Reliable Broadcasting

Luisa Gargano*
Dipartimento di Informatica ed Applicazioni
Università di Salerno
84081 Baronissi (SA), Italy

Arthur L. Liestman†
School of Computing Science
Simon Fraser University
Burnaby, B.C. V5A 1S6, Canada

Joseph G. Peters‡
School of Computing Science
Simon Fraser University
Burnaby, B.C. V5A 1S6, Canada

Dana Richards
Department of Computer Science
University of Virginia
Charlottesville, Virginia 22901, U.S.A.

Abstract

Broadcasting is a process of information dissemination in a communications network whereby a message, originated by one member, is transmitted to all members of the network. By adding some redundant calls to the broadcasting scheme, the completion of the broadcast can be guaranteed in the presence of faulty components. We investigate the implications of transmission failures on broadcasting. In particular, we consider broadcasting when the number of transmission failures is bounded by a constant. We determine the time required to guarantee a broadcast in this model. We also study the number of links required in networks which allow reliable broadcasting.

1 Introduction

A graph $G = (V, E)$ represents a communications network in which the vertices in V correspond to the members of the network and the edges in E correspond to communication links connecting pairs of members. In *broadcasting*, one member of the network, the *originator*, has a message which is to be transmitted to all of the other members as quickly as possible by a series of calls over the network. Each of these calls requires one unit of time; any member of the network can participate in at most one call per time unit; and any member can only call an adjacent member.

If the network has some faulty components, it may be difficult to broadcast a message. However, by incorporating sufficient redundancy into the calling scheme, the completion of the broadcast can be guaranteed in the presence of a bounded number of failures. In this paper, we investigate broadcasting in the presence of a constant number of transmission failures. We assume that any particular message transmission may fail so that, although a message is sent along the link, it is not received at the other end. Such failures are assumed to be of short duration. In particular, we assume that if two messages are sent along the same link in consecutive time units, a single failure

*This research was supported by Progetto Finalizzato Sistemi Informatici e Calcolo Parallelo of C.N.R. under Grant No. 91.00939.PF69 and by The Italian Ministry of the University and Scientific Research, Project: Algoritmi, Modelli di Calcolo e Strutture Informative.

†This research was supported by the Natural Sciences and Engineering Research Council of Canada under Grant No. OGP0001734.

‡This research was supported by the Natural Sciences and Engineering Research Council of Canada under Grant No. OGP0000322.

cannot prevent both messages from arriving at the other end. Furthermore, we assume that at most k such failures occur during the broadcast, where k is a constant.

Previous research into fault-tolerant broadcasting has focused on other types of faults. Several papers have investigated broadcasting in the presence of link failures of long duration [6, 12, 13, 16, 17, 19]. Berman and Hawrylycz considered broadcasting from a single specified originator in the presence of link failures [3]. Farley [8] and Farley and Proskurowski [9] studied broadcasting in the presence of non-adjacent link and/or vertex failures. Bienstock considered broadcasting in a model in which each edge is independently faulty with probability p [4]. Pelc [18] has studied a model in which both nodes and calls can fail. Several other papers have investigated broadcasting in faulty hypercubes [5, 10, 11, 20]. For a survey of work on broadcasting and related problems, see [14].

2 Definitions

Given a graph $G = (V, E)$, the *k-reliable broadcast time of a vertex* u of G, denoted $t_k(u)$, is the minimum time required to guarantee broadcast from u in the presence of up to k transmission failures. The *k-reliable broadcast time of a graph* G, denoted $t_k(G)$, is the maximum k-reliable broadcast time of any vertex u in G. For $k = 0$, these definitions are the standard definitions of broadcast time. We are interested in determining $T_k(n)$, the minimum value of $t_k(G)$ over all graphs G on n vertices, that is, $T_k(n) = \min\{t_k(G)|G \text{ is a graph on } n \text{ vertices}\}$.

It is easy to see that for any vertex u in a connected graph G with n vertices, $t_k(u) \geq \lceil \log n \rceil$, since the number of vertices which have received the message can at most double during each time unit. It is also easy to see that $t_0(K_n) = \lceil \log n \rceil$ where K_n is the complete graph on n vertices. Thus, $T_0(n) = \lceil \log n \rceil$. However, for $k \geq 1$, the exact value of $T_k(n)$ is not immediately obvious.

Although it is clear that the complete graph will allow the fastest possible k-reliable broadcast, the number of edges makes it impractical as a network structure. It would be useful to determine those graphs with the fewest edges among those which allow minimum time k-reliable broadcasting from any originator. The *k-reliable broadcast function*, $B_k(n)$, is the minimum number of edges in any graph G on n vertices with $t_k(G) = T_k(n)$. A *k-reliable minimum broadcast graph* (k-rmbg) is a graph G on n vertices with $B_k(n)$ edges for which $t_k(G) = T_k(n)$. ¿From an applications perspective, these graphs represent the cheapest possible communication networks in which reliable broadcasting can be accomplished, from any vertex, as fast as theoretically possible.

A (0-reliable) *broadcast scheme* for a particular originator u in a graph G is a spanning tree of G rooted at u with a positive integer label on each edge. The label on an edge indicates the time at which the message is sent on that edge. Such a message transmission is termed a *call*. A vertex v learns the message from the originator u if and only if the labelled spanning tree contains a *calling path* from u to v, that is, a path from u to v with increasing labels. The message arrives at v at time t where t is the label on the last edge of the calling path to v. A broadcast scheme is a collection of calling paths, one to each vertex, from the originator. The *broadcast time of a broadcast scheme* is the latest time at which any vertex learns the message.

