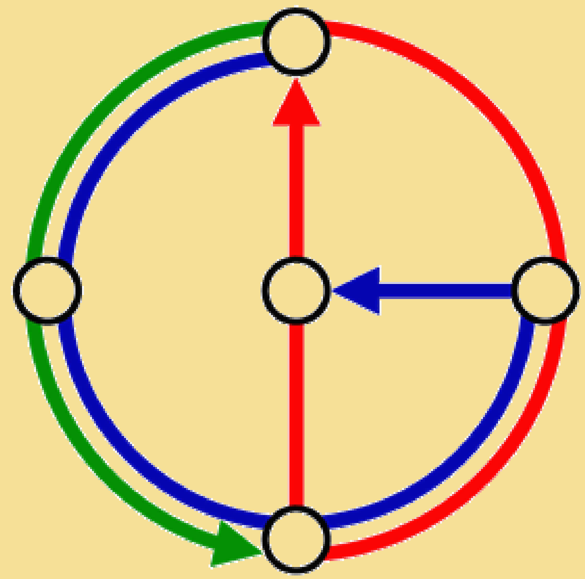
Asymptotically Optimal Mobile Ad-Hoc Routing

Fabian Kuhn

Roger Wattenhofer

Aaron Zollinger

ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich



Dept. of Computer Science
Distributed Computing Group

Overview

- Introduction
- Model
- Face Routing
- Adaptive Face Routing
- Lower Bound
- Conclusion

Model I (Ad-Hoc Network)

- Nodes are points in $\mathbb{R}^2$
- All nodes have the same transmission range (normalized to 1)
  - $\Rightarrow$ network is a unit disk graph
- Distance between any two nodes is lowerbounded by a constant d_0
  - $\Rightarrow$ $\Omega(1)$-model

Model II (Geometric Routing)

- Nodes know the geometric positions of themselves and of their neighbors
- Source s knows the coordinates of destination t
- Nodes not allowed to store anything
- In the message, only $O(\log n)$ additional bits can be stored

Cost Model I

- Cost of sending a message over a link (edge) e is $c(e)$
- Cost of a path p is the sum over the costs of its edges
- Cost of a routing algorithm $\mathcal{A}$ is the sum over the costs of the traversed edges

Cost Model II

3 different cost metrics:

- Link distance metric ($c_\ell(e) \equiv 1$)
- Euclidean distance ($c_d(e)$)
- Energy metric ($c_E(e) := c_d^2(e)$)

  more general: $c_E(e) := c_d^\alpha(e)$ for an $\alpha \geq 2$

Costs are Equivalent

Lemma:

In the $\Omega(1)$-model the link, Euclidean, and energy metrics of a path or an algorithm are equivalent up to a constant factor.

Face Routing

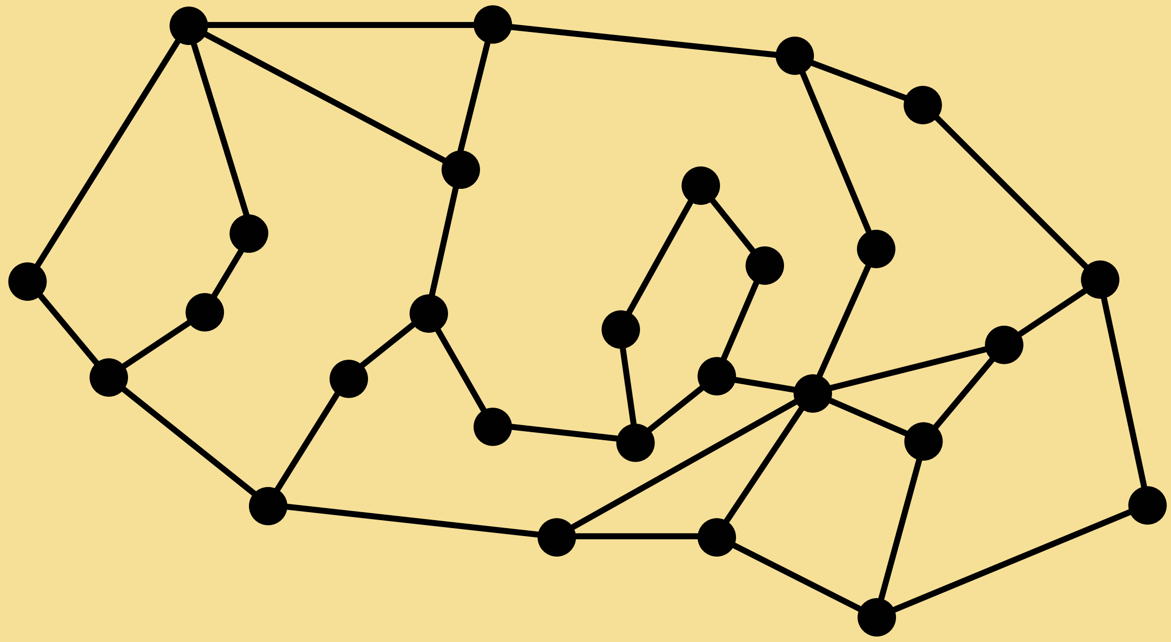
geometric routing algorithm for planar graphs

Compass Routing on Geometric Networks
[Kranakis, Singh, Urrutia; 1999]

Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Distributed Computing Group

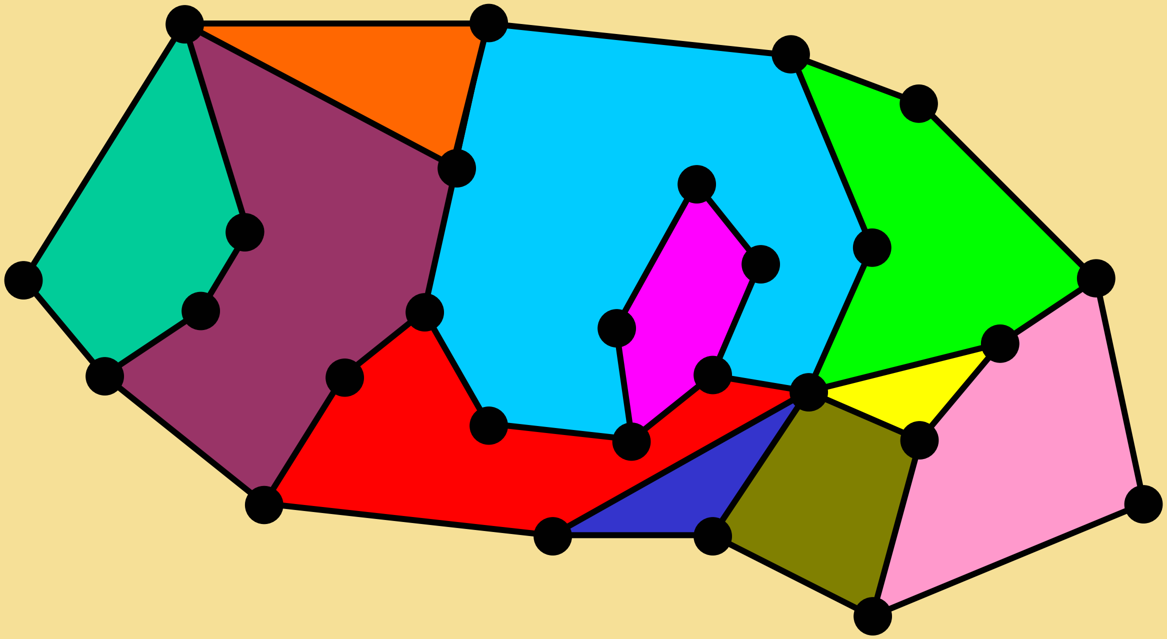
Face Routing



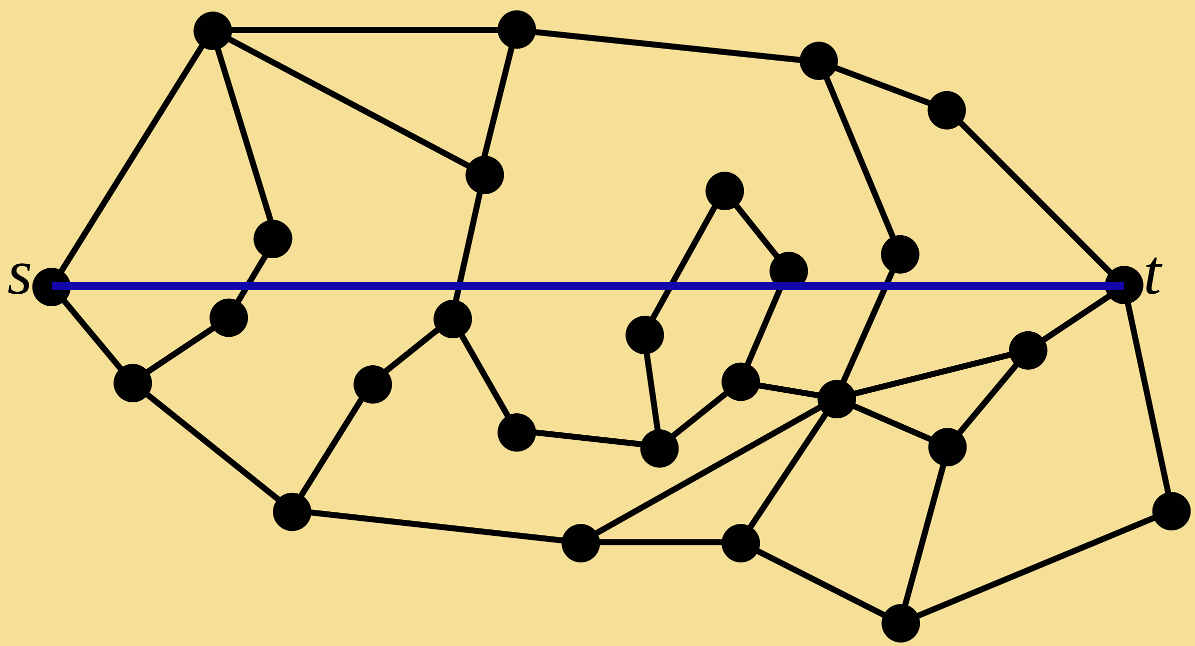
ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Distributed Computing Group

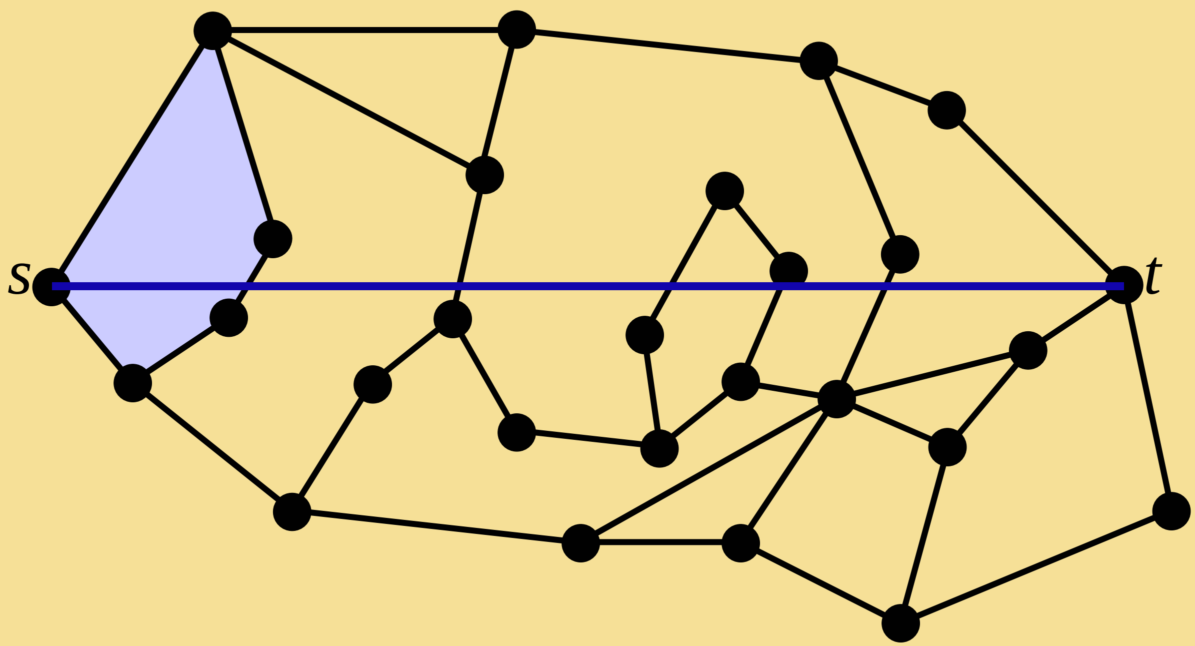
Face Routing (Faces)



