Neighbourhood Broadcasting in Hypercubes *

Jean-Claude Bermond,† **Afonso Ferreira,**†‡ **Stéphane Pérennes** †
Project MASCOTTE, CNRS - UNSA - INRIA
2004 Route des Lucioles, BP 93
F-06902 Sophia-Antipolis Cedex, France

Joseph G. Peters §
School of Computing Science
Simon Fraser University
Burnaby, British Columbia, Canada, V5A 1S6

Revised: July 2006

Abstract

In the broadcasting problem, one node needs to broadcast a message to all other nodes in a network. If nodes can only communicate with one neighbour at a time, broadcasting takes at least $\lceil \log_2 N \rceil$ rounds in a network of N nodes. In the *neighbourhood broadcasting* problem, the node that is broadcasting only needs to inform its neighbours. In a binary hypercube with N nodes, each node has $\log_2 N$ neighbours, so neighbourhood broadcasting takes at least $\lceil \log_2 \log_2(N+1) \rceil$ rounds. In this paper, we present asymptotically optimal neighbourhood broadcast protocols for binary hypercubes.

Keywords: broadcasting, hypercubes, neighbourhood communication

---

*This research was done while the first author was visiting Simon Fraser University and while the fourth author was visiting Project MASCOTTE - Sophia-Antipolis.

†Supported by the European Project FET IST AEOLUS.

‡On leave from CNRS. Currently Science Officer for Telecommunications & IST at COST, Brussels.

§Supported by NSERC of Canada, Université de Nice - Sophia-Antipolis, CNRS, and INRIA.

1 Introduction

In the broadcasting problem, a single *originator* is required to disseminate a piece of information to all other nodes of a network (modelled as a graph) as quickly as possible. In the *unit-cost single-port* communication model, each message transmission requires one time unit or *round*, and each node can communicate with at most one adjacent node (*neighbour*) at any given time. It is well-known that broadcasting in an n-dimensional binary hypercube, or *n-cube*, under this model requires $n = \log_2 N$ rounds of communication to inform all $N = 2^n$ nodes and that this is optimal. In this paper, we address a variant of this problem called *neighbourhood broadcasting* in which the originator only needs to inform its n neighbours in a hypercube. We show that this can be accomplished exponentially faster than normal (complete) broadcasting. A lower bound on the number of rounds for a neighbourhood broadcast is $\lceil \log_2(n+1) \rceil = \lceil \log_2 \log_2(N+1) \rceil$. We present two neighbourhood broadcast protocols and prove that the second protocol achieves the lower bound asymptotically. More precisely, we prove that a neighbourhood broadcast can be completed in at most $log_2 n + \lceil \sqrt{2 log_2 n} \rceil$ rounds (so the ratio of the upper bound for the second protocol and the lower bound tends to 1 as n tends to infinity). The exact analyses of our protocols are difficult, so, for each protocol, we introduce a sequence of truncated protocols and prove that their performances approach the lower bound.

The neighbourhood broadcasting problem was introduced by Cosnard and Ferreira [3] who outlined a simple $O(\log_2 n)$ protocol. They proved that the number of neighbours of the originator informed by their protocol after t rounds satisfies a Fibonacci recurrence and is proportional to $(1.618)^t$. Thus, the number of rounds to complete a neighbourhood broadcast using their protocol is proportional to $1.4404 \log_2 n$. In Section 2, we generalize the protocol from [3] to obtain the first of our new protocols called Protocol **A**. We were unable to find a closed form expression for the performance of Protocol **A**, but we can give generalized Fibonacci recurrence relations for truncated versions of Protocol **A**. The truncated protocol $\mathbf{A}_k$, $k \geq 2$, is obtained from Protocol **A** by discarding all communications that involve a node at distance greater than k from the originator. Protocol $\mathbf{A}_2$ is the protocol from [3]. For Protocol $\mathbf{A}_3$, the number of neighbours of the originator informed after t rounds is proportional to $(1.839)^t$, for Protocol $\mathbf{A}_4$ it is proportional to $(1.913)^t$, and for Protocol $\mathbf{A}_{12}$ it is $(1.991)^t$.

In Section 3, we describe and analyze a more sophisticated, and more efficient, protocol called Protocol **B**. We show that for any fixed $\epsilon > 0$ and sufficiently large t, the number of neighbours of the originator informed after t rounds of Protocol **B** is at least $(2-\epsilon)^t$. We also derive recurrence relations for the truncated protocols $\mathbf{B}_k$, $k \geq 2$. For example, the number of neighbours of the originator informed after t rounds of Protocol $\mathbf{B}_5$ is proportional to $(1.999)^t$. We think that Protocol **B** is not just asymptotically optimal, but that it is optimal or near-optimal in the sense that no protocol can inform the neighbours of the originator faster. Unfortunately, our attempts to significantly improve the lower bound have not succeeded, so improving the lower bound and determining the optimal performance exactly remain as open problems.

The protocol in Section 2 was first presented at a workshop in 1991 [1], including the closed form solution for a truncated version of the protocol and empirical evidence that the (un-truncated) protocol is asymptotically optimal. An incomplete manuscript [2] of the present paper, including the protocols in Sections 2 and 3 and parts of the analysis, has been in circulation since 1998. The workshop presentation and the manuscript have stimulated considerable interest in neighbourhood communication problems [6, 7, 10, 11, 12, 13, 16, 19, 20].

Hypercubes are Cayley graphs and many of the ideas in this paper can be modified or extended to other classes of Cayley graphs such as *star graphs*, which are Cayley graphs on permutation groups. The first bounds for broadcasting in star graphs appeared in [10]. The bounds were improved in [19], and an alternative protocol (with a weaker bound) was presented in [20]. The best current upper bounds for neighbourhood broadcasting in star graphs are $1.3125 \log_2 n + \mathrm{O}(\log_2 \log_2 \mathrm{n})$ [11] and $\log_2 n + \mathrm{O}(\sqrt{\log_2 \mathrm{n}})$ [12]. A larger class of Cayley graphs on permutation groups is studied in [16].

Neighbourhood gossiping in hypercubes, was studied in [13]. In the neighbourhood gossiping problem, each node starts with a unique piece of information and must learn the information of all of its neighbours. Normal (complete) gossiping in an n-cube takes at least $1.44n+\mathrm{O}(1)$ rounds [4, 18] and at most $1.88n+\mathrm{O}(1)$ rounds [17] using *half-duplex* links, and exactly n rounds using *full-duplex* (i.e., bi-directional) links (see [14]). The bounds in [13] on the numbers of rounds, $h_1(n)$ and $h_2(n)$, for half-duplex and full-duplex neighbourhood gossiping in an n-cube respectively, are $2.88 \log_2 n + \mathrm{O}(1) \leq \mathrm{h}_1(\mathrm{n}) \leq 3.76 \log_2 \mathrm{n} + \mathrm{O}(1)$ and $h_2(n) = 2 \log_2 n + \mathrm{O}(1)$. The ideas in [13] were extended to star graphs in [10]. Note that while the distinction between the half-duplex and full-duplex links is important for gossiping problems, it can be ignored for broadcasting problems because the (single) message in a broadcast protocol never needs to traverse any link in both directions.

