Discrete Applied Mathematics 159 (2011) 322–327

Contents lists available at ScienceDirect

Discrete Applied Mathematics

journal homepage: www.elsevier.com/locate/dam

Messy broadcasting — Decentralized broadcast schemes with limited knowledge

Hovhannes A. Harutyunyan [a], Pavol Hell [b], Arthur L. Liestman [b,*]

[a] Department of Computer Science and Software Engineering, Concordia University, Montreal, QC, H3G 1M8, Canada

[b] School of Computing Science, Simon Fraser University, Burnaby, B.C., V5A 1S6, Canada

ARTICLE INFO

Article history:
Received 21 August 2009
Received in revised form 25 October 2010
Accepted 1 December 2010
Available online 28 December 2010

Keywords:
Broadcasting
Messy model
Upper bound
Trees

ABSTRACT

We consider versions of broadcasting that proceed in the absence of information about the network. In particular, the vertices of the network do not know the structure of the network or the starting time, originator, or state of the broadcast. Furthermore, the protocols are not coordinated. This synchronous anonymous communication model has been called messy broadcasting. We perform a worst case analysis of three variants of messy broadcasting. These results also provide upper bounds on broadcasting where every vertex simply calls each of its neighbors once in random order. We prove exact bounds on the time required for broadcasting under two variants and give a conjectured value for the third.

1. Introduction

Broadcasting is an information dissemination problem in a connected network in which one vertex, called the *originator*, must distribute a message to all other nodes by a series of calls along the communication lines of the network. This is assumed to take place in discrete time units. Assuming the structure of the network is known, the broadcast is to be completed as quickly as possible, where each call involves at least one informed vertex and each informed vertex may call at most one other vertex per unit of time. (A vertex may receive several calls at the same time unit.)

For a given originator vertex x, we define the *broadcast time*, denoted $t(x)$, to be the minimum number of time units required to complete broadcasting from x. Note that $t(x) \geq \lceil \log_2 n \rceil$ for any vertex x in a connected graph G on n vertices, since during each time unit the number of informed vertices can at most double. (All logarithms in this paper are base 2 and we will omit the subscript 2 below.) The *broadcast time of the graph G*, denoted $t(G)$, is $\max\{t(x) | x \in V\}$. These definitions provide the starting point for many interesting investigations.

For surveys of results on broadcasting and related problems, see [17,12,18,19]. Many recent papers have been written on various aspects of broadcasting with different assumptions. For approximation algorithms, see, e.g., [7], for lower bounds on broadcast time, see, e.g., [10]; see also [3]. Randomized rumor spreading was considered in several papers (see, for example, [6,13,20,22]).

Messy broadcasting is a concept introduced by Ahlswede et al. [1] and further examined by various authors [4,15,16,21]. (A more descriptive phrase might be synchronous anonymous broadcasting, but for historical reasons we will continue to use the term "messy".) Here, it is assumed that the vertices of the network do not know the network structure. When broadcasting a message, the participants (other than the originator) do not know the originator or the time elapsed since the broadcast began. Further, the participants have restricted information about which of their neighbors are informed. Thus, they must make local decisions to forward the message, acting as independent agents with a limited view of the network.

* Corresponding author. Fax: +1 778 782 3045.
E-mail address: art@cs.sfu.ca (A.L. Liestman).

0166-218X/$ – see front matter
doi:10.1016/j.dam.2010.12.002

Our network is modeled as a connected graph $G = (V, E)$ where V is the set of vertices and E is the set of communication lines (edges). We consider a synchronized communication protocol under which message transmission time is assumed to be constant (independent of the message). Each vertex can transmit a message to at most one of its neighbors in a given time unit, but can receive information from any number of its neighbors simultaneously. In the messy model, an *informed* vertex may have neighbors that it knows have received the message. From the point of view of that vertex, the other neighbors may not know the message and are assumed to be *uninformed*. In each time unit, every informed vertex with uninformed neighbors must transmit the message to one of those uninformed neighbors. Since knowledge of its local neighborhood is limited, it may, in fact, send the message to an informed vertex.

The following three variants provide each vertex with slightly different views of their local neighborhood.

Model M_1: At each unit of time, every vertex knows exactly which of its neighbors are informed and which are uninformed.

Model M_2: At each unit of time, every informed vertex knows from which vertex (or vertices) it received the message and to which neighbors it has sent the message. Thus, it knows that these vertices are informed and must assume that other neighbors are uninformed.

