Information Processing Letters 129 (2018) 57–60

Contents lists available at ScienceDirect

Information Processing Letters

www.elsevier.com/locate/ipl

Deterministic improved round-trip spanners

Chun Jiang Zhu [a,b,*], Kam-Yiu Lam [b]

[a] *Research Department, Wisers Information Limited, Hong Kong, China*
[b] *Department of Computer Science, City University of Hong Kong, Hong Kong, China*

ARTICLE INFO

Article history:
Received 24 December 2016
Received in revised form 11 August 2017
Accepted 21 September 2017
Available online 25 September 2017
Communicated by Tsan-sheng Hsu

Keywords:
Graph algorithms
Graph spanners
Round-trip spanners
Deterministic algorithms
Shortest distances

ABSTRACT

In this paper, we study the deterministic construction of round-trip spanners for weighted directed graphs. We propose a deterministic algorithm which constructs, for any n-vertex graph $G(V, E)$, a round-trip spanner $H(V, E' \subseteq E)$ of stretch $2k+\epsilon$ and size $O((k/\epsilon)\cdot n^{1+1/k}\log(nw))$, where w is the maximum edge weight of G. Notably, this is the first deterministic construction of round-trip spanners and its stretch-size trade-off even improves the previous state-of-the-art randomized algorithm by Roditty et al. More specifically, the size is asymptotically reduced by a factor of k while the stretch factor remains the same. The result is the first clear improvement on round-trip spanners after about ten years and re-raises the open question that how best we can hope for the stretch-size trade-off of round-trip spanners in digraphs.

1. Introduction

Sparse spanners of undirected graphs, since they were introduced in 1989 [12], have received considerable research in the theory community. A sparse spanner of an undirected graph approximates distances in the graph such that the distance between any pair of vertices in the spanner is no larger than k times of that in the graph, and k is called the *stretch* factor. It is well known that for any n-vertex graph, there exists a spanner of stretch $2k-1$ and size (the number of edges) $O(n^{1+1/k})$. This is optimal if we believe the Erdos's Conjecture [10]. See [3,15] for the concrete constructions. A lot of research efforts were then devoted to the *additive spanners*, where the distance between any vertex pair is no larger by an additive term β instead of a multiplicative factor. Here the spanner is called a $+\beta$-spanner. See [2,4,17,5] for the different constructions of +2-, +6-, +4-spanners of size $O(n^{3/2})$, $O(n^{4/3})$, $O(n^{7/5})$ respectively. Abbound et al. [1] showed that the additive term must be polynomial instead of a constant term using the size bound $o(n^{4/3})$. At the same time, mixed spanners, such as (α, β)-spanners in which the distance in the spanner is at most $\alpha \cdot d + \beta$ with d being the distance in the graph, receive a lot of research as well [9,16,13,4].

The definition of spanners can be extended to digraphs naturally but it becomes trivial to study sparse spanners. Consider an n-vertex bipartite digraph $G(LHS, RHS)$ where LHS and RHS both have $n/2$ vertices and each vertex in LHS contains edges to all vertices in RHS. A spanner by definition requires to include all the edges, even only to preserve the connectivity, and thus has size $\Omega(n^2)$. Therefore, instead of studying one-way distance in digraphs, Cowen et al. [7,8] considered *round-trip distance* between two vertices. The round-trip distance between vertex u and v is the sum of the one-way distance from u to v and the one-way distance from v to u. In the papers, they studied the round-trip routing schemes and their results imply *round-trip spanners*, the data structure approximating round-trip distances instead of one-way distances in digraphs, of stretch 2^k-1 and size $\tilde{O}(n^{1+1/k})$. The round-trip spanner is close to being optimal when $k=2$

* Corresponding author.
E-mail addresses: chunjizhu2-c@my.cityu.edu.hk (C.J. Zhu), cskylam@cityu.edu.hk (K.-Y. Lam).

https://doi.org/10.1016/j.ipl.2017.09.008
0020-0190/© 2017 Elsevier B.V. All rights reserved.

but when k is a large integer, the stretch becomes extremely large.

Later, Roditty et al. [14] proposed the state-of-the-art randomized algorithm for constructing round-trip spanners of stretch $2k+\epsilon$ and size $\min\{O((k^2/\epsilon)\cdot n^{1+1/k}\log(nw)), O((k/\epsilon)^2 n^{1+1/k}\log^{2-1/k} n)\}$, where w is the maximum edge weight in the input graph G. The results significantly improve the stretch 2^k-1 of Cowen et al. [7,8] to $2k+\epsilon$, much closer to the optimal stretch $2k-1$ of spanners of undirected graphs. In their paper, they did not provide any deterministic algorithm for round-trip spanners, and it is still unclear whether the stretch-size trade-off of round-trip spanners can be further improved, e.g. reducing the stretch factor to $2k-1$ to be consistent with the trade-off of spanners of undirected graphs. Unfortunately, there have been as far as we know no clear improvement on the round-trip spanners by Roditty et al. [14] published for about ten years. Zhu and Lam [18] studied the *source-wise round-trip spanners* where only the round-trip distances between a given source vertex set and all vertices are approximated. Pachocki et al. [11] designed faster constructions of round-trip spanners of stretch $O(k\log n)$ and size $O(n^{1+1/k}\log^2 n)$ in $\tilde{O}(mn^{1/k})$ time with m being the number of edges of the original graph. They also constructed additive round-trip spanners of additive term $O(n^{\alpha})$ and size $\tilde{O}(n^{2-\alpha})$. However, they did not clearly improve the stretch-size trade-off of round-trip spanners by Roditty et al. [14].

