Slack Spanners

Applications etc.

< □ > < 同 > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > <

Spanners with Slack

Michael Dinitz Joint work with Hubert Chan and Anupam Gupta

Computer Science Department Carnegie Mellon University

Workshop on Flexible Network Design Bertinoro October 6, 2006 Slack Spanners

Applications etc.

Conclusion

Outline

Introduction

- Spanners
- Slack

2 Slack Spanners

- Main Result
- Gracefully Degrading Spanners

3 Applications etc.

- Distance Oracles
- Distance Labelings
- Other slack spanner results



< □ > < 同 > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > <

Spanner Definition

• Main problem: small representation of metric space

Definition

Give a metric (V, d), a *t-spanner* H = (V, E) is a weighted graph such that for all $u, v \in V$, $d(u, v) \le d_H(u, v) \le t \cdot d(u, v)$

- t is the stretch or the distortion
- |E| measures how sparse or small the spanner is. *Really* want |E| = O(n)
- Want to minimize |E| and t, i.e. create a low-stretch sparse spanner

Slack Spanners

Applications etc.

Conclusion

◆□▶ ◆□▶ ◆臣▶ ◆臣▶ 三臣 - のへぐ

Spanner example



Slack Spanners

Applications etc.

Conclusion

Spanner example





◆□▶ ◆□▶ ◆臣▶ ◆臣▶ 三臣 - のへぐ

Slack Spanners

Applications etc.

Conclusion

Spanner example





Slack Spanners

Applications etc.

Conclusion

Spanner example





◆□▶ ◆□▶ ◆臣▶ ◆臣▶ 三臣 - のへぐ

Slack Spanners

Applications etc.

Conclusion

Spanner example



Stretch = $d_H(x, y)/d(x, y) = 3/1 = 3$

▲ロト ▲帰 ト ▲ヨト ▲ヨト - ヨ - の々ぐ

Research on Spanners

• Classic research:

- Awerbuch '85: Inspired study of spanners
- Peleg & Schaffer '89
- Althofer, Das, Dobkin, Joseph, & Soares: Sparse spanners for weighted graphs
- Euclidean spanners

Research on Spanners

- Classic research:
 - Awerbuch '85: Inspired study of spanners
 - Peleg & Schaffer '89
 - Althofer, Das, Dobkin, Joseph, & Soares: Sparse spanners for weighted graphs
 - Euclidean spanners
- New research
 - Baswana et al: Sparse additive spanners
 - Lower bounds for additive and Euclidean spanners

Slack Spanners

Applications etc.

Conclusion

◆□ > ◆□ > ◆臣 > ◆臣 > ○臣 ○ のへ⊙

Simple Algorithm

Theorem (Althofer et al.)

For any integer k, a (2k - 1)-spanner with $O(n^{1+1/k})$ edges can be constructed efficiently

Slack Spanners

Applications etc.

Simple Algorithm

Theorem (Althofer et al.)

For any integer k, a (2k - 1)-spanner with $O(n^{1+1/k})$ edges can be constructed efficiently

Use a Kruskal-like algorithm:

- Initialize H to be the empty graph
- Let $\{u, v\}$ be shortest edge we haven't looked at yet
- If $d_H(u, v) > (2k 1)d(u, v)$, put $\{u, v\}$ in H
- Otherwise discard $\{u, v\}$ and repeat

Slack Spanners

Applications etc.

Conclusion

◆□▶ ◆□▶ ◆□▶ ◆□▶ □□ - のへぐ



Slack Spanners

Applications etc.

Conclusion

◆□▶ ◆□▶ ◆臣▶ ◆臣▶ 三臣 - のへぐ



Slack Spanners

Applications etc.

Conclusion

◆□▶ ◆□▶ ◆□▶ ◆□▶ □□ - のへぐ



Slack Spanners

Applications etc.

Conclusion

◆□▶ ◆□▶ ◆□▶ ◆□▶ □□ - のへぐ



Slack Spanners

Applications etc.

