Faster Centralized Communication in Radio Networks

Ferdinando Cicalese[1,⋆], Fredrik Manne[2,⋆⋆], and Qin Xin[2,⋆⋆]

[1] AG Genominformatik, Technische Fakultät, Universität Bielefeld, Germany
nando@cebitec.uni-bielefeld.de
[2] Department of Informatics, The University of Bergen, Norway
{fredrikm, xin}@ii.uib.no

Abstract. We study the communication primitives of broadcasting (one-to-all communication) and gossiping (all-to-all communication) in known topology radio networks, i.e., where for each primitive the schedule of transmissions is precomputed based on full knowledge about the size and the topology of the network. We show that gossiping can be completed in $O(D+\frac{\Delta\log n}{\log\Delta-\log\log n})$ time units in any radio network of size n, diameter D and maximum degree $\Delta=\Omega(\log n)$. This is an almost optimal schedule in the sense that there exists a radio network topology, such as: a Δ-regular tree in which the radio gossiping cannot be completed in less than $\Omega(D+\frac{\Delta\log n}{\log\Delta})$ units of time. Moreover, we show a $D+O(\frac{\log^3 n}{\log\log n})$ schedule for the broadcast task. Both our transmission schemes significantly improve upon the currently best known schedules in Gąsieniec, Peleg and Xin [PODC'05], i.e., a $O(D+\Delta\log n)$ time schedule for gossiping and a $D+O(\log^3 n)$ time schedule for broadcast. Our broadcasting schedule also improves, for large D, a very recent $O(D+\log^2 n)$ time broadcasting schedule by Kowalski and Pelc.

Keywords: Centralized radio networks, broadcasting, gossiping.

1 Introduction

We consider the following model of a radio network: an undirected connected graph $G=(V,E)$, where V represents the set of nodes of the network and E contains unordered pairs of distinct nodes, such that $(v,w)\in E$ iff the transmissions of node v can directly reach node w and vice versa (the reachability of transmissions is assumed to be a symmetric relation). In this case, we say that the nodes v and w are *neighbours* in G. Note that in a radio network, a message transmitted by a node is always sent to all of its neighbors.

The *degree* of a node w is the number of its neighbours. We use Δ to denote the *maximum degree* of the network, i.e., the maximum degree of any node in the network. The *size of the network* is the number of nodes $n=|V|$.

Communication in the network is synchronous and consists of a sequence of communication steps. In each step, a node v either transmits or listens. If v transmits, then

⋆ Supported by the Sofja Kovalevskaja Award 2004 of the Alexander von Humboldt Stiftung.
⋆⋆ Supported by the Research Council of Norway through the SPECTRUM project.

T. Asano (Ed.): ISAAC 2006, LNCS 4288, pp. 339–348, 2006.
© Springer-Verlag Berlin Heidelberg 2006

the transmitted message reaches each of its neighbours by the end of this step. However, a node w adjacent to v successfully receives this message iff in this step w is listening and v is the only transmitting node among w's neighbors. If node w is adjacent to a transmitting node but it is not listening, or it is adjacent to more than one transmitting node, then a *collision* occurs and w does not retrieve any message in this step.

The two classical problems of information dissemination in computer networks are the *broadcasting* problem and the *gossiping* problem. The broadcasting problem requires distributing a particular message from a distinguished *source* node to all other nodes in the network. In the gossiping problem, each node v in the network initially holds a message m_v, and the aim is to distribute all messages to all nodes. For both problems, one generally considers as the efficiency criterion the minimization of the time needed to complete the task.

In the model considered here, the running time of a communication schedule is determined by the number of time steps required to complete the communication task. This means that we do not account for any internal computation within individual nodes. Moreover, no limit is placed on the length of a message which one node can transmit in one step. In particular, this assumption plays an important role in the case of the gossiping problem, where it is then assumed that in each step when a node transmits, it transmits all the messages it has collected by that time. (i.e., the ones received and its own one.)

Our schemes rely on the assumption that the communication algorithm can use complete information about the network topology. Such topology-based communication algorithms are useful whenever the underlying radio network has a fairly stable topology/infrastructure. As long as no changes occur in the network topology during the execution of the algorithm, the tasks of broadcasting and gossiping will be completed successfully. In this extended abstract we do not touch upon reliability issues. However, we remark that it is possible to increase the level of fault-tolerance in our algorithms, at the expense of some small extra time consumption. We defer this issue to the extended version of this paper.