In *k-neighbourhood communication* problems, nodes that are at distance at most k are required to communicate. The neighbourhood broadcasting and gossiping problems are examples of 1-neighbourhood communication. Bounds for k-neighbourhood broadcasting and gossiping in paths, trees, cycles, 2-dimensional grids, and 2-dimensional tori were derived in [6, 7]. The results are optimal in most cases and within an additive constant of optimal in the other cases.

There are many papers describing protocols that minimize the time for a normal (complete) broadcast on various interconnection networks such as hypercubes and meshes. See [15] for a discussion of models and results for broadcasting and gossiping with unit-cost models and [5, 14] for comprehensive surveys.

2 A Simple Protocol

In Cosnard and Ferreira's neighbourhood broadcast protocol [3], the originator in a hypercube sends its message to a new neighbour during each round. Each informed neighbour of the originator broadcasts to its neighbours (except the originator, of course). These neighbours of the neighbours do not need to know the message, but each of them can inform one new neighbour of the originator. It is not difficult to show directly that this protocol takes $O(\log_2 n)$ rounds to inform all neighbours of the originator in an n-cube, but we will take the opportunity to introduce some notation that we will use to analyze our new protocols.

We will identify each vertex in an n-cube by a binary string of length n. Without loss of generality, the originator is labelled with a string of n 0's: $00\ldots00$. Each neighbour of the originator has exactly one 1 in its label. Each neighbour of a neighbour of the originator (except the originator) has two 1's in its label. In general, a node at (Hamming) distance k from the originator has exactly k 1's in its label. We will say that nodes at distance k from the originator are at *level* k. In the neighbourhood broadcasting problem, all level 1 nodes must be informed, and we want to do this as quickly as possible.

It will often be convenient to have a compact way to write node labels. When we write $\delta_1\delta_2\delta_3\delta_4$, $\delta_1 < \delta_2 < \delta_3 < \delta_4$, we mean that the label contains 1's in the indicated positions and 0's in all other positions, so this is a level 4 node. The label $\delta_1\bar{\delta_2}\delta_3$ has 1's in positions δ_1 and δ_3, a 0 in position δ_2, and 0's elsewhere, so this is a level 2 node. We will sometimes insert commas into labels to avoid ambiguity For example, 1,4,21 is the level 3 node shown in Figure 1 with 1's in positions 1, 4, and 21.

In our figures, we will draw the originator on the left and Hamming distance from the originator will increase from left to right. When we say that a node is informed *from the left* or *from the right*, we are referring to this left to right arrangement of increasing levels.

To analyze our protocols, we use the following notation:

$L_k^t(\mathbf{P})$	maximum number of level k nodes informed by level $k-1$ nodes (i.e., from the left) during round t of Protocol $\mathbf{P}$
$R_k^t(\mathbf{P})$	maximum number of level k nodes informed by level $k+1$ nodes (i.e., from the right) during round t of Protocol $\mathbf{P}$
$N_k^t(\mathbf{P})$	$= L_k^t(\mathbf{P}) + R_k^t(\mathbf{P})$: maximum total number of level k nodes informed during round t of Protocol $\mathbf{P}$
$T_k^t(\mathbf{P})$	$= \sum_{i=1}^{t} N_k^i(\mathbf{P})$: maximum total number of level k nodes informed during the first t rounds of Protocol $\mathbf{P}$

We will often omit the name of the protocol to simplify the notation when the protocol $\mathbf{P}$ is clear from the context.

In the analyses of our protocols, we will show several things. For each protocol $\mathbf{P}$, we will develop recurrence relations for $T_k^t(\mathbf{P})$. The value of $T_k^t(\mathbf{P})$ is an upper bound on the number of informed level k nodes after t rounds of Protocol $\mathbf{P}$. To prove that Protocol $\mathbf{P}$ achieves these bounds, we need to show that it informs *exactly* $T_k^t(\mathbf{P})$ level k nodes during the first t rounds. We do this by showing that all newly informed nodes are distinct and that all level 1 nodes are eventually informed. We will then determine the rate at which Protocol $\mathbf{P}$ informs level 1 nodes as a function of t. We do this by determining the value of the largest root a_k of the associated polynomial of the recurrence relation $T_1^t(\mathbf{P})$. The number of level 1 nodes informed by Protocol $\mathbf{P}$ is proportional to a_k^t.

We will begin by considering the protocol from [3]. We will call this Protocol $\mathbf{A}_2$ because it is a truncated version of Protocol $\mathbf{A}$, the first of our new protocols which we will introduce later in this section. If x is a node that is informed during round t of Protocol $\mathbf{A}_2$, then x informs uninformed nodes as follows:

Protocol $\mathbf{A}_2$ [3]

- If x is the originator, inform type L_1 nodes during rounds $t+1, t+2, \ldots$
- If x is a level 1 node, inform type L_2 nodes during rounds $t+1, t+2, \ldots$
- If x is a level 2 node, inform a type R_1 node during round $t+1$

The next theorem and corollary from [3] are restated using our notation.

Theorem 1 [3] $T_1^t(\mathbf{A}_2) = T_1^{t-1}(\mathbf{A}_2) + T_1^{t-2}(\mathbf{A}_2) + 1.$

Proof: First, we get $L_1^t = 1$, $t \geq 1$ because the originator informs one neighbour during each round. We also have $L_2^t = T_1^{t-1}$, $t \geq 2$, because each level 1 node that was informed during the first $t-1$ rounds can potentially inform a new level 2 node during round t. Finally, $R_1^t = L_2^{t-1}$, $t \geq 3$, because each informed level 2 node can potentially inform one new neighbour of the originator immediately after it receives the message. Thus, $R_1^t = T_1^{t-2}$, and for $t \geq 3$, we can write $T_1^t = T_1^{t-1} + N_1^t = T_1^{t-1} + L_1^t + R_1^t = T_1^{t-1} + T_1^{t-2} + 1$. □

Corollary 1 [3] $T_1^t(\mathbf{A}_2) \sim 1.618^t$.

Proof: Since $T_1^1 = 1$ and $T_1^2 = 2$, we get $T_1^t = F_{t+2} - 1$, where F_i is the i^{th} Fibonacci number (with starting values $F_1 = F_2 = 1$). The associated polynomial of T_1^t is $x^2 - x - 1 = 0$ and its largest root is $a_2 = \frac{1+\sqrt{5}}{2}$. It follows that the potential number of informed neighbours of the originator after t rounds is proportional to $\left(\frac{1+\sqrt{5}}{2}\right)^t \sim 1.618^t$. □

To show that the bound of Theorem 1 can be attained, we need to show that every level 1 node is informed and that no nodes are informed more than once. To do this, we have to specify which nodes are informed during each round. We use the following method: During round t, the originator (which we will refer to as node 0) will inform node $T_1^{t-1}+1$ at level 1 (i.e., the node whose label has a 1 in position $T_1^{t-1}+1$), and any level 1 node δ, $1 \leq \delta \leq T_1^{t-1}$, that was informed during the first $t-1$ rounds will inform node $\delta, \delta + T_1^t + 1$ at level 2 if $\delta + T_1^t + 1 \leq n$. If $\delta + T_1^t + 1 > n$, then we can assume that node δ is idle because communications to the right will not result in any more informed level 1 nodes before the end of the protocol. Then, during round $t+1$, each level 2 node $\delta, \delta + T_1^t + 1$ that was informed during round t will inform node $\delta + T_1^t + 1$ at level 1. Figure 1 shows how this can be done for $n \leq T_1^6 = 20$. (In Figure 1, the three bold arcs and the nodes with 21 in their labels are not part of Protocol $\mathbf{A}_2$ and should be ignored at this point.) The following lemma establishes the correctness of this pattern.