Conclusion

◆□▶ ◆□▶ ◆□▶ ◆□▶ □□ - のへぐ



Slack Spanners

Applications etc.

Conclusion

◆□▶ ◆□▶ ◆□▶ ◆□▶ □□ - のへぐ



Slack Spanners

Applications etc.

Conclusion

◆□▶ ◆□▶ ◆□▶ ◆□▶ □□ - のへぐ



Slack Spanners

Applications etc.

Conclusion

◆□▶ ◆□▶ ◆□▶ ◆□▶ □□ - のへぐ



Slack Spanners

Applications etc.

Conclusion

◆□▶ ◆□▶ ◆□▶ ◆□▶ □□ - のへぐ



Slack Spanners

Applications etc.

Conclusion

Althofer Example (k = 2)



◆□▶ ◆□▶ ◆臣▶ ◆臣▶ 三臣 - のへぐ

Slack Spanners

Applications etc.

Conclusion

Althofer Example (k = 2)



◆□▶ ◆□▶ ◆臣▶ ◆臣▶ 三臣 - のへぐ

Slack Spanners

Applications etc.

Conclusion

Althofer Example (k = 2)



◆□▶ ◆□▶ ◆□▶ ◆□▶ □□ - のへぐ

Slack Spanners

Applications etc.

Conclusion

◆□ > ◆□ > ◆豆 > ◆豆 > ・豆 ・ の Q @ >



Introduction	Slack Spanners	Applications etc.	Conclusion
000000000	00000000000	0000000	
Correctness			

◆□ ▶ ◆□ ▶ ◆三 ▶ ◆三 ▶ ● ○ ○ ○ ○

Need to prove that stretch is at most 2k - 1 and that $|E| = O(n^{1+1/k})$

Introduction	Slack Spanners	Applications etc.	Conclusion
0000000000	00000000000	0000000	
Correctness			

▲□▶ ▲□▶ ▲□▶ ▲□▶ ▲□ ▶ ④ ●

Need to prove that stretch is at most 2k - 1 and that $|E| = O(n^{1+1/k})$

• Stretch: by construction.

Introduction	Slack Spanners	Applications etc.	Conclusion
000000000	00000000000	0000000	
Correctnoss			

Need to prove that stretch is at most 2k - 1 and that $|E| = O(n^{1+1/k})$

- Stretch: by construction.
- Sparse:
 - Suppose edge $e = \{u, v\}$ creates a cycle C
 - Every other edge on C shorter than e
 - Without e, the distance between u and v was more than (2k-1)length(e)
 - So at least 2k 1 other edges on C
 - Girth at least 2k 1
- Well-known graph theory theorem: Girth of 2k 1 implies $|E| = O(n^{1+1/k})$

・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・
・

< □ > < 同 > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > <

Althofer is optimal

- Erdos girth conjecture: For every k, there is a graph with $\Omega(n^{1+1/k})$ edges and girth 2k-1
- Implies that the Althofer spanner is tight (well, at least for subgraph spanners...)
- So if we want O(n) edges, we need stretch of $\Omega(\log n)!$

▲ロト ▲帰 ト ▲ヨト ▲ヨト - ヨ - の々ぐ

What Now?

- In practice:
 - log n stretch is too large
 - On't need low stretch for all pairs
- Use 2 to fix 1
- How well can we do? Ignore 5% of pairs and get $O(\sqrt{\log n})$ stretch on the rest? $O(\log \log n)$? O(1)?

▲ロト ▲帰 ト ▲ヨト ▲ヨト - ヨ - の々ぐ

What Now?

- In practice:
 - log n stretch is too large
 - On't need low stretch for all pairs
- Use 2 to fix 1
- How well can we do? Ignore 5% of pairs and get $O(\sqrt{\log n})$ stretch on the rest? $O(\log \log n)$? O(1)?
- Ignoring a constant fraction of pairs lets us prove constant distortion on the rest!

