

Routing in Distributed Systems

- Lecture 3 -

Cyril Gavoille

University of Bordeaux, France

April 6-9, 2009
Valparaíso, Chile



- ① Routing with Succinct Tables
- ② Constructing Sparse Spanners
- ③ Routing v.s. Spanners

- Lecture 3 -

Routing v.s. Spanners

Outline

The Question and the Answer

A Naive Approach

The Proof

Outline

The Question and the Answer

A Naive Approach

The Proof

Summary

stretch	size	
$2k - 1$	$O(n^{1+1/k})$	Greedy Algorithm
$2k - 1$	$O(kn^{1+1/k})$	Tree Cover
$4k - 5$	$\tilde{O}(kn^{1/k})$	Routing (using T.C.)
$+ 2$	$O(n^{3/2})$	Tree Cover
$+ 6$	$O(n^{4/3})$	Spanner (T.C.?)
$+ f(k)$	$O(n^{1+1/k})$???	for $k > 3$

Recall: a tree cover has stretch s if, for all nodes x, y of G , there exists a T in the cover such that $d_T(x, y) \leq s \cdot d_G(x, y)$. Its size is the number of edges in the subgraph induced by all its trees.

The Question

Question:

Can we make additively stretched spanners routable?

The Question

Question:

Can we make additively stretched spanners routable?

!!! It might be the wrong question !!!

The Question

Question:

Can we make additively stretched spanners routable?

!!! It might be the wrong question !!!

Question (v2):

Is there a universal routing scheme with sublinear space and additive stretch?

Yes or No?

Yes or No?

PRO: Numerology!

Spanner: stretch-3 for size $O(n^{3/2})$

Routing: stretch-3 for size $\tilde{O}(n^{1/2})$

Spanner: stretch- $O(k)$ for size $O(n^{1+1/k})$

Routing: stretch- $O(k)$ for size $\tilde{O}(n^{1/k})$

Is it really just a coincidence?

Yes or No?

PRO: Numerology!

Spanner: stretch-3 for size $O(n^{3/2})$

Routing: stretch-3 for size $\tilde{O}(n^{1/2})$

Spanner: stretch- $O(k)$ for size $O(n^{1+1/k})$

Routing: stretch- $O(k)$ for size $\tilde{O}(n^{1/k})$

Is it really just a coincidence?

- There exist spanners of size $o(n^2)$ with constant additive stretch (ex: size $n^{3/2}$ or $n^{4/3}$ for stretch +2 or +6).
- It should exist sublinear compact routing scheme with constant additive stretch!!!

Yes or No?

CON: spanners do not tell us how to **route** on sparse graphs.

The problem is:

- Spanner: prove \exists a short path
- Routing: construct a short path

An Impossibility Result

Theorem (2009)

Every routing strategy providing, for each unweighted connected n -node graph, a labeled routing scheme with tables and addresses $\leq \mu$ bits, produces, for some graph, an additive stretch $\Omega(n^{1/3}/\mu^{2/3})$.

An Impossibility Result

Theorem (2009)

Every routing strategy providing, for each unweighted connected n -node graph, a labeled routing scheme with tables and addresses $\leq \mu$ bits, produces, for some graph, an additive stretch $\Omega(n^{1/3}/\mu^{2/3})$.

Corollary: The additive stretch of every universal routing strategy with tables and addresses in $o(\sqrt{n})$ is unbounded.

In particular, the stretch-7 routing scheme with $\tilde{O}(n^{1/3})$ -bit tables must have an additive stretch of $n^{\Omega(1)}$.

Outline

The Question and the Answer

A Naive Approach

The Proof

Idea: Amplify Detours

- 1 Consider a graph G for which every stretch- s routing scheme requires tables of size $\geq \mu$.
[if $|\text{tables}| < \mu$, then the additive stretch is $> s - 1$]

Idea: Amplify Detours

- 1 Consider a graph G for which every stretch- s routing scheme requires tables of size $\geq \mu$.
[if $|\text{tables}| < \mu$, then the additive stretch is $> s - 1$]
- 2 Construct G' from G by replacing each edge by a path of length δ .