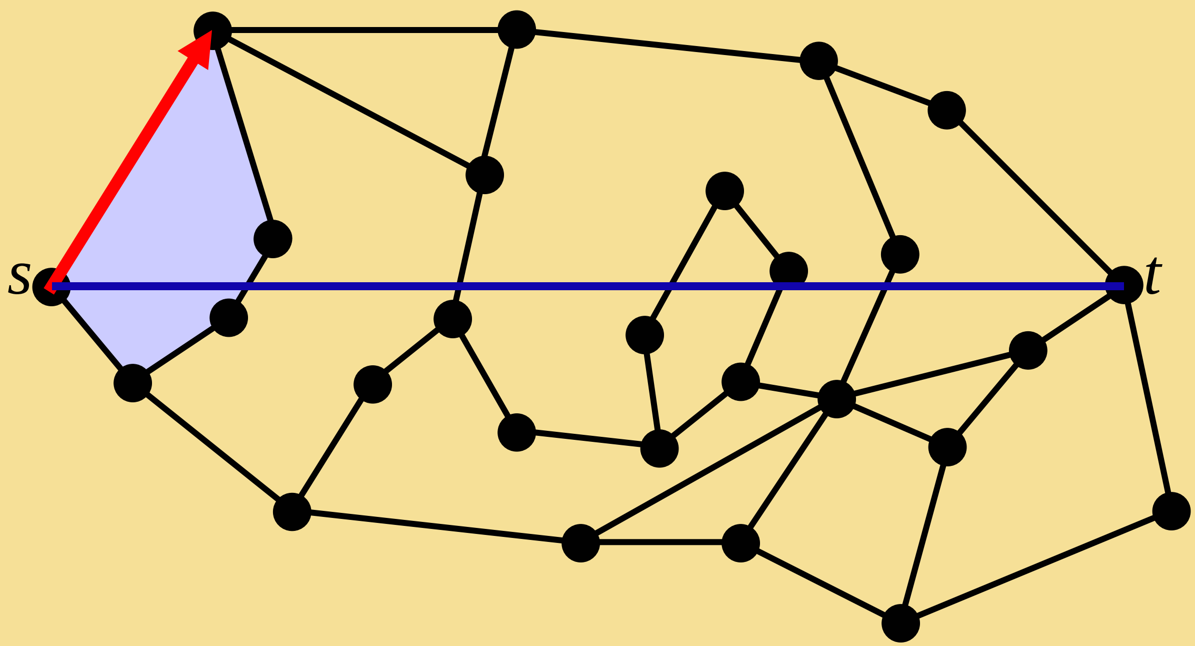
Face Routing



Face Routing



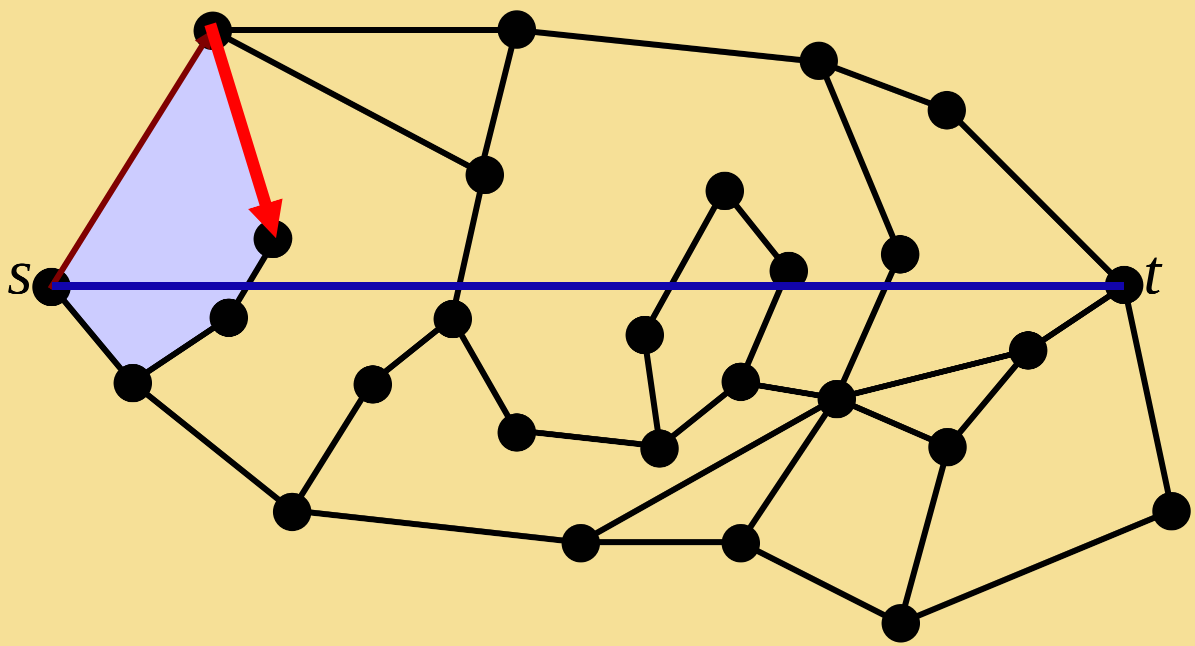
Face Routing



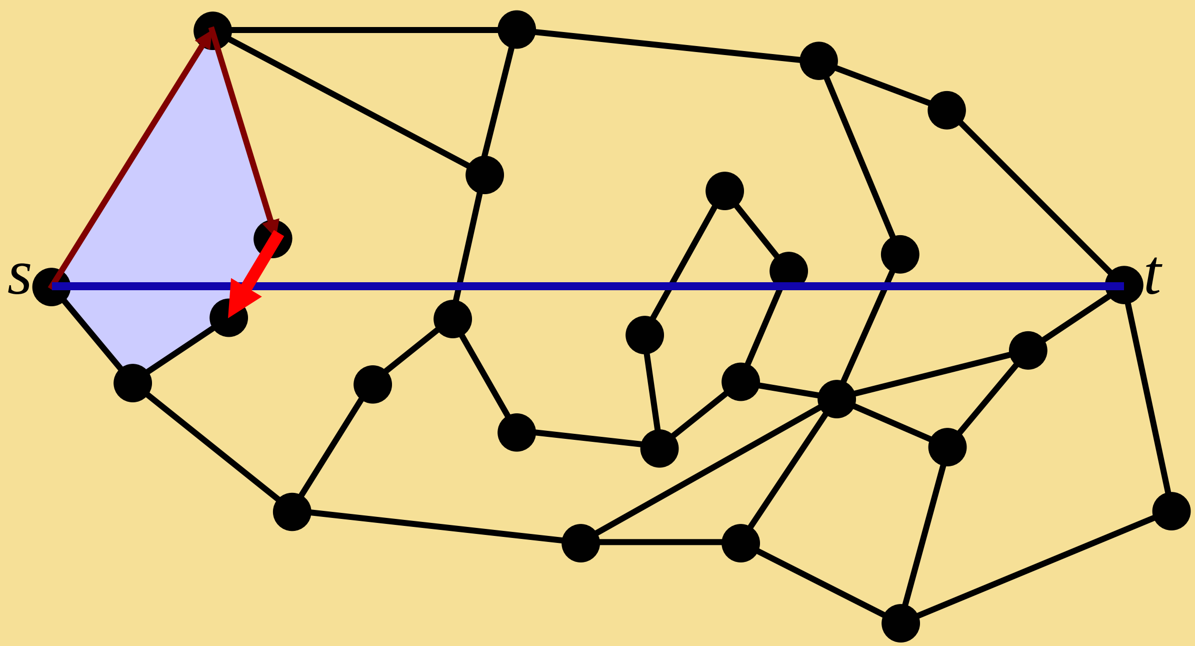
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Distributed Computing Group

Face Routing



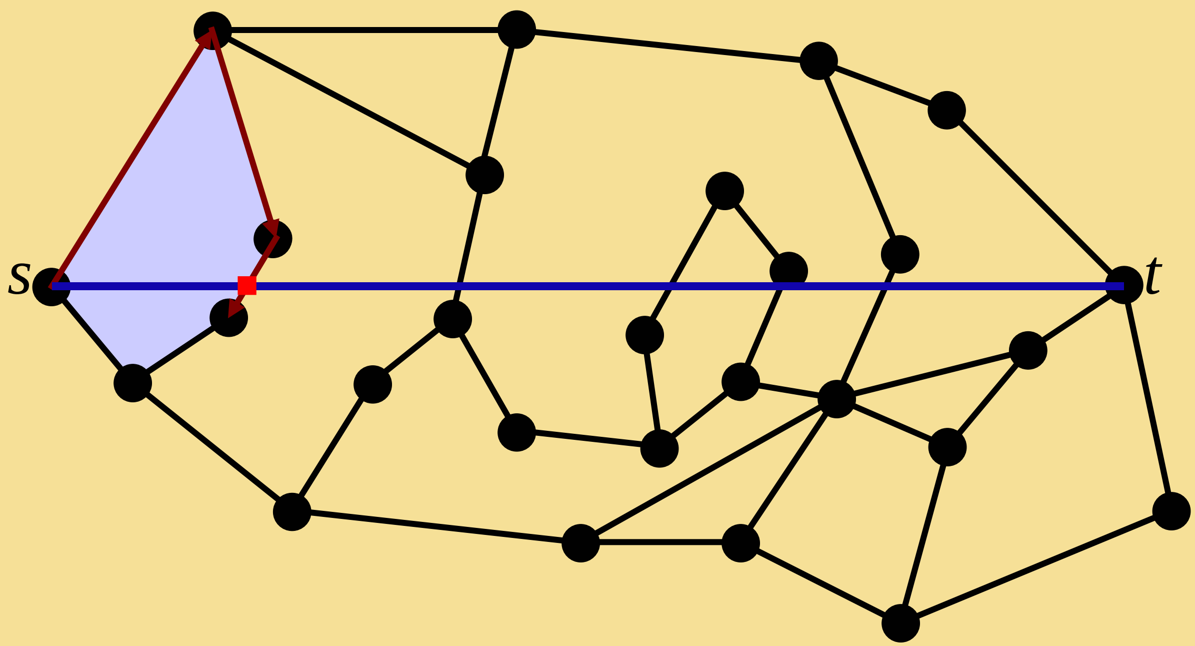
Face Routing



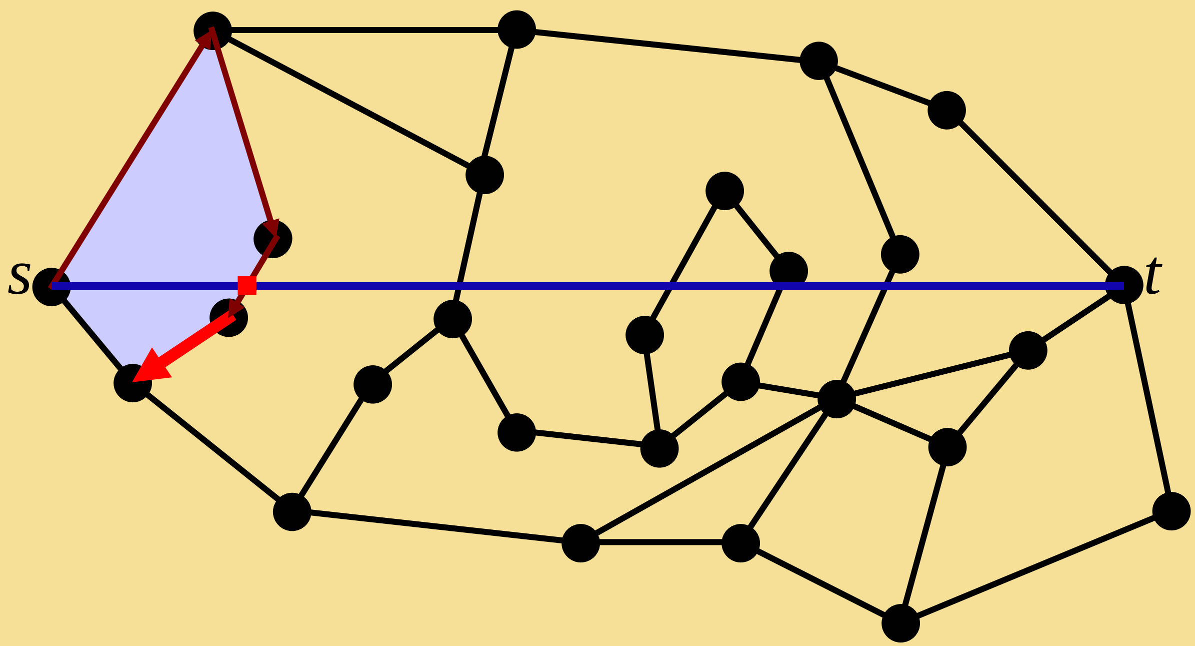
ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Distributed Computing Group