0000000 000 000000000 000000	-Neighborh	ods		
Introduction Slack Spanners Applications etc. Conclu	ntroduction	Slack Spanners 00000000000	Applications etc.	Conclusio

• Basic idea: ignoring small distances helps with large distances

Definition

Given ϵ , for any point $v \in V$, the ϵ -neighborhood $N_{\epsilon}(v)$ consists of the closest ϵn points to v

- $R(v,\epsilon) = \min\{r : |B(v,r)| \ge \epsilon n\}$
- v is ϵ -far from u if $d(u, v) \ge R(u, \epsilon)$



Introduction	Slack Spanners	Applications etc.	Conclusion
000000000	00000000000	0000000	
Slack definition	าร		

• Original work on slack was on slack embeddings into ℓ_p spaces

◆□▶ ◆□▶ ◆臣▶ ◆臣▶ 三臣 - のへぐ

• KSW '04, ABCDGKNS '05, ABN '06

000000000			
Introduction	Slack Spanners	Applications etc.	Conclusion

• Original work on slack was on slack embeddings into ℓ_p spaces

- KSW '04, ABCDGKNS '05, ABN '06
- Basic definition:

uennilions

Definition (Slack Spanner)

Given a metric (V, d), a *t*-spanner H = (V, E) has ϵ -slack if $d(u, v) \le d_H(u, v) \le t \cdot d(u, v)$ for all but ϵn^2 pairs $\{u, v\}$

000000000			
Introduction	Slack Spanners	Applications etc.	Conclusion

- Original work on slack was on slack embeddings into ℓ_p spaces
- KSW '04, ABCDGKNS '05, ABN '06
- Basic definition:

Definition (Slack Spanner)

Given a metric (V, d), a *t*-spanner H = (V, E) has ϵ -slack if $d(u, v) \leq d_H(u, v) \leq t \cdot d(u, v)$ for all but ϵn^2 pairs $\{u, v\}$

• More restrictive definition:

Definition (Uniform Slack)

Given a metric (V, d), a *t*-spanner H = (V, E) has ϵ -uniform slack if for all $u, v \in V$ such that v is ϵ -far from u, $d(u, v) \leq d_H(u, v) \leq t \cdot d(u, v)$

Slack Spanners

Applications etc.

Conclusion

◆□▶ ◆□▶ ◆三▶ ◆三▶ →三 ● ● ●

Conversion Theorem

Theorem

Suppose there exists an algorithm to construct a t(n)-stretch spanner with h(n) edges for any metric. Then we can find an ϵ -slack spanner with $5 + 6t(\frac{1}{\epsilon})$ stretch and $n + h(\frac{1}{\epsilon})$ edges.
Slack Spanners

Applications etc.

・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・

Conversion Theorem

Theorem

Suppose there exists an algorithm to construct a t(n)-stretch spanner with h(n) edges for any metric. Then we can find an ϵ -slack spanner with $5 + 6t(\frac{1}{\epsilon})$ stretch and $n + h(\frac{1}{\epsilon})$ edges.

We can apply this to the Althofer spanner:

Corollary

For any metric, for any $0 < \epsilon < 1$, for any integer k > 0, there exists a (12k - 1)-spanner with ϵ -slack of size $n + O((\frac{1}{\epsilon})^{1+1/k})$

Introduction 000000000	Slack Spanners 00000000000	Applications etc.	Conclusion
Density Net			

- Intuition: Small set of points that approximates the metric
- Recall that $R(u, \epsilon) = \min\{r : |B(u, r)| \ge \epsilon n\}$

Definition

An ϵ -density net is a subset N of V such that

1 For all $x \in V$, there is some $y \in N$ s.t. $d(x, y) \leq 2R(x, \epsilon)$

$$|N| \leq \frac{1}{\epsilon}$$



Slack Spanners

Applications etc.