A k-reliable broadcast scheme must have sufficient redundancy so that a 0-reliable broadcast scheme remains when any k calls are removed. Thus a k-reliable broadcast scheme must contain at least $k + 1$ calling paths to each vertex from the originator. Furthermore, these $k + 1$ calling paths must be pairwise *call disjoint*, that is, no two calling paths use the same edge at the same time. In such a scheme, a vertex v is guaranteed to learn the message by time t if there is a calling path from u to v with final label at most t when at most k calls are removed from the scheme. The broadcast time of this scheme is the latest time at which any vertex is guaranteed to learn the message. Note that the edges used in a k-reliable broadcast scheme do not necessarily form a spanning tree if $k > 0$, and edges can be used to make calls in two directions (at different times, of course). In this paper, we will present k-reliable broadcast schemes by explicitly describing all of the calls.

Our goals in this paper are to determine the time required for k-reliable broadcasting and to find graphs which have smaller numbers of edges than K_n and which allow minimum time k-reliable broadcasting. In Section 3 we determine lower bounds on the time required for k-reliable broadcasting for $k \geq 1$ and we give a 1-reliable broadcast scheme for K_n to show that our bound is exact for $k = 1$. In Section 4, we show that two families of minimum broadcast graphs - the hypercubes and a family of Cayley graphs with $n = 2^m - 2$ vertices, are k-reliable broadcast graphs for any $k \geq 0$, and we prove that the hypercubes are k-rmbg's when $k < n/2$. We extend these results to obtain minimum time k-reliable broadcast schemes for K_n for some other values of n and k. We conclude with a short discussion of open problems in Section 5.

3 Time Required to Broadcast Reliably

Lemma 3.1 *For any $k \geq 0$, $(k+1)n - (k+1)$ calls are necessary for k-reliable broadcasting in any n vertex graph.*

PROOF A 0-reliable broadcast scheme must include at least $n-1$ calls to reach all vertices. For $k > 0$, each vertex other than the originator must have at least $k+1$ incoming calls in any k-reliable broadcast scheme. If some vertex had fewer than $k+1$ incoming calls, then removal of these calls would leave a scheme which is not 0-reliable. □

Theorem 3.2 *For $k \geq 0$,* $T_k(n) \geq \begin{cases} \lfloor \log n \rfloor + 2k + 1 & \textit{for odd } n \\ \lfloor \log n \rfloor + 2k + \lceil \frac{2n-2k-2^{\lfloor \log n \rfloor + 1}}{n} \rceil & \textit{for even } n \end{cases}$

PROOF By Lemma 3.1, the total number of calls must be at least $(n-1)(k+1)$.

At most $2^{\lfloor \log n \rfloor} - 1$ calls can be made in the first $\lfloor \log n \rfloor$ time units, informing at most $2^{\lfloor \log n \rfloor} - 1$ vertices. In each subsequent time unit, at most $\lfloor \frac{n}{2} \rfloor$ calls can be made. Thus, the time required to make $(n-1)(k+1)$ calls is at least $\lfloor \log n \rfloor + \lceil \frac{(n-1)(k+1)-(2^{\lfloor \log n \rfloor}-1)}{\lfloor \frac{n}{2} \rfloor} \rceil$.

When n is odd, this becomes $\lfloor \log n \rfloor + \lceil \frac{2(n-1)(k+1)-2^{\lfloor \log n \rfloor+1}+2)}{n-1} \rceil = \lfloor \log n \rfloor + 2k + \lceil \frac{2n-2^{\lfloor \log n \rfloor+1}}{n-1} \rceil$ $= \lfloor \log n \rfloor + 2k + 1$, since $n - 1 \geq 2n - 2^{\lfloor \log n \rfloor + 1} > 0$ for odd n.

For even n, the bound is $\lfloor \log n \rfloor + \lceil \frac{2(n-1)(k+1)-2^{\lfloor \log n \rfloor+1}+2)}{n} \rceil = \lfloor \log n \rfloor + 2k + \lceil \frac{2n-2k-2^{\lfloor \log n \rfloor+1}}{n} \rceil$. □

Corollary 3.2a $T_k(n) \geq \log n + \lceil \frac{2k(n-1)}{n} \rceil$ *when $n = 2^m$.*

PROOF At most $n-1$ calls can be made in the first $\log n$ time units and at most $\frac{n}{2}$ calls can be made in each subsequent time unit. Thus, the time to complete $(n-1)(k+1)$ calls is at least $\log n + \lceil \frac{(n-1)(k+1)-(n-1)}{\frac{n}{2}} \rceil = \log n + \lceil \frac{2k(n-1)}{n} \rceil$. □

Theorem 3.3 $T_1(n) = \lceil \log n \rceil + 2$.