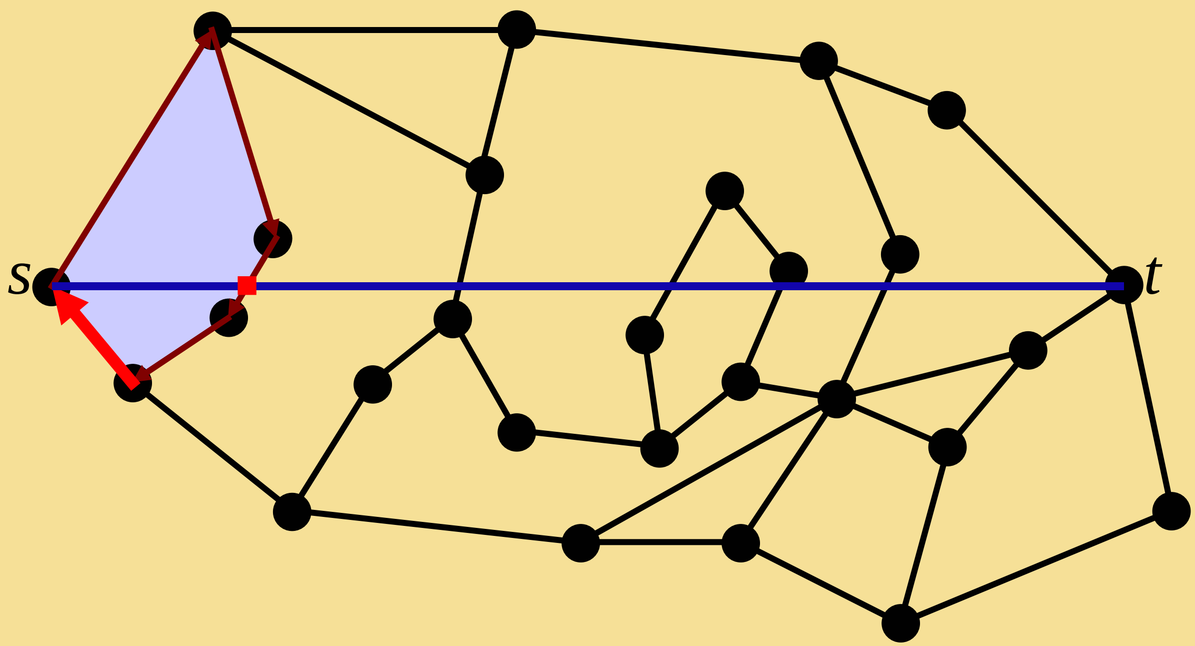
Face Routing



Face Routing



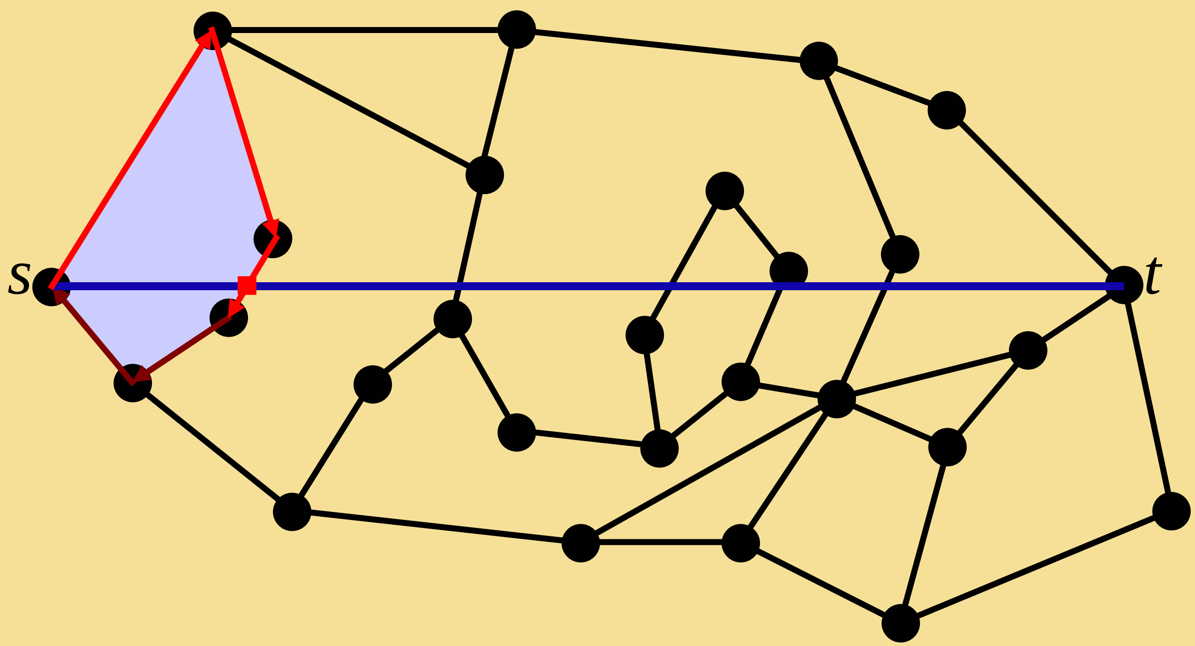
Face Routing



ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Distributed Computing Group

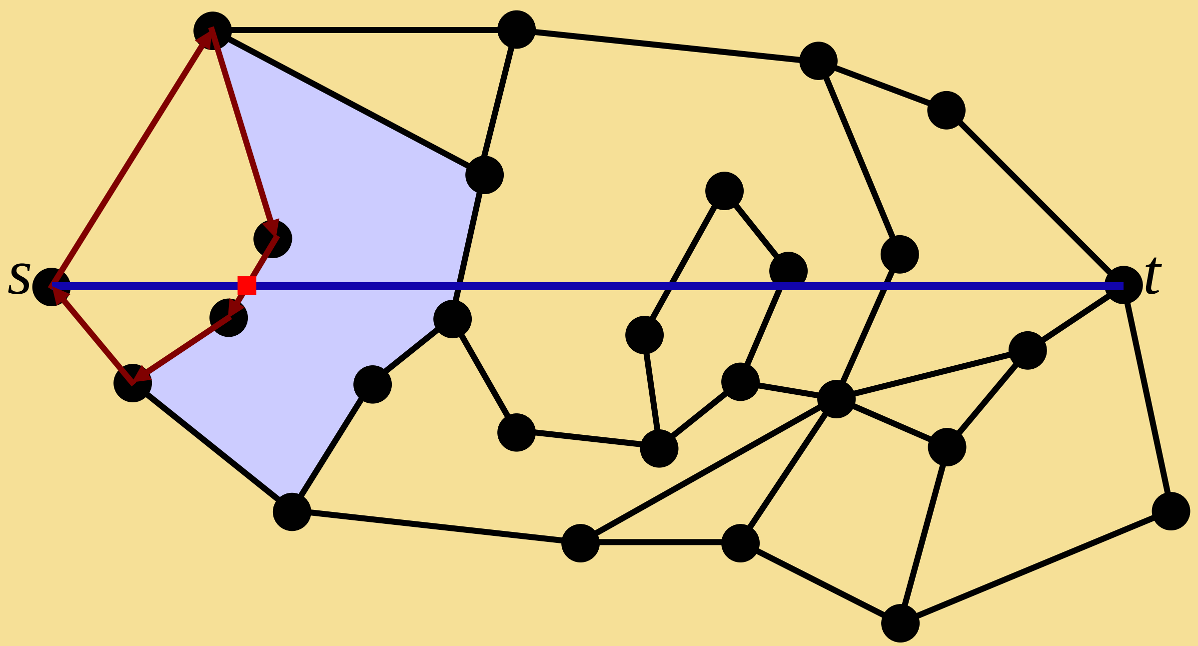
Face Routing



ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Distributed Computing Group

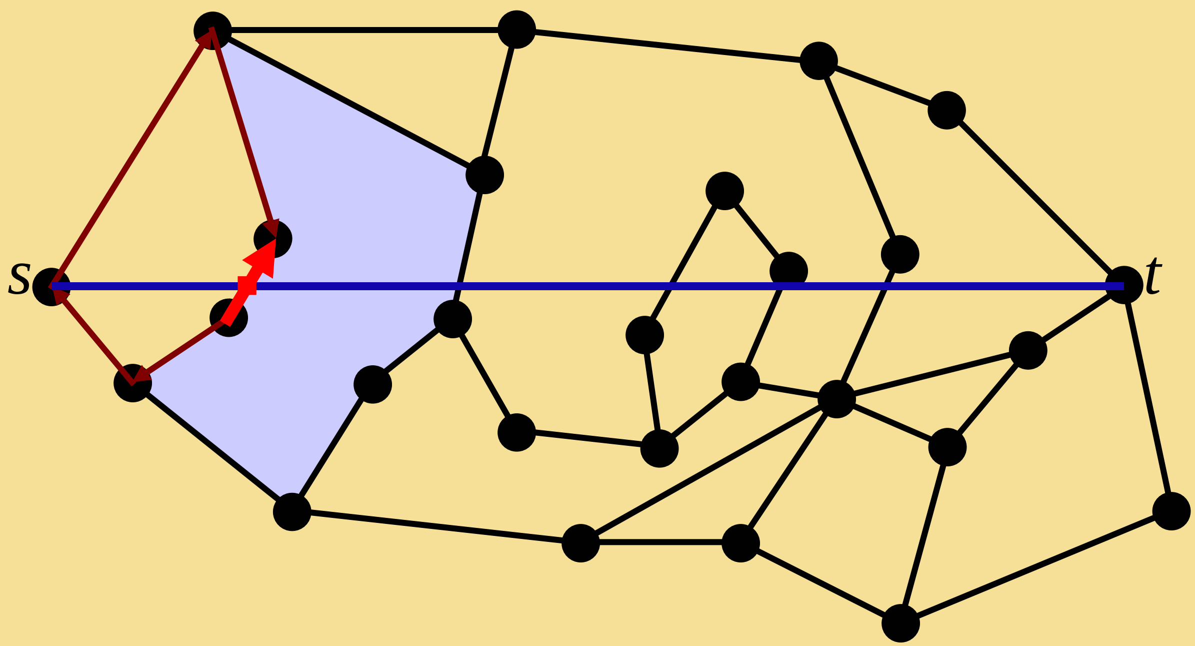
Face Routing



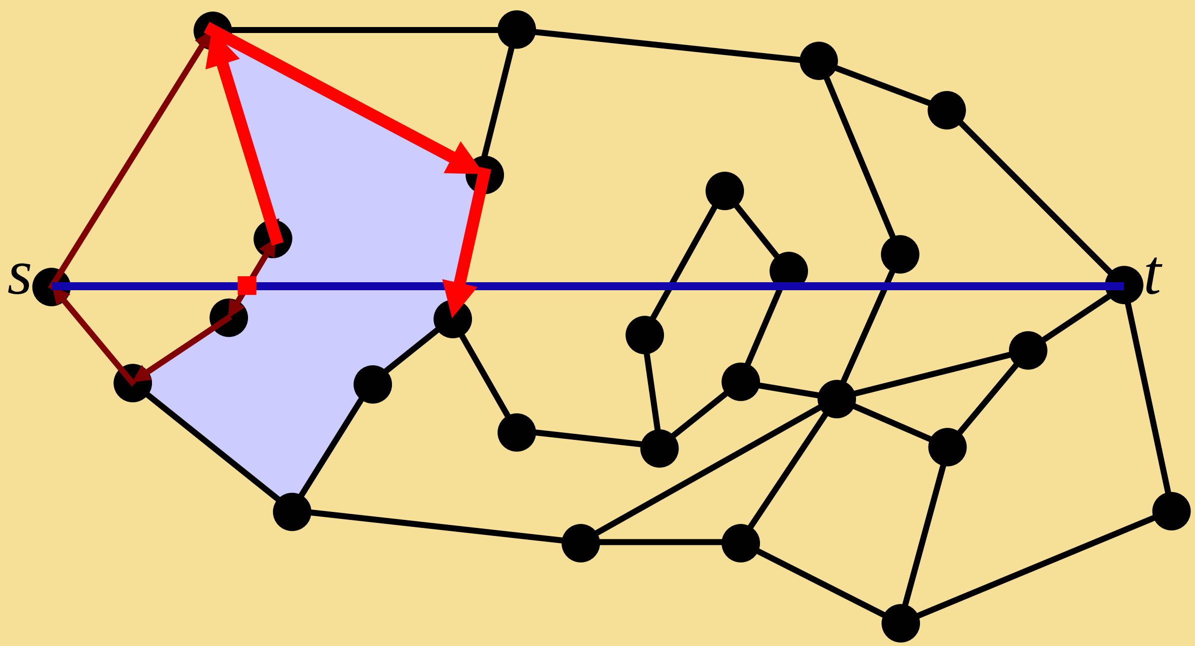
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Distributed Computing Group

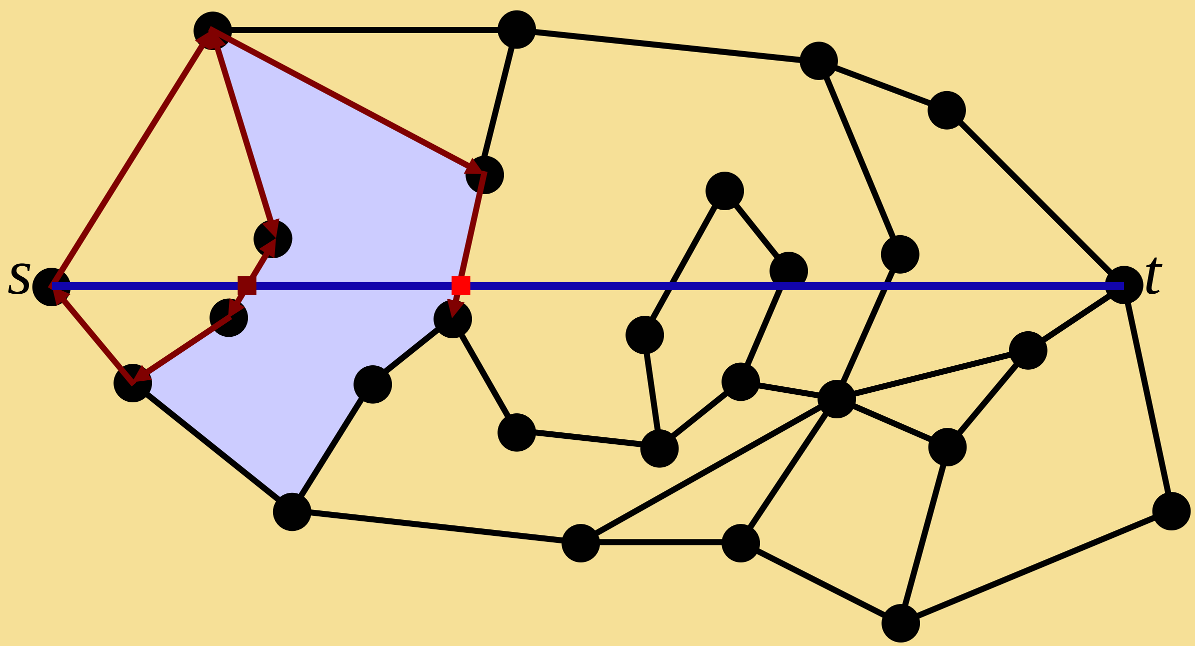
Face Routing



Face Routing



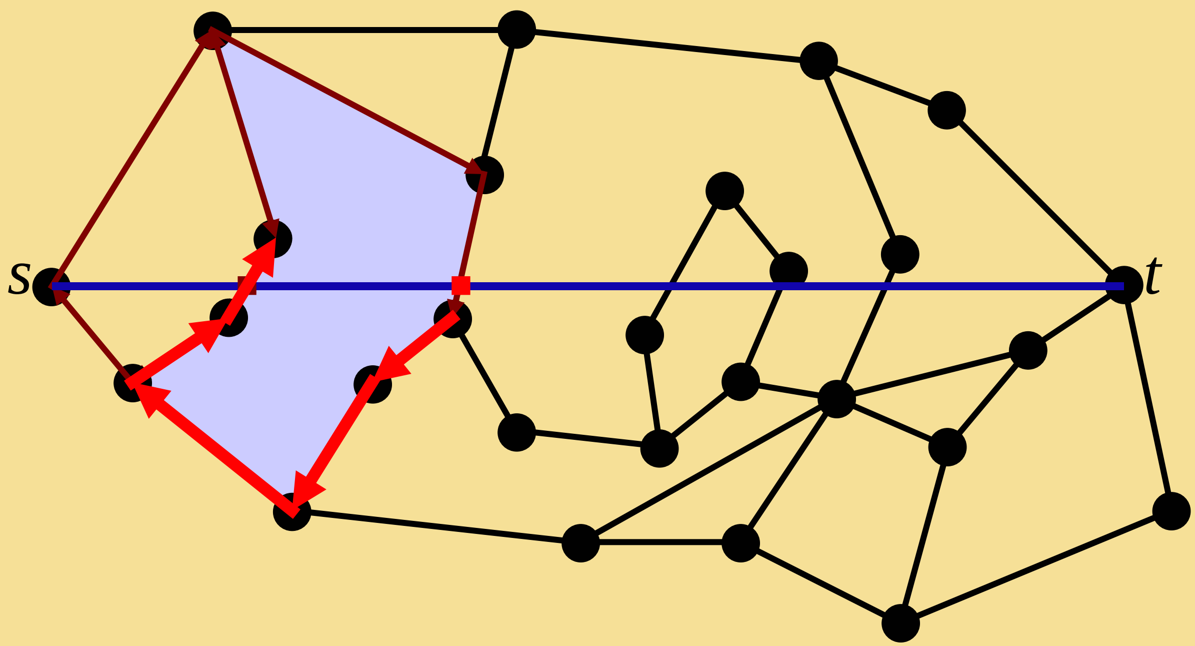
Face Routing



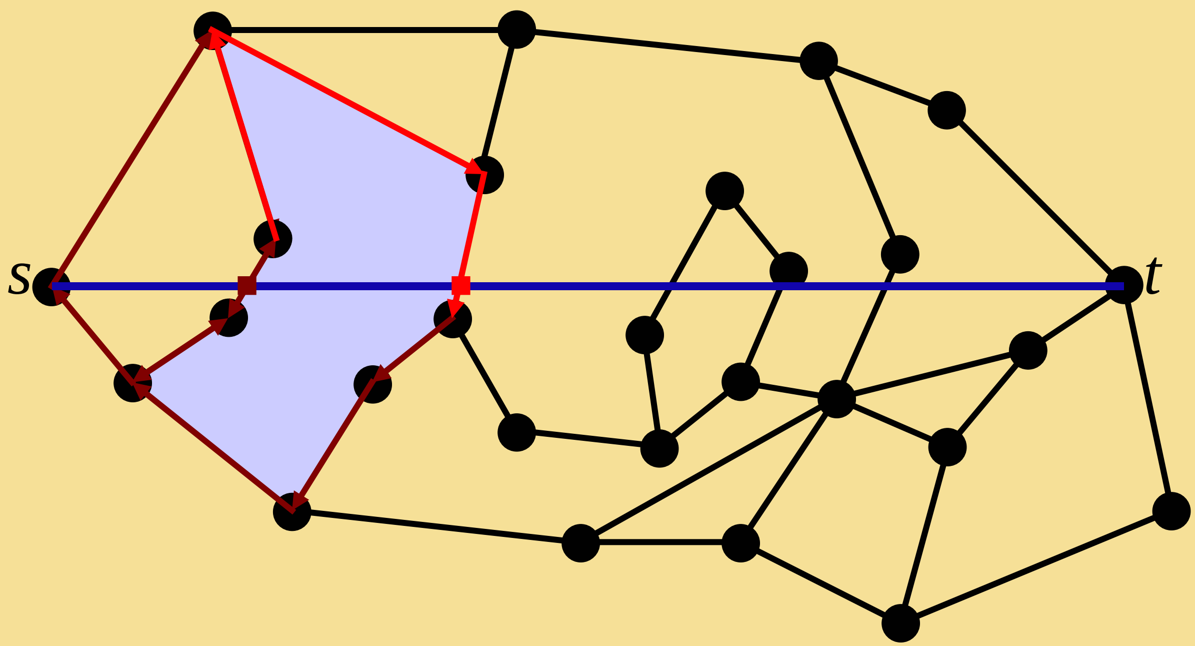
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Distributed Computing Group