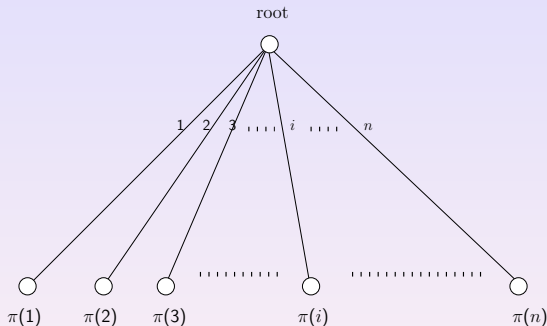
Idea: Amplify Detours

- 1 Consider a graph G for which every stretch- s routing scheme requires tables of size $\geq \mu$.
[if $|\text{tables}| < \mu$, then the additive stretch is $> s - 1$]
- 2 Construct G' from G by replacing each edge by a path of length δ .

Our Guess: Every stretch- s routing scheme on G' with tables of size $< \mu$ must have an additive stretch $> \delta(s - 1)$.

Example: Lower Bound for Stars

Restriction: addresses are fixed by an adversary, and cannot be changed/optimized by the designer of the routing scheme (name-independent).

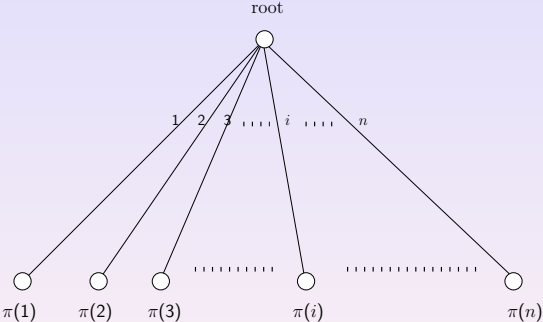


Example: Lower Bound for Stars

Restriction: addresses are fixed by an adversary, and cannot be changed/optimized by the designer of the routing scheme (name-independent).



Claim: if $\mu < n \log n$,
then stretch $\geq +2$.

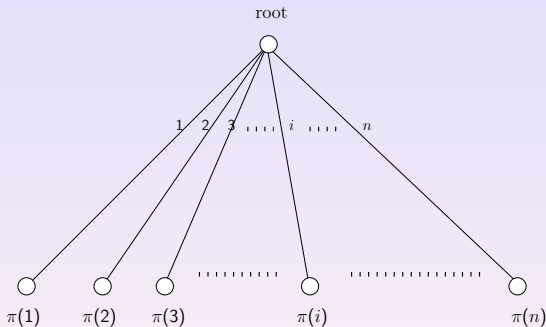


Example: Lower Bound for Stars

Restriction: addresses are fixed by an adversary, and cannot be changed/optimized by the designer of the routing scheme (name-independent).



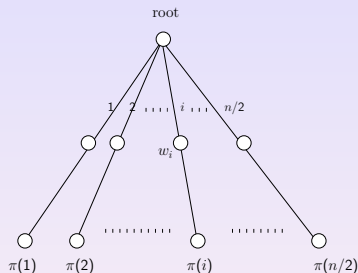
Claim: if $\mu < n \log n$,
then stretch $\geq +2$.



Proof: if stretch $< +2$, then the number of distinct routing tables at the root must be $\geq n!$. \Rightarrow a star needs at its root $\mu \geq \log(n!) \sim n \log n$.

Longer Detour

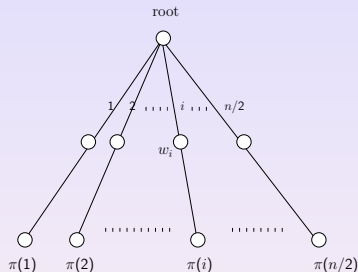
Let us show that for some small enough constant $c > 0$:
if $\mu < cn \log n$, then stretch $> +2$.