Our results. We provide a new (efficiently computable) deterministic schedule that uses $O(D + \frac{\Delta \log n}{\log \Delta - \log \log n})$ time units to complete the gossiping task in any radio network of size n, diameter D and maximum degree $\Delta = \Omega(\log n)$. This significantly improves on the previously known best schedule, i.e., the $O(D + \Delta \log n)$ schedule of [10]. Remarkably, our new gossiping scheme constitutes an almost optimal schedule in the sense that there exists a radio network topology, specifically a Δ-regular tree, in which the radio gossiping cannot be completed in less than $\Omega(D + \frac{\Delta \log n}{\log \Delta})$ units of time.

For the broadcast task, we show a new (efficiently computable) radio schedule that works in time $D + O(\frac{\log^3 n}{\log \log n})$, improving the currently best published result for arbitrary topology radio networks, i.e., the $D + O(\log^3 n)$ time schedule proposed by Gąsieniec *et al.* in [10]. It is noticeable that for large D, our scheme also outperforms the very recent (asymptotically optimal) $O(D + \log^2 n)$ time broadcasting schedule by Kowalski and Pelc in [12]. This is because of the significantly larger coefficient of the D term hidden in the asymptotic notation. In fact, in our case the D term comes with coefficient 1.

Related work. The work on communication in known topology radio networks was initiated in the context of the broadcasting problem. In [3], Chlamtac and Weinstein prove that the broadcasting task can be completed in time $O(D \log^2 n)$ for every n-vertex radio network of diameter D. An $\Omega(\log^2 n)$ time lower bound was proved for the family of graphs of radius 2 by Alon *et al* [1]. In [5], Elkin and Kortsarz give an efficient deterministic construction of a broadcasting schedule of length $D + O(\log^4 n)$ together with a $D + O(\log^3 n)$ schedule for planar graphs. Recently, Gąsieniec, Peleg and Xin [10] showed that a $D + O(\log^3 n)$ schedule exists for the broadcast task, that works in *any* radio network. In the same paper, the authors also provide an optimal randomized broadcasting schedule of length $D + O(\log^2 n)$ and a new broadcasting schedule using fewer than $3D$ time slots on planar graphs. A $D+O(\log n)$-time broadcasting schedule for planar graphs has been showed in [13] by Manne, Wang and Xin. Very recently, a $O(D + \log^2 n)$ time deterministic broadcasting schedule for any radio network was proposed by Kowalski and Pelc in [12]. This is asymptotically optimal unless $NP \subseteq BPTIME(n^{\mathcal{O}(\log \log n)})$ [12]. Nonetheless, for large D, our $D + O(\frac{\log^3 n}{\log \log n})$ time broadcasting scheme outperforms the one in [12], because of the larger coefficient of the D term hidden in the asymptotic notation describing the time evaluation of this latter scheme.

Efficient radio broadcasting algorithms for several special types of network topologies can be found in Diks *et al.* [4]. For general networks, however, it is known that the computation of an optimal (radio) broadcast schedule is NP-hard, even if the underlying graph is embedded in the plane [2, 15].

Radio gossiping in networks with known topology was first studied in the context of radio communication with messages of limited size, by Gąsieniec and Potapov in [8]. They also proposed several optimal or close to optimal $O(n)$-time gossiping procedures for various standard network topologies, including lines, rings, stars and free trees. For general topology radio network a $O(n \log^2 n)$ gossiping scheme is provided and it is proved that there exists a radio network topology in which the gossiping (with unit size messages) requires $\Omega(n \log n)$ time. In [14], Manne and Xin show the optimality of this bound by providing an $O(n \log n)$-time gossiping schedule with unit size messages in any radio network. The first work on radio gossiping in known topology networks with arbitrarily large messages is [9], where several optimal gossiping schedules are shown for a wide range of radio network topologies. For arbitrary topology radio networks, an $O(D + \Delta \log n)$ schedule was given by Gąsieniec, Peleg and Xin in [10]. To the best of our knowledge no better result is known to date for arbitrary topology.