▲ロト ▲帰 ト ▲ヨト ▲ヨト - ヨ - の々ぐ

Constructing a Density Net

- **1** Put points in a list *L* by non-decreasing value of $R(\cdot, \epsilon)$
- 2 Initialize N := 0.
- While *L* is non-empty:
 - Remove first point v from L
 - **2** If there exists $u \in N$ s.t. $N_{\epsilon}(v)$ and $N_{\epsilon}(u)$ intersect, then discard v; otherwise add v to N

Slack Spanners

Applications etc.

Conclusion



Slack Spanners

Applications etc.

Conclusion



Slack Spanners

Applications etc.

Conclusion



Slack Spanners

Applications etc.

Conclusion



Slack Spanners

Applications etc.

Conclusion



Slack Spanners

Applications etc.

Conclusion



Slack Spanners

Applications etc.

Conclusion



Slack Spanners

Applications etc.

Conclusion



Slack Spanners

Applications etc.

Conclusion



Slack Spanners

Applications etc.

Conclusion

◆□▶ ◆□▶ ◆三▶ ◆三▶ 三三 のへぐ

Density Net Proof

- Need to prove:
 - If all x ∈ V there is some y ∈ N such that d(x, y) ≤ 2R(x, ε)
 |N| ≤ 1/ε

Slack Spanners

Applications etc.

▲ロト ▲帰 ト ▲ヨト ▲ヨト - ヨ - の々ぐ

Density Net Proof

- Need to prove:
 - For all $x \in V$ there is some $y \in N$ such that $d(x, y) \leq 2R(x, \epsilon)$
 - $|N| \leq \frac{1}{\epsilon}$
- Net property:
 - If $x \in N$ then we're good.
 - Else there is $y \in N$ before x s.t. $N_{\epsilon}(x)$ and $N_{\epsilon}(y)$ intersect. So $d(x, y) \leq R(x, \epsilon) + R(y, \epsilon) \leq 2R(x, \epsilon)$

Slack Spanners

Applications etc.

Conclusion

Density Net Proof

- Need to prove:
 - For all $x \in V$ there is some $y \in N$ such that $d(x, y) \leq 2R(x, \epsilon)$
 - $|N| \leq \frac{1}{\epsilon}$
- Net property:
 - If $x \in N$ then we're good.
 - Else there is $y \in N$ before x s.t. $N_{\epsilon}(x)$ and $N_{\epsilon}(y)$ intersect. So $d(x, y) \leq R(x, \epsilon) + R(y, \epsilon) \leq 2R(x, \epsilon)$
- Size property:
 - For different $u,v\in {\sf N},\ {\sf N}_\epsilon(u)$ and ${\sf N}_\epsilon(v)$ are disjoint
 - Each $|N_{\epsilon}(u)| \ge \epsilon n$, so $|N| \le \frac{1}{\epsilon}$

Slack Spanners

Applications etc.

Conclusion

・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・

Conversion Algorithm

- Given metric (V, d),construct ϵ -density net N. Note that $|N| \leq \frac{1}{\epsilon}$
- **2** Construct $t(\frac{1}{\epsilon})$ -spanner with $h(\frac{1}{\epsilon})$ edges on N
- **③** For all $u \in V \setminus N$, add an edge to the nearest point in N

Applications etc.

・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・

Conversion Algorithm

- Given metric (V, d),construct ϵ -density net N. Note that $|N| \leq \frac{1}{\epsilon}$
- **2** Construct $t(\frac{1}{\epsilon})$ -spanner with $h(\frac{1}{\epsilon})$ edges on N
- **③** For all $u \in V \setminus N$, add an edge to the nearest point in N

Obviously sparse: $O(n + h(\frac{1}{\epsilon}))$ edges

Slack Spanners

Applications etc.

Conclusion



Slack Spanners 000000000000



Slack Spanners 000000000000



Slack Spanners

Applications etc.