Proof #1: Consider routing from root to any leaf $y = \pi(i)$.
If $\mu < \log((n/2)!) \sim \frac{1}{2}n \log n$, then stretch $\geq +4???$

Longer Detour

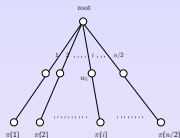
Let us show that for some small enough constant $c > 0$:
if $\mu < cn \log n$, then stretch $> +2$.



Proof #1: Consider routing from root to any leaf $y = \pi(i)$.
If $\mu < \log((n/2)!) \sim \frac{1}{2}n \log n$, then stretch $\geq +4$???

No! because we traverse new nodes (possibly two) before selecting the right branch. These nodes have μ bits of information and might change the decision at the root.

Longer Detour



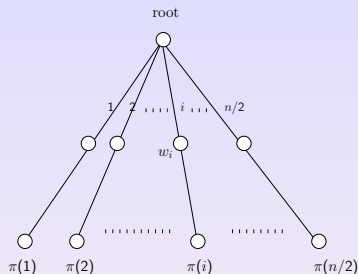
Proof #2: Set μ low enough: $2\mu < \log((n/2)!)$, or $\mu \sim \frac{1}{4}n \log n$. If stretch $\leq +2$ for all y , then the route to y has to traverse a new node w not on the y 's branch. By the choice of μ ,

$$\mu(w) + \mu(\text{root}) \leq 2\mu < \log((n/2)!)$$

We know that, after a detour of $+2$, the root has not enough information yet to correctly route to all y . So, the route requires a second detour of $+2$.

$$\Rightarrow \text{If } \mu < \frac{1}{4}n \log n - O(n), \text{ then stretch } \geq +4.$$

WRONG! WRONG! WRONG! WRONG! WRONG!



Routing scheme with $O(\log n)$ -bit tables and stretch $+2$:

Middle node w_i in the branch of $\pi(i)$ stores integer p such that $\pi(p) = i$. If we arrive at w_i , it means we want to route to the leaf named " i ".

Routing from root $\rightarrow i \in [1, n/2]$: 1) Route to port i ; 2) Read p at w_i ; 3) Come back to the root; 4) use port p . Stretch is $+2$.

Conclusion of this Story

One cannot design lower bounds on Information Theory based on arguments like:

“I know that node x does not know information I ,
so it has to store it.

Thus node x must store $|I|$ bits.”

Proving that x stores at least $|I|$ bits requires a proof.

Outline

The Question and the Answer

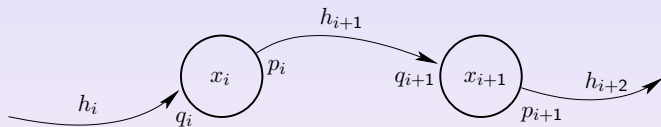
A Naive Approach

The Proof

Preliminaries

Let R be a routing scheme on G , i.e.,

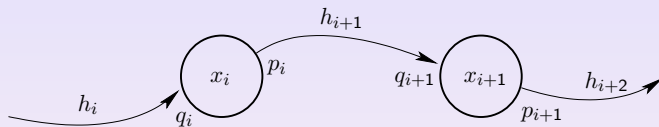
$$R : (x_i, h_i, q_i) \mapsto (p_i, h_{i+1})$$



Preliminaries

Let R be a routing scheme on G , i.e.,

$$R : (x_i, h_i, q_i) \mapsto (p_i, h_{i+1})$$



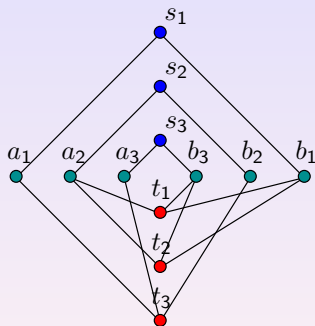
Definition

The “memory requirements” of R at x is the size in bits of the smallest program (say in \mathbb{C}) implementing function $R(x, \cdot, \cdot)$.