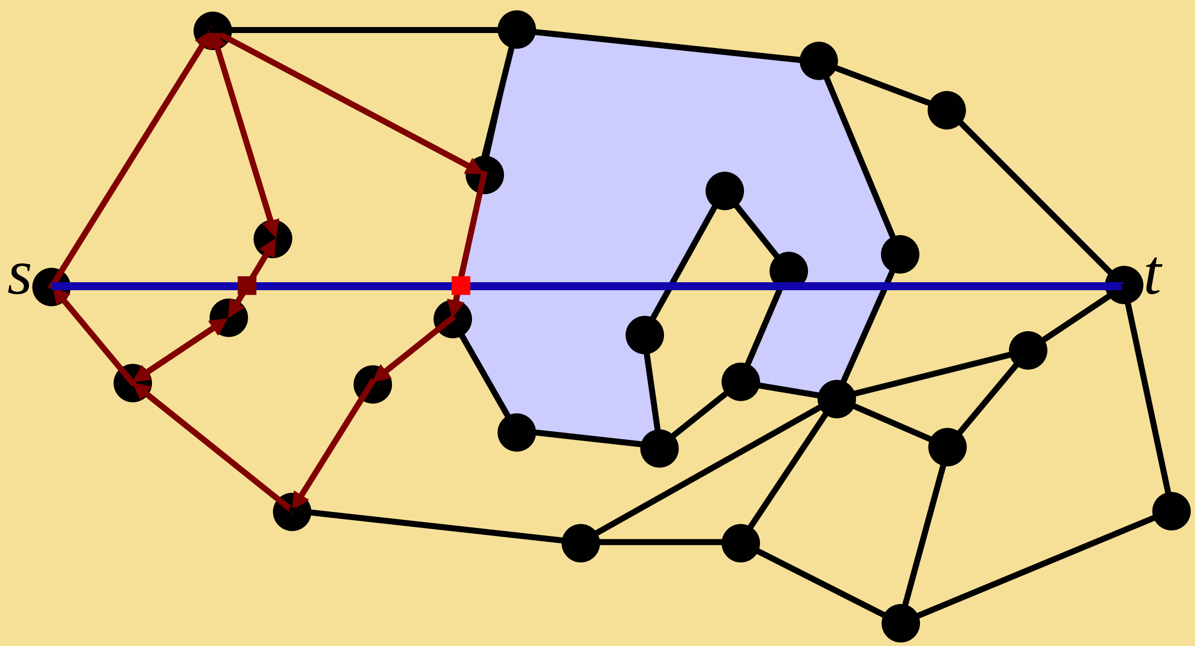
Face Routing



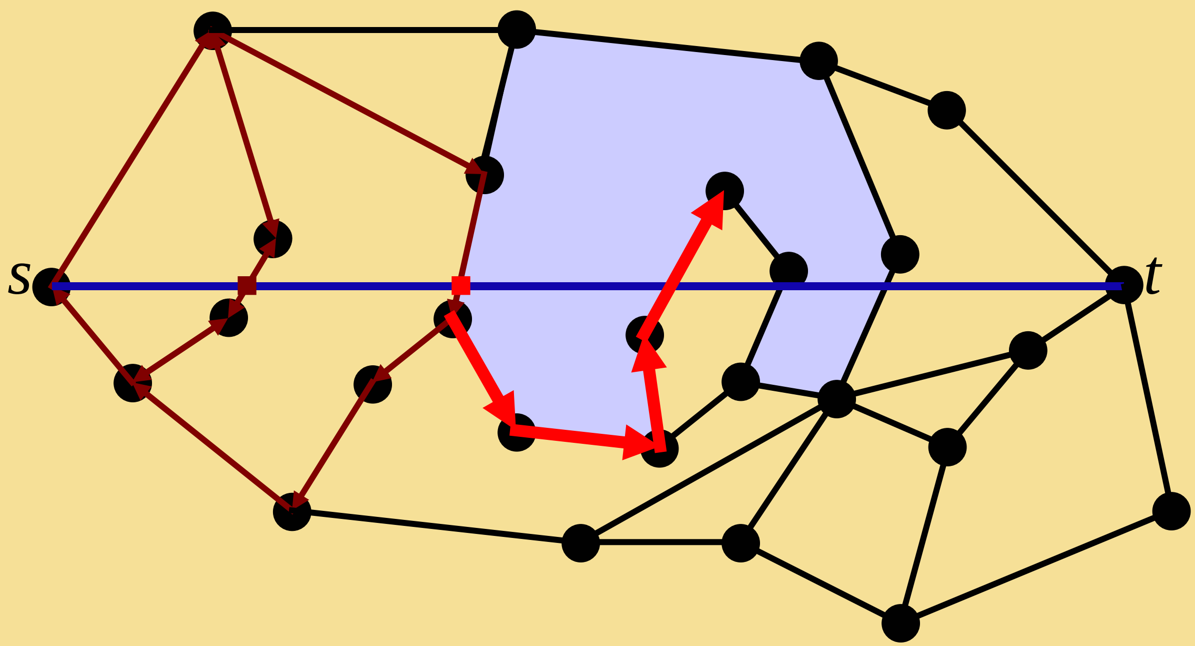
Face Routing



Face Routing



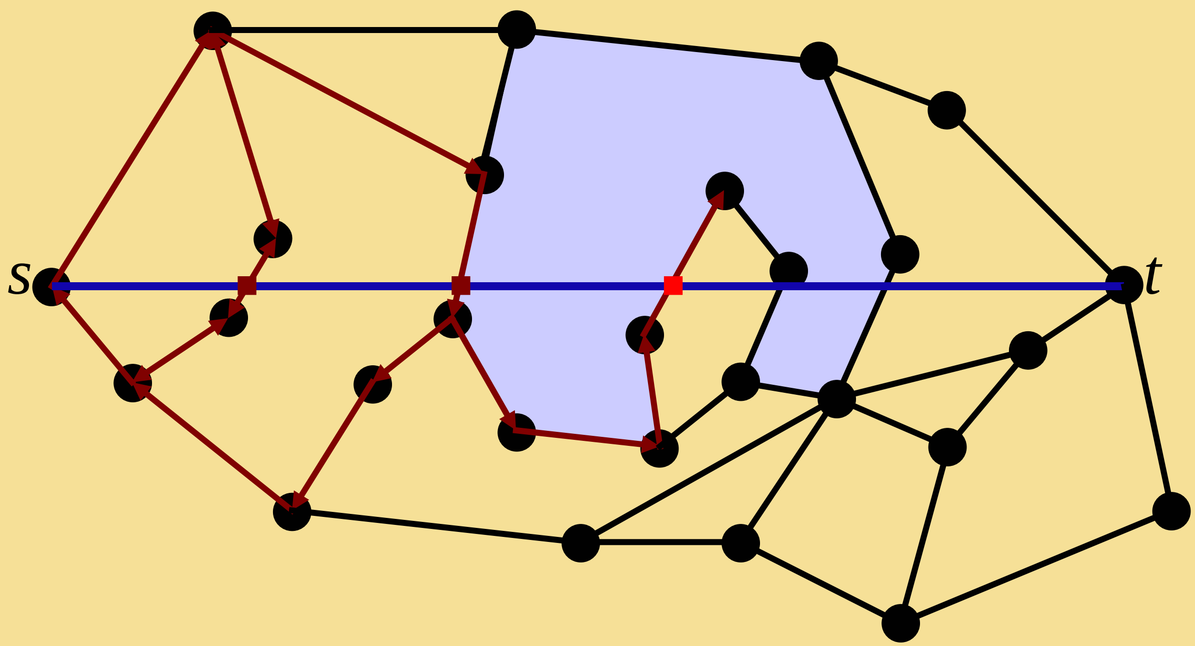
Face Routing



ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Distributed Computing Group

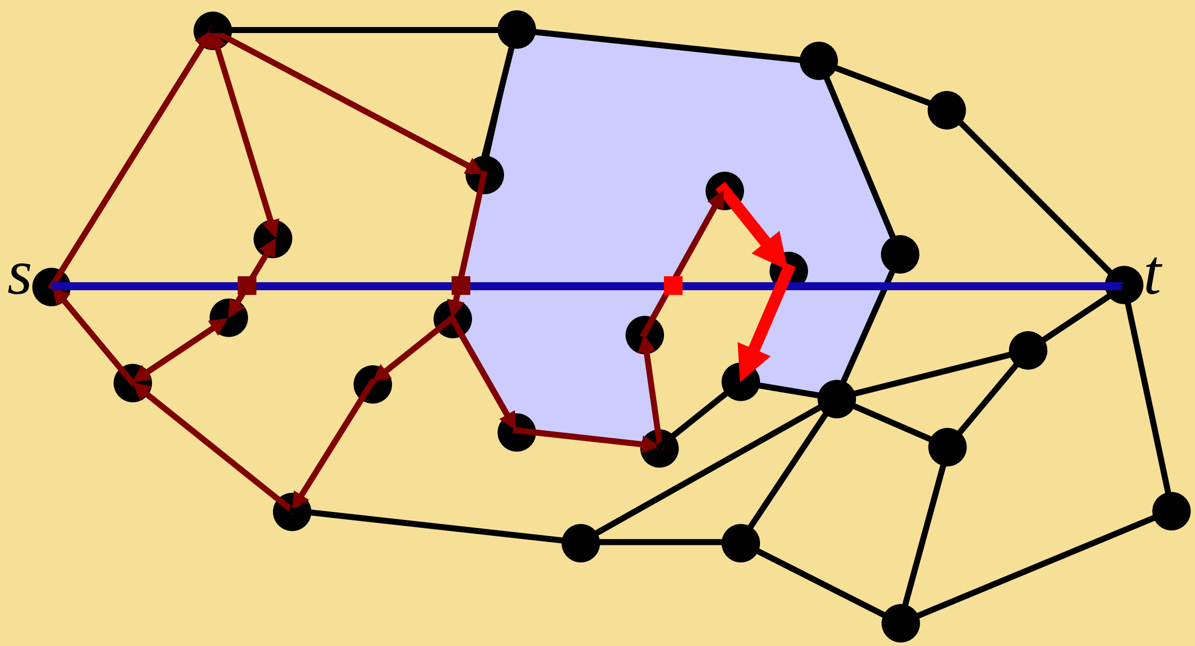
Face Routing



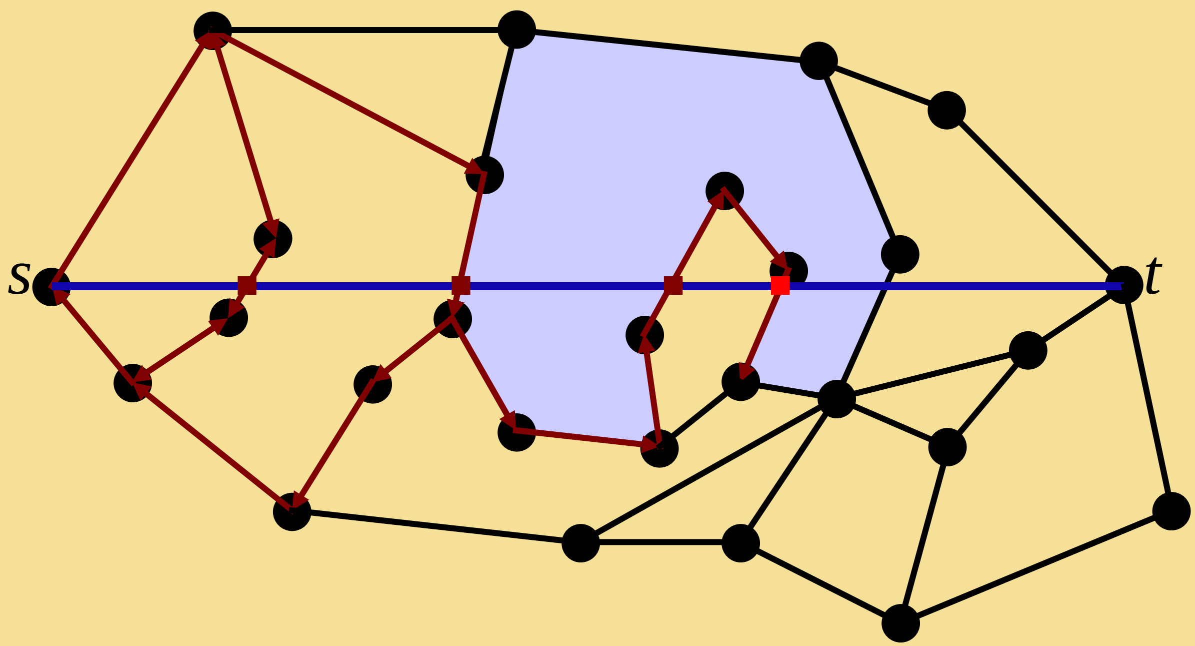
ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Distributed Computing Group

Face Routing



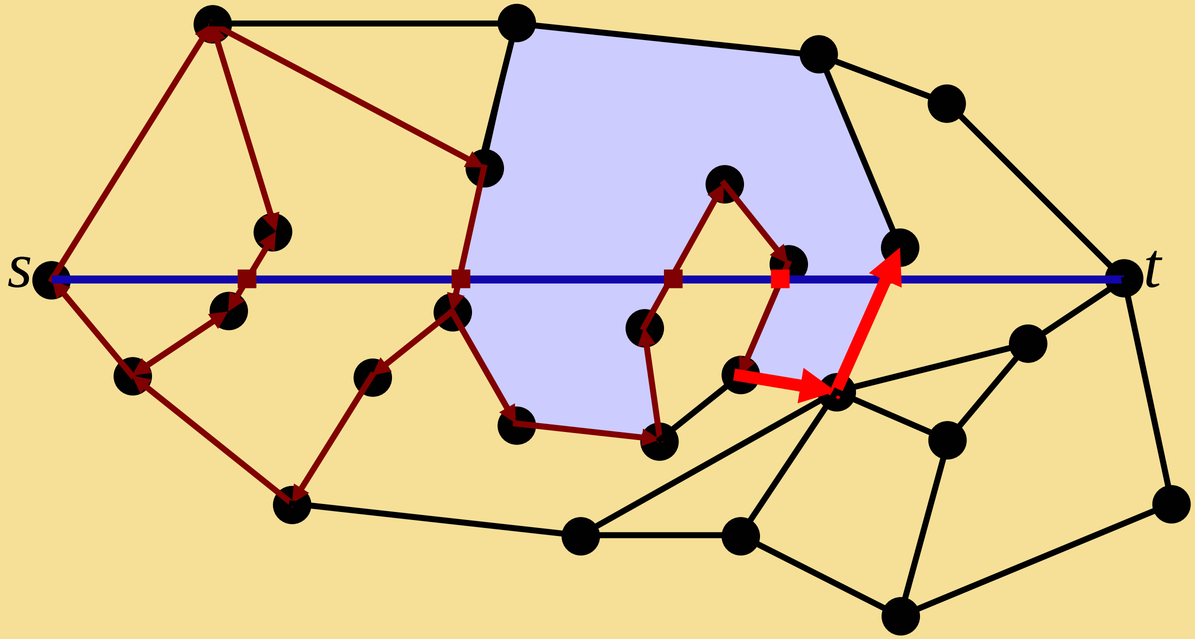
Face Routing



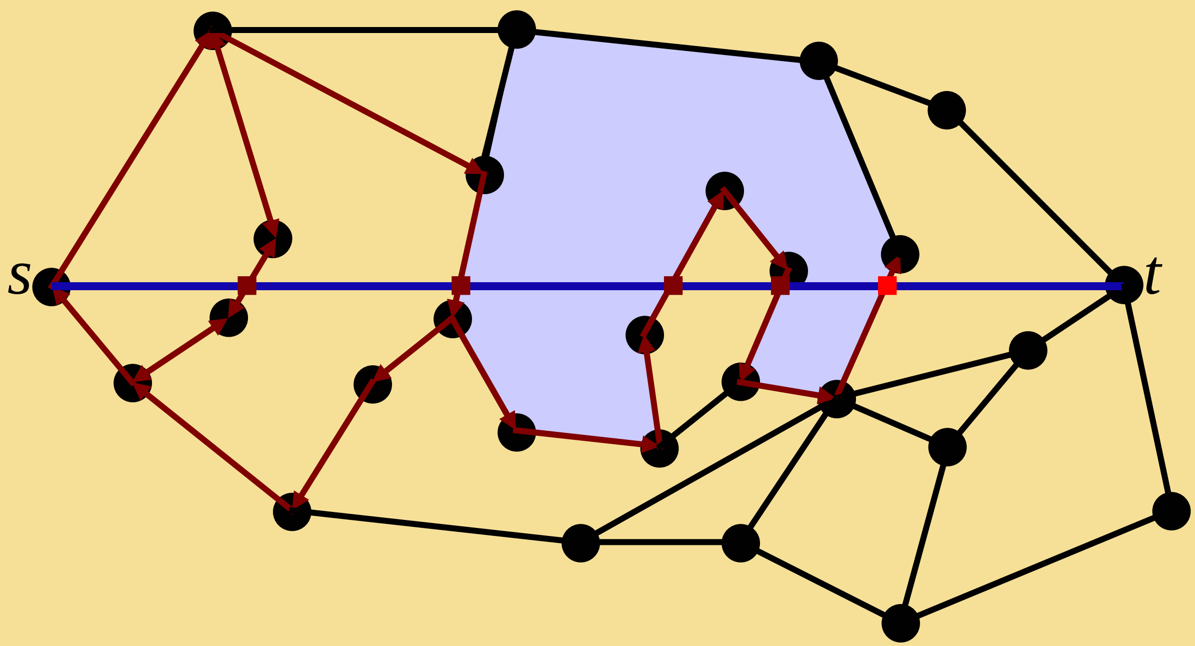
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Distributed Computing Group