Lemma 1 *All level 1 nodes δ with $1 \leq \delta \leq \min(n, T_1^t(\mathbf{A}_2))$ are informed in t rounds.*

Proof: By induction. The claim is true for $t = 1$ and $t = 2$. Now, suppose that the claim is true after round t. If $n \leq T_1^t$, we are done. If $n > T_1^t$, then the new level 1 nodes informed during round $t+1$ are node $T_1^t + 1$ which is informed by node 0, and all nodes δ with $T_1^t + 2 \leq \delta \leq \min(n, T_1^t + T_1^{t-1} + 1)$ which are informed by the level 2 nodes $\delta, \delta + T_1^t + 1$ with $1 \leq \delta \leq T_1^{t-1}$. By Theorem 1, $T_1^{t+1} = T_1^t + T_1^{t-1} + 1$, so the new level 1 nodes informed during round $t+1$ are all nodes δ with $T_1^t + 1 \leq \delta \leq \min(n, T_1^{t+1})$. □

The first of our new protocols is a natural generalization of the protocol from [3]. Each node x that is informed during round t informs the following uninformed nodes:

Protocol **A**

- If x is the originator, inform type L_1 nodes during rounds $t+1, t+2, \ldots$
- If x is a level 1 node, inform type L_2 nodes during rounds $t+1, t+2, \ldots$
- If x is a level $k \geq 2$ node, inform a type R_{k-1} node during round $t+1$ and type L_{k+1} nodes during rounds $t+2, t+3, \ldots$

In Protocol **A**, each newly informed node at level $k \geq 2$ immediately informs one level $k-1$ node and then informs level $k+1$ nodes until the protocol terminates. The intuition is that each communication to the right can introduce a new dimension, which can eventually result in a new level 1 node being informed. So, in Protocol **A**, a node that has been informed from the left immediately initiates a path of communications going back to the level 1 node with the new dimension. Newly informed nodes that have received the message from the right, continue to forward the message left towards the level 1 node. Additional communications to the left will not lead directly to more informed nodes at level 1 because no new dimensions are being introduced. (We will see later in Protocol **B** how more new dimensions can be introduced indirectly.)

Protocol **A** informs level 1 nodes faster than Protocol $\mathbf{A}_2$. Figure 1 shows that Protocol **A** can inform 21 level 1 nodes during the first six rounds while Protocol $\mathbf{A}_2$ can inform at most 20. The third protocol, $\mathbf{A}_3$, shown in Figure 1 will be described later. Protocol $\mathbf{A}_3$ can inform the same number of level 1 nodes as Protocol **A** during the first six rounds, but eventually (when the number of rounds is nine or greater) Protocol **A** informs level 1 nodes faster than Protocol $\mathbf{A}_3$.

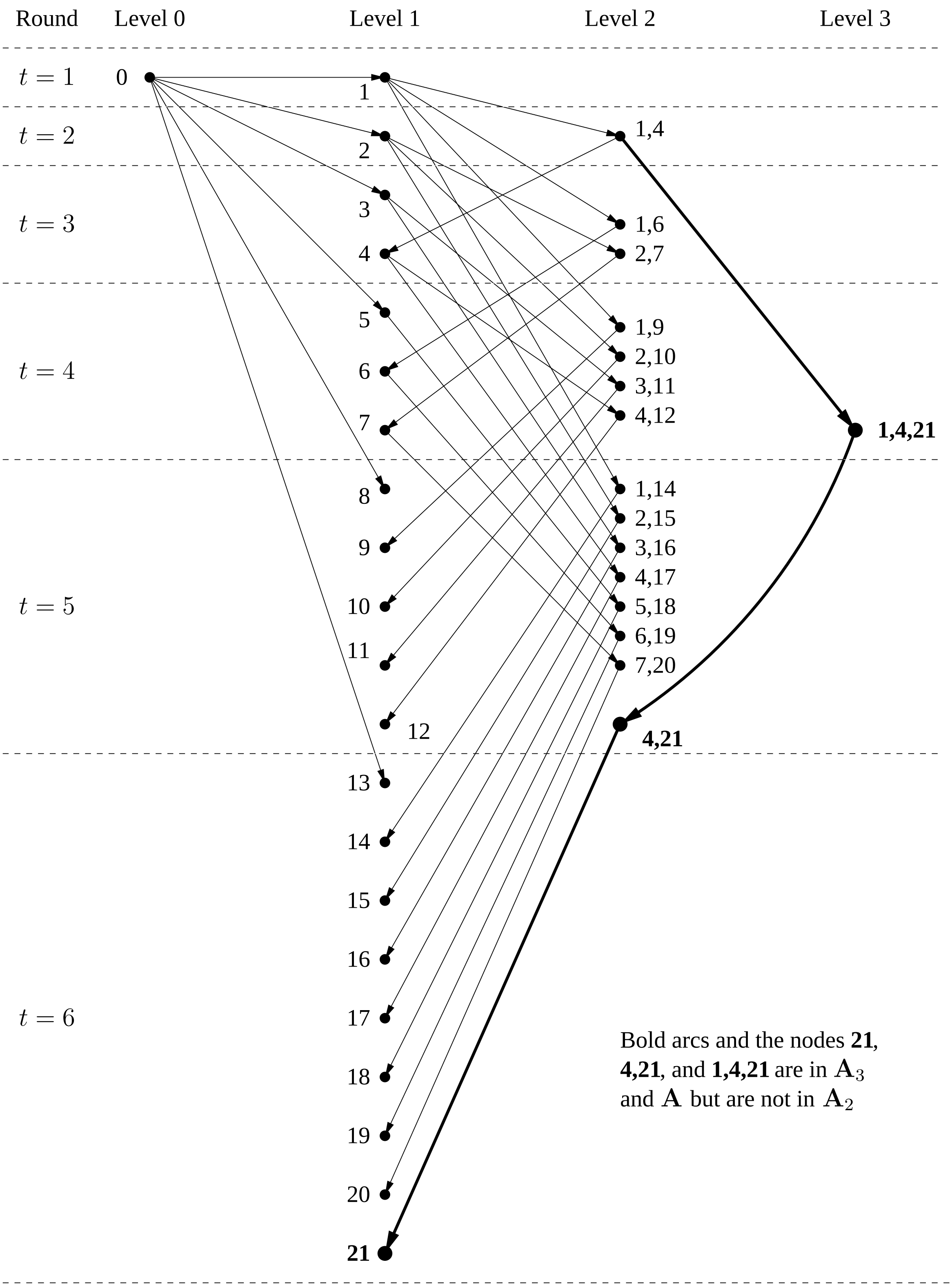

Figure 1: Node labels during the first 6 rounds of Protocols $\mathbf{A}_2$, $\mathbf{A}_3$, and $\mathbf{A}$.

