Multiple-source shortest paths [ACE]

Similar to planar algorithm —
Move sources around “outer” face 0, maintain shortest path tree T

T changes when \( \text{slack}(u \rightarrow v) = \text{dist}(u) + w(u \rightarrow v) - \text{dist}(v) \leq 0 \)

Dart is active if slack is decreasing

• How to find next pivot

Recall dual cut graph \( K^* = (G \setminus T)^* \)

red cut graph \( R^* \) — remove “hair” from \( K^* \)

\( \Rightarrow O(g) \) cut paths meeting at \( O(g) \) cut vertices

All active darts are in \( R^* \)

For each cut path, either all darts in one direction are active or no darts are active

Red faces of \( G^* \)
\( = \) red vertices of \( G \)
\( \text{dist} \uparrow \)

Blue faces of \( G^* \)
\( = \) blue vertices of \( G \)
\( \text{dist} \downarrow \)

Green edges separate red/blue faces

Dart \( d \) active \( \iff \) tail\((d)\) blue \( + \) head\((d)\) red
\( \iff \) left\((d*)\) blue \( + \) right\((d*)\) red

Grove decomposition: Partition \( K^* \) into edge-disjoint trees
each containing one cutpath + hair,
mmeet at cut vertices

Find next pivot:
for each tree \( T_i \) in grove
if \( T_i \) is green
Find min slack in cut path \( T_i \)
\( \Rightarrow O(g \log n) \) with tree DS.
Perform Next Pivot

Update slacks of active cut paths \( O(g \log n) \)
Add edge to \( K^* \)
Remove edge from \( K^* \)

with more effort:
\( O(g \log n) \) per vertex
of outer face
\( + O(\log n) \) per pivot

How many pivots?

Lemma: Each dart pivots into \( T \) at most \( O(g) \) times

Proof: Let \( Y \) be tree of shortest paths ending at \( y \).
\( X \) = subtree of \( Y \) rooted at \( x \)
\( X \cap \emptyset = \) finite # intervals

Two shortest paths to \( x \) cannot cross.
Any two shortest path from different intervals to \( x \)
are not homotopic rel \( \emptyset \)

Max # non-crossing non-homotopic paths from \( z \) to \( x \)
is \( O(g) \).

Defines system of quads!

Bipartite graph, every face has degree 4

\[
V - E + F = 2 - 2g \\
2E = 4F \Rightarrow E = 2F \\
2 - F = 2 - 2g \Rightarrow F = 2g \Rightarrow [E = 4g]
\]
Minimum Cut / Homology

An even subgraph is a subgraph with all degrees even.

- cycles
  (Connected even = Eulerian)

A boundary subgraph is body of union of faces.

\[ \Rightarrow \text{even} \]

Two subgraphs \( A, B \) are homologous if \( A + B \) is boundary.

Even subgraphs define a vector space \( \mathbb{Z}_2(G) = \mathbb{Z}_2^{E-V+1} \)

Boundary

\[ \mathcal{B}_1(G) = \mathbb{Z}_2^{F-1} \]

Homology classes

\[ H_1(G) = \mathcal{Z}_1/\mathcal{B}_1 = \mathbb{Z}_2^{2-V+E-F} = \mathbb{Z}_2^{B_1} \]

With boundary: \( \mathbb{Z}_2^{B_1 - 1} \times \beta = \text{First Betti} \)

\[ \text{min}(s, t) - \text{cut} \quad \iff \quad \text{min even subgraph homologous to } \partial \mathcal{S}_x^*(s, t^*) \]

Reduce to: Given even subgraph \( \gamma \subseteq G \), find min even subgraph homologous to \( \gamma \).

Wlog surface has no boundary — identify \( x^* \) and \( t^* \)

Homology signatures = cohomology

[Dual system of cycles from tree-cotree]

\( \mathbb{Z}_2 \)-homology cover

Shortest cycle through \( x \) in class \( h = \text{shortest path from } (x, 0) \) to \( (x, h) \)
Any non-null-hom cycle must cross homology basis at least once

$\Rightarrow$ 2g instances of M3SP in G

$2^{\log n \text{loglog } n}$ time

To find min subgraph, find min cycles in every class
and assemble via DP:

$2^{\log n \text{loglog } n + \log g}$

This problem is NP-hard!