Model M_3: Every informed vertex knows only to which neighbors it has sent the message. It does not know from which neighbor it received the message. Thus, it must assume that all neighbors are uninformed except those to which it has sent the message.

We are concerned with the worst case performance of broadcast schemes in these models. In other words, we investigate how slow the broadcast can be under the respective rules.

If we were to consider the expected time, instead of the worst case time, under the respective rules, we would obtain a randomized model for which our worst case results obviously give an upper bound.

We define the *broadcast time of vertex x* in graph G using model M_i, denoted $t_i(x)$, for $i = 1, 2, 3$, to be the maximum number of time units required to complete broadcasting from vertex x over all broadcast schemes for x. Broadcasting is *complete* when all vertices are informed. To clarify, the broadcast time of a vertex x in graph G under model M_i is the first time unit at which there are no uninformed vertices. The *broadcast time of graph G* using model M_i, denoted $t_i(G)$, for $i = 1, 2, 3$, is the maximum broadcast time for any vertex x of G. That is, $t_i(G) = \max\{t_i(x)|x \in V\}$. From the definitions, it is clear that $t(G) \le t_1(G) \le t_2(G) \le t_3(G)$ for any connected graph G.

Some other approaches have been used to investigate broadcasting where vertices are assumed to have limited knowledge of the structure of the network. Gargano et al. [14] give worst case results for a model called blind broadcasting in which vertices know only their immediate neighbors. De Marco and Pelc [5] give worst case results on broadcasting time for a model in which each vertex knows the structure of the network within a given distance r. This model was first proposed by Awerbuch et al. [2] in order to formally investigate the tradeoff between information and complexity, measuring the number of messages used rather than the broadcasting time. The tradeoffs between a priori knowledge and efficiency have been investigated by Fraigniaud et al. [11]. Feige et al. [9] (and later Elsässer et al. [8]) consider a version of broadcasting where a vertex only knows how many neighbors it has. They propose a procedure in which each informed vertex sends the message to one of its neighbors chosen uniformly at random in each time period. In contrast to our model, this requires no local memory but may result in sending the message repeatedly to the same neighbor. They analyze the expected behavior of their procedure.

There are obvious lower and upper bounds on the broadcast time $t(G)$ for any graph $G = (V, E)$ on n vertices: $\lceil \log n \rceil \le t(G) \le n - 1$. For messy broadcasting, such bounds seem to be more difficult to establish. Ahlswede et al. considered the problem of constructing graphs with the smallest possible messy broadcast times [1]. They proved the existence of graphs G and H on n vertices with messy broadcast times $t_1(G) \le t_2(G) \le \frac{3}{\log 3} \log n$ and $t_3(H) \le 2.5 \log n$ for large n. In this paper we are concerned with the *greatest* possible broadcast times, i.e., we study the functions $u_i(n) = \max\{t_i(G)|G = (V, E), |V| = n\}$ for $i = 1, 2, 3$.

This paper is organized as follows. In Section 2, we determine the exact value of $u_3(n)$, showing $u_3(n) = 2n - 3$ and describing a class of graphs for which this value is attained. In Section 3, we present the exact value of $u_1(n)$, showing $u_1(n) = n - 1$ and again describing a class of graphs for which the value is attained. In that same section, we also conjecture the exact value of $u_2(n)$ to be $u_2(n) = 2n - \lceil \log n \rceil - 2$. We prove that this is a lower bound for those values $n = 2^k + 1$ and describe a class of graphs for which this conjectured value is attained. We conclude the paper with some remarks including the statement of tight upper bounds on the messy broadcast time for trees.

The worst case bounds in the three models M_1, M_2, and M_3 differ at most by a multiplicative factor of 2. One may ask whether this holds for all individual graphs, as well. Although we do not know the answer, there is some evidence that it may not hold for every graph. In particular, the n-dimensional hypercube Q_n has $t_2(Q_n) = n(n - 1)/2$ [16]. We know that $t_1(Q_n) \ge 2n - 2$ and it seems possible that that $t_1(Q_n) = \theta(n)$ while $t_2(Q_n) = \theta(n^2)$.

2. Model M_3

Theorem 2.1. $u_3(n) = 2n - 3$.

Proof. Let $G = (V, E)$ be a graph on n vertices with $t_3(G) = u_3(n)$. We construct a sequence of n vertex graphs $G^1, G^2, G^3, \ldots, G^p$ such that $u_3(n) = t_3(G) = t_3(G^1) = \cdots = t_3(G^p) = 2n - 3$.