2 Gossiping in General Graphs with Known Topology

The gossiping task can be performed in two consecutive phases. During the first phase we gather all individual messages in one (central) point of the graph. Then, during the second phase, the collection of individual messages is broadcast to all nodes in the network. We start this section with the presentation of a simple gathering procedure that works in time $O((D + \Delta)\frac{\log n}{\log \Delta - \log \log n})$ in free trees. Later we show how to choose a spanning breadth-first (BFS) tree in an arbitrary graph G in order to gather (along its branches) all messages in G also in time $O((D+\Delta)\frac{\log n}{\log \Delta - \log \log n})$, despite the additional edges in G which might potentially cause additional collisions. Finally, we show how the gathering process can be pipelined and sped up to run in $O(D + \frac{\Delta \log n}{\log \Delta - \log \log n})$ time.

A super-ranking procedure. Given an arbitrary tree, we choose its central node c as the root. Then, the nodes in the tree (rooted at c) are partitioned into consecutive layers $L_i = \{v \mid dist(c, v) = i\}$, for $i = 0, .., r$ where r is a radius of the tree. We denote the size of each layer L_i by $|L_i|$.

We use a non-standard approach for ranking the nodes in a rooted tree, which we call *super-ranking*. The super-ranking depends on an integer parameter $2 \leq x \leq \Delta$, that for our purposes will be optimized later. Specifically, for every leaf v we define $\texttt{rank}(v, x) = 1$. Then, for a non-leaf node, v with children $v_1, \ldots, v_k$, we define $\texttt{rank}(v, x)$ as follows. Let $\hat{r} = \max_{i=1,\ldots,k}\{\texttt{rank}(v_i, x)\}$. If at least x of the children of v have rank $\hat{r}$, then $\texttt{rank}(v, x) = \hat{r} + 1$ otherwise $\texttt{rank}(v, x) = \hat{r}$.

For each $x \geq 2$, we define $r_{max}^{[x]} = \max_{v \in T} \texttt{rank}(v, x)$. As an immediate consequence of the definition of $\texttt{rank}(\cdot, \cdot)$ we have the following.

Lemma 1. *Let T be a tree with n nodes of maximum degree Δ. Then, $r_{max}^{[x]} \leq \lceil \log_x n \rceil$, for each $2 \leq x \leq \Delta$.*

Note that when $x = 2$ we obtain the standard ranking procedure, that has been employed in the context of radio communication in known topology networks in [6, 9, 10]. Previously this same ranking had been used to define the *Strahler number* of binary trees, introduced in hydrogeology [16] and extensively studied in computer science (cf. [17] and the references therein).

The schedule for gathering messages at the root is now defined in stages using the super-ranked tree under the assumption that the value of the parameter x has been fixed. For the sake of the analysis, we will optimize its value later. We partition the nodes of the tree into different *rank sets* that are meant to separate the stages in which nodes are transmitting, i.e., nodes from different rank sets transmit in different stages. For $y \geq 2$, let $r_{max}^{[y]}$ be the maximum rank for a node of T according to the super-ranking with parameter y. Recall that $r_{max}^{[y]} \leq \lceil \log_y n \rceil$. Then, let $R_i(y) = \{v \mid \texttt{rank}(v, y) = i\}$, where $1 \leq i \leq r_{max}^{[y]}$.

We use the above rank sets to partition the node set as follows. In particular, we shall use the ranking of the nodes both for the parameter y set to a fixed parameter $x > 2$ and to 2.

Definition 1. *We partition the set of nodes as follows:*
The fast transmission set is given by $F_j^k = \{v \mid v \in L_k \cap R_j(2) \text{ and } parent(v) \in R_j(2)\}$. *Also define* $F_j = \bigcup_{k=1}^{D} F_j^k$ *and* $F = \bigcup_{j=1}^{r_{max}^{[2]}} F_j$.
The slow transmission set is given by $S_j^k = \{v \mid v \in L_k \cap R_j(2)$ *and* $parent(v) \in R_p(2)$, *for some* $p > j$; *and* $\texttt{rank}(v, x) = \texttt{rank}(parent(v), x), x > 2\}$. *Also define* $S_j = \bigcup_{k=1}^{D} S_j^k$ *and* $S = \bigcup_{j=1}^{r_{max}^{[2]}} S_j$.
The super-slow transmission set is given by $SS_j^k = \{v \mid v \in L_k \cap R_j(x) \text{ and } parent(v) \in R_i(x), i > j\}$. *Accordingly, define* $SS_j = \bigcup_{k=1}^{D} SS_j^k$ *and* $SS = \bigcup_{j=1}^{r_{max}^{[x]}} SS_j$.

