Brief Announcement: Faster 1-Shot Broadcast in Known Graphs

Marek Klonowski[1], Dariusz Kowalski[2], and Dominik Pająk[⋆1]

[1] Department of Fundamentals of Computer Science,
Wrocław University of Science and Technology, Poland
[2] School of Computer and Cyber Sciences, Augusta University, USA

Abstract. We present a new deterministic algorithm for broadcast in radio networks in a model where each station can transmit at most once and the topology of the network is known to the algorithm. Our algorithm works in time $D+O(\sqrt{n\log n})$ and the best previous algorithm for this problem has complexity $D+O(\sqrt{n}\log n)$. We also present a new technique to transfer algorithms satisfying a certain monotonicity assumption working only for small diameter networks into an algorithm working in any graphs.

1 Introduction

Model We have an undirected graph $G=(V,E)$ with a distinguished node $r\in V$. Node r initially holds a message (we assume that the contents of the message can be delivered in a single transmission). Time is divided into discrete steps (or rounds). In each step each station can be in one of three states: it can transmit some message, listen to the channel or stay idle. When exactly one neighbor of some listening station is transmitting in a given round, then the transmitted message is received by this node. We say that that a *collision* occurs, if two neighbors of some station are transmitting in the same round. We assume that the topology of the network is **known** to the algorithm.

Broadcast problem The objective in the 1-Shot problem is to deliver the message from r to all the nodes of the network with an additional restriction that each station can transmit in at most one round.

Related work An algorithm for 1-Shot presented in Gasieniec *et al.* [1] has complexity $D+O(\sqrt{n}\log n)$, which given an $\Omega(D+\sqrt{n})$ [1] lower bound leaves a gap of multiplicative factor of $\log n$ for $D<\sqrt{n}$.

Our results We present a deterministic $O(D+\sqrt{n\log n})$ algorithm for 1-Shot.

Notation We denote for any $v\in V$ by d_v the length of a shortest paths between r and v. We denote by D, the diameter of the graph and by $D_r=\max_{v\in V} d_v$ the eccentricity of node r. Note that $D_r\le D\le 2\cdot D_r$.

Definition 1. *A broadcast algorithm is called* monotone *if two conditions are satisfied:*
1. for any two nodes such that u receives the message before v we have $d_u\le d_v$,
2. in every step, the set of broadcasting nodes B have identical distance to r.

⋆ Supported by Polish National Science Center grant UMO-2019/33/B/ST6/02988

2 1-Shot algorithm

The problem can be solved for bipartite graphs in optimal time $O(\sqrt{n})$ using the algorithm presented in [1]. We reuse that algorithm to get a monotone algorithm for graphs with small diameter. For each $i=0,...,D_r$, we denote $n_i=|\{u\in V:d_u=i\}|$ and we will say that the nodes $\{u\in V:d_u=i\}$ constitute the i-th layer of the graph.

Run the algorithm for bipartite graphs from [1, Theorem 3.4] in sequence between each pair of layers (first between layers 0 and 1, then between layers 1 and 2 etc).

Algorithm 1: SmallDiam

Lemma 1. *Algorithm SmallDiam solves the 1-Shot problem in time* $O(\sqrt{nD})$.

(All omitted proofs are provided in the Appendix.) SmallDiam is monotone and by Lemma 1 it solves 1-Shot in time $O(\sqrt{n\log n})$ for graphs with diameter $O(\log n)$. SmallDiam can be used to derive a time-efficient algorithm for 1-Shot problem for graph with large diameter.

In order to achieve this, we use a standard definition of ranking, based on the *Strahler Number* introduced in hydro-geology [3], used in a number of papers in the context of radio communication in known topology networks (e.g., [2]).

We use the procedure Gathering-Spanning-Tree from [2], which defines the explicit construction of spanning tree T and the ranking with properties (see [2, Lemma 2.4]):

(P0) $D_r=D_r^{(T)}$, where $D_r^{(T)}$ is the eccentricity of node r, when using only edges of tree T,
(P1) $\mathsf{rank}(r)\leq\log_2 n+1$,
(P2) for any two nodes v,w with $\mathsf{rank}(v)=\mathsf{rank}(w)$ and $d_v=d_w$, parent of w in tree T is not a neighbor of v in graph G.

