, v)$, $f(e) = 0$, ow

While there is a node (other than t) with positive excess
    Pick a node v with excess(v) > 0
    If there is an edge (v, w) in E_f s. t. push(v, w) applies
        Push(v, w)
    Else
        Relabel(v)

Push(v, w):
Applies if excess(v) > 0, $h(w) < h(v)$
    $q = \min(\text{excess}(v), c_f(v,w))$
    Add q to f(v, w)

Relabel(v):
Applies if excess(v) > 0 and for all w s.t (v, w) in E_f, $h(w) >= h(v)$
    Increase h(v) by 1

Invariant: Preflow f and labeling h are always compatible over the Preflow-Push algorithm

Fact: When Preflow-push stops, f is a **flow**

Proof: Why does Preflow-push stop?
- No valid push or relabel operation:

Preflow f and labeling h are compatible if:
1. $h(s) = n$, $h(t) = 0$
2. For all edges (v, w) in G_f, $h(v) <= h(w) + 1$

Property 1: If preflow f and labeling h are compatible, then there is no s-t path in G_f

Property 2: If **flow** f and labeling h are compatible, then f is a max flow

PreFlow Push: Correctness

Start with labeling: $h(s) = n, h(t) = 0, h(v) = 0$, for other v
Start with preflow f: $f(e) = c(e)$ for $e = (s, v)$, $f(e) = 0$, ow

While there is a node (other than t) with positive excess
    Pick a node v with excess(v) > 0
    If there is an edge (v, w) in E_f s. t. push(v, w) applies
        Push(v, w)
    Else
        Relabel(v)

Push(v, w):
Applies if excess(v) > 0, h(w) < h(v)
    q = min(excess(v), $c_f(v,w)$)
    Add q to f(v, w)

Relabel(v):
Applies if excess(v) > 0 and for all w s.t (v, w) in E_f, h(w) >= h(v)
    Increase h(v) by 1

Invariant: Preflow f and labeling h are always compatible over the Preflow-Push algorithm

Fact: When Preflow-push stops, f is a **flow**

Proof: Why does Preflow-push stop?
- No valid push or relabel operation:
We can always relabel or push if excess(v) > 0 for some v

Preflow f and labeling h are compatible if:
1. $h(s) = n$, $h(t) = 0$
2. For all edges (v, w) in G_f, $h(v) \le h(w) + 1$

Property 1: If preflow f and labeling h are compatible, then there is no s-t path in G_f

Property 2: If **flow** f and labeling h are compatible, then f is a max flow

PreFlow Push: Correctness

Start with labeling: $h(s) = n, h(t) = 0, h(v) = 0$, for other v
Start with preflow f: $f(e) = c(e)$ for $e = (s, v)$, $f(e) = 0$, ow

While there is a node (other than t) with positive excess
    Pick a node v with excess(v) > 0
    If there is an edge (v, w) in E_f s. t. push(v, w) applies
        Push(v, w)
    Else
        Relabel(v)

Push(v, w):
Applies if excess(v) > 0, $h(w) < h(v)$
    $q = \min(\text{excess}(v), c_f(v,w))$
    Add q to f(v, w)

Relabel(v):
Applies if excess(v) > 0 and for all w s.t (v, w) in E_f, $h(w) >= h(v)$
    Increase h(v) by 1

Invariant: Preflow f and labeling h are always compatible over the Preflow-Push algorithm

Fact: When Preflow-push stops, f is a **flow**

Proof: Why does Preflow-push stop?
- No valid push or relabel operation:
  We can always relabel or push if excess(v) > 0 for some v
- No node v with excess(v) > 0:

Preflow f and labeling h are compatible if:
1. $h(s) = n$, $h(t) = 0$
2. For all edges (v, w) in G_f, $h(v) <= h(w) + 1$

Property 1: If preflow f and labeling h are compatible, then there is no s-t path in G_f

Property 2: If **flow** f and labeling h are compatible, then f is a max flow

PreFlow Push: Correctness

Start with labeling: $h(s) = n, h(t) = 0, h(v) = 0$, for other v
Start with preflow f: $f(e) = c(e)$ for $e = (s, v)$, $f(e) = 0$, ow

While there is a node (other than t) with positive excess
    Pick a node v with excess(v) > 0
    If there is an edge (v, w) in E_f s. t. push(v, w) applies
        Push(v, w)
    Else
        Relabel(v)