Face Routing



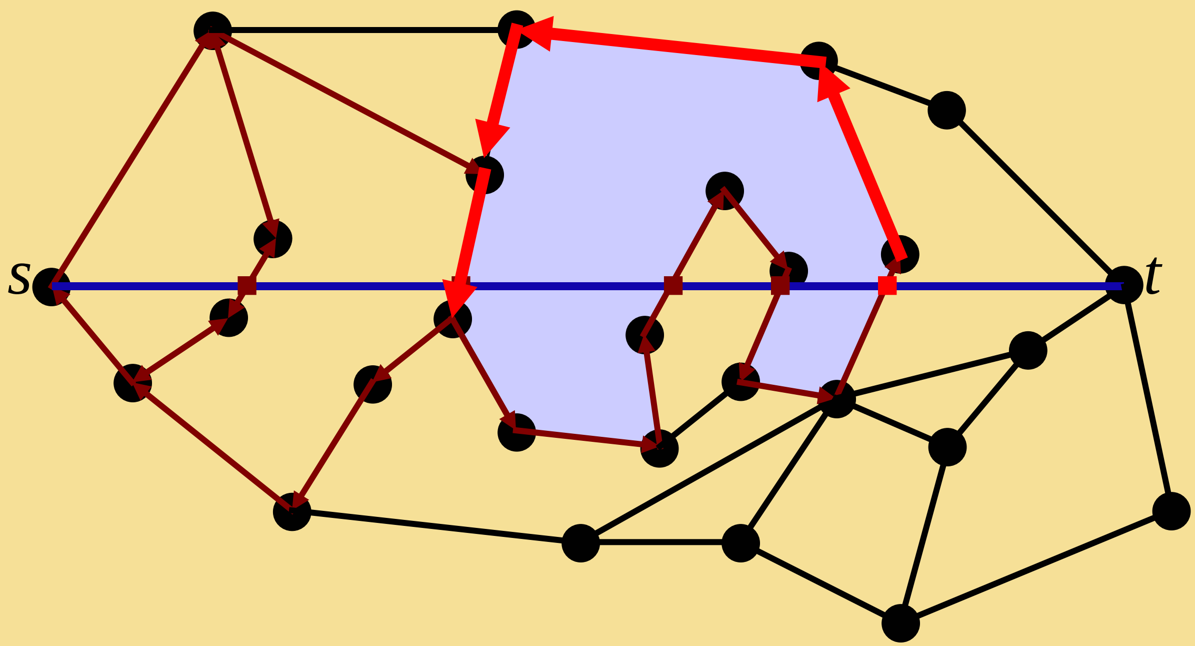
Face Routing



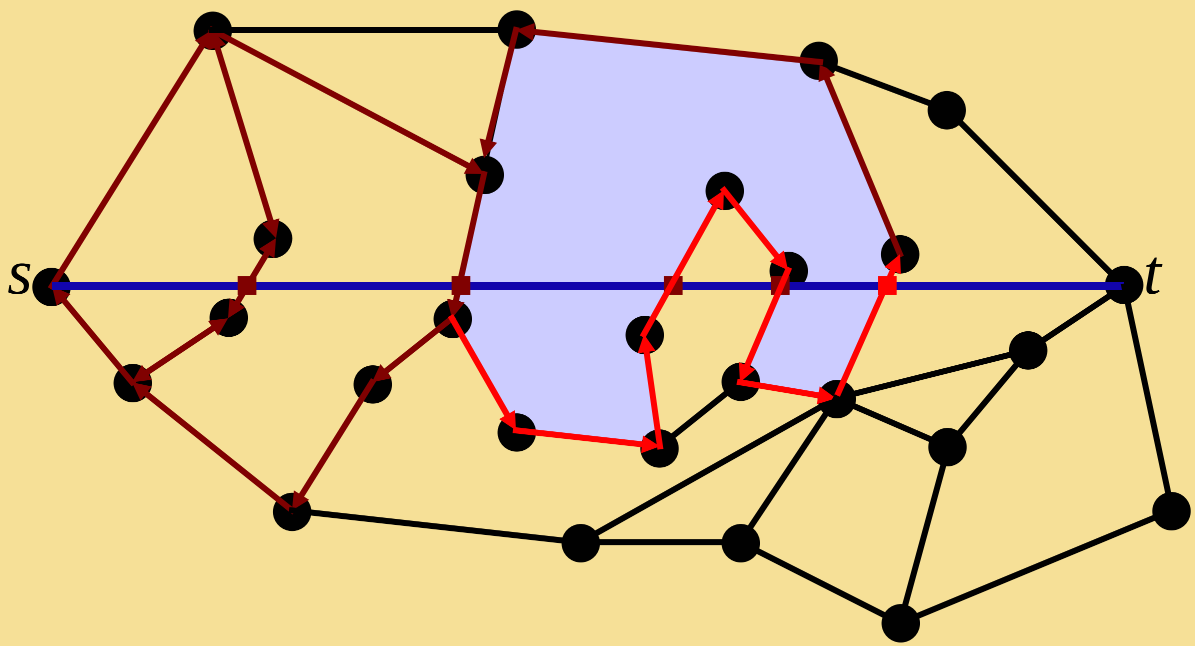
ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Distributed Computing Group

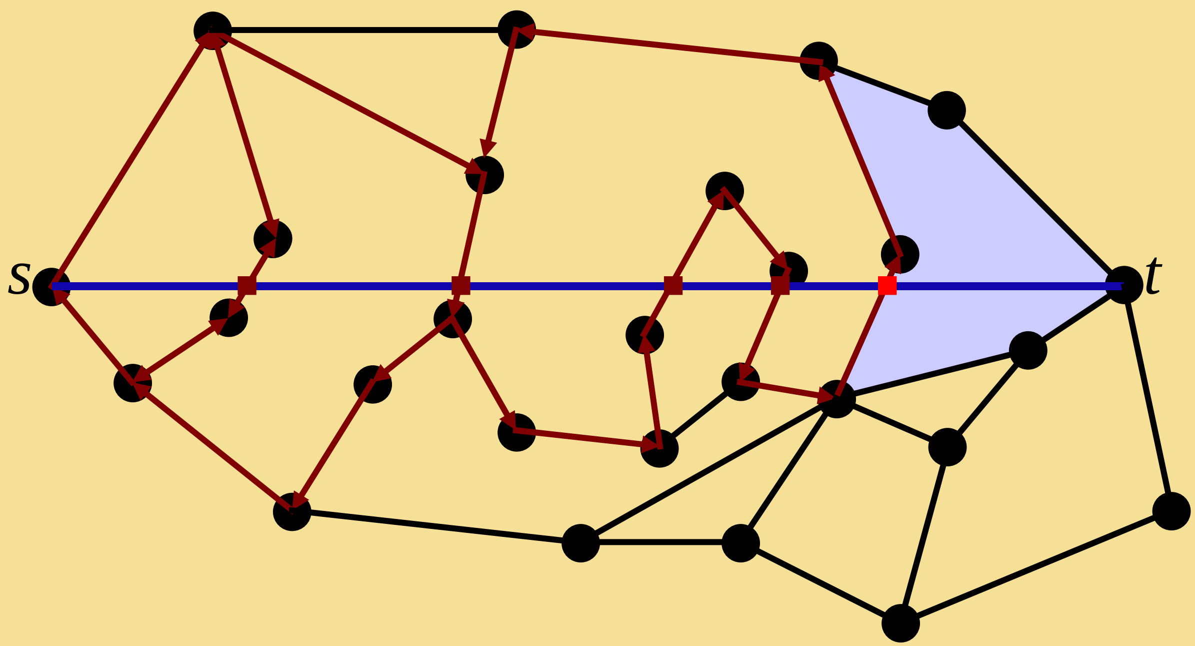
Face Routing



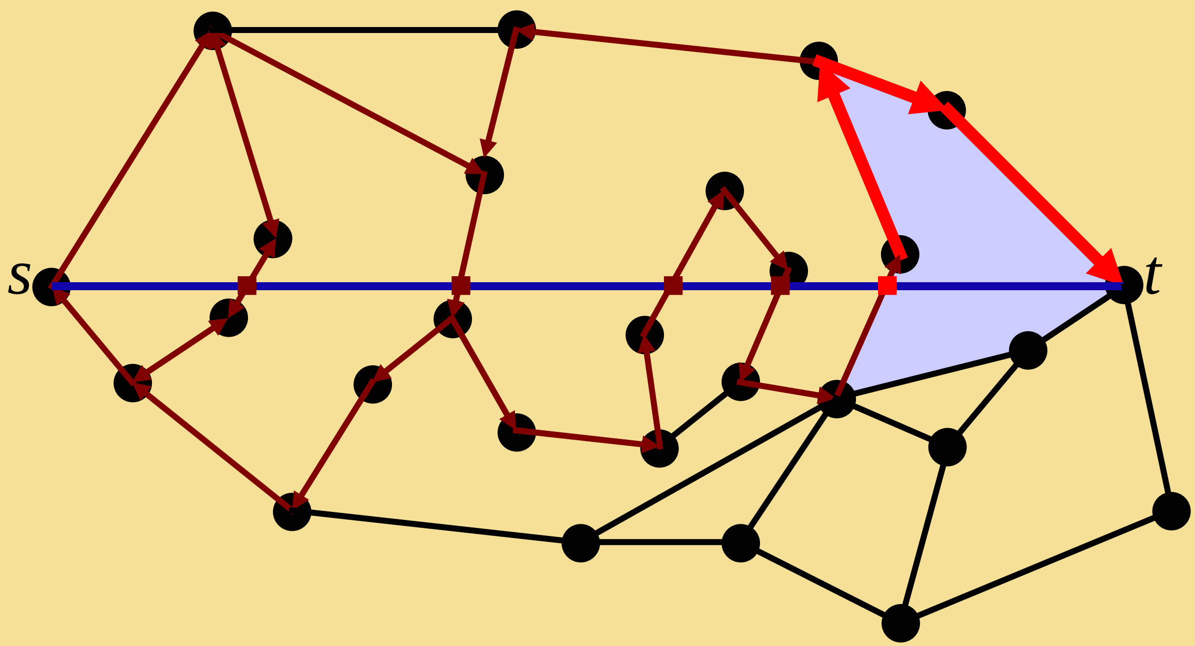
Face Routing



Face Routing



Face Routing



Face Routing (Analysis)

Lemma:

Face Routing always finds a path to the destination. The total cost of Face Routing is $O(n)$.

Face Routing (Analysis)

Proof Sketch:

- each face is explored at most once

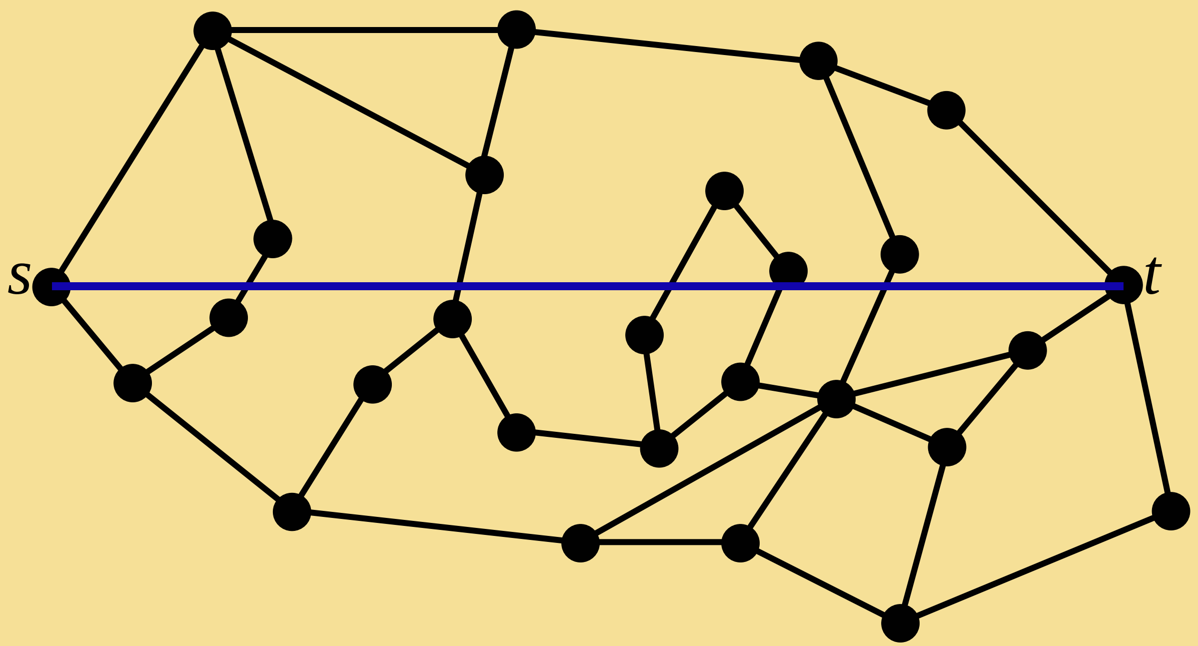
  $\Rightarrow$ each edge is traversed at most four times

- there are at most $3n$-6 edges (Euler formula)

Problem of Face Routing

- Face Routing always reaches destination
- However, even if source and destination are close to each other, Face Routing can take $O(n)$ steps.
- We would like to have an algorithm, whose cost is a function of the cost of an optimal path.

