Noname manuscript No.
(will be inserted by the editor)

Insert your title here

Do you have a subtitle? If so, write it here

First Author · Second Author

Received: date / Accepted: date

Abstract **Keywords**

1 Introduction

The minimum broadcast time consists of a set of communication nodes with a subset of source nodes. The task is to disseminate a signal to every node in a shortest possible time while abiding by communication rules. An *informed* node is a node that has received the signal. Otherwise, a node is *uninformed*. At the beginning, the set of informed nodes is exactly the set of sources. An informed node u can send the signal to an uninformed node v if u and v are located within a communication vicinity of each other.

The continuous time is divided into discrete time steps. At each time step, every informed node can forward the signal to at most one uninformed neighbor. The number of informed nodes at some time step can therefore be up to double the number of informed nodes at the previous time step.

2 Network Model and Notation

The communication network is represented by a connected undirected unweighted graph $G = (V, E)$ and a subset of nodes $S \subseteq V$, where $|V| = n$ and $|S| = s$. Broadcasting is defined as a sequence of sets $S = V_0 \subseteq \cdots \subseteq V_k = V$ where each V_i represents the nodes informed after time step i, $0 \leq i \leq k$.

F. Author
first address
Tel.: +123-45-678910
Fax: +123-45-678910
E-mail: fauthor@example.com

S. Author
second address

For each node $v \in V_i \setminus V_{i-1}$, there exists a single node $p(v) \in V_{i-1}$ adjacent to v, which forwarded the signal to v. Also, for every two distinct nodes $u, v \in V_i \setminus V_{i-1}$ we have $p(u) \neq p(v)$. The optimization problem in question is defined as follows [2,3]:

Problem 1 Given $G = (V, E)$ and $S \subseteq V$, find a sequence $S = V_0 \subseteq \cdots \subseteq V_k = V$ of minimum length k and a mapping $p : V \setminus S \to V$, such that for each $v \in V \setminus S : \{v, p(v)\} \in E$, and for each $u, v \in V_i \setminus V_{i-1} : p(u) \neq p(v) \Leftrightarrow u \neq v$.

We also define the set $A = \{(i, j), (j, i) : \{i, j\} \in E\}$ that consists of all arcs that can be derived by directing edges in E. A degree, in-degree and out-degree of node v in G are denoted by $\deg_G(v)$, $\deg_G^-(v)$ and $\deg_G^+(v)$, respectively. Whenever there is no danger of confusion, the subscript G is omitted. The set of neighbors of $v \in V$ in G is denoted by $N(v)$.

For convenience we further define a graph $F_i = (V_i, E_i)$ with the edge set $E_i = \{\{v, p(v)\} : v \in V_i\}$. F_i is a forest containing pruned binomial trees defined by the mapping p.

3 Lower bounds

In this section, we study lower bounds on delay k for several restrictions of input graphs. An optimal solution is obtained by solving a sequence of decision problems with varying k. It is therefore desirable to determine tight lower and upper bounds in order to arrive in the optimum after solving as few decision problems as possible. Obvious bounds for a general graph instance are given by

Observation 1 *For an instance (G,S) of Problem 1,*

$$\left\lceil \log \frac{n}{s} \right\rceil \leq k \leq n - s.$$

Consider a d-regular graph with one source. The forest H_k consists of a single tree. We investigate the number of leaves in H_k and derive a lower bound on the dealy for this graph class. For $k = 1, 2$, $L(H_1) = L(H_2) = 2$. For $k \geq 3$, $L(H_k)$ corresponds to the number of nodes with degree $1, \ldots, d-1$ in H_{k-1}. It can also be interpreted as the sum of number of leaves in $H_{k-d+1}, \ldots, H_{k-1}$, which leads to the following formula

$$L(H_k) = \sum_{i=k-d+1}^{d-1} L(H_i). \tag{1}$$

Eq. (1) exactly corresponds to the recursive definition of Fibonacci sequence of order $d - 1$. As each of the two base cases $L(H_1)$ and $L(H_2)$ equal double the base cases of the Fibonacci sequence, the number of leaves in time step k is calculated as

$$L(H_k) = 2F_k^{d-1}. \tag{2}$$

Since each node in H_k had been a leaf in exactly one time step, the number of nodes in H_k can be expressed as

$$|V_k| = 2\sum_{i=1}^{k} F_i^{d-1}. \tag{3}$$

Proposition 1 *For a d-regular graph on n and s source, the lower bound on delay is*

$$\left\lceil \min\{k : 2\sum_{i=1}^{k} F_i^{d-1} \geq n\}/s \right\rceil. \tag{4}$$