PROOF $T_1(n) \geq \lceil \log n \rceil + 2$ by Theorem 3.2. To establish the matching upper bound, we construct a 1-reliable broadcast scheme for K_n with broadcast time $\lceil \log n \rceil + 2$ and $2n - 2$ calls. In the case of odd n, create a broadcast tree (i.e., a 0-reliable broadcast scheme for one of the vertices) on n vertices with $\frac{n-1}{2}$ vertices in the subtree rooted at the vertex called at time 1 by the originator (call this subset of vertices A), and the remaining $\frac{n-1}{2}$ vertices in subtrees rooted at vertices called after time 1 by the originator (call this subset of vertices B). Add a perfect matching between A and B. (The originator is left out of this matching.) During time step $\lceil \log n \rceil + 1$, the vertices in A use the edges of the matching to call the vertices in B, and during time step $\lceil \log n \rceil + 2$ the vertices in B call the vertices in A. In the case of even n, create a broadcast tree on n vertices with $\frac{n}{2}$ vertices in the subtree rooted at the vertex called at time 1 (call this subset of vertices A), and $\frac{n}{2} - 1$ vertices in subtrees rooted at vertices called after time 1 (call this subset of vertices plus the originator B). As in the odd case, add a perfect matching between A and B and make calls from A to B and from B to A during time steps $\lceil \log n \rceil + 1$ and $\lceil \log n \rceil + 2$ respectively. □

4 Broadcasting Reliably

We begin this section by showing that hypercubes are k-reliable broadcast graphs for any $k \geq 0$ and that they are k-reliable minimum broadcast graphs for $0 \leq k < \frac{n}{2}$ (where n is the number of vertices). Then we establish that a family of Cayley graphs with $n = 2^m - 2$ vertices is also a family of k-reliable broadcast graphs for any $k \geq 0$. We conclude with extensions of our constructions that give k-reliable broadcast graphs for several other values of n and k.

An m-dimensional hypercube, or m-cube, can be viewed as a set of 2^m vertices with vertex identifiers $\{x_{m-1}x_{m-2}\ldots x_0 | x_i = 0 \text{ or } 1 \text{ for all } 0 \leq i < m\}$ and with an edge between any pair of vertices whose identifiers differ in exactly one bit. Such a pair of vertices whose identifiers differ in only bit x_i are considered to be neighbors in *dimension* i.

Theorem 4.1 *The hypercube on* $n = 2^m$ *vertices allows* k*-reliable broadcasting from any originator in* $\log n + 2k$ *time units.*

PROOF A simple $\log n = m$ time 0-reliable broadcast scheme for the hypercube can be obtained by having each informed vertex call its dimension j neighbor at time $j + 1$ where $0 \leq j \leq m - 1$. That is, if vertex x with identifier $x_{m-1}\ldots x_j \ldots x_0$ learns the message at time $i \leq j$, it informs its neighbor $x_{m-1}\ldots x_{j+1}$ $((x_j + 1) \bmod 2)$ $x_{j-1}\ldots x_0$ at time $j + 1$.

To construct a k-reliable broadcast scheme for $k > 0$, we will add k sets of pairs of calls. At time $m+1$ every vertex $x_{m-1}\ldots x_1 1$ calls $x_{m-1}\ldots x_1 0$ and at time $m+2$ every vertex $x_{m-1}\ldots x_1 0$ calls $x_{m-1}\ldots x_1 1$. That is, each vertex calls its dimension 0 neighbor during one of the two time units. In general, at time $m + 2i + 1$, $0 \leq i \leq k - 1$, $x_{m-1}\ldots x_{q+1} 1 x_{q-1}\ldots x_0$ calls $x_{m-1}\ldots x_{q+1} 0 x_{q-1}\ldots x_0$ where $q \equiv i \pmod m$ and at time $m + 2i + 2$, $0 \leq i \leq k - 1$, $x_{m-1}\ldots x_{q+1} 0 x_{q-1}\ldots x_0$ calls $x_{m-1}\ldots x_{q+1} 1 x_{q-1}\ldots x_0$. That is, each vertex calls its dimension q neighbor during one of the two time units. We now show that this yields a k-reliable broadcast scheme by showing that there are $k + 1$ call disjoint paths from the originator to each other vertex.

One way to view the scheme is as a series of steps, each of which involves calls in a particular dimension. The first m steps correspond to the 0-reliable scheme; note that these steps are 1 time unit long. The k subsequent steps involve calls made in dimension $i \bmod m$ during the pairs of time units $m + 2i + 1$ and $m + 2i + 2$ for $i = 0, 1, 2, \ldots k - 1$. Note that each informed vertex makes exactly one call during each step.

Let T_i^j denote the tree rooted at the vertex v_i called by the originator during step i of the scheme and containing all vertices reached from v_i by paths of calls made in steps $i + 1$ through j

where $j \leq i+m$. V_i^j will be used to denote the set of vertices of T_i^j. Let d_i denote $(i-1) \bmod m$. The identifier of vertex v_i differs from the identifier of the originator only in bit d_i.

To simplify the presentation, we will assume that the originator is $(00\ldots0)$. Then, v_i is $(00\ldots010\ldots0)$ with the 1 in position d_i. The set V_i^{i+m-1} consists of all vertices whose labels contain a 1 in position d_i. In general, the set V_i^j, $j < i+m$ consists of all vertices whose labels contain a 1 in position d_i and 0's in positions $d_i + (j-i) + 1$ through $d_i - 1$ (where, here and throughout the proof, such arithmetic is $\bmod m$).

Consider the set of trees T_i^{i+m-1}, T_{i+1}^{i+m-1}, ..., T_{i+m-1}^{i+m-1}. The union of the sets of vertices in these trees together with the originator contains all n vertices. Furthermore, these vertex sets are pairwise disjoint and do not contain the originator. It follows that these trees are call disjoint.