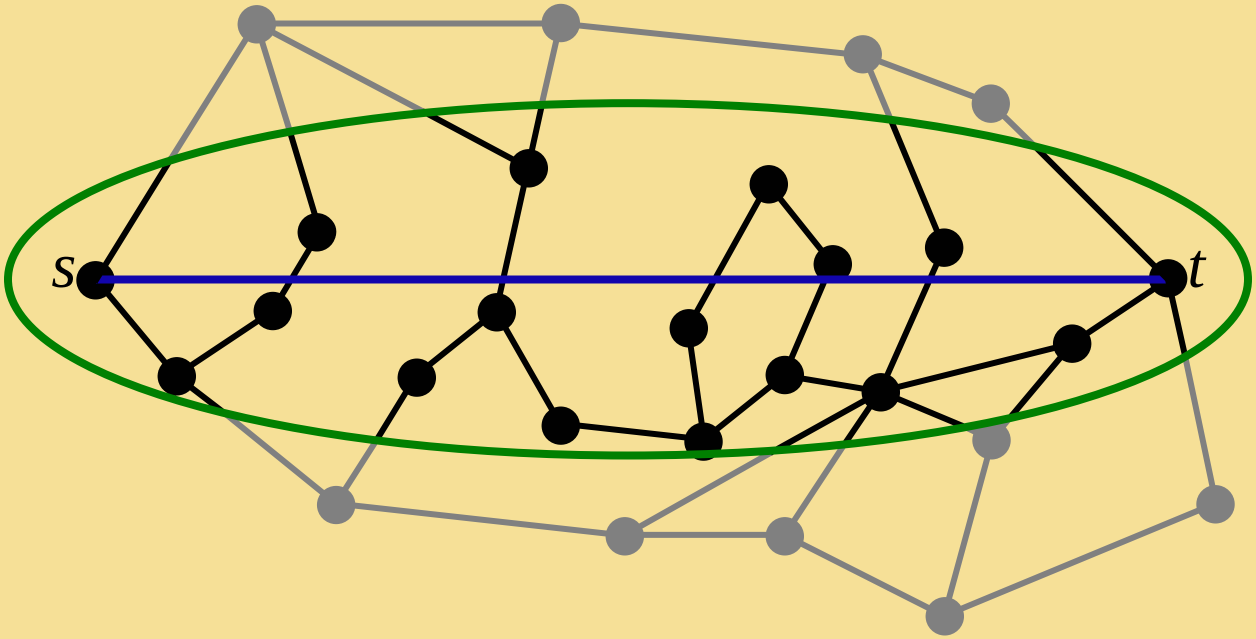
Adaptive Face Routing



ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Distributed Computing Group

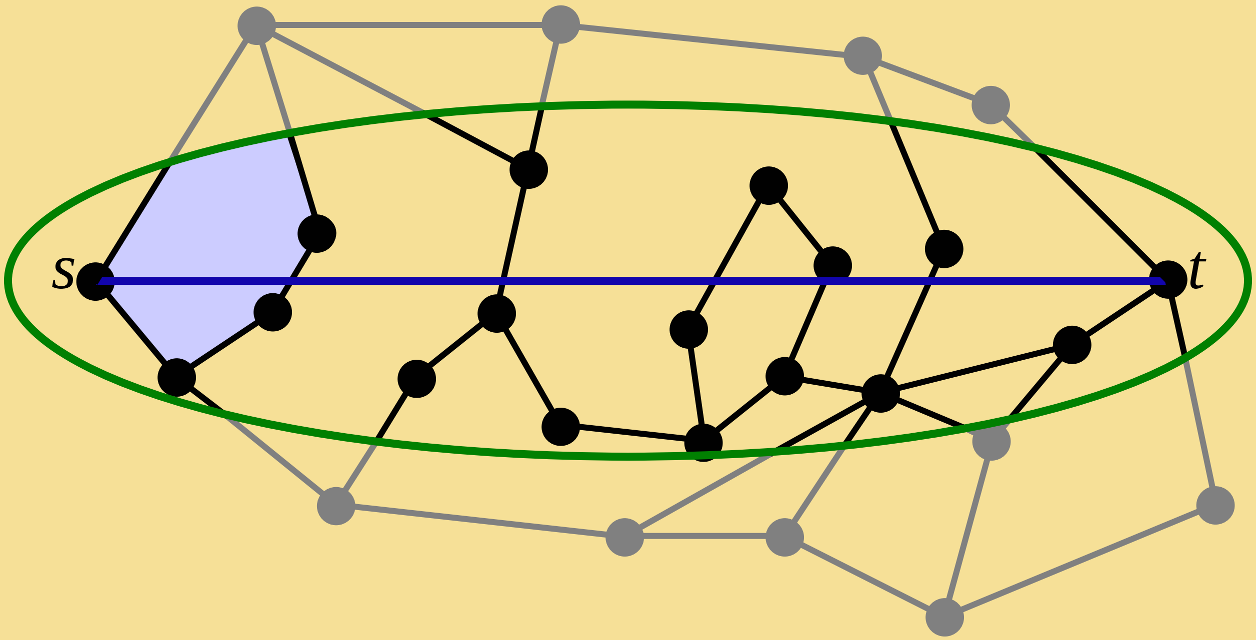
Adaptive Face Routing



ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Distributed Computing Group

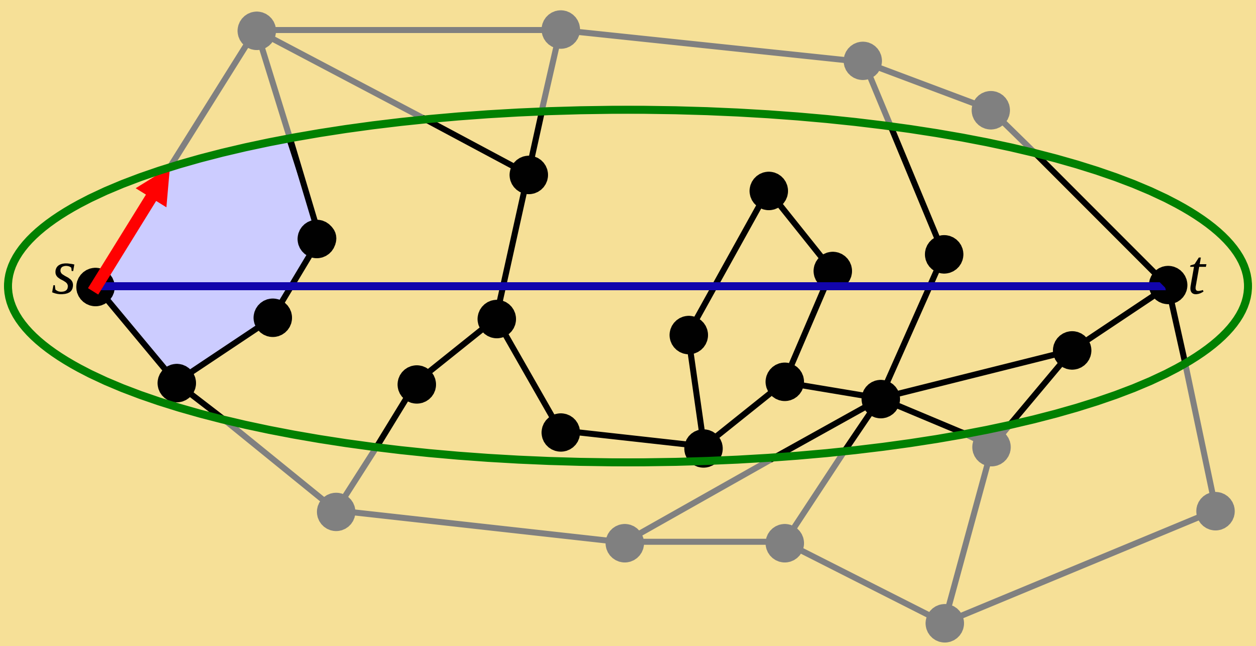
Adaptive Face Routing



Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Distributed Computing Group

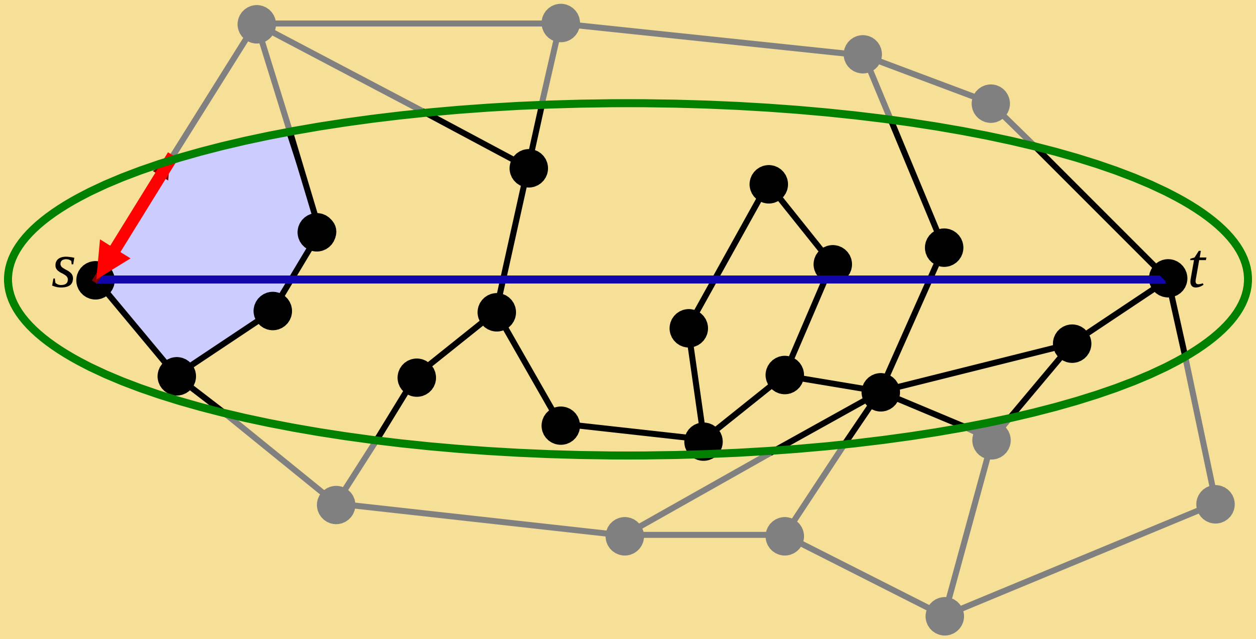
Adaptive Face Routing



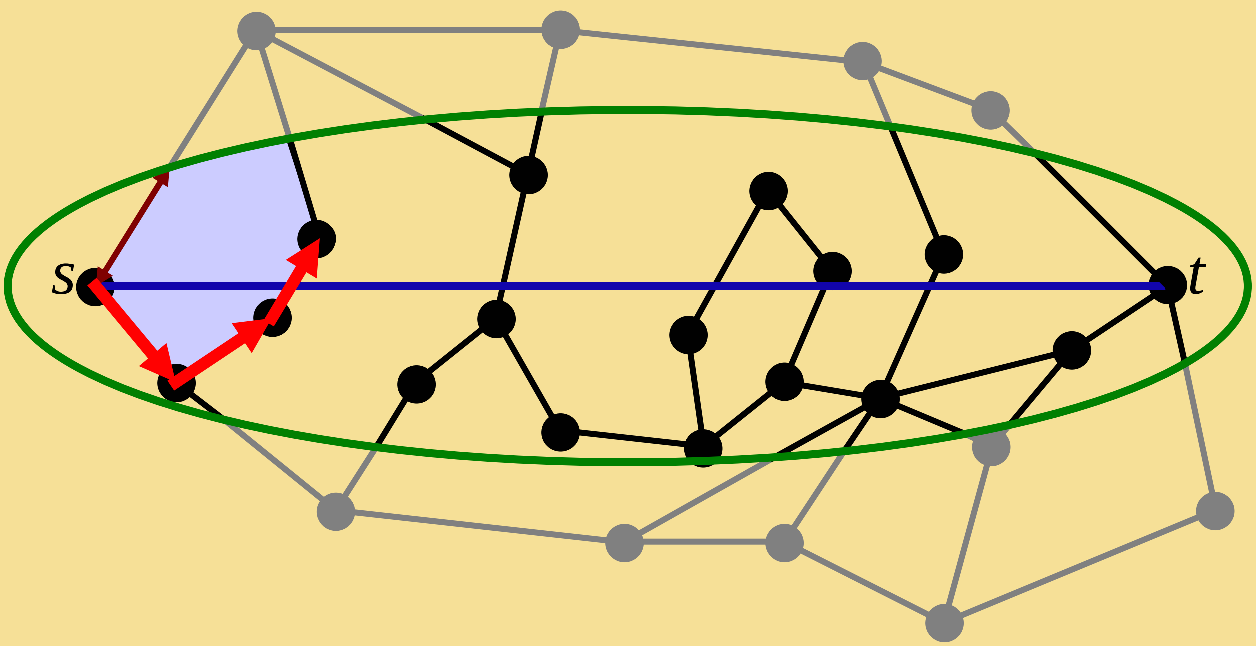
ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Distributed Computing Group