Proof □

An additional knowledge of a degree sequence of G can be exploited. Alg. 1 calculates a lower bound on the delay when given number of nodes, sources and a degree sequence in G as an input. The algorithm iteratively updates possible node degrees in F_i in each time step i, and records the maximum potential number of nodes in V_i. For the purpose of finding lower bounds, it is assumed that each node $v \in V_i$ with $\deg_{F_i}(v) < \deg_G(v)$ informs a new uninformed node. The number of iterations is then the lowest possible delay for given input. Once a node v reaches its maximum degree, i.e., when $\deg_{F_i} = \deg_G(v)$ for some i, v does not inform any other node in the next time steps.

Data: $n, s, d_1, \dots, d_n \in \mathbb{N}, s \leq n,$
$1 \leq d_n \leq \dots \leq d_2 \leq d_1$

$a_1, \dots, a_{2s} \leftarrow 1;$
for $i = 2s+1, \dots, n$ **do**
  $a_i \leftarrow 0;$
end
$c \leftarrow 2s;$`// every source informes a new node`
$k \leftarrow 1;$
while $c < n$ **do**
  $k \leftarrow k+1;$
  $c_n \leftarrow 0;$
  for $i = 1, \dots, c$ **do**
    if $a_i < d_i$ **then**
      $a_i \leftarrow a_i + 1;$
      $c_n \leftarrow c_n + 1;$
      if $c + c_n < n$ **then**
        $a_{c+c_n} \leftarrow 1;$`// Newly informed node`
      end
    end
  end
  $c \leftarrow c + c_n;$
end
return $k;$

Algorithm 1: Lower bound exploiting distribution of degrees

The input is assumed to be correct in the sense that a graph with given degree sequence exists, and by definition, the degree sequence is ordered non-increasingly. For each iteration k, Alg. 1 stores degrees of nodes in F_k in variables $a_1, \ldots, a_n$. Note that the forest F_k is not actually constructed. The algorithm operates merely with potential degrees of nodes in F_k. Next, variable c keeps the value $|V_k|$, i.e., the number nodes informed within k steps. Finally, c_n stores $|V_k \setminus V_{k-1}|$, thus the number of nodes newly informed in time step k.

Proposition 2 *If G is an arbitrary graph with n nodes, s sources and node degrees d_i, $1 \leq i \leq n$, Alg. 1 calculates a lower bound for the delay k from Problem 1 in G.*

Proof (Idea:) The most optimistic scenario is when F_k consists of full binomial trees. In such a case, for $u \in V_i$ and $v \in V_j$ we have $\deg_{F_k}(v) \leq \deg_{F_k}(u) \Leftrightarrow i \leq j$. We can therefore assert that in an optimal solution, $\deg_G(u) \leq \deg_G(v) \Leftrightarrow i \leq j$. Whenever $\deg_{F_k}(v) < \deg_G(v)$, v informs a new node in $V \setminus V_k$ with maximum degree in G, which is ensured by the decreasing order of input node degrees. For every $v \in V$, $\deg_{F_k}(v) \leq \deg_G(v)$ always holds. Further, when the condition $c < n$ is tested on line 7, c contains the correct value of number of nodes in V_k. The returned value is therefore a lower bound on the delay. $\square$

References

1. Chu, X., Chen, Y., Time division intersatellite link topology generation problem: Modeling and solution, International Journal of Satellite Communications and Networking, 194 – 206, 36 (2017)
2. Jansen, K., Müller, H., The minimum broadcast time problem for several processor networks, Theoretical Computer Science, 69 – 85, 147 (1995)
3. Middendorf, M., Minimum broadcast time is NP-complete for 3-regular planar graphs and deadline 2, Information Processing Letters, 281 – 287, 46 (1993)