Efficiently Maintaining the Edge List

Discharge(v)

```
while e(v) > 0 and (v has not been relabeled)
    w = current-edge(v)
    if (v, w) is admissible
        push(v, w)
    elseif (v, w) is not the last edge in v's list
        Advance current-edge(v)
    else relabel(v)
        current-edge(v) = first-edge(v)
```

Observations

- In one call to discharge, all pushes except possibly the last one are saturating.
- Nonsaturating pushes cause discharge to terminate
- Each time the current edge point advances through the entire list v is is relabeled.

Conclusions

- Time spent advancing pointer is propotional to time spent relabeling.
- Data structure overhead = $O(nm)$.

FIFO Algorithm

Algorithm: Keep active vertices in a queue. Call discharge from the vertex at the head of the queue, and add newly activated vertices to the rear of the queue.

Phases

- Phase 0: vertices added during initialization
- Phase i: vertices added during Phase $i-1$.

Claim: The number of phases is $O(n^2)$

Proof: Via potential function:

$$\Phi = \max\{d(v) : e(v) > 0\}$$

FILL IN PROOF

Bounding the number of non-saturating pushes

- At most 1 non-saturating push per Discharge
- At most n Discharges per phase
- At most $O(n^2)$ phases

Conclusion: At most $O(n^3)$ non-saturating pushes.
Total run time =

$$O(nm + n^3) = O(n^3)$$

Excess Scaling

Ideas:

- $e_{\max} = \max_v \{e(v)\}$.
- **Only push flow from vertices with** $e(v) \approx e_{\max}$.
- **Gradually decrease** $e_{\max}$.

Details:

- Δ **is an upper bound on** $e_{\max}$.
- **A node with** $e(v) \geq \Delta/2$ **has large excess.**
- **A node with** $e(v) < \Delta/2$ **has small excess.**

Invariants to maintain

- **Only push flow from nodes with large excess.**
- **Never let $e(v) > \Delta$ for any vertex.**

Node selection rule: Choose, among vertices with large excess, the one with minimum distance label.

Implication: Any push goes from a vertex of large excess to one of small excess.

Excess Scaling Algorithm

Excess Scaling Algorithm

Preprocess
$\Delta = 2^{\lceil lgU \rceil}$
while $(\Delta \geq 1)$
    while some node has large excess
        Let v **be the min-dist label node of large excess**
        $w = \textit{current-edge}(v)$
        if $push(v, w)$ **is applicable**
            push $\delta = \min\{e(v), c_f(v, w), \Delta - e(w)\}$ **units of flow on** (v, w)
        elseif (v, w) **is not the last edge in v's list**
            $advance(w)$
        else $relabel(v)$

    $\Delta = \Delta/2$

Analysis

Lemma:

- **Each non-saturating push send at least** $\Delta/2$ **units of flow**
- **No excess ever exceeds** Δ

Lemma: **There are** $O(n^2)$ **non-saturating pushes per scaling phass.**

Proof: **Use potential function**

$$\Phi = \sum_v e(v)d(v)/\Delta$$

.

Summary

- $O(n^2)$ **relabels**
- $O(nm)$ **time relabeling**
- $O(nm)$ **saturating pushes**
- $\lceil \lg U \rceil$ **scaling phases, each does** $O(n^2)$ **non-saturating pushes.**

Total Time: $O(nm + n^2 \lg U)$