Adaptive Face Routing



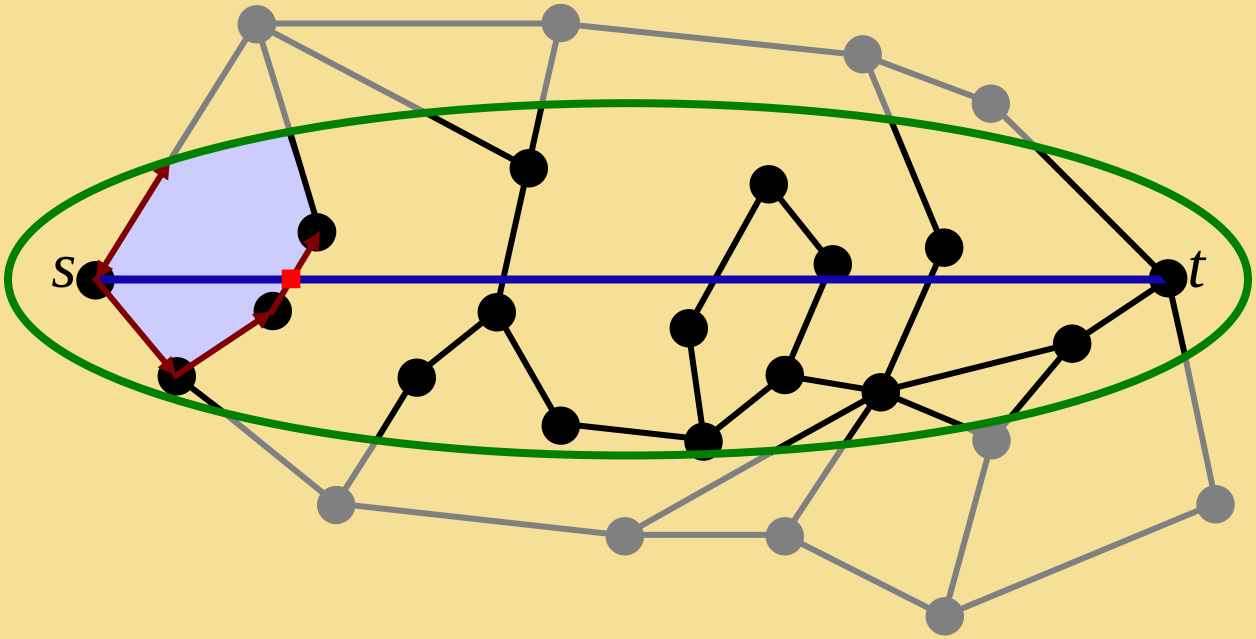
Adaptive Face Routing



ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Distributed Computing Group

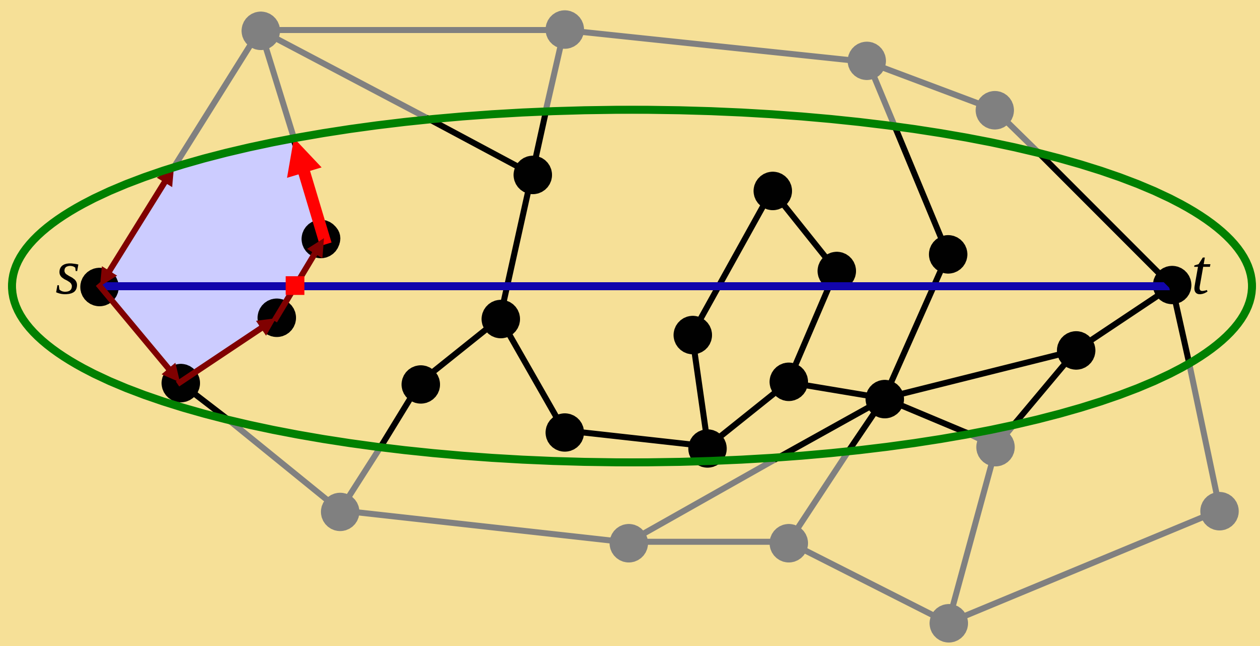
Adaptive Face Routing



ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Distributed Computing Group

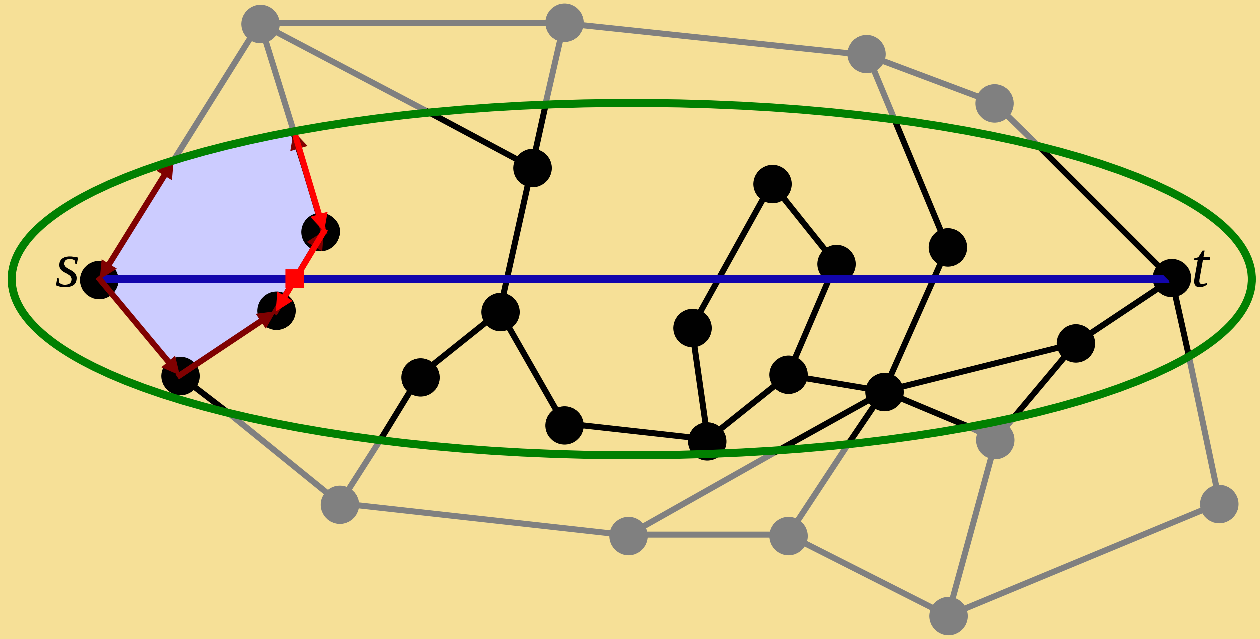
Adaptive Face Routing



ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Distributed Computing Group

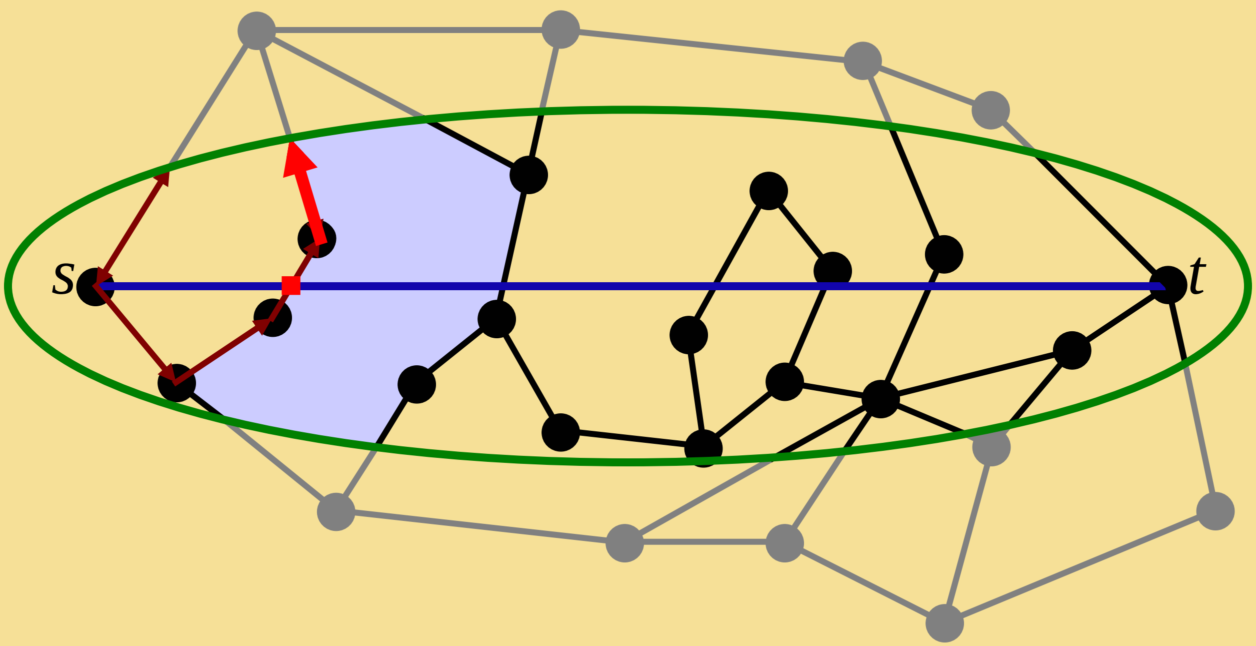
Adaptive Face Routing



ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Distributed Computing Group

Adaptive Face Routing

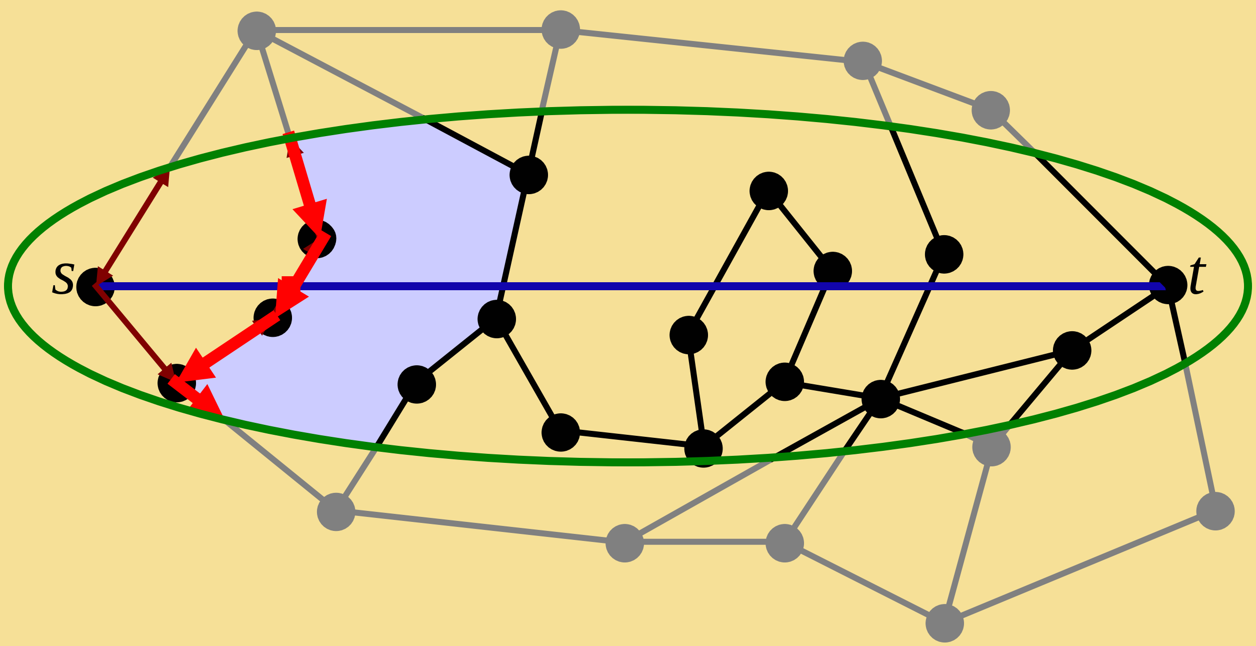


ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

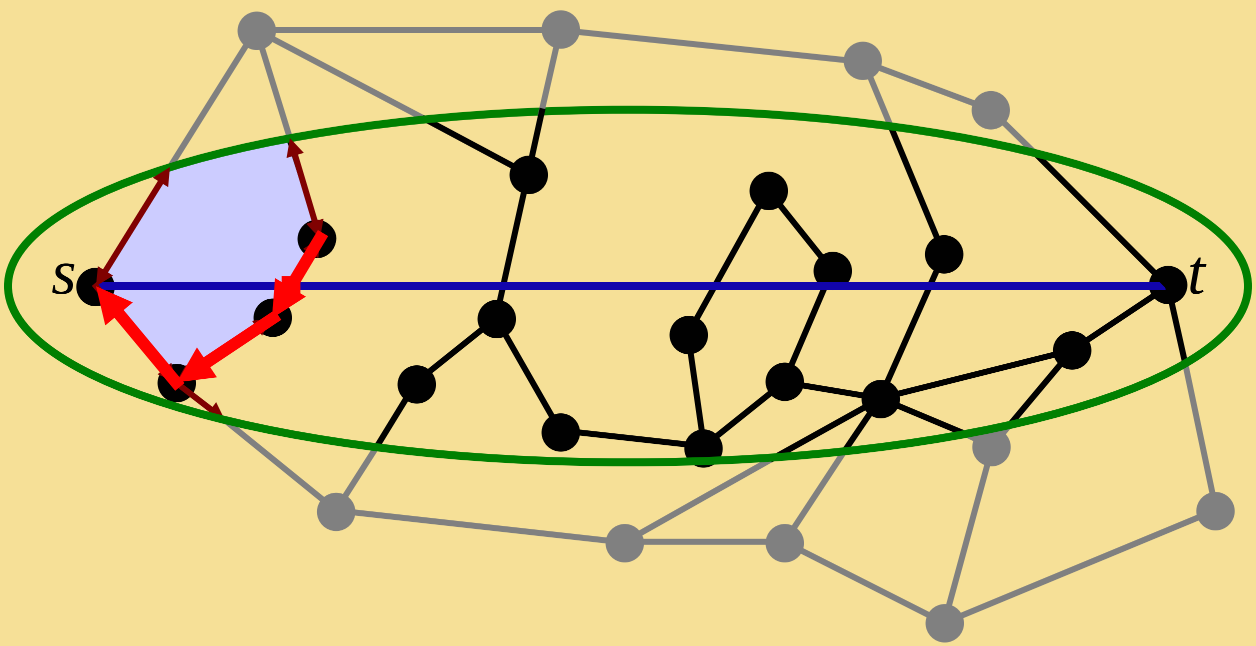
Distributed Computing Group

Adaptive Face Routing

Ellipse is too small, go back!



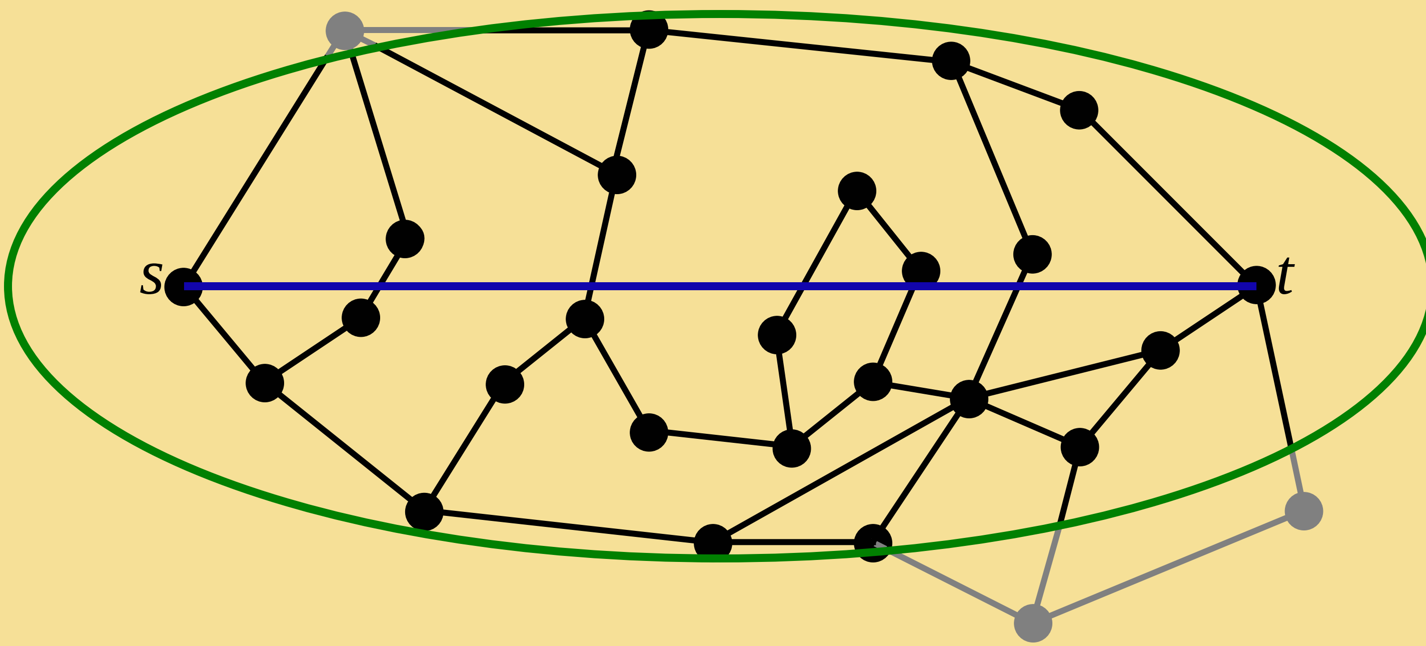
Adaptive Face Routing



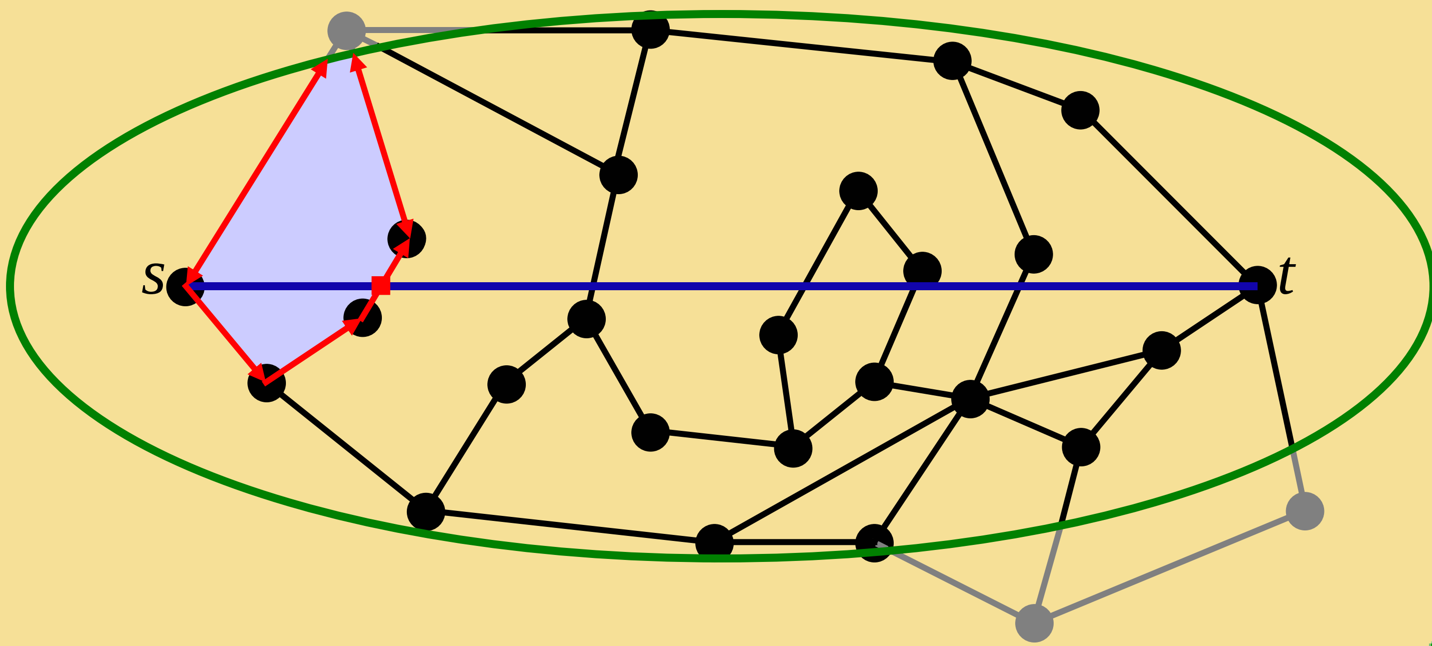
ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Distributed Computing Group

