Online Motion Planning MA-INF 1314
Online TSP, Shortcut algorithm

Elmar Langetepe
University of Bonn
Online TSP Problem

• Unknown graph $G = (V, E)$
• Visit all vertices and return
• Traffic sign model
• Planar graphs
• DFS search
• Problem of large edges
Shortcut algorithm

- Vertices: Visited, Boundary, Unknown
- Edges: Explored, Boundary, Unknown
- Shortest path $d_G(v, w)$
Shortcut algorithm

- Boundary edge \((x, y)\) blocks boundary edge \((v, w)\)

1. \(|xy| < |vw|\) and
2. \(d_G(v, x) + |xy| < (1 + \delta)|vw|\)

- Later: You eventually move from \(y\) over \(x\) to \(v\)!
Shortcut algorithm

- **Shortcut** non-blocked boundary edge \((x, y)\), charge edge \((x, y)\)
- Blocked edge \((v, w)\) becomes unblocked!
- Artificial edge \(j(y, w)\) of length \(d_G(v, w) < (2 + \delta)|vw|\)

1. \(|xy| < |vw|\) and
2. \(d_G(v, x) + |xy| < (1 + \delta)|vw|\)
Shortcut algorithm

- Use non-blocked DFS edge \((x, y)\), charge edge \((x, y)\)
- Use jump DFS edge \(j(y, w)\), charge edge \((v, w)\)

1. \(|xy| < |vw|\) and
2. \(d_G(v, x) + |xy| < (1 + \delta)|vw|\)
**Pseudocode: Shortcut**\( (x, y, G) \)

Traveling from \( x \), we have visited \( y \) for the first time

**for all** Boundary edges \((v, w)\) do

  if Visiting \( y \) caused \( Block((v, w)) \) to become empty then
    Build jump edge \( j(y, w) \), append jump edge on \( Incident(y) \) and \( Incident(w) \)
  end if

end for

**for all** Edges \((y, z) \in Incident(y)\) do

  if \( z \) is boundary vertex and \((y, z)\) shortcut then
    Traverse the edge \( (y, z) \); Shortcut\( (y, z, G) \)
  else if \( z \) is boundary vertex and \((y, z)\) is jump edge then
    Move to \( z \) along the shortest known path; Shortcut\( (y, z, G) \)
  end if

end for

Return to \( x \) along the shortest known path
Shortcut algorithm: Analysis

**Lemma:** (structural properties)

1. Any vertex of a graph $G$ will be visited.
2. If during execution boundary edge $(x, y)$ blocks boundary edge $(v, w)$, then $(x, y)$ blocks $(v, w)$ until either $y$ or $w$ is visited.
3. If after traversing a boundary edge $(x, y)$ another boundary edge $(v, w)$ became a shortcut and a jump edge $j(y, w)$ was added, we have $d_G(y, w) < (2 + \delta)|vw|$. 

1. and 3. by construction!
2. Bound. edges until neither $y$ nor $w$ is visited, condition remains
Shortcut algorithm: Analysis

**Theorem:** For planar graphs the above Shortcut algorithm is 16-competitive.

**Proof:**

- Charged edges! Shortcuts, jump edges, set $P$
- $2(2 + \delta)|P|$ in total! Factor 2 DFS forth and back!
- MST of $G$, $|OPT| \geq |MST|$
- Show $|P| \leq (1 + 2/\delta)|MST|$
- $2(2 + \delta)|P| \leq 2(2 + \delta)(1 + 2/\delta)|MST| \leq 2(2 + \delta)(1 + 2/\delta)|OPT|$
- Minimize $2(2 + \delta)(1 + 2/\delta)$, $\delta = 2, 16!$
Shortcut algorithm: Analysis

• Charged edges! Shortcuts, jump edges, set $P$
• Proof: $|P| \leq (1 + 2/\delta)|MST|$
• $|MST|$ with maximal number of edges form $P$!
• $MST \cup P$, consider ring $R$ chord set $C$, $C = MST \setminus P$
• Show: $|C| \leq |R|/\delta$, $|R| = 2|MST|$
Shortcut algorithm: Analysis

- Proof: $|P| \leq (1 + 2/\delta)|MST|$
- Show: $|C| \leq |R|/\delta, C = MST \setminus P$
- Open the ring accordingly
- Process chords from inside to outside, budget $|R|$
Shortcut algorithm: Analysis

- Show: $|C| \leq |R|/\delta$, $C = MST \setminus P$
- Exchange $R(a, c)$ by $(a, c)$, $R'$
- Exchange $R'(a, d)$ by $(a, d)$, $R''$
- And so on from inside to outside
Shortcut algorithm: Analysis

• Show: $|C| \leq |R|/\delta$, $C = MST \setminus P$

• Assume: $(x, y)$ inner chord for $R(x, y)$, Blackboard

• Show: $|R(x, y)| \geq (1 + \delta)|xy|$, good chords!