Lemma 2. *Fix positive integers $i \leq r_{max}^{[x]}$, $j \leq r_{max}^{[2]}$ and $k \leq D$. Then, during the ith stage, all nodes in F_j^k can transmit to their parents simultaneously without any collisions.*

Proof. Consider any two distinct nodes u and v in F_j^k, and suppose they interfere with each other. This is true if they have a neighbor in L_{k-1} in common. Obviously, u and v are on the same level and must therefore have the same parent y in the tree. Moreover, according to the definition of the fast transmission set F_j^k, $u, v, y \in R_j(2)$. However, according to the definition of the super-ranking procedure, if $\texttt{rank}(u, 2) = \texttt{rank}(v, 2) = j$ then $\texttt{rank}(y, 2)$ must be at least $j+1$. Hence the nodes u and v cannot both belong to F_j^k, which leads to a contradiction. ∎

Lemma 3. *Fix positive integers $i \leq r_{max}^{[x]}$, $j \leq r_{max}^{[2]}$ and $k \leq D$. Then, all messages from nodes in $S_j^k \cap R_i(x)$ can be gathered in their parents in at most $x - 1$ time units.*

Proof. By Definition 1 we have: For each node v in $S_j^k \cap R_i(x)$ we have that $parent(v)$ has at most $x - 1$ children in $S_j^k \cap R_i(x)$, for $i = 1, 2, ..., r_{max}^{[x]} \leq \lceil \log_x n \rceil$, $j = 1, 2, ..., r_{max}^{[2]} \leq \lceil \log n \rceil$ and $k = 1, .., D$. Now, using the above claim, the desired result is achieved by simply letting each parent of nodes in $S_j^k \cap R_i(x)$ collect messages from one child at a time. ∎

We shall use the following result from [10].

Proposition 1. *[10] There exists a gathering procedure Γ such that in any graph G of maximum degree Δ_G and diameter D_G the gossiping task, and in particular the gathering stage, can be completed in time $O(D_G + \Delta_G \log n)$.*

The following procedure moves messages from all nodes v with $\texttt{rank}(v, x) = i$ into their lowest ancestor u with $\texttt{rank}(u, x) \geq i + 1$, where $x > 2$, using the gathering procedure Γ from the previous proposition.

Procedure SUPER-GATHERING(i);

1. Move messages from nodes in $(F \cup S) \cap R_i(x)$ to SS_i;
   using the gathering procedure Γ in Proposition 1.
2. Move messages from nodes in SS_i to their parents;
   all parents collect their messages from their children in SS_i one by one.

Note that the subtrees induced by the nodes in $R_i(x)$ have maximum degree $\leq x$. Thus, by Proposition 1 and Lemma 3, we have that the time complexity of step 1 is $O(D + x \log n)$. The time complexity of step 2 is bounded by $O(\Delta)$, where Δ is the maximum degree of the tree. By Lemma 1, $r_{max}^{[x]} \leq \lceil \log_x n \rceil$. Thus, we have that the procedure SUPER-GATHERING completes the gathering stage in time $O((D+\Delta+x \log n) \log_x n)$. Since we can follow this with the trivial broadcasting stage following in time $O(D)$, we have proved the following.

Theorem 1. *In any tree of size n, diameter D and maximum degree Δ, the gossiping task can be completed in time $O((D + \Delta + x \log n) \log_x n)$, where $2 < x \leq \Delta$. In* *particular when* $\Delta = \Omega(\log n)$, *by choosing* $x = \frac{\Delta}{\log n}$, *we obtain the bound* $O((D + \Delta)\frac{\log n}{\log \Delta - \log\log n})$.