In this paper, we study the derandomization of the state-of-the-art randomized algorithm and propose a deterministic algorithm which notably exhibits a better stretch-size trade-off than the previous state-of-the-art. Our main result is:

Theorem 1. *For any n-vertex digraph with maximum edge weight w, there exists a deterministic construction of a round-trip spanner of stretch $2k+\epsilon$ and size $O((k/\epsilon)\cdot n^{1+1/k}\log(nw))$.*

The size is reduced by a factor of k compared to the previous state-of-the-art asymptotically while the stretch factor remains the same. Our results are obtained by combining ideas from both Cohen [6] and Roditty et al. [14].

In Section 2, we define the terms and notations used in the paper. We provide the details of the deterministic algorithm for round-trip spanners and prove its stretch and size bound in Section 3. Finally, we conclude the paper with a short discussion on future work in Section 4.

2. Notations and definitions

We consider weighted digraphs in this paper. In a graph $G(V,E,W)$ (W assigns weights to edges in E and can be omitted for a succinct presentation), the subgraph of G induced by vertices in $U\subseteq V$ is denoted as $G(U)$. A *path* P from vertex u to vertex v in G is a sequence of edges in G traversing from u to v. The *distance* of P is the sum of edge weights of all the edges on P. The *one-way shortest path* from u to v in G is the path with the shortest distance amongst all the paths from u to v in G, and its distance is called the *one-way distance* from u to v in G.

A *round-trip shortest path* between u and v in a graph H consists of a one-way shortest path from u to v in H and a one-way shortest path from v to u in H, and its distance is called the *round-trip distance* $d_{RT}(u,v,H)$. We do not have to assume unique shortest paths (and thus unique round-trip shortest paths) between any vertices and there can exist multiple shortest paths between them. A *round-trip spanner* of stretch k of a digraph $G(V,E)$ is a subgraph $H(V,E')$ such that for vertices $u,v\in V$, $d_{RT}(u,v,H)\le 2k\cdot d_{RT}(u,v,G)$.

A *(round-trip) ball* $Ball_U(u,R)$ of G is a set of vertices whose round-trip distance from u in $G(U)$ is less than or equal to the *radius* R. For a ball $B=Ball_U(u,R)$, let its *round-trip tree* $RT\text{-}Tree(B)$ be the union of the shortest path tree in $G(U)$ from u to every vertex in B, and the shortest path tree in $G(U)$ from every vertex in B to u. We use $|B|$ to denote the number of vertices in B. Because of the fact that each vertex on a shortest path from u to $v\in B$ or from v to u must be in B as well, the number of edges in $RT\text{-}Tree(B)$ is at most $2(|B|-1)$.

3. An improved round-trip spanner algorithm

The state-of-the-art randomized algorithm for round-trip spanners has not yet been derandomized, although its base work [6] includes some deterministic results. We apply the derandomization technique of [6] in the round-trip spanners [14], and prove the size and stretch bounds independently.

The core of the derandomization is to derandomize the construction of (k,R)-*cover*, which we use the same definition as [14].

Definition 1. In a digraph $G(V,E)$, a collection C of round-trip balls is a (k,R)-*cover* of G if and only if each ball in C has radius at most kR, and for every $u,v\in V$ with their round-trip distance $d_{RT}(u,v,G)\le R$, there is a round-trip ball $B\in C$ such that $u,v\in B$.

We define (and construct) the *cluster* ball $cluster(u,U)$ of a vertex u and its *core* ball $core(u,U)$, with respect to an arbitrary vertex set U as follows by iterations. Initially, $cluster(u,U)$ is set to u itself, which can be considered as $Ball_U(u,0)=Ball_U(u,0\cdot R)$. In each iteration, we first assign $cluster(u,U)$ as the core ball $core(u,U)$, and then increase the radius of $cluster(u,U)$ by R, e.g., from $Ball_U(u,(i-1)\cdot R)$ to $Ball_U(u,i\cdot R)$. This process continues until the size of the cluster is less than or equal to $n^{1/k}$ times that of its core ball in terms of the number of vertices. More formally, the algorithm is summarized in Algorithm 1.

Theorem 2. *The cluster ball $cluster(u,U)$ output by Algorithm 1 has radius at most $k\cdot R$.*

Proof. We prove that the construction stops with that $i\le k+1$. Then the cluster ball $cluster(u,U)$, which is $Ball_U(u,(i-1)\cdot R)$ at the termination has radius at most $k\cdot R$.

Note that before the last iteration, the number of vertices in $G(cluster(v,U))$ is larger than that of $G(core(v,U))$