At step $i+m$, there are two sets of calls across dimension $d_{i+m} = d_i$. The first set of calls is from the set V_i^{i+m-1} of vertices with 1 in bit position d_i to all other vertices in the hypercube (including the originator). Note that after these calls have been made, every vertex in the hypercube has received the message along a path which began with a call from the originator to v_i at step i. The tree T_i^{i+m} is the union of all such paths. This tree is call disjoint from all of the trees T_{i+1}^{i+m-1}, T_{i+2}^{i+m-1}, ..., T_{i+m-1}^{i+m-1}. T_i^{i+m} is also call disjoint from the second set of calls at step $i+m$, that is, the set of calls from the originator and each of the vertices in V_{i+1}^{i+m-1}, V_{i+2}^{i+m-1}, ..., V_{i+m-1}^{i+m-1} to the vertices of V_i^{i+m-1}. Therefore, T_i^{i+m} is call disjoint from the trees T_{i+1}^{i+m}, T_{i+2}^{i+m}, ..., T_{i+m-1}^{i+m}. Since any extensions of the paths of the trees T_{i+1}^{i+m}, T_{i+2}^{i+m}, ..., T_{i+m-1}^{i+m} involve calls that are made after time $i+m$, the tree T_i^{i+m} is call disjoint from any tree $T_{i'}^{i'+m}$, $i' > i$.

The scheme continues through step $m+k$. After the completion of step $m+k$, each vertex has received the message through each of the trees T_i^{i+m}, $1 \leq i \leq k$ and each vertex except the originator has received the message through one of the trees T_{k+1}^{k+m}, T_{k+2}^{k+m}, ...,T_{m+k}^{k+m}. Since all of these trees are call disjoint, the scheme contains $k+1$ call disjoint paths from the originator to each other vertex. □

Corollary 4.1a *$T_k(n) = \log n + 2k$ when $n = 2^m$ and $n > 2k$.*

PROOF ¿From Theorem 4.1, we know that k-reliable broadcasting can be done in $\log n + 2k$ time in the hypercube on $n = 2^m$ vertices. This matches the lower bound of $\log n + \left\lceil \frac{2k(n-1)}{n} \right\rceil$ time units from Theorem 3.2 when $n > 2k$. □

Corollary 4.1b *The hypercube on $n = 2^m$ vertices is a k-rmbg for any k, $0 \leq k < \frac{n}{2}$.*

PROOF Note that each vertex must receive at least $k+1$ calls. To achieve this in $\log n + \left\lceil \frac{2k(n-1)}{n} \right\rceil$ time units requires that $n-1$ calls are made in the first m time units (see proof of Corollary 3.2a) which requires that the originator must have degree at least m. Since the hypercube is regular with degree m and allows a minimum time k-reliable broadcast scheme, it is a k-rmbg. □

Another family of broadcast graphs on which we can broadcast reliably is a family of Cayley graphs with $2^m - 2$ vertices. The Cayley graph from the dihedral group D_{2^m-1} is a graph G_m with degree m, vertex set $V(G_m) = \{x^i,\ \alpha x^i \mid i = 0, \ldots, 2^m-2\}$, and generators αx^{2^i-1}, $i = 0, \ldots, m-1$, where 1 is the identity element, $\alpha^2 = x^{2^m-1} = x^0 = 1$, and $\alpha x \alpha^{-1} = x^{-1}$. We say that two nodes z and y are dimension i neighbors if $y = z\alpha x^{2^i-1}$. This family of graphs was recently shown to be a family of (0-reliable) minimum broadcast graphs [7, 15]. In this paper, we have adopted the notation from [7].

In the $m+1$ time 0-reliable broadcast scheme for G_m presented in [15] and [7], each informed vertex calls its dimension j neighbor at time $j+1$ where $0 \le j \le m-1$, and at time $m+1$ each informed vertex, except the originator and its dimension 0 neighbor, calls its dimension 0 neighbor. We need a more general result about 0-reliable broadcast schemes for G_m before proving that the dihedral Cayley graphs allow minimum time k-reliable broadcasting.

Lemma 4.2 *For each $i \ge 0$,*
a. $\{2^i r \bmod (2^m-1) \mid r = 0,\ldots,2^m-2\} = \{0,\ldots,2^m-2\}$
b. $\{(2^i s - 1) \bmod (2^m-1) \mid s = 1,\ldots,2^m-1\} = \{0,\ldots,2^m-2\}$

PROOF We prove the part **a** by induction on i. The proof of part **b** is similar. For $i=0$, the statement is trivially true. Suppose that the hypothesis holds for $i-1$, that is, $\{2^{i-1} r \bmod (2^m - 1) \mid r = 0,\ldots,2^m-2\} = \{0,\ldots,2^m-2\}$. We can write

$$
\begin{aligned}
&\{2^i r \bmod (2^m-1) \mid r = 0,\ldots,2^m-2\}\\
&\quad= \{2(2^{i-1} r \bmod (2^m-1)) \bmod (2^m-1) | r = 0,\ldots,2^m-2\}\\
&\quad= \{2s \bmod (2^m-1)\ | s = 0,\ldots,2^m-2\} \text{ (by ind. hyp.)}\\
&\quad= \{0,2,\cdots,2(2^{m-1}-1), 2(2^{m-1}) \bmod (2^m-1),\cdots,2(2^m-2) \bmod (2^m-1)\}\\
&\quad= \{0,2,\cdots,2(2^{m-1}-1),1,3,\cdots,2^m-3\}\\
&\quad= \{0,1,\cdots,2^m-2\}
\end{aligned}
$$

□

Lemma 4.3 *The scheme in which each informed vertex of G_m calls its dimension $(i+j) \bmod m$ neighbor at time $j+1$, $0 \le j \le m-1$, and each informed vertex, except the originator and its dimension i neighbor, calls its dimension i neighbor at time $m+1$, is a 0-reliable broadcast scheme for G_m for any $0 \le i \le m-1$.*

