Multiple source shortest paths

Implicitly compute distance from every \( s_i \) to every vertex \( v \).

Preprocess \( \Sigma \) \( \Rightarrow O(S(n) \log n) \) time

Query \( \Rightarrow O(\log n) \)

**Preprocess**

- Planar map \( \Sigma \)
  - \( l(u \rightarrow v) + l(v \rightarrow u) \)
  - Source vertices \( s_1 \rightarrow s_n \)
    in order on outer face
  - \( \Sigma \backslash S \) strongly connected
  - Shortest paths unique

---

**Divide-and-Conquer**

\( MSSP\text{-}Prep(H, i, k) \)

\( H' = Filter(H, i, k) \)

\[ j \leq \left\lceil \frac{i + k}{2} \right\rceil \]

\( MSSP\text{-}Prep(H', i, j) \)

\( MSSP\text{-}Prep(H', j, k) \)

\( H = \text{minor of } \Sigma \)
Filter \((H, i, k)\)

\(T_j = \text{sh. path tree rooted at } s_j\)

1. Compute \(T_i\) and \(T_k\) \((O(5n))\) time

\(u \rightarrow v\) is properly shared by \(T_i\) and \(T_k\)

\[\rightarrow \text{pred}_i(v) = \text{pred}_k(v) = u\]

\[\rightarrow \text{if } \text{pred}_i(u) = \text{pred}_k(u), \text{ then } \text{pred}_i(u) \rightarrow u\]

\[-\text{ otherwise}\]

\[\text{pred}_i(u), v, \text{ pred}_k(u)\] are ordered clockwise around \(u\).

\(u \rightarrow v\) is an edge in \(T_j\) for all \(i \leq j \leq k\).
$h \rightarrow v$ is shared by all $T_j$'s but not prop. shared by $T_i$ and $T_j$

**Contraction**

- $	ext{rep}(v) \leq u$
- $l(v) \leq l(u \rightarrow v)$

For every $w \rightarrow v$, set $l = \infty$
For every $v \rightarrow w$

- $l(v \rightarrow w) = l(u \rightarrow w) + l(u \rightarrow v)$
- If $\text{pred}(v) = v$, then $\text{pred}(v) = u$

Resolve parallel edges at end
Discover + contract properly shared edges via preorder traversal of Ti

Key Invariant:
\[ \text{dist}_j(w) = \text{dist}_j(\text{rep}(v)) + \text{off}(w) \]

Filter takes \( \mathcal{O}(s(u)) \) time

Query (Rec, j, v)
\[ \text{dist}(s_j, v) \text{ in } \text{Rec}.H \]

if \( j = \text{Rec}.i \) return \( \text{Rec}.\text{dist}_i(v) \)
if \( j = \text{Rec}.k \) return \( \text{Rec}.\text{dist}_k(v) \)
if \( j \leq \text{Rec}.\text{left}.k \)
    return Query (Rec.left, j, Rec.off[v])
else -- right

\( \mathcal{O}(\log s) \)
Analysis key claim:

Sum of # vertices represented at Rec's at any level of the tree is $O(n)$

Contracting one properly edge doesn't create or destroy other properly shared edges.

$\text{Filter}(H, i, k) = \text{Filter}(Z, i, k)$

"properly shared" is wlog wrt $Z$

Index $j$ is interesting for $u$ if $\text{pred}(j(u)) - u$ is NOT prop. shared by $T_j + T_{j+1}$

$\begin{align*}
1 & \text{ } \text{ } u \text{ is stored in Rec } \\
& \text{ iff } j \text{ is interesting to } u \\
& \text{ where } \text{Rec}.\text{parent}.i \leq j \leq \text{Rec}.\text{parent}.k
\end{align*}$

$\begin{align*}
2 & \text{ each vertex } u \text{ has } \leq \deg(u) \text{ interesting } j. \\
\Rightarrow & \text{ at any level, } u \text{ is stored in } \leq 2 \cdot \deg(u) \text{ Recs. } \\
\sum_{u} 2 \cdot \deg(u) = 4 \cdot E \leq 4 \cdot (3n - 6) < 12n
\end{align*}$