The recurrence equations for Protocol $\mathbf{A}$ are:

$$\begin{aligned}
L_1^t(\mathbf{A}) &= 1 && t \geq 1 \\
L_2^1(\mathbf{A}) &= 0 && \\
L_2^t(\mathbf{A}) &= \sum_{i=1}^{t-1}(L_1^i(\mathbf{A}) + R_1^i(\mathbf{A})) = T_1^{t-1}(\mathbf{A}) && t \geq 2 && (1) \\
L_k^t(\mathbf{A}) &= 0 && t \leq 2k-3,\ k \geq 2 \\
L_k^t(\mathbf{A}) &= \sum_{i=1}^{t-2}(L_{k-1}^i(\mathbf{A}) + R_{k-1}^i(\mathbf{A})) && t \geq 2k-2,\ k \geq 3 && (2) \\
R_k^t(\mathbf{A}) &= 0 && t \leq 2k,\ k \geq 1 \\
R_k^t(\mathbf{A}) &= L_{k+1}^{t-1}(\mathbf{A}) + R_{k+1}^{t-1}(\mathbf{A}) && t \geq 2k+1,\ k \geq 1 && (3) \\
N_k^t(\mathbf{A}) &= L_k^t(\mathbf{A}) + R_k^t(\mathbf{A}) && t \geq 1,\ k \geq 1 && (4) \\
T_k^t(\mathbf{A}) &= \sum_{i=1}^{t} N_k^i(\mathbf{A}) = \sum_{i=1}^{t}(L_k^i(\mathbf{A}) + R_k^i(\mathbf{A})) && t \geq 1,\ k \geq 1
\end{aligned}$$

We begin our analysis of Protocol $\mathbf{A}$ by simplifying the expression for $T_1^t(\mathbf{A})$. We can express $N_1^t(\mathbf{A})$ as a function of the $L_k^t(\mathbf{A})$ by using equation (3) repeatedly:

$$N_1^t = L_1^t + R_1^t = 1 + L_2^{t-1} + R_2^{t-1} = 1 + L_2^{t-1} + L_3^{t-2} + L_4^{t-3} + \cdots + L_k^{t-k+1} + \cdots . \tag{5}$$

Then we use $T_1^t = T_1^{t-1} + N_1^t$, equation (5), and $L_2^{t-1} = T_1^{t-2}$ (from equation (1)) to obtain:

$$T_1^t = T_1^{t-1} + T_1^{t-2} + 1 + \sum_{i \geq 3} L_i^{t-i+1} \tag{6}$$

To show that this bound for $T_1^t(\mathbf{A})$ is attained by Protocol $\mathbf{A}$, we have to specify which nodes are informed during each round. We also have to show that no nodes are informed more than once, and that every neighbour of the originator is informed.

During round t, node 0 (the originator), will inform node $T_1^{t-1} + 1$ at level 1. Each level 1 node δ, $1 \leq \delta \leq T_1^{t-1}$, that was informed during the first $t-1$ rounds will inform node $\delta, \delta + T_1^t + 1$ if $\delta + T_1^t + 1 \leq n$ and will be idle if $\delta + T_1^t + 1 > n$. Once a node becomes idle, it remains idle until the end of the protocol.

To describe the behaviour of the level 2 nodes during round t, let us rank the nodes $\delta_1\delta_2$, $\delta_1 < \delta_2$, that are informed during the first $t-2$ rounds in increasing order according to the value of δ_2. (We will prove below that there are exactly $T_2^{t-2} = L_3^t$ such nodes and that they all have different values of δ_2.) If $\delta_1\delta_2$ is the j^{th} node in this ranking, it will inform the level 3 node $\delta_1\delta_2\delta_3$, where $\delta_3 = T_1^{t+1} + 1 + L_2^{t+1} + j$, if $\delta_3 \leq n$ and will be idle otherwise.

To describe the pattern by which level $k-1$ nodes inform level k nodes during round t (and the way that new dimensions are introduced), let us rank the level $k-1$ nodes $\delta_1\delta_2\dots\delta_{k-1}$, $\delta_1 < \delta_2 < \cdots < \delta_{k-1}$, that are informed during the first $t-2$ rounds in increasing order according to the value of δ_{k-1}. (We will prove below that there are exactly $T_{k-1}^{t-2} = L_k^t$ such nodes and that they all have different values of δ_{k-1}.) Then, if $\delta_1\delta_2\dots\delta_{k-1}$ is the j^{th} node in this ranking, it will inform the level k node $\delta_1\delta_2\dots\delta_{k-1}\delta_k$, where $\delta_k = T_1^{t+k-2} + 1 + L_2^{t+k-2} + L_3^{t+k-3} + \cdots + L_{k-1}^{t+1} + j$, if $\delta_k \leq n$ and will be idle otherwise.

Finally, each level $k \geq 2$ node $\delta_1\delta_2\dots\delta_k$, $\delta_1 < \delta_2 < \cdots < \delta_k$, that is informed during round $t-1$ will inform the level $k-1$ node $\rho_1\rho_2\dots\rho_{k-1} = \delta_2\delta_3\dots\delta_k$ during round t. (I.e., we always delete the leftmost index from the label of the level k node to obtain the label of the level $k-1$ node.)

Claim 1 *During round t, the nodes informed by Protocol* **A** *are:*

- *all level 1 nodes δ_1 such that $\delta_1 = T_1^{t-1}(\mathbf{A}) + j$, where $1 \leq j \leq N_1^t(\mathbf{A})$;*
- *all level 2 nodes $\delta_1\delta_2$, $\delta_1 < \delta_2$ such that $\delta_2 = T_1^t(\mathbf{A}) + 1 + j$, where $1 \leq j \leq N_2^t(\mathbf{A})$;*
- *all level k nodes $\delta_1\delta_2\dots\delta_k$, $\delta_1 < \delta_2 < \cdots < \delta_k$ such that $\delta_k = T_1^{t+k-2}(\mathbf{A}) + 1 + L_2^{t+k-2}(\mathbf{A}) + L_3^{t+k-3}(\mathbf{A}) + \cdots + L_{k-1}^{t+1}(\mathbf{A}) + j$, where $1 \leq j \leq N_k^t(\mathbf{A})$.*

Proof: First let us prove that if the claim is true, then the level k nodes informed during round t have a different rightmost index than the nodes informed during the first $t-1$ rounds, so $T_k^t = \sum N_k^t$. For level 1, it is clear that $\delta_1 > T_1^{t-1}$. The level 2 nodes informed before round t have $\delta_2 \leq T_1^{t-1} + 1 + N_2^{t-1}$ and the nodes informed during round t have $\delta_2 \geq T_1^t + 2 = T_1^{t-1} + N_1^t + 2 = T_1^{t-1} + R_1^t + 3 = T_1^{t-1} + N_2^{t-1} + 3$. The level k nodes informed before round t have $\delta_k \leq T_1^{t+k-3} + 1 + L_2^{t+k-3} + \cdots + L_{k-1}^t + N_k^{t-1} \leq T_1^{t+k-2}$ and the nodes informed during round t have $\delta_k \geq T_1^{t+k-2} + 2 + L_2^{t+k-2} + \cdots + L_{k-1}^{t+1} > T_1^{t+k-2}$.