PROOF We will prove the result assuming that the originator is the identity element 1. The vertex-transitivity of Cayley graphs guarantees that the scheme can be used for any originator. Let I_j be the set of all informed vertices at time j.

$$
\begin{aligned}
I_0 &= \{x^0 = 1\}\\
I_1 &= \{1, \alpha x^{2^i-1}\}\\
I_2 &= I_1 \cup \{\alpha x^{2^{i+1}-1}, x^{2^i}\}\\
&= \{1, x^{2^i}, \alpha x^{2^i-1}, \alpha x^{2\cdot 2^i-1}\}\\
&\ \vdots\\
I_j &= I_{j-1} \cup I_{j-1}\cdot \alpha x^{2^{i+j-1}-1}\\
&= \bigcup_{k=0}^{2^{j-2}-1} \{x^{k\cdot 2^i}, \alpha x^{(k+1)\cdot 2^i-1}, x^{k\cdot 2^i}\cdot \alpha x^{2^{i+j-1}-1}, \alpha x^{(k+1)\cdot 2^i-1}\cdot \alpha x^{2^{i+j-1}-1}\}\\
&= \bigcup_{k=0}^{2^{j-2}-1} \{x^{k\cdot 2^i}, \alpha x^{(k+1)\cdot 2^i-1}, \alpha x^{(2^{i+j-1}-1)-k\cdot 2^i}, x^{(2^{i+j-1}-1)-((k+1)\cdot 2^i-1)}\}
\end{aligned}
$$

$$
\begin{aligned}
&= \bigcup_{k=0}^{2^{j-1}-1} \{x^{k\cdot 2^i}, \alpha x^{(k+1)\cdot 2^i-1}\} \\
&\vdots \\
I_{m+1} &= I_m \cup I_m \cdot \alpha x^{2^i-1} \\
&= \bigcup_{k=0}^{2^{m-1}-1} \{x^{k\cdot 2^i}, \alpha x^{(k+1)\cdot 2^i-1}, \alpha x^{(2^i-1)-k\cdot 2^i}, x^{(2^i-1)-((k+1)\cdot 2^i-1)}\}
\end{aligned}
$$

Using $x^{2^m-1}=1$, we get

$$
\begin{aligned}
I_{m+1} &= \bigcup_{k=0}^{2^{m-1}-1} \{x^{k\cdot 2^i}, \alpha x^{(k+1)\cdot 2^i-1}, \alpha x^{(2^m-1)+(2^i-1)-k\cdot 2^i}, x^{(2^m-1)+(2^i-1)-((k+1)\cdot 2^i-1)}\} \\
&= \bigcup_{k=0}^{2^m-1} \{x^{k\cdot 2^i}, \alpha x^{(k+1)\cdot 2^i-1}\} \\
&= \bigcup_{k=0}^{2^m-2} \{x^{k\cdot 2^i}, \alpha x^{(k+1)\cdot 2^i-1}\}
\end{aligned}
$$

By Lemma 4.2, $I_{m+1} = \{x^k, \alpha x^k \mid k = 0, 1, \ldots, 2^m-2\} = V(G_m)$. □

Theorem 4.4 *The dihedral Cayley graph on $n = 2^{m+1}-2$ vertices allows k-reliable broadcasting from any originator in $\lceil \log n \rceil + 2k$ time units.*

PROOF This proof is similar to the proof for hypercubes although the details here are somewhat more involved. We construct a k-reliable broadcast scheme for $k > 0$, by adding k sets of pairs of calls to a 0-reliable broadcast scheme and then show that there are $k+1$ sets of call disjoint paths from the originator to all other vertices. We will prove the result when the originator is the identity element. Vertex-transitivity implies the result for other originators.

Start with the $m+1$ time 0-reliable broadcast scheme for vertex 1 which uses dimension i (generator αx^{2^i-1}) at time $i+1$, $0 \le i \le m-1$, and dimension 0 again at time $m+1$. The remaining k steps are each 2 time units long and continue to rotate through the dimensions in the same order as the 0-reliable scheme. In particular, during the first time unit of step $m+1+i$, $1 \le i \le k$, every vertex x^j calls $x^j \alpha x^{2^i-1} = \alpha x^{2^i-1-j}$, $j = 0, \ldots, 2^m-2$, and during the second time unit every vertex αx^j calls $\alpha x^j \alpha x^{2^i-1} = x^{2^i-1-j}$. Note that each informed vertex makes exactly one call during each step.

Let T_i^j denote the tree rooted at the vertex $\alpha x^{2^{i-1}-1}$ called by the originator during step i of the scheme and containing all vertices reached from $\alpha x^{2^{i-1}-1}$ by paths of calls made in steps $i+1$ through j where $j \le i+m+1$. V_i^j is the set of vertices of T_i^j. Our $k+1$ sets of call disjoint paths from the originator to all other vertices will be based on the trees T_i^{i+m+1}, $1 \le i \le k$, and the collection of trees $\bigcup_{i=k+1}^{k+m} T_i^{k+m+1}$. By Lemma 4.3 and vertex-transitivity, each of the trees T_i^{i+m+1}, $1 \le i \le k$ is a broadcast scheme for G_m with originator $\alpha x^{2^{i-1}-1}$ (with a redundant call to the dimension i neighbor of $\alpha x^{2^{i-1}-1}$ during step $i+m+1$). As we will see, the tree with root (originator) 1 and subtrees T_i^{k+m+1}, $k+1 \le i \le k+m$ is also a broadcast scheme for G_m (and also contains a redundant call during step $k+m+1$). Unfortunately, this set of trees is not call disjoint, so we will have to modify them.