Gathering messages in arbitrary graphs. We start this section with the introduction of the novel concept of a *super-gathering spanning tree* (SGST). These trees play a crucial role in our gossiping-scheme for arbitrary graphs. We shall show an $O(n^3)$-time algorithm that constructs a SGST in an arbitrary graph G of size n and diameter D. In the concluding part of this section, we propose a new more efficient schedule that completes message gathering in time $O(D + \frac{\Delta \log n}{\log \Delta - \log\log n})$.

A *super-gathering spanning tree* (*SGST*) for a graph $G = (V, E)$ is any BFS spanning tree T_G of G, ranked according to the super-ranking above and satisfying[1]

(1) T_G is rooted at the central node c of G,
(2) T_G is ranked,
(3) all nodes in F_j^k of T_G are able to transmit their messages to their parents simultaneously without any collision, for all $1 \le k \le D$ and $1 \le j \le r_{max}^{[2]} \le \lceil \log n \rceil$
(4) every node v in $S_j^k \cap R_i(x)$ of T_G has following property: $parent(v)$ has at most $x - 1$ neighbours in $S_j^k \cap R_i(x)$, for all $i = 1, 2, ..., r_{max}^{[x]} \le \lceil \log_x n \rceil$, $j = 1, 2, ..., r_{max}^{[2]} \le \lceil \log n \rceil$ and $k = 1, .., D$.

Any BFS spanning tree T_G of G satisfying only conditions (1),(2), and (3) above is called a *gathering spanning tree*, or simply *GST*. Figure 1 shows an example of a *GST*. We recall the following result from [10].

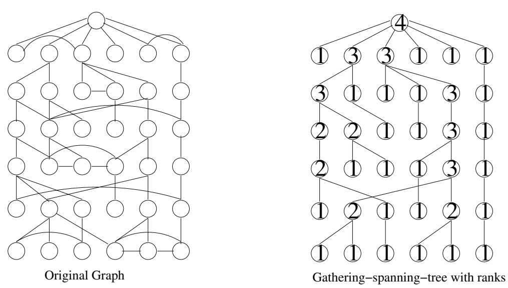

Fig. 1. Creating a gathering spanning tree

Theorem 2. *There exists an efficient (*$O(n^2 \log n)$ *time) construction of a* GST *on an arbitrary graph G. (see Theorem 2.5 in [10])*

The procedure SUPER-GATHERING-SPANNING-TREE constructs a super-gathering-spanning-tree SGST $\subseteq G$ on the basis of a GST$\subseteq G$ using a *pruning process*. The pruning process is performed layer by layer starting from the bottom (layer D) of the

[1] We use the definition 1 of the ranking partitions given above.

GST. For each layer we gradually *fix* the parents of all nodes which violate condition (4) above, i.e., each v in $S_j^k \cap R_i(x)$ of *GST*, such that $parent(v)$ has at least x neighbours in $S_j^k \cap R_i(x)$. In fact, for our gathering-scheme, v is a node which is potentially involved in collisions. In each layer, the pruning process starts with the nodes of highest rank in the current layer. We use $NB(v)$ to denote the set of neighbours of the node v in the original graph G. In Figure 2, we show the output of the SUPER-GATHERING-SPANNING-TREE procedure when it is run on the GST presented in Figure 1.

Procedure SUPER-GATHERING-SPANNING-TREE(*GST*);
(1) Fix $\texttt{rank}(w, 2)$ for every node $w \in V$;
(2) For $k := D$ down to 1 do
(3) For $i := r_{max}^{[x]}$ down to 1 do
(4) For $j := r_{max}^{[2]}$ down to 1 do
(5) While $\exists v \in S_j^k \cap R_i(x)$ in *GST* such that $|S_j^k \cap R_i(x) \cap NB(parent(v))| \geq x$ do
(6) $\texttt{rank}(parent(v), x) = i + 1$; //$\texttt{rank}(v, x) = i$
(7) $UPDATE = \{u | u \in S_j^k \cap R_i(x) \cap NB(parent(v))\}$;
(8) $SS^k_{\texttt{rank}(v,x)} = SS^k_{\texttt{rank}(v,x)} \cup UPDATE$;
(9) $E_{GST} = E_{GST} - \{(u, parent(u)) | u \in UPDATE\}$;
(10) $E_{GST} = E_{GST} \cup \{(u, parent(v)) | u \in UPDATE\}$;
(11) $S_j^k = S_j^k - \{u | u \in UPDATE\}$;
(12) re-set $\texttt{rank}(w, x)$ for each $w \in V$;
(13) recompute the sets S and SS in GST

We now prove that Procedure SUPER-GATHERING-SPANNING-TREE constructs the SGST of an arbitrary graph $G = (V, E)$ in time $O(n^3)$. The following technical lemma is easily proved by induction.