Conclusion



000000000	0000000000000	0000000	Conclusion
Low Stretch			



- Let $u, v \in V$ s.t. $v \notin N_{\epsilon}(u)$
- Let u', v' be the closest points in N to u and v respectively

◆□ > ◆□ > ◆臣 > ◆臣 > ○臣 ○ のへ⊙

Introduction	Slack Spanners	Applications etc.	Conclusion
Low Stretch			



- Let $u, v \in V$ s.t. $v \notin N_{\epsilon}(u)$
- Let u', v' be the closest points in N to u and v respectively

•
$$d(u, u') \leq 2R(u, \epsilon) \leq 2d(u, v)$$

•
$$d(v, v') \le d(v, u') \le d(v, u) + d(u, u') \le 3d(u, v)$$

• $d(u', v') \le d(u', u) + d(u, v) + d(v, v') \le 6d(u, v)$

ntroduction 000000000	Slack Spanners	Applications etc.	Conclusion
low Stretch			



• Let
$$u, v \in V$$
 s.t. $v \notin N_{\epsilon}(u)$

• Let u', v' be the closest points in N to u and v respectively

•
$$d(u, u') \leq 2R(u, \epsilon) \leq 2d(u, v)$$

•
$$d(v, v') \le d(v, u') \le d(v, u) + d(u, u') \le 3d(u, v)$$

- $d(u', v') \le d(u', u) + d(u, v) + d(v, v') \le 6d(u, v)$
- By spanner on N, $d_H(u',v') \le t(\frac{1}{\epsilon})d(u',v') \le 6t(\frac{1}{\epsilon})d(u,v)$
- So

 $d_H(u,v) \leq d(u,u') + d_H(u',v') + d(v',v) \leq (5+6t(\frac{1}{\epsilon}))d(u,v)$

Slack Spanners

Applications etc.

◆□▶ ◆□▶ ◆三▶ ◆三▶ 三三 のへぐ

Gracefully Degrading

• Previous result of the form "You give me an $\epsilon,$ I give you an $\epsilon\text{-slack spanner"}$

Applications etc.

Gracefully Degrading

- Previous result of the form "You give me an $\epsilon,$ I give you an $\epsilon\text{-slack spanner"}$
- Could ask for something stronger:
- "I give you a spanner that works simultaneously for all ϵ "
- Called a gracefully degrading spanner

Applications etc.

Gracefully Degrading

- Previous result of the form "You give me an $\epsilon,$ I give you an $\epsilon\text{-slack spanner"}$
- Could ask for something stronger:
- "I give you a spanner that works simultaneously for all ϵ "
- Called a gracefully degrading spanner

Theorem

For any metric, there is a spanner H with O(n) edges s.t. for any $0 < \epsilon < 1$, H is a $O(\log \frac{1}{\epsilon})$ -spanner with ϵ -slack.

Introduction Slack Spanners Applications etc. Conclus

◆□▶ ◆□▶ ◆三▶ ◆三▶ 三三 のへぐ

Gracefully Degrading Construction

Intuition: layers of slack spanners for various value of ϵ .

Slack Spanners

Applications etc.

Conclusion

▲ロト ▲帰 ト ▲ヨト ▲ヨト - ヨ - の々ぐ

Gracefully Degrading Construction

Intuition: layers of slack spanners for various value of ϵ . Actually much simpler – only 2 layers needed:

Applications etc.

・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・

Gracefully Degrading Construction

Intuition: layers of slack spanners for various value of $\epsilon.$ Actually much simpler – only 2 layers needed:

- Let $\epsilon_0 = n^{-1/2}$, and construct a ϵ_0 -density net N of V
- 2 Connect every vertex to the closest point in N
- Solution Create a 1-spanner H_0 (e.g. a clique) on N (uses O(n) edges)
- Use Althofer to make a log *n*-spanner H' on V
- Set H to be the union of H_0 and H', together with edges that connect each point in V to its closest point in N

Applications etc.