Consider the tree T with root 1 and subtrees T_i^{i+m}, T_{i+1}^{i+m}, ..., T_{i+m-1}^{i+m} where $1 \leq i \leq k$. By the same reasoning as in the proof of Lemma 4.3, T contains all vertices of G_m. Notice that T_i^{i+m} contains an unnecessary call from its root $\alpha x^{2^{i-1}-1}$ to the originator 1 during step $i+m$. Let S_i^{i+m} denote the tree obtained from T_i^{i+m} by removing the unnecessary call to 1. By Lemma 4.3, the tree T' with root 1 and subtrees S_i^{i+m}, T_{i+1}^{i+m}, ..., T_{i+m-1}^{i+m} is a broadcast scheme for G_m with originator 1. Furthermore, the sets of vertices of the trees S_i^{i+m}, T_{i+1}^{i+m}, ..., T_{i+m-1}^{i+m} are pairwise disjoint so these trees are pairwise call disjoint.

During the two time units of step $i+m+1$, every vertex calls its neighbor in dimension $(i+m) \bmod m$. Clearly these calls at step $i+m+1$ are disjoint from the calls in S_i^{i+m}, T_{i+1}^{i+m}, ..., T_{i+m-1}^{i+m}. Let S_i^{i+m+1} denote the tree S_i^{i+m} extended to step $i+m+1$ by including the calls from vertices in S_i^{i+m} to their dimension $(i+m) \bmod m$ neighbors. We know that T_i^{i+m+1} contains all vertices of G_m so the tree S_i^{i+m+1} contains all vertices of G_m except 1 and the vertex αx^{2^i-1} which is the dimension $(i+m) \bmod m$ neighbor of 1. Put another way, to every vertex, except 1 and αx^{2^i-1}, there is a path in S_i^{i+m+1} which began with a call from the originator to $\alpha x^{2^{i-1}-1}$ at step i and S_i^{i+m+1} is call disjoint from the trees T_{i+1}^{i+m+1}, ..., T_{i+m-1}^{i+m+1}. The tree T_{i+m+1}^{i+m+1} consists of one vertex, αx^{2^i-1}, which is called by 1 in step $i+m+1$. S_i^{i+m+1} does not contain this call, so S_i^{i+m+1} is also call disjoint from T_{i+m+1}^{i+m+1}. Unfortunately, S_i^{i+m+1} is not call disjoint from T_{i+m}^{i+m+1} since both trees contain a call from $\alpha x^{2^{i-1}-1}$ to its dimension $(i+m) \bmod m$ neighbor in step $i+m+1$. However, since S_i^{i+m+1} contains a call from $\alpha x^{2^{i-1}-1}$ to this same neighbor in step $i+1$, the call in step $i+m+1$ is not needed by S_i^{i+m+1} and can be removed to obtain a tree R_i^{i+m+1}. R_i^{i+m+1} does not contain any calls after step $i+m+1$ so it is call disjoint from every tree $T_{i'}^{i'+m+1}$ with $i' > i$.

We can repeat this construction to obtain the k call disjoint trees R_i^{i+m+1}, $1 \leq i \leq k$. Each of these trees is missing the originator 1 and one other vertex. In particular, R_i^{i+m+1} does not contain a call to αx^{2^i-1}, the dimension $(i+m) \bmod m$ neighbor or 1. Notice that vertex 1 calls αx^{2^i-1} during the first time unit of step $i+m$, but both vertices are idle during the second time unit because all useless calls from αx^{2^i-1} to 1 have been removed. We can therefore add a second call from 1 to αx^{2^i-1} during step $i+m$ for $i = 1, 2, ..., k$ to complete the k sets of call disjoint paths from the originator to all other vertices. By Lemma 4.3, the trees T_i^{k+m+1}, $k+1 \leq i \leq k+m$ provide the $k+1^{st}$ set of call disjoint paths.

In summary, the k-reliable broadcast scheme consists of $k+m+1$ steps. The first $m+1$ steps have one time unit each and the remaining k steps have two time units. During each step $i+1$, each informed vertex, except the originator and its dimension i neighbor, calls its dimension i neighbor. During step $i+1$, the originator calls its dimension i neighbor once if $0 \leq i \leq m$ and twice if $m+1 \leq i \leq k+m$. □

Corollary 4.4a *$T_k(n) = \lceil \log n \rceil + 2k$ when $n = 2^m - 2$ and $2(k+1) < n$.*

PROOF The lower bound from Theorem 3.2 matches the upper bound from Theorem 4.4 when $2(k+1) < n$. □

We now show how to obtain k-reliable broadcast schemes for some additional values of n and k by augmenting some of the broadcast schemes above. We can augment the broadcast scheme for hypercubes from Theorem 4.1 to obtain a $\lfloor \log n \rfloor + 2k + 1$ time k-reliable broadcast scheme for several values of n and k. We will do this separately for even and odd values of n.

Lemma 4.5 *There exists a* $\lfloor \log n \rfloor + 2k + 1$ *time* k*-reliable broadcast scheme for* K_n *for any* $n = 2^m + 2i$ *and* $k \leq 2i - 1 \leq \lfloor \log n \rfloor$.

PROOF We obtain such a scheme by augmenting the scheme for 2^m vertices from Theorem 4.1. Start by choosing a subgraph H of K_n that includes the originator and that is isomorphic to the hypercube on 2^m vertices. Let K_{2i} be the (complete) subgraph of K_n that contains the $2i$ vertices not in H. Perform a 0-reliable broadcast in H using the first m steps of the scheme in the proof of Theorem 4.1. Next, choose $2i$ vertices from among the originator and the m vertices of H that it called at times 1 through m. At time $m+1$, each of these $2i$ chosen vertices calls one of the $2i$ uninformed vertices of K_{2i}. This extends the 0-reliable broadcast scheme to all $2^m + 2i$ vertices of K_n.