Lemma 4. *After completing the pruning process at layer k in GST, the structure of edges in GST between layers $k-1, .., D$ is fixed, i.e., each node v within layers $k, .., D$ in all sets $S_j^k \cap R_i(x)$, satisfy the following property: $parent(v)$ has at most $x - 1$ neighbours in $S_j^k \cap R_i(x)$, for $i = 1, .., r_{max}^{[x]} \leq \lceil \log_x n \rceil$ and $j = 1, .., r_{max}^{[2]} \leq \lceil \log n \rceil$.*

By the above lemma, Theorem 2 and the fact that procedure SUPER-GATHERING-SPANNING-TREE preserves the property of the GST it starts with, we get

Theorem 3. *For an arbitrary graph there exists an $O(n^3)$ time construction of a $SGST$.*

$O((D + \Delta)\frac{\log n}{\log \Delta - \log \log n})$-time gossiping. Using the ranks computed on the $SGST$, the nodes of the graph are partitioned into distinct *rank sets* $R_i = \{v | \texttt{rank}(v, x) = i\}$, where $1 \leq i \leq r_{max}^{[x]} \leq \lceil \log_x n \rceil$. This allows the gathering of all messages into the central node c, stage by stage, using the structure of the $SGST$ as follows. During the ith stage, all messages from nodes in $(F \cup S) \cap R_i(x)$ are first moved to the nodes in SS_i. Later, we move all messages from nodes in SS_i to their parents in $SGST$. In order to avoid collisions between transmissions originating at neighbouring BFS layers we divide the sequence of transmission time slots into three separate (interleaved)

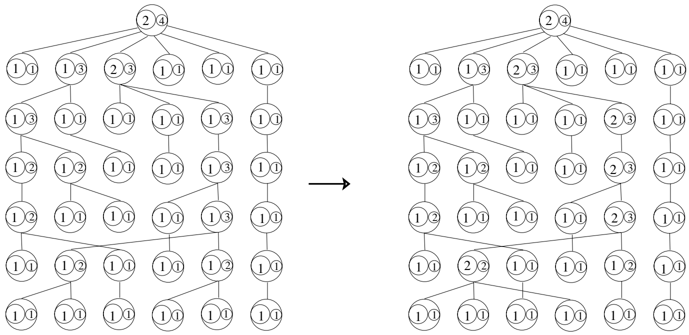

Fig. 2. From gathering-spanning-tree to super-gathering-spanning-tree

subsequences of time slots. Specifically, the nodes in layer L_j transmit in time slot t iff $t \equiv j \pmod 3$.

Lemma 5. *In stage i, the nodes in the set SS_i of the $SGST$ transmit their messages to their parents in time $O(\Delta)$.*

Proof. By [9, Lemma 4], one can move all messages between two partitions of a bipartite graph with maximum degree Δ (in this case two consecutive BFS layers) in time Δ. The solution is based on the use of the *minimal covering set*. Note that during this process a (possibly) combined message m sent by a node $v \in SS_i$ may be delivered to the parent of another transmitting node $w \in SS_i$ rather than to $parent(v)$. But this is fine, since now the time of delivery of the message m to the root of the tree is controlled by the delivery mechanism of the node w. Obviously this flipping effect can be observed a number of times in various parts of the tree, though each change of the route does not change the delivering mechanism at all.

In order to avoid extra collisions caused by nodes at neighbouring BFS layers, we use the solution with three separate interleaved subsequences of time slots incurring a slowdown with a multiplicative factor of 3. ∎

When the gathering stage is completed, the gossiping problem is reduced to the broadcasting problem. We distribute all messages to every node in the network by reversing the direction and the time of transmission of the gathering stage. In section 3 we prove that the broadcasting stage can be performed faster in graphs with large Δ, i.e., in time $D + O(\frac{\log^3 n}{\log \log n})$.