Now suppose that the claim is true until round $t-1$. We prove that the claim is true for round t by induction on t. We showed above that $T_k^{t-1} = \sum N_k^{t-1}$ if the claim is true for round $t-1$. The level 1 nodes that are informed during round t are node $T_1^{t-1} + 1$ which is informed by the originator, and each node $\rho_1 = \bar{\delta}_1\delta_2$ such that $\delta_1\delta_2$ is a level 2 node that was informed during round $t-1$. By the induction hypothesis, these nodes informed by level 2 nodes are of the form $\rho_1 = T_1^{t-1} + 1 + j$, where $1 \leq j \leq N_2^{t-1}$. So, altogether, the level 1 nodes informed during round t are the nodes $T_1^{t-1} + j$, where $1 \leq j \leq 1 + N_2^{t-1} = N_1^t$. This last equation is true because $L_1^t = 1$ and $N_2^{t-1} = R_1^t$ by equations (3) and (4).

The level 2 nodes that are informed during round t are:

- every node $\delta_1\delta_2$ informed by a level 1 node δ_1 such that $\delta_2 = T_1^t + 1 + j$, where $1 \leq j \leq T_1^{t-1} = L_2^t$ (by equation 1);
- every node $\rho_1\rho_2$ informed by a level 3 node $\delta_1\delta_2\delta_3$ which was informed during round $t-1$ such that $\rho_2 = \delta_3$ where $\rho_2 = T_1^t + 1 + L_2^t + j$, $1 \leq j \leq N_3^{t-1}$ by the induction hypothesis (at level 3).

Altogether, the level 2 nodes informed during round t are the nodes with rightmost index $T_1^t + 1 + j$, where $1 \le j \le L_2^t + N_3^{t-1} = N_2^t$. This last equation is true because $N_3^{t-1} = R_2^t$ and $L_2^t + R_2^t = N_2^t$ by equations (3) and (4).

The level k nodes that are informed during round t are:

- every node $\delta_1\delta_2\cdots\delta_k$ informed by a level $k-1$ node which was informed during round $t-1$ such that $\delta_k = T_1^{t+k-2} + 1 + L_2^j$, where $1 \le j \le T_1^{t-1} = L_2^{t+k-2} + L_3^{t+k-3} + \cdots + L_{k-1}^{t+1} + j$, $1 \le j \le T_{k-1}^{t-2} = L_k^t$ (by equation (1));
- every node $\rho_1\rho_2\cdots\rho_k$ informed by a level $k+1$ node $\delta_1\delta_2\cdots\delta_{k+1}$ which was informed during round $t-1$ such that the rightmost index $\rho_k = \delta_{k+1}$ satisfies $\rho_k = T_1^{t+k-2} + 1 + L_2^{t+k-2} + L_3^{t+k-3} + \cdots + L_{k-1}^{t+1} + j$, $1 \le j \le N_{k+1}^{t-1}$ by the induction hypothesis.

Altogether, the level k nodes informed during round t are the nodes with rightmost index $T_1^{t+k-2} + 1 + L_2^{t+k-2} + L_3^{t+k-3} + \cdots + L_{k-1}^{t+1} + j$, where $1 \le j \le L_k^t + N_{k+1}^{t-1} = N_k^t$. This last equation is true because $N_{k+1}^{t-1} = R_k^t$ and $L_k^t + R_k^t = N_k^t$ by equations (3) and (4). □

If we *truncate* Protocol **A** at some level k, that is, we discard all parts of the protocol involving levels greater than k, then we get a Protocol $\mathbf{A}_k$ that approximates Protocol **A**. In fact, Protocol $\mathbf{A}_2$ is exactly the protocol from [3]. Figure 1 shows the first six rounds of Protocol $\mathbf{A}_3$. Notice that Protocol $\mathbf{A}_3$ can inform one more level 1 node than Protocol $\mathbf{A}_2$ in six rounds (using the bold arcs). The sequence $\mathbf{A}_2$, $\mathbf{A}_3$, $\mathbf{A}_4$, ... is a sequence of increasingly accurate approximations of Protocol **A**. We will solve the recurrence equations for Protocol $\mathbf{A}_k$, but, unfortunately, we have not been able to solve the recurrence equations for Protocol **A** without truncation.

Now, let us show how to find an expression for $T_1^t(\mathbf{A}_k)$ for the truncated protocol $\mathbf{A}_k$. First, note that for Protocol $\mathbf{A}_k$ we have to truncate equation (6) at level k. This is done by deleting the terms $L_i^{t-i+1}(\mathbf{A})$ for all $i > k$. Our aim will therefore be to express $T_1^t(\mathbf{A})$ for any k as the sum of two functions, the first depending on the $L_i^{t-i+1}(\mathbf{A})$ for $i \le k$, and the second depending on the $L_i^{t-i+1}(\mathbf{A})$ for $i > k$. Furthermore, we will show how to express the first function as a polynomial in the $T_1^j(\mathbf{A})$ for $j \le t-1$.

In summary, we want to obtain $T_1^t(\mathbf{A}) = P_k^t + g_k^t$ where P_k^t is a polynomial in the $T_1^j(\mathbf{A})$ with $j \le t-1$ and g_k^t is a function of the $L_i^{t-i+1}(\mathbf{A})$ with $i > k$. Therefore, for $\mathbf{A}_k$ we will obtain $T_1^t(\mathbf{A}_k) = Q_k^t$, where Q_k^t is the polynomial obtained from P_k^t by replacing the $T_1^j(\mathbf{A})$ by the $T_1^j(\mathbf{A}_k)$, $j \le t-1$. $T_1^t(\mathbf{A}_k)$ satisfies a generalized Fibonacci type of recurrence relation for which the asymptotic behaviour is determined by the largest root of the associated polynomial.

For $k = 2$, equation (6) gives $P_2^t = T_1^{t-1} + T_1^{t-2} + 1$ and $g_2^t = \sum_{i \ge 3} L_i^{t-i+1}$, so we obtain $T_1^t(\mathbf{A}_2) = T_1^{t-1}(\mathbf{A}_2) + T_1^{t-2}(\mathbf{A}_2) + 1$ which is Theorem 1.

For $k \ge 3$, we have to compute the L_i^{t-i+1} as functions of the T_1^j. This cannot be done directly, but it can be done using differences. For this purpose, we introduce a difference operator D such that for any function $f(t)$, $D[f(t)] = f(t) - f(t-1)$.

Using $T_1^t = T_1^{t-1} + D[T_1^t]$, equation (6) becomes

$$T_1^t = T_1^{t-1} + D[P_2^t] + \sum_{i \geq 3} D[L_i^{t-i+1}]. \tag{7}$$

Using $D[P_2^t] = T_1^{t-1} - T_1^{t-3}$ we get $T_1^t = P_3^t + g_3^t$ where

$$P_3^t = 2T_1^{t-1} - T_1^{t-3} + D[L_3^{t-2}] \text{ and } g_3^t = \sum_{i \geq 4} D[L_i^{t-i+1}]. \tag{8}$$

By (2) and (3), $D[L_3^{t-2}] = L_3^{t-2} - L_3^{t-3} = L_2^{t-4} + R_2^{t-4} = R_1^{t-3}$. By (4),

$$D[L_3^{t-2}] = R_1^{t-3} = N_1^{t-3} - L_1^{t-3} = T_1^{t-3} - T_1^{t-4} - 1. \tag{9}$$

Using (9) in equation (8), we get $P_3^t = 2T_1^{t-1} - T_1^{t-4} - 1$. This gives the following result:

Theorem 2 $T_1^t(\mathbf{A}_3) = 2T_1^{t-1}(\mathbf{A}_3) - T_1^{t-4}(\mathbf{A}_3) - 1.$

This is a generalized Fibonacci sequence. The largest root of the associated polynomial $x^4 - 2x^3 + 1 = 0$ is $a_3 \approx 1.839$. Thus:

Corollary 2 $T_1^t(\mathbf{A}_3) \sim 1.839^t$.