It remains to complete k additional call disjoint paths to each vertex. The 2^m vertices of H and the $2i$ vertices of K_{2i} do this independently. Partition the time units $m+2, m+3, \ldots, m+2k+1$ into k steps of two time units each as in the scheme of Theorem 4.1. The vertices of H complete a k-reliable hypercube scheme using steps $m+1$ through $m+k$ from the scheme in the proof of Theorem 4.1. Now, note that the paths in the extended 0-reliable scheme that inform the $2i$ vertices of K_{2i} are edge disjoint. To complete the k additional paths to any particular vertex of K_{2i} we only need to ensure that it receives calls from k other vertices of K_{2i}. Since the edges of any complete graph on $2i$ vertices can be partitioned into $2i-1$ 1-factors, we can complete up to $2i-1$ additional paths to each vertex of K_{2i} by making two calls (one in each direction) between each pair of vertices in one of these 1-factors during each of the last k steps. □

Corollary 4.5a $T_k(n) = \lfloor \log n \rfloor + 2k + 1$ *for any* $n = 2^m + 2i$ *and* $k \leq 2i - 1 \leq \lfloor \log n \rfloor$.

PROOF The time achieved in the scheme of Lemma 4.5 matches the lower bound from Theorem 3.2 □

We can modify the scheme of Lemma 4.5 to obtain a similar scheme for odd values of n.

Lemma 4.6 *There exists a* $\lfloor \log n \rfloor + 2k + 1$ *time* k*-reliable broadcast scheme for* K_n *for any* $n = 2^m + 2i + 1 > 5$ *and* $k \leq 2i - 1 \leq \lfloor \log n \rfloor$.

PROOF Observe that in the scheme of Lemma 4.5, the originator receives calls in each of the last k steps. Since these are wasted calls, we can omit them and let the originator make other calls during these steps. To obtain a scheme for $n = 2^m + 2i + 1$ vertices, we can augment the scheme for $2^m + 2i$ vertices from Lemma 4.5 by adding calls to inform one additional vertex x. In particular, any of the first 2^m vertices of H which is idle at time $m+1$ calls x at time $m+1$. In each of the subsequent k steps, the originator calls x. This completes $k+1$ call disjoint paths to x. □

Corollary 4.6a $T_k(n) = \lfloor \log n \rfloor + 2k + 1$ *for any* $n = 2^m + 2i + 1 > 5$ *and* $k \leq 2i - 1 \leq \lfloor \log n \rfloor$.

PROOF The time achieved in the scheme of Lemma 4.6 matches the lower bound from Theorem 3.2 □

The idea used in the schemes of Lemma 4.5 and Lemma 4.6 can be extended slightly. During time unit $m+1$, additional new vertices could be called from the first 2^m vertices. However, the number of vertices used from each of the m trees T_1^m, T_2^m, ...,T_m^m restricts the number of matchings that can be used to form call disjoint paths. Hence, such extensions can only be done for relatively small values of k.

For some values of n and k, Lemma 4.5 and Lemma 4.6 do not apply, but we can still exploit the observation that the originator unnecessarily receives calls from other vertices in the scheme of Theorem 4.1.

For example, when $n = 10$ and $k = 2$, our lower bound says that we need at least 27 calls and at least 7 time units. Furthermore, at most 7 calls can be made in the first 3 time units, so we must make 5 calls in each of the last 4 time units. Lemma 4.5 does not apply because $k > 2i - 1$. The construction in Lemma 4.5 gives a 2-reliable broadcast scheme, but the scheme requires 8 time units. To accomplish 2-reliable broadcasting in 7 time units, we will reorder the calls that occur in k sets of pairs during the last $2k$ time units of the scheme from the construction in Lemma 4.5. When the unnecessary calls to the originator are eliminated from this modified scheme, there is enough room in the last $2k$ time units for the two calls that are made at time $m+1$ in the scheme from the construction in Lemma 4.5.

Start by choosing a subgraph H isomorphic to a hypercube on 8 vertices and perform a 0-reliable broadcast scheme. To simplify the following description, assume that the originator is vertex 000, and let 001, 010, and 100 be the vertices called by the originator at times 1, 2, and 3. At time 4, the vertices of H with a 1 in bit position 0 call their neighbors in dimension 0, except 001 which calls a vertex u that is not in H instead of calling the originator. Since the originator is not being called by 001, it is free at time 4 to call the other vertex v that is not in H. Similarly, at time 5, the vertices of H with a 1 in bit position 1 call their neighbors in dimension 1, except 010 which calls v instead of the originator. This leaves the originator free to call u at time 5. At times 6 and 7, u and v exchange calls, while the vertices of H with 0's in positions 0 and 1, respectively, call their neighbors in those dimensions.

A similar scheme can be found for $n = 10$ and $k = 3$. In this case, the lower bound of 9 time units is achievable by reordering the calls in the last $3k$ time units of the scheme from the construction in Lemma 4.5. In a similar manner to the previous example, calls from vertices of H with 1's in positions 0, 1, and 2 are made at times 4, 5, and 6, respectively, and calls from vertices with 0's in positions 0, 1, and 2 are made at times 7, 8, and 9, respectively. Vertex u receives calls from 001, 000, and 100 at times 4, 5, and 6. Vertex v receives calls from 000 at times 4 and 6 and from 010 at time 5. Vertices u and v exchange calls at times 7 and 8 and are idle at time 9.