Theorem 4. *In any graph G with $\Delta = \Omega(\log n)$, the gossiping task can be completed in time $O((D + \Delta)\frac{\log n}{\log \Delta - \log \log n})$.*

Proof. During the ith stage, all messages from $(F \cup S) \cap R_i(x)$ are moved to SS_i. Because of property (4) of the SGST, Proposition 1 assures that this can be achieved in time $O(D + x \log n)$. By Lemma 5, all nodes in the set SS_i can transmit their messages to their parents in $SGST$ in time $O(\Delta)$. By Lemma 1, this process is repeated at most $\log_x n$ times. Thus, the gossiping time can be bounded by $O((D + \Delta + x \log n) \log_x n)$. The desired result follows directly by setting $x = \frac{\Delta}{\log n}$. ∎

$O(D + \frac{\Delta \log n}{\log \Delta - \log \log n})$ **-time gossiping.** The result of Theorem 4 is obtained by a transmission process consisting of $\lceil \log_x n \rceil$ separate stages, each costing $O(D+\Delta+x \log n)$ units of time. We shall now show that the transmissions of different stages can be pipelined and a new gossiping schedule obtained of length $O(D + \frac{\Delta \log n}{\log \Delta - \log \log n})$.

The communication process will be split into consecutive blocks of 9 time units each. The first 3 units of each block are used for fast transmissions from the set F, the middle 3 units are reserved for slow transmissions from the set S and the remaining 3 are used for super-slow transmissions of nodes from the set SS. We use 3 units of time for each type of transmission in order to prevent collisions between neighbouring BFS layers, like we did in the last section. Recall that we can move all messages between two consecutive BFS layers in time Δ [9, Lemma 4]. Moreover, the same result in [9] together with property (4) of the GSTS, allows us to move all messages stored in $S_j^k \cap R_i(x)$ to their parents in $SGST$ within time $x - 1$.

We compute for each node $v \in S_j \cap R_i(x)$ at layer k the number of a step $1 \leq s(v) \leq x - 1$ in which node v can transmit without interruption from other nodes in $S_j \cap R_i(x)$ also in layer k. We also compute for each node $u \in SS_i$ at layer k the number of a step $1 \leq ss(u) \leq \Delta$ in which the node u can transmit without interruption from other nodes in SS_i also in layer k.

Let v be a node at layer k and with $\mathtt{rank}(v, 2) = j$ and $\mathtt{rank}(v, x) = i$, in SGST. Depending on if v belongs to the set F, to the set S or to the set SS, it will transmit in the time block $t(v)$ given by:

$$t(v) = \begin{cases} (D-k+1) + (j-1)(x-1) + (i-1)\left(\Delta + (x-1)\log n\right) & \text{if } v \in F \\ (D-k+1) + (j-1)(x-1) + s(v) + (i-1)\left(\Delta + (x-1)\log n\right) & \text{if } v \in S \\ (D-k+1) + \log n(x-1) + (i-1)\left(\Delta + (x-1)\log n\right) + ss(v) & \text{if } v \in SS \end{cases}$$

We observe that any node v in the $SGST$ requires at most D fast transmissions, $\log n$ slow transmissions and $\log_x n$ super-slow transmissions to deliver its message to the root of the $SGST$ if there is no collision during each transmission. Moreover, the above definition of $t(v)$ results in the the following lemma, whose proof is deferred to the full version of the paper.

Lemma 6. *A node v transmits its message as well as all messages collected from its descendants towards its parent in SGST successfully during the time block allocated to it by the transmission pattern.*

Since the number of time blocks used is $\leq D + (x \cdot \log n + \Delta) \cdot (\log_x n + 1)$, we have

Theorem 5. *In any graph G, the gossiping task can be completed in time $O(D + (x \cdot \log n + \Delta) \log_x n)$, where $2 \leq x \leq \Delta$. In particular when $\Delta = \Omega(\log n)$, by setting $x = \frac{\Delta}{\log n}$ the bound becomes $O(D + \frac{\Delta \log n}{\log \Delta - \log \log n})$.*

By employing the solution of the equation $\Delta = x \log x$ one can obtained an improved $O(D + \frac{\Delta \log n}{\log \Delta - \log \log n + \log \log \log^\star n})$-time gossiping schedule. Moreover, a recursive procedure can be employed to attain the bound $O(D + \frac{\Delta \log n}{\log \Delta - \log \log n + \log^c \log \log^\star n})$, where c is some constant.