We will compute the polynomials for $k \geq 4$ using the following theorem:

Theorem 3 $P_k^t = T_1^{t-1}(\mathbf{A}) + P_{k-1}^t - P_{k-1}^{t-1} + T_1^{t-3}(\mathbf{A}) - T_1^{t-4}(\mathbf{A}) - P_{k-2}^{t-3} + P_{k-2}^{t-4}$, $k \geq 4$.

Proof: First, we prove by induction that

$$P_k^t = T_1^{t-1} + D[P_{k-1}^t] + D^{k-2}[L_k^{t-k+1}] \text{ and } g_k^t = \sum_{i \geq k+1} D^{k-2}[L_i^{t-i+1}]. \tag{10}$$

This is true for $k = 2$ by equations (1) and (6) and for $k = 3$ by equation (8). Suppose that it is true for k. Then using $T_1^t = T_1^{t-1} + D[T_1^t]$, we obtain $T_1^t = T_1^{t-1} + D[P_k^t] + D^{k-1}[L_{k+1}^{t-k}] + \sum_{i \geq k+2} D^{k-1}[L_i^{t-i+1}]$, so $P_{k+1}^t = T_1^{t-1} + D[P_k^t] + D^{k-1}[L_{k+1}^{t-k}]$ and $g_{k+1}^t = \sum_{i \geq k+2} D^{k-1}[L_i^{t-i+1}]$.

Note that the formula of the theorem can be rewritten as

$$P_k^t = T_1^{t-1} + D[P_{k-1}^t] + D[T_1^{t-3} - P_{k-2}^{t-3}]. \tag{11}$$

So, using equation (10), the theorem can be proved by proving that

$$D^{k-2}[L_k^{t-k+1}] = D[T_1^{t-3} - P_{k-2}^{t-3}]. \tag{12}$$

For $k \geq 3$, we can use (2) and (3) to obtain $D[L_k^t] = L_k^t - L_k^{t-1} = L_{k-1}^{t-2} + R_{k-1}^{t-2} = R_{k-2}^{t-1}$. So, for $k \geq 4$ we can use $D[L_{k-1}^{t+1}] = L_{k-2}^{t-1} + R_{k-2}^{t-1}$ to obtain

$$D[L_k^t] = D[L_{k-1}^{t+1}] - L_{k-2}^{t-1}. \tag{13}$$

By (13),

$$D^{k-2}[L_k^{t-k+1}] = D^{k-2}[L_{k-1}^{t-k+2}] - D^{k-3}[L_{k-2}^{t-k}] = D[D^{k-3}[L_{k-1}^{t-k+2}] - D^{k-4}[L_{k-2}^{t-k}]]. \tag{14}$$

By induction, equation (12) with $k-1$ substituted for k gives

$$D^{k-3}[L_{k-1}^{t-k+2}] = D[T_1^{t-3} - P_{k-3}^{t-3}], \tag{15}$$

and equation (12) with $k-2$ substituted for k and $t-3$ substituted for t gives

$$D^{k-4}[L_{k-2}^{t-k}] = D[T_1^{t-6} - P_{k-4}^{t-6}]. \tag{16}$$

Equation (11) with $k-2$ substituted for k and $t-3$ substituted for t gives

$$P_{k-2}^{t-3} = T_1^{t-4} + D[P_{k-3}^{t-3}] + D[T_1^{t-6} - P_{k-4}^{t-6}]. \tag{17}$$

Combining equations (15), (16), and (17), we obtain

$$\begin{aligned} D^{k-3}[L_{k-1}^{t-k+2}] - D^{k-4}[L_{k-2}^{t-k}] &= D[T_1^{t-3} - P_{k-3}^{t-3}] - P_{k-2}^{t-3} + T_1^{t-4} + D[P_{k-3}^{t-3}] \\ &= T_1^{t-3} - P_{k-2}^{t-3}. \end{aligned}$$

□

Using Theorem 3, we are able to compute all of the polynomials P_k^t for $k \geq 4$ and therefore the recurrence relations for $T_1^k(\mathbf{A}_k)$. For example, we obtain:

Theorem 4 $T_1^t(\mathbf{A}_4) = 3T_1^{t-1}(\mathbf{A}_4) - 2T_1^{t-2}(\mathbf{A}_4) + T_1^{t-3}(\mathbf{A}_4) - 3T_1^{t-4}(\mathbf{A}_4) + T_1^{t-5}(\mathbf{A}_4) + T_1^{t-6}(\mathbf{A}_4)$.

Theorem 5 $T_1^t(\mathbf{A}_5) = 4T_1^{t-1}(\mathbf{A}_5) - 5T_1^{t-2}(\mathbf{A}_5) + 4T_1^{t-3}(\mathbf{A}_5) - 7T_1^{t-4}(\mathbf{A}_5) + 6T_1^{t-5}(\mathbf{A}_5) - T_1^{t-8}(\mathbf{A}_5)$.

The following table shows the value of the largest root a_k of the associated polynomial of $T_1^t(\mathbf{A}_k)$ for $k \leq 13$. The number of level 1 nodes informed by Protocol $\mathbf{A}_k$ is proportional to a_k^t.

Protocol	Largest Root	Protocol	Largest Root	Protocol	Largest Root
$\mathbf{A}_2$	1.61803	$\mathbf{A}_6$	1.96277	$\mathbf{A}_{10}$	1.98703
$\mathbf{A}_3$	1.83929	$\mathbf{A}_7$	1.97297	$\mathbf{A}_{11}$	1.98933
$\mathbf{A}_4$	1.91286	$\mathbf{A}_8$	1.97948	$\mathbf{A}_{12}$	1.99107
$\mathbf{A}_5$	1.94552	$\mathbf{A}_9$	1.98390	$\mathbf{A}_{13}$	1.99241

Table 1: Asymptotic Values for Protocol $\mathbf{A}_k$.

3 A Sophisticated Protocol

In Protocol **A**, each newly informed node at level $k \geq 3$ only informs one level $k-1$ node before broadcasting to the right. This leaves some nodes at levels 2 through $k-1$ uninformed. The idea of our second protocol, Protocol **B**, is to inform as many nodes as possible at the lower levels, because these nodes can introduce new dimensions by communicating to the right and this will lead to new level 1 nodes. A new dimension introduced by a level k node in a communication during round t can result in a newly informed node at level 1 as early as round $t+k$.