The Graph Family $\mathcal{F} = \mathcal{F}(p, \delta)$

Graphs of \mathcal{F} are constructed from all $p \times p$ boolean matrices.

Sets of p nodes: $S = \{s_i\}$, $A = \{a_i\}$, $B = \{b_i\}$, and $T = \{t_j\}$.



$$M = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

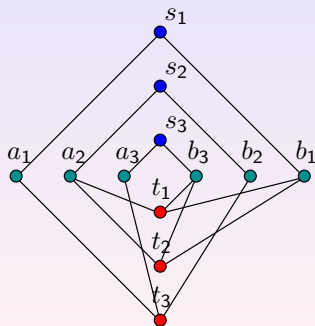
Connect a path of length δ between: $s_i \rightsquigarrow a_i$, $s_i \rightsquigarrow b_i$, and $t_j \rightsquigarrow a_i$ if $M[i, j] = 1$, and $t_j \rightsquigarrow b_i$ if $M[i, j] = 0$.

Properties

Consider any $G = G(M) \in \mathcal{F}$.

Property

Every **walk** in G from s_i to t_j of length $\rho < d_G(s_i, t_j) + 2\delta$ contains node a_i if and only if $M[i, j] = 1$.



$$M = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Consider any routing strategy for \mathcal{F} producing, for each $G \in \mathcal{F}$, a routing scheme R with addresses and tables of size $\leq \mu$ and additive stretch $\beta < 2\delta$.

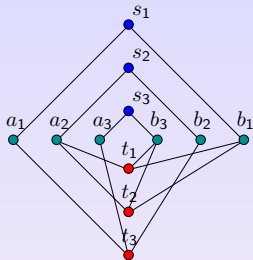
Observation: Property applies to the route length of R from s_i to t_j .

Consider any routing strategy for \mathcal{F} producing, for each $G \in \mathcal{F}$, a routing scheme R with addresses and tables of size $\leq \mu$ and additive stretch $\beta < 2\delta$.

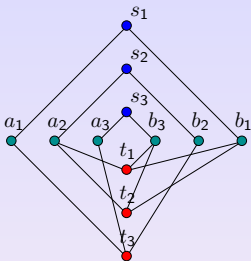
Observation: Property applies to the route length of R from s_i to t_j .

Let $K(M)$ be the length of the shortest program that outputs M and that stops (Kolmogorov Complexity).

There must exist M with $K(M) \geq p^2$.



Consider the program P that simulates, for all (s_i, t_j) , the first routing decisions from s_i to t_j until a_i or b_i is left for ever.



Consider the program P that simulates, for all (s_i, t_j) , the first routing decisions from s_i to t_j until a_i or b_i is left for ever.

- ① P outputs M [because of the Property, as $\beta < 2\delta$]
- ② $|P| \leq 3\delta p \mu$ [because P uses addresses of t_j 's, algorithms $R(x, \cdot, \cdot)$ for all x of the paths from s_i to a_i and b_i .]

Since P outputs M , we must have $|P| \geq K(M)$. Choose M such that $K(M)$ is maximal, $K(M) \geq p^2$. So,

$$3\delta p \mu \geq |P| \geq K(M) \geq p^2 .$$

So, if $\beta < 2\delta$ (i.e., Property holds), then $3\delta\mu \geq p$.

Choose, $\delta = \frac{1}{4}p/\mu$.

Then, $3\delta\mu < p$, and thus we have that $\beta \geq 2\delta = \frac{1}{2}p/\mu$.

Number of nodes in G :

$$n = \delta(2p + p^2) \leq 3\delta p^2 \leq \frac{3}{4}p^3/\mu.$$

And thus $p \geq (\mu n)^{1/3}$.

We have therefore proved that:

$$\beta \geq \frac{1}{2}(\mu n)^{1/3}/\mu = \frac{1}{2}n^{1/3}/\mu^{2/3}$$

QED

THANK YOU!