Oblivious Routing on Geometric Networks

Costas Busch, Malik Magdon-Ismail and Jing Xi
{buschc,magdon,xij2}@cs.rpi.edu

July 20, 2005.
Oblivious Routing: Background and Our Contribution

- The Algorithm: Oblivious Routing with Single Intermediate Node
- Good Geometric (Metric) Embeddings; Examples
- Routing Result; Examples
- Discussion
Routing: construct “good” paths given sources and destinations.

- Communication Networks – eg. Internet.
- Ad-hoc Networks – eg. sensor networks.
- Parallel Architectures – eg. Mesh.
- ...
Routing: construct “good” paths given sources and destinations.

- Communication Networks – eg. Internet.
- Ad-hoc Networks – eg. sensor networks.
- Parallel Architectures – eg. Mesh.
- ...

Wish List: simple, scalable, efficient, near-optimal, general
Oblivious Routing

A packet’s path is specified independently of other packets’ paths.

- Suffices to specify algorithm for any single packet to select its path.
- Every packet uses this algorithm independent of other packets.

distributed, hence scalable;
applies to dynamic (online) setting with streaming packets;

( A packet $\pi$ is a source-destination pair $\langle s, t \rangle$)
Optimality of Paths

Congestion

$C(v)$: number of paths using node $v$

$C(v) = 6$

Congestion ($C$): $\max_e C(e)$

$C^*$: optimal congestion

Stretch

$\text{stretch}(\pi): \frac{D(\pi)}{d(\pi)}$ ← packet's path length

stretch: $\max_\pi \text{stretch}(\pi)$

$\text{stretch} = 2$

Optimal: $C = O(C^*)$; stretch $= O(1)$

(Similarly can define w.r.t. edge congestion $C_{edge}$.)

Srinivasan et al. [STOC97]: Near-optimal; offline; non-oblivious.
**Related Work – Opt.** $C$

### $d$-dim Mesh:

$C = O(C^* \cdot d \cdot \log n)$

(Maggs et al. [FOCS97])

(Also gave a lower bound $\Omega(C^* \cdot \frac{1}{d} \cdot \log n)$)

### Arbitrary:

$C = O(C^* \log^3 n)$

Räcke [FOCS02], (non-constructive)

Azar et al. [STOC03]

Harrelson et al. [SPAA03]

Bienkowski et al. [SPAA03] (Polynomial-time, constructive)

Bansal et al. [SPAA03], (On-line version)

Oblivious algorithms with near-optimal $C$; **Unbounded** stretch

*simple, scalable, efficient, near-optimal, general*
Congestion–Stretch Trade Off

Good stretch, Bad C  Bad stretch, Good C

We want **Good Stretch, Good C**
Related Work – Opt. $C, \text{stretch}$

$d$-dim Mesh: $C = O(C^* \cdot d^2 \cdot \log n)$; stretch $= O(d^2)$;
(Busch et al. [IPDPS05])
(Also lower bound $\Omega(\log d(\pi))$ random bits per packet)
(Scheidler (class notes) indep. considered $d = 2$)

Arbitrary: Not Possible (constructive).

simple, scalable, efficient, near-optimal, general
Hierarchical Decompositions

Existing algorithms use hierarchical network decompositions
Hierarchical Decompositions

Existing algorithms use hierarchical network decompositions
Hierarchical Decompositions

Existing algorithms use **hierarchical network decompositions**
Hierarchical Decompositions

Existing algorithms use **hierarchical network decompositions**
Hierarchical Decompositions

Existing algorithms use **hierarchical network decompositions**
Hierarchical Decompositions

Existing algorithms use hierarchical network decompositions
The Gap

simple, scalable, efficient, near-optimal, general
↓
simple, scalable, efficient, near-optimal, general
↓?
simple, scalable, efficient, near-optimal, general

(Not Possible)
Our Contribution

simple, scalable, efficient, near-optimal, general

↓

simple, scalable, efficient, near-optimal, general

↓

simple, scalable, efficient, near-optimal, general

Link Routing to Finite Metric Embedding
Our Contribution

**Simple:** Single intermediate point algorithm, not based on hierarchical decomposition.

**Near-Optimal:** For networks that have low distortion embeddings, eg. Mesh, sensor networks.

**General:** Result holds for all networks.
General Idea: Diffusive Routing

Imagine the network in space [Network Embedding]

source: $s$; destination: $t$. 
General Idea: Diffusive Routing

Imagine the network in space [Network Embedding]