To describe Protocol **B** more precisely, we need to extend the notation used for Protocol **A**. We will distinguish nodes informed from the right by a node x according to the number of communications to the left that have been made by x. If x is a node at level k, then the first node that it informs at level $k-1$ is a type $R_{k-1,1}$ node, the second node that it informs at level $k-1$ is a type $R_{k-1,2}$ node, and so on. This gives the following notation:

$L_k^t(\mathbf{P})$ maximum number of level k nodes informed by level $k-1$ nodes during round t of Protocol **P**

$R_{k,1}^t(\mathbf{P})$ maximum number of level k nodes informed during round t of Protocol **P** by level $k+1$ nodes which have not communicated to the left before round t

$R_{k,j}^t(\mathbf{P})$ maximum number of level k nodes informed during round t of Protocol **P** by level $k+1$ nodes which have informed exactly $j-1$ level k nodes before round t

$R_k^t(\mathbf{P})$ $= \sum_{j=1}^{k} R_{k,j}^t(\mathbf{P})$: maximum total number of level k nodes informed by level $k+1$ nodes during round t of Protocol **P**

$N_k^t(\mathbf{P})$ $= L_k^t(\mathbf{P}) + R_k^t(\mathbf{P})$: maximum total number of level k nodes informed during round t of Protocol **P**

$T_k^t(\mathbf{P})$ $= \sum_{i=1}^{t} N_k^i(\mathbf{P})$: maximum total number of level k nodes informed during the first t rounds of Protocol **P**

Now we can describe Protocol **B** precisely. If x is a node that is informed during round

t, then x informs the following uninformed nodes:

Protocol **B**

- If x is the originator, inform type L_1 nodes during rounds $t+1, t+2, \ldots$
- If x is a level 1 node, inform type L_2 nodes during rounds $t+1, t+2, \ldots$
- If x is a type L_k node, $k \geq 2$, inform a type $R_{k-1,1}$ node during round $t+1$, a type $R_{k-1,2}$ node during round $t+2$, $\ldots$, a type $R_{k-1,k-1}$ node during round $t+k-1$, and type L_{k+1} nodes during rounds $t+k, t+k+1, \ldots$
- If x is a type $R_{k,j}$ node, $k \geq 2$, $1 \leq j \leq k$, inform a type $R_{k-1,j}$ node during round $t+1$, a type $R_{k-1,j+1}$ node during round $t+2$, $\ldots$, a type $R_{k-1,k-1}$ node during round $t+k-j$, and type L_{k+1} nodes during rounds $t+k-j+1, t+k-j+2, \ldots$

Before we analyze Protocol **B**, it will be helpful to look at an example of part of the protocol. Figure 2 shows a path from the originator to a level 5 node labelled $\delta_1\delta_2\delta_3\delta_4\alpha$. The tree of all communications to the left from node $\delta_1\delta_2\delta_3\delta_4\alpha$ is also shown, but all other communications have been omitted to keep the figure simple. In our example, dimension α is introduced in the communication right from node $\delta_1\delta_2\delta_3\delta_4$ to node $\delta_1\delta_2\delta_3\delta_4\alpha$ during round t. The rounds during which other nodes are informed and the types of the nodes are indicated in the figure.

Figure 2 illustrates some properties that we will use in our analysis. First, consider the path of type L_k nodes from the originator to node $\delta_1\delta_2\delta_3\delta_4\alpha$ along the top of the diagram. Each of the communications to the right shown in the figure introduces a new dimension, but the rounds during which these communications occur are not consecutive because communications to the left by the type L_k nodes are done before communications to the right. The type L_2 node labelled $\delta_1\delta_2$ makes one communication to the left (to a type $R_{1,1}$ node) before informing the type L_3 node $\delta_1\delta_2\delta_3$, the type L_3 node makes two communications to the left, and in general, a type L_k node, $k \geq 2$, will make $k-1$ communications to the left before communicating to the right. So, a type L_k node will receive the message $\sum_{i=1}^{k-1} i = \frac{k(k-1)}{2}$ rounds after the originator initiates the path to the right.

Next, we can consider node $\delta_1\delta_2\delta_3\delta_4\alpha$ to be the root of a broadcast tree, which we denote $T_{\delta_1\delta_2\delta_3\delta_4\alpha}$, going left and starting in round $t+1$. The tree $T_{\delta_1\delta_2\delta_3\delta_4\alpha}$, is a complete binomial tree of depth 4 and contains all nodes at levels 1 through 5 with a 1 in position α. Notice that the number of level $i+1$ nodes in $T_{\delta_1\delta_2\delta_3\delta_4\alpha}$ is $\binom{4}{i}$, $1 \leq i+1 \leq 5$. In particular, $T_{\delta_1\delta_2\delta_3\delta_4\alpha}$ contains one new level 1 node. Another useful property of $T_{\delta_1\delta_2\delta_3\delta_4\alpha}$ is that all $\sum_{i=0}^{4} \binom{4}{i} = 2^4$ nodes, including $\delta_1\delta_2\delta_3\delta_4\alpha$, finish their communications to the left during the same round $t+4$, so they can all start communicating to the right simultaneously in round $t+5$. Each of

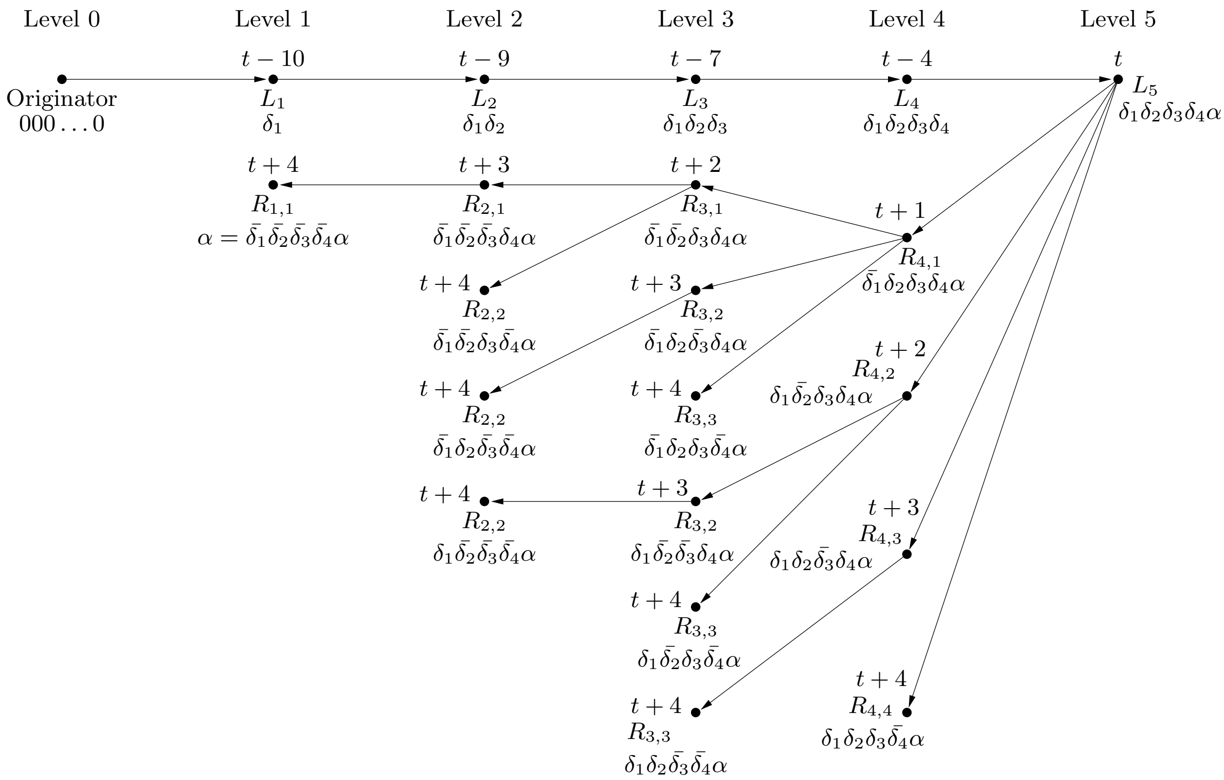

Figure 2: The broadcast tree of $T_{\delta_1\delta_2\delta_3\delta_4\alpha}$

these communications to the right introduces a new dimension, and each node that receives the message from the left during round $t+5$ is the root of a broadcast tree going left that contains a new level 1 node. In general, the broadcast tree of a type L_k node that is informed during round t contains 2^{k-1} nodes, including one level 1 node that is informed during round $t+k-1$, and all nodes of this tree communicate to the right during round $t+k$ introducing 2^{k-1} new dimensions.