For any node $v\in V\setminus\{r\}$ we define by $\mathsf{parent}(v)$, the parent of v in tree T.

Graph of paths The ranking can be seen as partition of the tree into subpaths of nodes with the same ranking. In our construction we treat each subpath (of tree T) with the same ranking as a super-node in a graph $G'=(V',E')$.

Definition 2 (Paths). *Given graph $G=(V,E)$ and the ranked spanning tree T, we first find heads $h\in V$ of each path with the property that $\mathsf{rank}(\mathsf{parent}(h))>\mathsf{rank}(h)$. For each such node we construct a path of maximum length in tree T (by the definition of the ranking there is always at most one child with the same ranking). This creates a partition of the set of vertices V into a family of disjoint subsets (call them $P_1,P_2,...$. For any such path P, directly from the definition we obtain the following properties:*

1. *Any two nodes $u,v\in P$ satisfy $\mathsf{rank}(u)=\mathsf{rank}(v)$.*
2. *If $P=(p_1,p_2,...)$ we have $d_{p_i}=d_{p_{i+1}}-1$ (i.e., the distance to r is always increasing when walking along the path).*

Definition 3 (Graph of paths). *Given graph $G=(V,E)$ and the ranked spanning tree T and the partition into paths $P_1,P_2,...$ we denote the ranking of all the nodes in P as $\mathsf{rank}(P)$ and its node that is closest to r by $\mathsf{head}(P)=\operatorname{argmin}_{v\in P}d_v$. The vertex set of the graph G' will be the set of paths $\{P_1,P_2,...\}$. We define the edges E' of graph G' as follows. There is an edge between P and P' if for some $u\in P'$, $(u,\mathsf{head}(P))\in E$. Note that due to property $(P1)$, the diameter of G' is $O(\log n)$.*

If we run algorithm SmallDiam on graph G', each vertex $v'\in V'$ would transmit in some step $a(v')$. For any node $v\in V$, we denote by $\mathsf{super}(v)$ a node from G' that contains v and we assign for each $v\in V$, $a(v)=a(\mathsf{super}(v))$.

For any $v \in V$ let function $b(v)$ equal to 1 if node v is the last node (*i.e.,* furthest from r) of its path (as defined in Defintion 2) and 0 otherwise. We are ready to define our algorithm for 1-Shot in any graph.

Transmit in step $t(v) = d_v + 3 \cdot (2 \cdot a(v) + b(v))$

Algorithm 2: OneShot

Lemma 2. *If in algorithm OneShot, for some nodes $u, u' \in V$, neighbors of node v (in graph G), we have $t(u) = t(u')$, then:*

$$1.\ d_u = d_{u'}, \quad 2.\ a(u) = a(u'), \quad 3.\ b(u) = b(u')$$

Theorem 1. *Algorithm OneShot solves 1-Shot problem in time $D_r + O(\sqrt{n \log n})$.*

Proof. Case 1: $\mathsf{rank}(v) = \mathsf{rank}(\mathsf{parent}(v))$. In this case we will prove that if $\mathsf{parent}(v)$ has the message in step $t(\mathsf{parent}(v))$, then the message will be delivered to v in this step. We need to show that in this step no other neighbor of v transmits a message.

Assume for contradiction that for some u, neighbor of v we have $t(u) = t(\mathsf{parent}(v))$ (and $u \neq \mathsf{parent}(v)$). By Lemma 2 we have $d_u = d_{\mathsf{parent}(v)}$. By the monotonicity of SmallDiam we also have that $\mathsf{rank}(u) = \mathsf{rank}(\mathsf{parent}(v))$. Since $\mathsf{parent}(v)$ is not the last node on its path, then $b(u) = 2$ hence also by Lemma 2 $b(u) = 2$. Thus there is another node w (we cannot have $v = w$ because T is a tree), with $d_w = d_v$ and $u = \mathsf{parent}(w)$. We get the contradiction because edge $(v, \mathsf{parent}(w))$ in graph G cannot exist by property $(P2)$.