Packet “diffuses” out from $s$ – congestion spreads.
General Idea: Diffusive Routing

Imagine the network in space [Network Embedding]

Packet “focuses” back to $t$. 
General Idea: Diffusive Routing

Imagine the network in space [Embedded Network]

Diffusion by random choice of intermediate node.
Outline

• Oblivious Routing: Background and Our Contribution

• The Algorithm: Oblivious Routing with Single Intermediate Node

• Good Geometric (Metric) Embeddings; Examples

• Routing Result; Examples

• Discussion
I: Embed the Network

Embedding function $f : v \in V \mapsto x \in A \subset \mathbb{R}^2$; \( \{v_1, \ldots, v_n\} \mapsto \{x_1, \ldots, x_n\} \)
II: Random Intermediate Point

\[ |\ell\perp| = |\ell| \]
Choose a node close to the random intermediate point.
III: Follow the Geodesic

Choose a path as close as possible to the geodesic.
Choose a path as close as possible to the geodesic.
III: Follow the Geodesic

Choose a path as close as possible to the geodesic.
Choose a path as close as possible to the geodesic.
Outline

• Oblivious Routing: Background and Our Contribution

• The Algorithm: Oblivious Routing with Single Intermediate Node

• Good Geometric (Metric) Embeddings; Examples

• Routing Result; Examples

• Discussion
What Can Go Wrong?

Entire diffusive area is not within the network.

\( \gamma \) (pseudo-convexity): fraction guaranteed to lie within network.

– Want \( \gamma \) to be large (note, \( \gamma \leq \frac{1}{2} \)).
What Can Go Wrong?

There is no node close to the random intermediate node $R$ (Coverage Radius): Maximum distance to an intermediate node.
– Want $R$ to be small.
What Can Go Wrong?

No Geodesic following path

\( \Delta \) (Deviation): Furtherest a geodesic path gets from the geodesic.
– Want \( \Delta \) to be small.
What Can Go Wrong?

Geodesic following paths have large stretch.

$\Sigma$ (Geodesic Stretch): Maximum stretch of a geodesic path.

– Want $\Sigma$ to be small.
What Can Go Wrong?

Using an intermediate node is costly (large stretch).

\[ \text{dist}_E(s, w) + \text{dist}_E(w, t) \leq \sqrt{2} \text{dist}_E(s, t). \]

Want \( \text{dist}_G \approx \text{dist}_E \).

**Distortion:** \( \alpha \leq \frac{\text{dist}_G(u, v)}{\text{dist}_E(u, v)} \leq \beta \)

(w.l.o.g. \( \alpha = 1 \))

\( \text{dist}_E = \text{Euclidean distance, dist}_G = \text{Graph distance} \)
## Graph Embedding Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>What is it?</th>
<th>Best If</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\gamma$</td>
<td>Pseudo-Convexity</td>
<td>Min. diffusive area in network</td>
</tr>
<tr>
<td>$R$</td>
<td>Coverage-Radius</td>
<td>Max. distance to intermediate node</td>
</tr>
<tr>
<td>$\Delta$</td>
<td>Deviation</td>
<td>Max. stray of geodesic path</td>
</tr>
<tr>
<td>$\Sigma$</td>
<td>Geodesic Stretch</td>
<td>Max. stretch of geodesic paths</td>
</tr>
<tr>
<td>$\beta$</td>
<td>Distortion</td>
<td>How closely $\text{dist}_G$ matches $\text{dist}_E$</td>
</tr>
</tbody>
</table>

Note: Embedding parameters are not independent. eg.  
- $\gamma$ and $R$ are interdependent.  
- Smaller deviation embedding may have a larger stretch.
Examples

Certain networks have natural embeddings:

Mesh, sensor networks (disc graphs), . . . .
Examples

Mesh Sensor Network Mesh with Hole

\[ \gamma = \frac{1}{2} \]
\[ R = \frac{1}{\sqrt{2}} \]
\[ \Delta = \frac{1}{\sqrt{2}} \]
\[ \Sigma = 1 \]
\[ \beta = \sqrt{2} \]

Geodesic following paths are shortest paths.
No two nodes are closer than $L$. Each unit square contains from 1 to $k = O(1/L^2)$ nodes. $r = 2\sqrt{2}$. (max. degree $\delta \leq 32k$.)