・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・

Gracefully Degrading Construction

Intuition: layers of slack spanners for various value of $\epsilon.$ Actually much simpler – only 2 layers needed:

• Let $\epsilon_0 = n^{-1/2}$, and construct a ϵ_0 -density net N of V

- 2 Connect every vertex to the closest point in N
- Solution Create a 1-spanner H_0 (e.g. a clique) on N (uses O(n) edges)
- Use Althofer to make a log *n*-spanner H' on V
- Set H to be the union of H_0 and H', together with edges that connect each point in V to its closest point in N

Each step creates O(n) edges, so there are only O(n) edges total

Slack Spanners

Applications etc.

Conclusion



Two cases for the stretch:



Applications etc.

Stretch

Two cases for the stretch:

- $\epsilon < \epsilon_0$: Use H' to get stretch $O(\log n) = O(\log n^{1/2}) = O(\log \frac{1}{\epsilon_0}) = O(\log \frac{1}{\epsilon})$ between every pair of points
- **2** $\epsilon \geq \epsilon_0$: Use H_0 . Same analysis as for slack spanner, except that stretch in the net is 1, so total stretch is at most 11.

Applications etc.

< □ > < 同 > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > <

Average Stretch

Gracefully degrading spanner automatically gives us a normal $O(\log n)$ -spanner that has O(1) average distortion!

$$\frac{1}{\binom{n}{2}} \sum_{\{x,y\} \in \binom{V}{2}} \frac{d_H(x,y)}{d(x,y)} = \frac{2}{n} \sum_{x \in V} \frac{1}{n-1} \sum_{y \neq x} \frac{d_H(x,y)}{d(x,y)}$$
$$\leq \frac{2}{n} \sum_{x \in V} \left(\frac{1}{n^{1/2}} O(\log n) + (1 - \frac{1}{n^{1/2}}) \cdot 11 \right)$$
$$= O(1)$$

・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・

Distance Oracles Overview

- Intuition: *all*-pairs shortest path is rarely necessary.
- Distance oracle: data structure/algorithm for computing approximate distances in a metric
- Want to minimize stretch, space, and query time
- First studied by Thorup and Zwick ('01): for any integer k ≥ 1, oracle with stretch 2k − 1, space O(kn^{1+1/k}), query time O(k)
- Implicitly created a spanner, clever way of doing queries based on special structure of spanner

Slack Spanners

Applications etc. $0 \bullet 0 \circ 0 \circ 0$

Conclusion

▲ロト ▲帰 ト ▲ヨト ▲ヨト - ヨ - の々ぐ

Oracles with Slack

Can create slack oracles using slack embeddings:

Theorem (ABN '06)

For any integer $k \ge 1$, there is an oracle with ϵ -slack, stretch 6k - 1, O(k) query time, and $O(n \log n \log \frac{1}{\epsilon} + k \log n(\frac{1}{\epsilon} \log \frac{1}{\epsilon})^{1+1/k})$ space
Slack Spanners

Applications etc.

< □ > < 同 > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > <

Oracles with Slack

Can create slack oracles using slack embeddings:

Theorem (ABN '06)

For any integer $k \ge 1$, there is an oracle with ϵ -slack, stretch 6k - 1, O(k) query time, and $O(n \log n \log \frac{1}{\epsilon} + k \log n(\frac{1}{\epsilon} \log \frac{1}{\epsilon})^{1+1/k})$ space

But slack spanners are better:

Theorem

For every integer $k \ge 1$, there is an oracle with ϵ -slack, stretch 12k - 1, O(k) query times, and $O(n + k(\frac{1}{\epsilon})^{1+1/k})$ space

Same method as used for slack spanners

Introduction 000000000 Slack Spanners

Applications etc.

Conclusion

Gracefully Degrading Oracles

Can do the same thing for gracefully degrading oracles.