Adaptive Face Routing



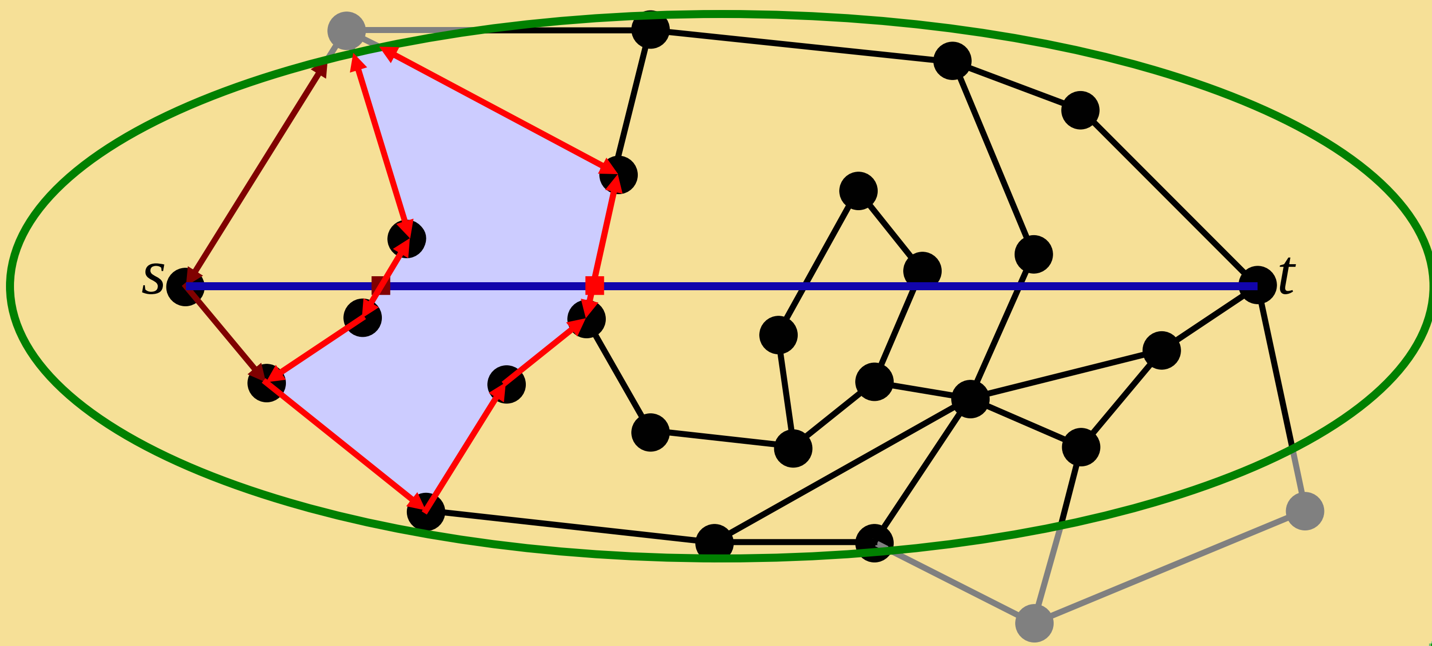
Adaptive Face Routing



ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Distributed Computing Group

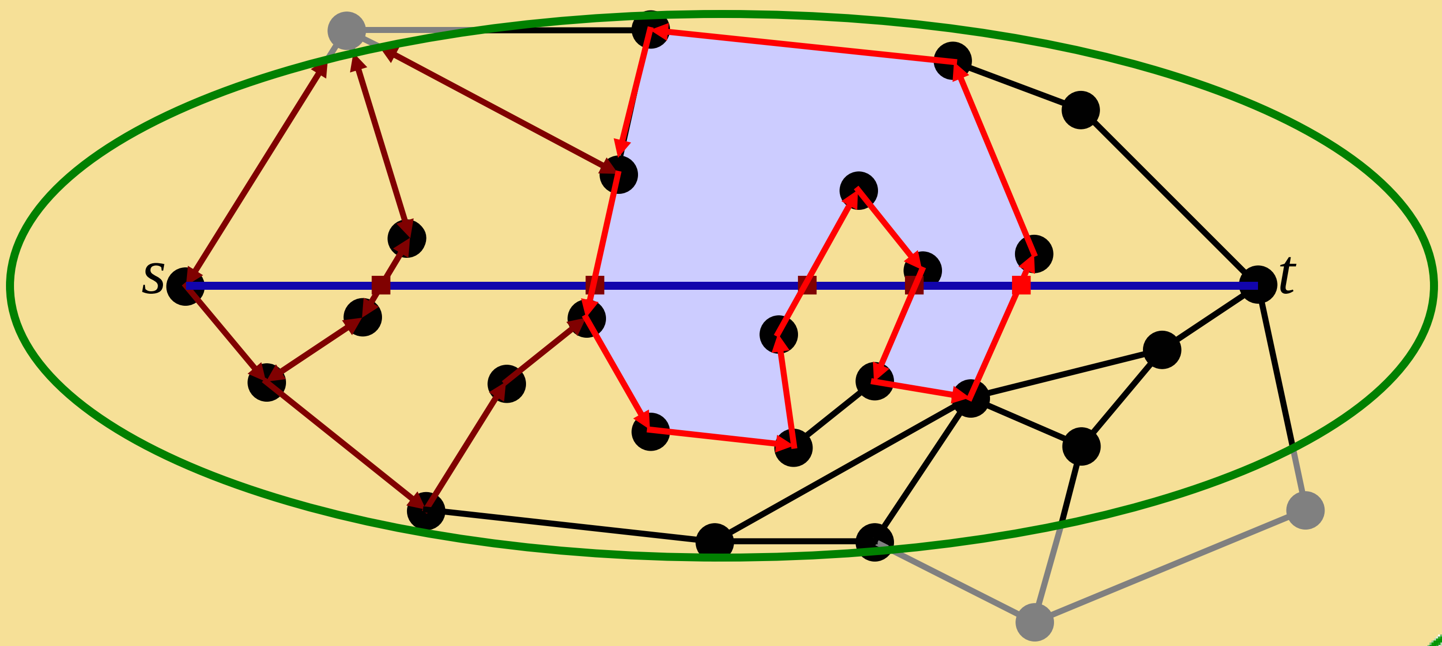
Adaptive Face Routing



Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Distributed Computing Group

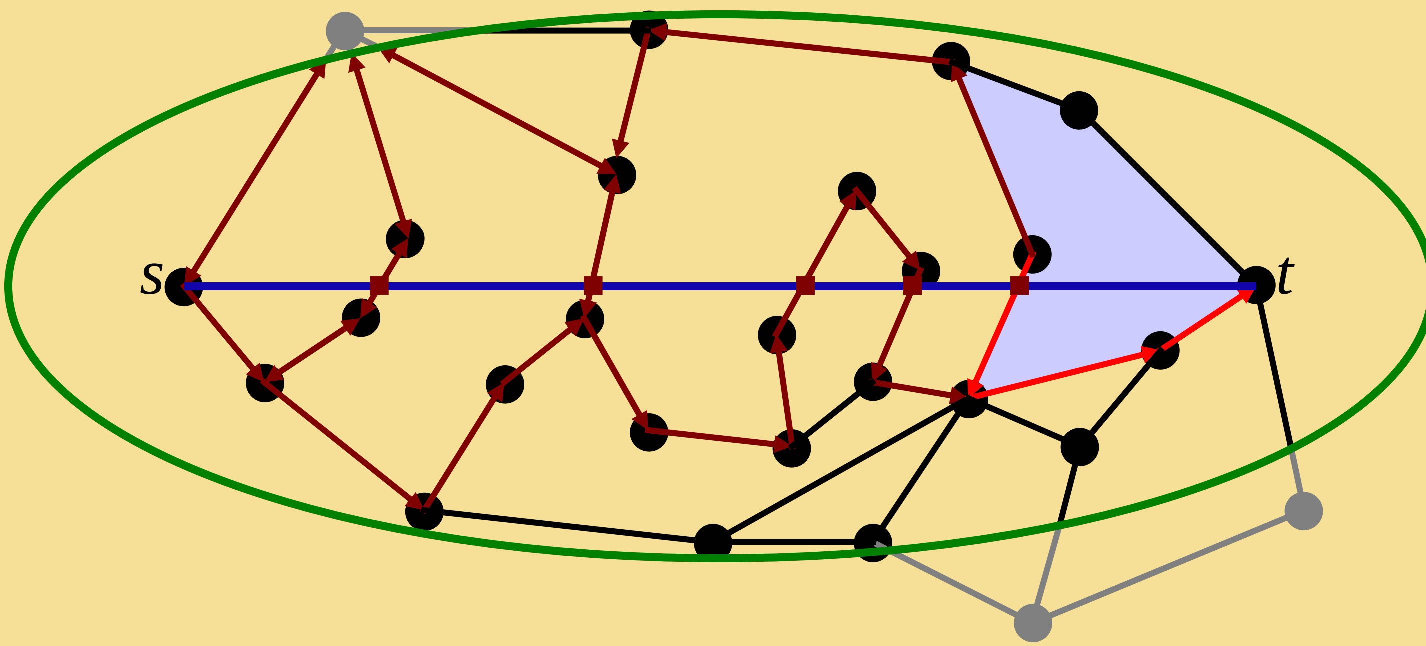
Adaptive Face Routing



ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Distributed Computing Group

Adaptive Face Routing



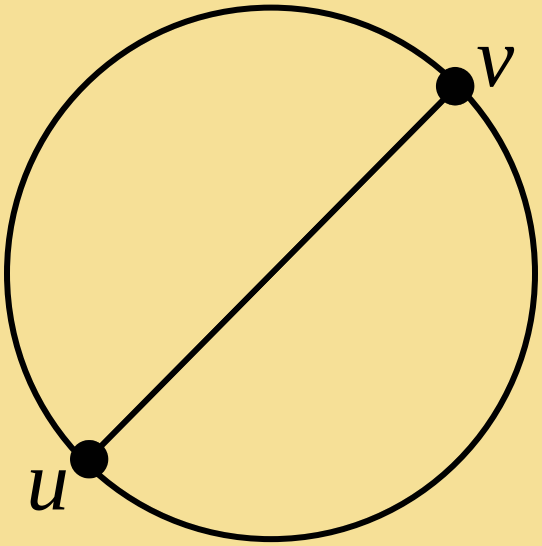
AFR on the Unit Disk Graph

- AFR needs a planar graph, UDG is not planar
- need a planar subgraph of UDG:
  - simple distributed construction
  - spanner for link distance, Euclidean, and energy metric

Gabriel Graph

Definition:

Two nodes u and v are connected by an edge iff the circle with $\overline{uv}$ as diameter contains no other node.



Properties of GG ∩ UDG

- For each pair of nodes, the GG (∩ UDG) contains an energy optimal path (on UDG).

  ⇒ spanner for link, Eucl. dist., energy ($\Omega(1)$-model)
- planar
- no additional communication

⇒ meets all our requirements

AFR Complexity

Theorem 1:

Let c_* be the cost (link, Euclidean, or energy) of an optimal path between two nodes on the UDG. Applying AFR on GG∩UDG then terminates with cost $O(c_*^2)$.

AFR Complexity, Proof I

Lemma:

For each used ellipse $\mathcal{E}$, the cost is linear in the number of nodes in $\mathcal{E}$.

Collorary:

In the $\Omega(1)$-model, for each used ellipse $\mathcal{E}$, the cost is linear in area covered by $\mathcal{E}$.

AFR Complexity, Proof II

Ellipses grow exponentially

$$\Downarrow$$

Lemma:

The cost, AFR needs to route a packet, is linear in the area covered by the last used ellipse.

AFR Complexity, Proof III

Lemma:

Using an ellipse $\mathcal{E}$, AFR finds a path from s to t iff there is such a path inside $\mathcal{E}$.

Lemma:

All paths of (Euclidean) length smaller or equal to c are inside an ellipse whose area is in $O(c^2)$.

AFR Complexity, Proof IV

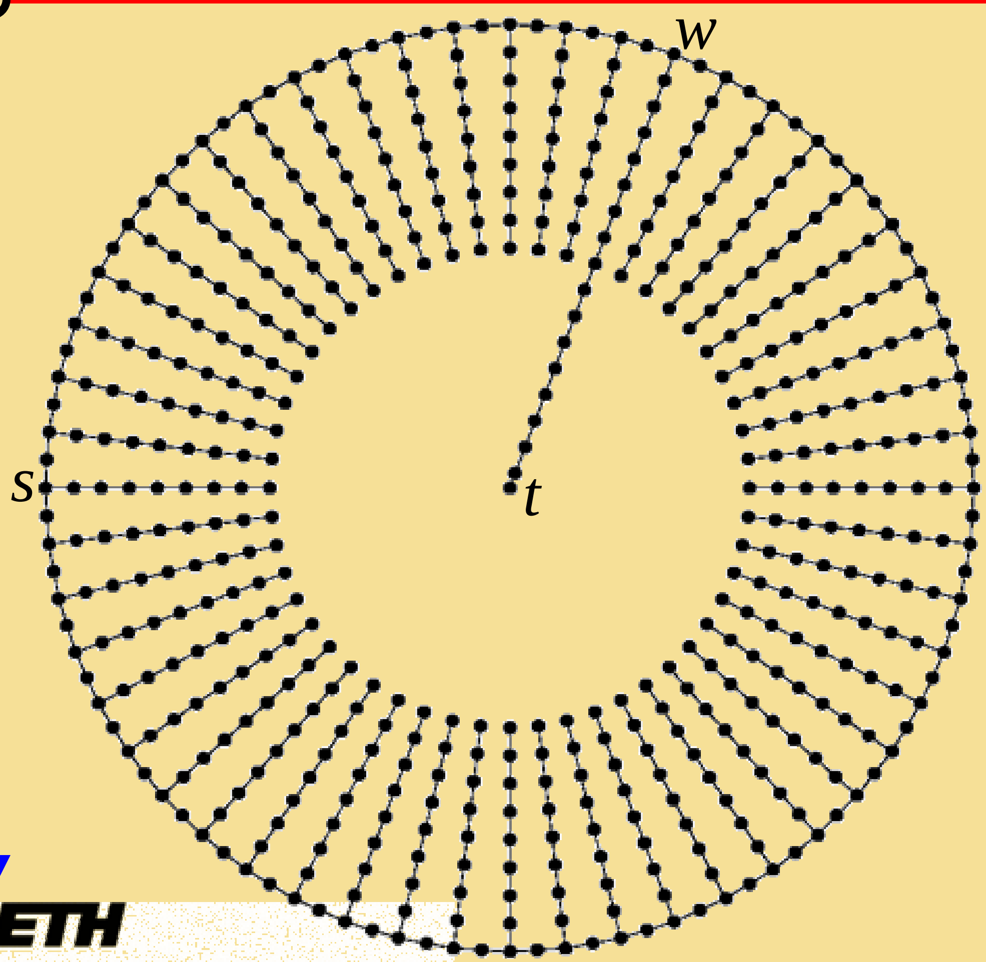
All the lemmas together now prove Theorem 1.

Lower Bound

Theorem 2:

Let c_* be the cost (link, Euclidean, or energy) of an optimal path between two nodes on a UDG $\mathcal{G}$. For each geometric routing algorithm, there is a graph for which the cost is $\Omega(c_*^2)$.

Lower Bound, Proof



Cost of optimal path:

$$c_* \leq (\pi + 1)R$$

Exp. cost for an algorithm $\mathcal{A}$:

$$\mathrm{E}[c(\mathcal{A})] \geq \frac{\pi R}{2} \cdot \Theta(R)$$
$$= \Theta(R^2)$$

Main Theorem

Theorem 3:

On the Unit Disk Graph in the $\Omega(1)$-model, AFR is asymptotically optimal.

(follows directly from Theorems 1 and 2)

Conclusion I

$\Omega(1)$ restriction can be dropped by clustering:

- works fine for link and Euclidean distance (still $\Theta(c_*{}^2)$)
- For energy, it can be shown that the cost of a geometric routing alg. cannot be bounded by a function of c_* alone.

Conclusion II

- The lower bound holds for all routing algorithms which have only local knowledge at the beginning.
- if coordinates of dest. are not known, but if each node can store some bits: Then there is a simple flooding variant which achieves $O(c_*^2)$.