Geodesic paths constructed from unit square path.
Examples

Mesh Sensor Network Mesh with Hole

Geodesic following paths are shortest paths.

\[ \gamma = \frac{1}{2} \]
\[ R \rightarrow \frac{1}{\sqrt{2}} + r \]
\[ \Delta \rightarrow \frac{1}{\sqrt{2}} + r \]
\[ \Sigma = 1 \]
\[ \beta \leq 5 \]
Examples

Mesh

Sensor Network

Mesh with Hole

\[ \sqrt{n} \]

\[ \sqrt{n} \]

\[ \gamma = O(1) \]
\[ \beta = O(1) \]
\[ \Delta = O(1) \]
\[ \Sigma = O(1) \]

\[ R = O(1) \]

\[ (L=\text{min. node sep.}) \]

\[ (r=\text{size of hole.}) \]
Outline

• Oblivious Routing: Background and Our Contribution

• The Algorithm: Oblivious Routing with Single Intermediate Node

• Good Geometric (Metric) Embeddings; Examples

• Routing Result; Examples

• Discussion
Theorem.

$$\text{stretch} = O(\beta \cdot R \cdot \Sigma)$$

The stretch depends on:
- Quality of the embedding: $\beta$;
- Coverage density: $R$;
- Geodesic stretch: $\Sigma$. 
\(-\) distance stretch is \(O(1+R)\).
\(-\) \(\beta\) links distances to graph distances.
\(-\) \(\Sigma\) is stretch introduced by geodesic paths.

\[ \beta \cdot R \cdot \Sigma \]
Congestion

Theorem.

\[ C \leq f(\gamma, R, \Delta, \beta; n) \cdot C^* \]

\[ f = O\left(\frac{\beta^2(R + \Delta)}{\gamma} \cdot ((\beta + \Delta)^2 + \log(n + R))\right) \]

\[ (f = O(\log n)) \]

The Congestion depends on:
- Optimal Congestion: \(C^*\);
- Extent of diffusion: \(\gamma\);
- Quality of the embedding: \(\beta\);
- Coverage density: \(R\);
- Deodesic deviation: \(\Delta\).
**Lemma 1.** Probability \((P)\) source at distance \(d\) uses node \(v\). \(P \sim \frac{1}{d}\). (so \(E[C(v)] \sim \sum d \frac{N_d}{d}\), where \(N_d = \text{Number of sources distance } d \text{ from } v\))
Lemma 2. \( C^* \sim \frac{N_d}{d} \).

(so \( E[C(v)] \sim \sum_d \frac{N_d}{d} \sim \sum_d C^* \sim C^* \log n \))

- destination is \( \sim 2d \) away.
- use at least \( \sim d \) nodes in \( 2d \)-disc.
- total node usage \( \sim N_d \cdot d \).
- number of nodes in disc \( \sim d^2 \).
- pigeonhole: \( \exists \) node used \( \sim \frac{N_d}{d} \) times.
Examples

Mesh

Sensor Network

Mesh with Hole

\[ \gamma = O(1) \]
\[ R = O(1) \]
\[ \Delta = O(1) \]
\[ \Sigma = O(1) \]
\[ \beta = O(1) \]

\[ \gamma = O(1) \]
\[ R = O(1) \]
\[ \Delta = O(1) \]
\[ \Sigma = O(1) \]
\[ \beta = O(\frac{1}{L}) \]

\[ \text{stretch: } O(1) \]
\[ C : O(C^* \log n) \]

\[ \gamma = O(1) \]
\[ R = O(r) \]
\[ \Delta = O(r) \]
\[ \Sigma = O(1) \]
\[ \beta \leq O(1) \]

\[ \text{stretch: } O(\frac{1}{L}) \]
\[ C : O(C^* \frac{\log n}{L^2}) \]

\[ \text{stretch: } O(r) \]
\[ C : O(C^* \cdot r \log n) \]

\[ (L=\text{min. node sep.}) \]
\[ (r=\text{size of hole.}) \]
Wrap Up

– Embedding Parameters: $\gamma, R, \Delta, \Sigma, \beta$.

– Good embeddings: Good embedding parameters.

– Diffusive Routing: stretch $= O(1)$; $C = O(C^* \log n)$.

simple, scalable, efficient, near-optimal, general

Ongoing: Can we remove the dependence on $\gamma$. 