3 Final Remarks: Broadcasting in Graphs with Known Topology

By exploiting the structure of the SGST it is possible to obtain a very efficient scheduling algorithm for completing the broadcasting task in a general known topology radio network. The following theorem summarizes our findings. Due to the space constaints we defer the details to the full version of the paper.

Theorem 6. *For any n node radio network of diameter D, a broadcasting schedule of length $D + O(\frac{\log^3 n}{\log \log n})$ can be deterministically constructed in polynomial time.*

References

1. N. Alon, A. Bar-Noy, N. Linial and D. Peleg. A lower bound for radio broadcast. *J. Computer and System Sciences* 43, (1991), 290 - 298.
2. I. Chlamtac and S. Kutten. On broadcasting in radio networks-problem analysis and protocol design. *IEEE Trans. on Communications* 33, (1985), pp. 1240-1246.
3. I. Chlamtac and O. Weinstein. The wave expansion approach to broadcasting in multihop radio networks. *IEEE Trans. on Communications* 39, (1991), pp. 426-433.
4. K. Diks, E. Kranakis, D. Krizanc and A. Pelc. The impact of information on broadcasting time in linear radio networks. *Theoretical Computer Science*, 287, (2002), pp. 449-471.
5. M. Elkin and G. Kortsarz. Improved broadcast schedule for radio networks. *Proc. 16th ACM-SIAM Symp. on Discrete Algorithms*, 2005, pp. 222-231.
6. I. Gaber and Y. Mansour. Broadcast in radio networks. *Proc. 6th ACM-SIAM Symp. on Discrete Algorithms*, 1995, pp. 577-585.
7. L. Gąsieniec, E. Kranakis, A. Pelc and Q. Xin. Deterministic M2M multicast in radio networks. *Proc. 31st ICALP*, 2004, LNCS 3142, pp. 670-682.
8. L. Gąsieniec and I. Potapov, Gossiping with unit messages in known radio networks. *Proc. 2nd IFIP Int. Conference on Theoretical Computer Science*, 2002, pp. 193-205.
9. L. Gąsieniec, I. Potapov and Q. Xin. Efficient gossiping in known radio networks. *Proc. 11th SIROCCO*, 2004, LNCS 3104, pp. 173-184.
10. L. Gąsieniec, D. Peleg and Q. Xin. Faster communication in known topology radio networks. *Proc. 24th Annual ACM SIGACT-SIGOPS PODC*, 2005, pp. 129-137.
11. L. Gąsieniec, T. Radzik and Q. Xin. Faster deterministic gossiping in ad-hoc radio networks. *Proc. 9th Scandinavian Workshop on Algorithm Theory*, 2004, LNCS 3111, pp. 397-407.
12. D. Kowalski and A. Pelc. Optimal deterministic broadcasting in known topology radio networks. *Distributed Computing*, (2006), to appear.
13. F. Manne, S. Wang and Q. Xin. Faster radio broadcast in planar graphs. *Manuscript, 2006.*
14. F. Manne and Q. Xin. Optimal gossiping with unit size messages in known radio networks. *Proc. 3rd Workshop on Combinatorial and Algorithmic Aspects of Networking*, to appear.
15. A. Sen and M.L. Huson. A new model for scheduling packet radio networks. *Proc. 15th Joint Conf. of IEEE Computer and Communication Societies*, 1996, pp. 1116-1124.
16. A.N. Strahler. Hypsometric (area-altitude) analysis of erosional topology. *Bull. Geol. Soc. Amer.* 63, (1952), pp. 117–1142.
17. X.G. Viennot. A Strahler bijection between Dyck paths and planar trees. *Discrete Mathematics* 246, (2002), pp. 317–329.