• Substitute $R(x, y)$ by $(x, y)$

• $-\delta|xy|$ from the budget of $|R|$, recursively!
Shortcut algorithm: Analysis

- Assume: \((x, y)\) inner chord for \(R(x, y)\), Blackboard
- Show: \(|R(x, y)| \geq (1 + \delta)|xy|\)
- By contradiction: Assume \(|R(x, y)| < (1 + \delta)|xy|\)
- Edge \((v, w) \in R(x, y)\) with \(|vw| \geq |xy|\) is called large
- Show 1: There is always a large edge!
- Consider point in time \(t\) where \((x, y)\) is charged! Must exist!
- There is a first boundary edge \((v, w)\) in \(R(x, y)\)
- \(d_G(x, v) + |vw| < (1 + \delta)|xy|\) because \(|R(x, y)| < (1 + \delta)|xy|\) was assumed
- \(|xy| > |vw|\), then \((x, y)\) was still blocked by \((v, w)\), contradiction!
- Thus \(|xy| \geq |vw|\) holds, \((v, w)\) is large
Shortcut algorithm: Analysis

- By contradiction: Assume $|R(x, y)| < (1 + \delta)|xy|$
- Edge $(v, w) \in R(x, y)$ with $|vw| \geq |xy|$ is called large.
- Show 2: There is large edge in $R(x, y)$ not charged!
- $(v, w) \in R(x, y)$ charged last point $t$ in time.
- Boundary edge on path $R(x, y) + (x, y) - (v, w)$ from $v$ to $w$ at $t$.
- $(a, b)$ first such while moving from $v$ to $w$.
- $|R(x, y)| < (1 + \delta)|xy|$ and $|xy| \leq |vw|$.
- Also $d_{G}(v, a) + |ab| \leq |R(x, y)| + |xy| - |vw|$.
- $d_{G}(v, a) + |ab| < (1 + \delta)|vw|$, $(v, w)$ not blocked.
- $|ab| \geq |vw|$, and $(a, b)$ is large!
Shortcut algorithm: Analysis

- By contradiction: Assume $|R(x, y)| < (1 + \delta)|xy|$
- Show 2: There is always a large edge $(a, b)$ not charged!
- $MST - (a, b) + (x, y)$ has smaller cost than $MST$ for $|ab| > |xy|$
- For $|ab| = |xy|$ the spanning tree $MST - (a, b) + (x, y)$ induces less chords
- Contradiction! $|R(x, y)| \geq (1 + \delta)|xy|$
Shortcut algorithm: Analysis

- Show: $|C| \leq |R|/\delta$, $C = MST \setminus P$
- Assume: $(x, y)$ inner chord for $R(x, y)$, Blackboard
- Show: $|R(x, y)| \geq (1 + \delta)|xy|$, good chords!
- Exchange $R(x, y)$ by $(x, y)$, $R'$
- Cost $-\delta|xy|$ from the budget $|R|$, recursively!
- Next inner chord $(x', y')$: $|R'(x', y')| \geq (1 + \delta)|x'y'|$
- $|R| - \delta|xy| - \delta|x'y'|$ and so on
- $|R| - \delta|C| \geq 0$
Shortcut algorithm: Analysis

**Theorem:** For planar graphs the above Shortcut algorithm is 16-competitive.

**Proof:**

- $P$ set of charged edges
- $2(2 + \delta)|P|$ in total! Factor 2 DFS forth and back!
- MST of G, $|OPT| \geq |MST|$
- $|P| \leq (1 + 2/\delta)|MST|$
- By $|C| \leq |R|/\delta$, $|R| = 2|MST|$, $C = MST \setminus P$
- $2(2 + \delta)|P| \leq 2(2 + \delta)(1 + 2/\delta)|MST| \leq 2(2 + \delta)(1 + 2/\delta)|OPT|$
- Minimize $2(2 + \delta)(1 + 2/\delta)$, $\delta = 2, 16$
Shortcut algorithm: Computational cost

Lemma: The computational cost of the Shortcut algorithm is bounded by $O(n^2 \log n)$ for a planar graph of $n$ vertices.

- Lists $\text{BlockedBy}(e)$ and $\text{Block}(e)$, cross pointers
- Detect boundary edge $(x, y)$: Init. $\text{BlockedBy}((x, y))$, $\text{Block}((x, y))$
- Single source Dijkstra for $x$
- All $(v, w)$ with $|xy| < |vw|$ and $d_G(v, x) + |xy| < (1 + \delta)|vw|$
- All $(a, b)$ with $|ab| < |xy|$ and $d_G(x, a) + |ab| < (1 + \delta)|xy|$
- $O(n^2 \log n)$, planar, $O(|E|) = O(|V|) = n$
- Update: $\text{BlockedBy}(e)$ and $\text{Block}(e)$
- Traverse jump edges: Again Dijkstra! $O(n)$ jump edges! $O(n^2 \log n)$
• Explored \((x, y)\) no longer blocks: for any \((v, w) \in \text{BlockedBy}((x, y))\) remove \((x, y)\) in \(\text{Block}((v, w))\), \(O(1)\)

• \(O(n)\) lists in total

• Any edge inserted, deleted once! \(O(n^2)\)