Push(v, w):
Applies if excess(v) > 0, h(w) < h(v)
    q = min(excess(v), $c_f(v,w)$)
    Add q to f(v, w)

Relabel(v):
Applies if excess(v) > 0 and for all w s.t (v, w) in E_f, h(w) >= h(v)
    Increase h(v) by 1

Invariant: Preflow f and labeling h are always compatible over the Preflow-Push algorithm

Fact: When Preflow-push stops, f is a **flow**

Proof: Why does Preflow-push stop?
- No valid push or relabel operation:
  We can always relabel or push if excess(v) > 0 for some v
- No node v with excess(v) > 0:
  Then f is a flow!

Preflow f and labeling h are compatible if:
1. $h(s) = n$, $h(t) = 0$
2. For all edges (v, w) in G_f, $h(v) \le h(w) + 1$

Property 1: If preflow f and labeling h are compatible, then there is no s-t path in G_f

Property 2: If **flow** f and labeling h are compatible, then f is a max flow

PreFlow Push: Correctness

Start with labeling: $h(s) = n, h(t) = 0, h(v) = 0$, for other v
Start with preflow f: $f(e) = c(e)$ for $e = (s, v)$, $f(e) = 0$, ow

While there is a node (other than t) with positive excess
  Pick a node v with excess(v) > 0
  If there is an edge (v, w) in E_f s. t. push(v, w) applies
    Push(v, w)
  Else
    Relabel(v)

Push(v, w):
Applies if excess(v) > 0, $h(w) < h(v)$
  $q = \min(\text{excess}(v), c_f(v,w))$
  Add q to f(v, w)

Relabel(v):
Applies if excess(v) > 0 and for all w s.t (v, w) in E_f, $h(w) >= h(v)$
  Increase h(v) by 1

Invariant: Preflow f and labeling h are always compatible over the Preflow-Push algorithm

Fact: When Preflow-push stops, f is a **flow**

Preflow f and labeling h are compatible if:
1. $h(s) = n, h(t) = 0$
2. For all edges (v, w) in G_f, $h(v) <= h(w) + 1$

Property 1: If preflow f and labeling h are compatible, then there is no s-t path in G_f

Property 2: If **flow** f and labeling h are compatible, then f is a max flow

PreFlow Push: Correctness

Start with labeling: $h(s) = n, h(t) = 0, h(v) = 0$, for other v
Start with preflow f: $f(e) = c(e)$ for $e = (s, v)$, $f(e) = 0$, ow

While there is a node (other than t) with positive excess
    Pick a node v with excess(v) > 0
    If there is an edge (v, w) in E_f s. t. push(v, w) applies
        Push(v, w)
    Else
        Relabel(v)

Push(v, w):
Applies if excess(v) > 0, h(w) < h(v)
    q = min(excess(v), $c_f(v,w)$)
    Add q to f(v, w)

Relabel(v):
Applies if excess(v) > 0 and for all w s.t (v, w) in E_f, h(w) >= h(v)
    Increase h(v) by 1

Invariant: Preflow f and labeling h are always compatible over the Preflow-Push algorithm

Fact: When Preflow-push stops, f is a **flow**

From **Property 2** of compatible flows, and **Invariant**, f is a max flow

Thus, Preflow-Push correctly outputs a maxflow

Preflow f and labeling h are compatible if:
1. $h(s) = n$, $h(t) = 0$
2. For all edges (v, w) in G_f, $h(v) \le h(w) + 1$

Property 1: If preflow f and labeling h are compatible, then there is no s-t path in G_f

Property 2: If **flow** f and labeling h are compatible, then f is a max flow

Correctness: Proof Outline

Three Steps:

- Compatibility: Show that the preflow f and the labeling h maintained by the algorithm always obeys a **compatibility** property

- If a flow f is compatible with some labeling, then f is a max-flow

- Preflow-push outputs a flow on termination

Pre-Flow Push

- Algorithm
- Correctness
- Running Time Analysis

Running Time Analysis: Outline

1. How many Push Ops? Relabel Ops?

2. How to implement Push and Relabel Ops efficiently?

Running Time Analysis: Outline

1. How many Relabel Ops?

   Main Idea: Bound the maximum value of h(v) for any node v, and bound #relabel ops through this