Theorem

For any integer k with $1 \le k \le O(\log n)$, there is a distance oracle with worst cast stretch of 2k - 1 and O(k) query time that uses $O(kn^{1+1/k})$ space such that the average distortion is O(1)

Improvement over ABN '06 if $k = o(\log n)$

Introduction 000000000	Slack Spanners 00000000000	Applications etc.	Conclusion
Distance Labelir	ng Overview		

- How can we assign each point a short label so that approximate distances can be computed quickly by just comparing labels?
 - Used in various networking applications
 - \bullet Embedding into ℓ_p very natural approach: size of a label is the dimension

< □ > < 同 > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > <

Distance Labeling Overview

- How can we assign each point a short label so that approximate distances can be computed quickly by just comparing labels?
- Used in various networking applications
- \bullet Embedding into ℓ_p very natural approach: size of a label is the dimension
- One of the original motivations for definition of slack in KSW '04:
 - In general can't have dimension less that $\Omega(\log n)!$
 - Seems to work better in practice

・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・

Distance Labeling Overview

- How can we assign each point a short label so that approximate distances can be computed quickly by just comparing labels?
- Used in various networking applications
- \bullet Embedding into ℓ_p very natural approach: size of a label is the dimension
- One of the original motivations for definition of slack in KSW '04:
 - In general can't have dimension less that $\Omega(\log n)!$
 - Seems to work better in practice
- Can we do better with spanners than with embeddings?

Slack Spanners

Applications etc.

Conclusion

▲ロト ▲帰 ト ▲ヨト ▲ヨト - ヨ - の々ぐ

Slack Labelings

Using embeddings:

Theorem (ABCDGKNS '05)

Any embedding $\varphi : V \to \ell_p$ with ϵ -(uniform) slack must have dimension that depends on log n

Slack Spanners

Applications etc.

Slack Labelings

Using embeddings:

Theorem (ABCDGKNS '05)

Any embedding $\varphi : V \to \ell_p$ with ϵ -(uniform) slack must have dimension that depends on log n

We get rid of all dependence on n by not using an embedding!

Theorem

For any integer k with $1 \le k \le \log \frac{1}{\epsilon}$, we can assign each point a label that uses $O((\frac{1}{\epsilon})^{1/k} \log^{1-1/k} \frac{1}{\epsilon})$ space so that if v is ϵ -far from u, their distance can be computed up to stretch 12k - 1 in O(k) time

Applications etc. ○○○○○●○

◆□▶ ◆□▶ ◆三▶ ◆三▶ 三三 のへぐ

Subgraph Spanner

- What if our input isn't a metric but a graph?
- Want our spanner to be a subgraph

・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・
 ・

Subgraph Spanner

- What if our input isn't a metric but a graph?
- Want our spanner to be a subgraph

Theorem

Given a weighted graph G = (V, E), for any integer k > 0 and any $0 < \epsilon < 1$, there exists a (12k - 1)-spanner with ϵ -slack and $O(n + \sqrt{n}(\frac{1}{\epsilon})^{1+1/k})$ edges.

 Uses pairwise distance preservers of Coppersmith and Elkin to make a subgraph that emulates the spanner on the net

Low Weight

Could also try to minimize the weight of the spanner.

Theorem

For any metric, there is an ϵ -slack spanner with $O(\log \frac{1}{\epsilon})$ stretch, $O(n + \frac{1}{\epsilon})$ edges, and weight $O(\log^2(\frac{1}{\epsilon})) \times wt(MST)$

Main idea: use LASTs (Light Approximate Shortest-path Trees) of Khuller, Raghavachari, and Young

▲ロト ▲帰 ト ▲ヨト ▲ヨト - ヨ - の々ぐ

Review

- Ignoring a constant fraction of distances gives us lots of power (e.g. constant stretch, linear size spanners)!
- Using
 e-density nets to represent metrics gives us good slack and gracefully degrading spanners, distance oracles, and distance labelings

Introduction 000000000 Slack Spanners

Applications etc.

Conclusion

◆□▶ ◆□▶ ◆臣▶ ◆臣▶ 三臣 - のへぐ

Future Research

- Slack version of (your favorite problem here)
- Additive spanners????

Thank You!