Furthermore we can observe that for v: $t(v) > t(\mathsf{parent}(v))$, because we have, that $\mathsf{rank}(v) \geq \mathsf{rank}(\mathsf{parent}(v))$ by the construction of tree T. This implies by the monotonicity of algorithm SmallDiam, that $a(\mathsf{parent}(v)) \leq a(v)$, hence $t(v) > t(\mathsf{parent}(v))$.

Case 2: $\mathsf{rank}(v) < \mathsf{rank}(\mathsf{parent}(v))$. In this case v is the beginning of its path (i.e., $v = \mathsf{head}(\mathsf{super}(v))$). Algorithm SmallDiam is correct by Lemma 1 hence in some step t^*, node $\mathsf{super}(v)$ receives the message from some node $w' \in V'$ in step $a(w')$. And by the construction of the graph of paths G' we have that v is adjacent to some node $w \in V$ for which $w' = \mathsf{super}(w)$. Now we want to show that in step $t(w)$, node w delivers the message to v. Assume for contradiction that the transmission of node w collides at v with the transmission of some other node u. We have by Lemma 2, that $d_u = d_w$ and $a(u) = a(w)$. Consider nodes $\mathsf{super}(w)$ and $\mathsf{super}(u)$. Since $d_u = d_w$ we have $\mathsf{super}(w) \neq \mathsf{super}(u)$ (by property 2 from Definition 2). This means that there would be a collision between $\mathsf{super}(w)$ and $\mathsf{super}(u)$. In effect $\mathsf{super}(v)$ would not receive the message in step $a(\mathsf{super}(w))$, a contradiction.

We also need to show that $t(v) > t(w)$ – which means that v is not required to transmit before it receives the message. We know that $d_w = d_v - 1$ and by the monotonicity of algorithm SmallDiam we have that $a(w) < a(v)$ hence we get $t(v) > t(w)$. □

References

1. Leszek Gasieniec, Erez Kantor, Dariusz R. Kowalski, David Peleg, and Chang Su. Time efficient k-shot broadcasting in known topology radio networks. *Distributed Computing*, 21(2):117–127, 2008. `doi:10.1007/s00446-008-0058-0`.
2. Leszek Gasieniec, David Peleg, and Qin Xin. Faster communication in known topology radio networks. *Distributed Computing*, 19(4):289–300, 2007. URL: `https://doi.org/10.1007/s00446-006-0011-z`, `doi:10.1007/s00446-006-0011-z`.
3. A Stahler. Hypsometric (area altitude) analysis of erosional topology. *Geol Soc Am Bull*, 63:1117–1142, 1952.

APPENDIX

Proof of Lemma 1

Proof. It is easy to see that algorithm 1 is a correct 1-Shot algorithm. To prove the time complexity observe that the time to transmit the message between layers i and $i+1$ is $O(\sqrt{n_i+n_{i+1}})$. The total time is $O\left(\sum_{i\in\{0,1,\ldots D_r\}}\sqrt{n_i}\right)$, hence using Jensen's inequality:

$$\sum_{i\in\{0,1,\ldots D_r\}}\sqrt{n_i}=(D_r+1)\sum_{i\in\{0,1,\ldots D_r\}}\frac{\sqrt{n_i}}{D_r+1}\leq(D_r+1)\sqrt{\frac{\sum_{i\in\{0,1,\ldots D_r\}}n_i}{D_r+1}}\quad\in O(\sqrt{nD})$$

□

Proof of Lemma 2

Proof. Since u and u' are both neighbors of v we have $|d_u-d_v|\leq 1$, $|d_{u'}-d_v|\leq 1$ hence

$$|d_u-d_{u'}|\leq 2. \tag{1}$$

Since $t(u)=t(u')$, then:

$$d_u=d_{u'} \pmod 3 \tag{2}$$

From (1) and (2) we get that $d_u = d_{u'}$. This implies that also $2\cdot a(u)+b(u) = 2\cdot a(u')+b(u')$. If we had $b(u)=1$ and $b(u')=0$, the left side of the equality would be odd and the right would be even. Thus $b(u)=b(u')$ and also $a(u)=a(u')$. □