5 Discussion

In this paper, we have investigated some of the implications for broadcasting of transmission failures of short duration. We have proved lower bounds on the time to broadcast reliably in the presence of k faults and have constructed k-reliable broadcast schemes that achieve these bounds for various values of n and k. Our most general constructions provide minimum time k-reliable broadcast schemes for $n = 2^m$ and $n = 2^m - 2$ for any k. The constructions use the family of hypercubes and a family of Cayley graphs based on dihedral groups. The hypercubes are the only non-trivial k-rmbg's that we have found. We believe that the Cayley graphs with $2^m - 2$ vertices are also k-rmbg's, but we have not found a proof.

It would be interesting to find k-rmbg's, or k-reliable graphs with small numbers of edges, for other values of n. In general, it seems to be difficult to verify whether or not a given graph is k-reliable for all k (or a large range of values of k), especially if the graph is not vertex-transitive. The star graphs are proposed in [1] as good alternatives to hypercubes for interprocessor communication. Like the hypercubes, the star graphs are Cayley graphs which are edge-transitive. The edgetransitivity makes it easy to modify the construction of Theorem 4.1 to find k-reliable broadcast schemes for star graphs for many values of k. Unfortunately, minimum time k-reliable broadcasting is impossible in star graphs, even for $k = 0$. This is because star graphs have $n = m!$ vertices and degree $m - 1$, and this degree is too small to permit 0-reliable broadcasting in $\log(m!)$ time. It would be interesting to determine the minimum time for k-reliable broadcasting in other popular interconnection networks such as de Bruijn graphs and shuffle-exchange graphs.

We close with the following conjecture.

Conjecture 5.1 *There are no values of n and k such that $B_{k+1}(n) < B_k(n)$. That is, there are no values of n and k such that a $k+1$-rmbg with n vertices has fewer edges than a k-rmbg with n vertices.*

References

[1] S. Akers and B. Krishnamurthy, A Group-Theoretic Model for Symmetric interconnection Networks, *IEEE Trans. on Computers* 38, pp. 555-566, 1989.

[2] A. Bagchi and S. L. Hakimi, Information Dissemination in Distributed Networks with Faulty Units, manuscript, 1990.

[3] K. A. Berman and M. Hawrylycz, Telephone Problems with Failures, *SIAM J. Alg. Disc. Meth.* 7, pp. 13-17, 1986.

[4] D. Bienstock, Broadcasting with Random Faults, *Disc. Appl. Math.* 20, pp. 1-7, 1988.

[5] J. Bruck, Optimal Broadcasting in Faulty Hypercubes via Edge-Disjoint Embeddings, tech report RJ7147 (67394) *IBM Almaden*,1989.

[6] S.-C. Chau and A. L. Liestman, Constructing Fault-tolerant Minimal Broadcast Networks, *J. Comb. Inf. and. Sci.* 11, pp. 1-18, 1986.

[7] M. Dinneen, M. Fellows, and V. Faber, Algebraic Constructions of Efficient Broadcast Networks, *Ninth Int. Symp. on Applied Algebra, Algebraic Algorithms and Error Correcting Codes*, Lecture Notes in Computer Science 539, pp. 152-158, 1991.

[8] A. Farley, Reliable Minimum-Time Broadcast Networks, *Proc. Eighteenth SE Conf. on Combinatorics, Graph Theory and Computing*, Utilitas Mathematica, 1987.

[9] A. Farley and A. Proskurowski, Networks Immune to Isolated Failures, *Networks* 11, pp. 255-268, 1981.

[10] P. Fraigniaud, Asymptotically Optimal Broadcasting and Gossiping in Faulty Hypercube Multicomputers, *IEEE Trans. Comp.*, to appear.

[11] P. Fraigniaud and C. Peyrat, Broadcasting in a Hypercube When Some Calls Can Fail, *Inf. Proc. Lett.* 39, pp. 115-119, 1991.

[12] L. Gargano, Tighter Time Bounds on Fault Tolerant Broadcasting and Gossiping, *Networks* 22, pp. 469-486, 1992.

[13] L. Gargano and U. Vaccaro, Minimum Time Broadcast Networks Tolerating a Logarithmic Number of Faults, *SIAM J. Discr. Math.* 5, pp. 178-198, 1992.

[14] S. T. Hedetniemi, S. M. Hedetniemi, and A. L. Liestman, A Survey of Broadcasting and Gossiping in Communication Networks, *Networks* 18, pp. 319-349, 1988.

[15] L. H. Khachatrian and O. S. Harutounian, Construction of New Classes of Minimal Broadcast Networks. *Conference on Coding Theory*, Armenia, 1990.

[16] A. L. Liestman, Fault-tolerant Broadcast Graphs, *Networks* 15, pp 159-171, 1985.

[17] G. Maddaluno, Algoritmi per la Costruzione di Networks Fault-tolerant, M.Sc. thesis, Universita di Salerno, 1987.

[18] A. Pelc, Broadcasting in Complete Networks with Faulty Nodes using Unreliable Calls, *Inf. Proc. Lett.* 40, pp. 169-174, 1991.

[19] D. Peleg and A. A. Schäffer, Time Bounds on Fault-Tolerant Broadcasting, *Networks* 19, pp 803-822, 1989.

[20] P. Ramanathan and K. G. Shin, Reliable Broadcasting in Hypercube Multicomputers, *IEEE Trans. Comp.* 37, pp. 1654-1657, 1988.