With this intuition, we can write the recurrence equations for Protocol **B**:

$$
\begin{array}{rcll}
L_1^t(\mathbf{B}) &=& 1 & t \geq 1 \\
L_k^t(\mathbf{B}) &=& 0 & t \leq \frac{k(k-1)}{2},\ k \geq 2 \\
L_k^t(\mathbf{B}) &=& \sum_{i=1}^{t-k+1} L_{k-1}^i(\mathbf{B}) + \sum_{j=1}^{k-1} \sum_{i=1}^{t-k+j} R_{k-1,j}^i(\mathbf{B}) & t \geq \frac{k(k-1)}{2} + 1,\ k \geq 2
\end{array} \quad (18)
$$

$$
\begin{array}{rcll}
R_{k,j}^t(\mathbf{B}) &=& 0 & 1 \leq k < j \\
R_{k,j}^t(\mathbf{B}) &=& 0 & t \leq \frac{k(k+1)}{2} + j,\ 1 \leq j \leq k \\
R_{k,j}^t(\mathbf{B}) &=& L_{k+1}^{t-j}(\mathbf{B}) + \sum_{i=1}^{j} R_{k+1,i}^{t-j+i-1}(\mathbf{B}) & t \geq \frac{k(k+1)}{2} + j + 1,\ 1 \leq j \leq k
\end{array} \quad (19)
$$

$$
\begin{array}{rcll}
R_k^t(\mathbf{B}) &=& 0 & t \leq \frac{k(k+1)}{2} + 1,\ k \geq 1 \\
R_k^t(\mathbf{B}) &=& \sum_{j=1}^{k} R_{k,j}^t(\mathbf{B}) & t \geq \frac{k(k+1)}{2} + 2,\ k \geq 1 \\
N_k^t(\mathbf{B}) &=& L_k^t(\mathbf{B}) + R_k^t(\mathbf{B}) & t \geq 1,\ k \geq 1 \\
T_k^t(\mathbf{B}) &=& \sum_{i=1}^{t} N_k^i(\mathbf{B}) = \sum_{i=1}^{t} (L_k^i(\mathbf{B}) + R_k^i(\mathbf{B})) & t \geq 1,\ k \geq 1
\end{array}
$$

Theorem 6 *Protocol* **B** *informs* 2^t *level 1 nodes no later than round* $t + \left\lceil \frac{\sqrt{8t+1}-1}{2} \right\rceil$.

Proof: During each round of Protocol **B**, each informed node informs an uninformed neighbour, so the total number of informed nodes after t rounds is 2^t. By equation (18), the most distant informed node from the originator after t rounds is at level at most k_t where $t \leq \frac{k_t(k_t+1)}{2}$. So, $k_t = \left\lceil \frac{\sqrt{8t+1}-1}{2} \right\rceil$.

From the discussion above, a type L_k node x that is informed during round t is the root of a broadcast tree T_x with 2^k nodes that are all informed during round $t+k-1$. In particular, T_x includes a level 1 node which we will call $f_1(x)$.

Now we will show that at time $t + k_t$ there are at least 2^t informed level 1 nodes. To prove this, we will associate with each of the 2^t nodes informed during the first t rounds, a level 1 node that is informed no later than round $t + k_t$.

If node x is of type $L_k^{t'}$, the associated level 1 node is $f_1(x)$ of the broadcast tree T_x, and $f_1(x)$ is informed no later than round $t' + k - 1 \leq t + k_t - 1$.

If node x is of type R_m, it belongs to the broadcast tree of a type L_k^{t-h} node $r(x)$ with $k \leq k_t$; therefore $m \leq k_t - 1$.

Case 1: If $h \geq k-1$, then all the nodes of the broadcast tree of $r(x)$ are informed during round t. During round $t+1$, x will inform a type L_{m+1}^{t+1} node y, which in turn informs the level 1 node $f_1(y)$ m rounds later, that is, during round $t+m+1 \leq t+k_t$.

Case 2: If $h < k-1$, then only $2^h - 1$ nodes of the broadcast tree of $r(x)$ are informed during the first t rounds. We will show that we can associate at least $2^h - 1$ informed level 1 nodes with this broadcast tree. Indeed, all the nodes of the broadcast tree of $r(x)$ are informed during round $t-h+k-1$. During round $t-h+k$, any type R_p node of the broadcast tree will inform a type L_{p+1} node which in turn informs a level 1 node during round $t-h+k+p$. So, no later than round $t+k_t$ we have at least as many informed level 1 nodes as the number of R_p nodes with $p \leq h$. The number of such R_p nodes is $1 + \binom{k-1}{1} + \cdots + \binom{k-1}{h-1} > 1 + \binom{h}{1} + \cdots + \binom{h}{h-1} = 2^h - 1$. □

Corollary 3 *In the hypercube with $N = 2^n$ nodes, neighbourhood broadcasting can be done in at most $log_2 n + \lceil \sqrt{2 log_2 n}\, \rceil$ rounds.*

Corollary 4 *For any fixed $\epsilon > 0$ and sufficiently large t, the number of level 1 nodes informed by Protocol* **B** *in t rounds is at least $(2-\epsilon)^t$.*

Proof: After $t = u + \sqrt{2u+1}$ rounds, we have 2^u level 1 nodes informed. Solving for u we get $u = t + 1 - \sqrt{2t+1}$. So, at time t there are at least $2^{t+1-\sqrt{2t+1}}$ informed level 1 nodes. For any fixed ϵ and sufficiently large t, $2^{t+1-\sqrt{2t+1}} \geq (2-\epsilon)^t$. □

We can *truncate* Protocol **B** at some level $k \geq 3$, in the same way that we truncated Protocol **A**, to get a sequence $\mathbf{B}_3$, $\mathbf{B}_4$, $\mathbf{B}_5$, ..., of increasingly accurate approximations of Protocol **B**. Protocol $\mathbf{B}_2$ is exactly the same as Protocol $\mathbf{A}_2$. We begin our analysis in the same way as we did for Protocol **A** (*cf.* equation (6)) by simplifying the expression for $T_1^t(\mathbf{B})$:

$$T_1^t = T_1^{t-1} + N_1^t = T_1^{t-1} + 1 + L_2^{t-1} + L_3^{t-2} + \cdots + L_k^{t-k+1} + \cdots .$$

Noting that $L_2^{t-1} = T_1^{t-2}$, we get

$$T_1^t = T_1^{t-1} + T_1^{t-2} + 1 + \sum_{i \geq 3} L_i^{t-i+1}. \tag{20}$$

Using the difference operator with equation (18), we get

$$D[L_k^t] = L_{k-1}^{t-k+1} + \sum_{j=1}^{k-1} R_{k-1,j}^{t-k+j}.$$

By repeated use of equation (19), we get

$$D[L_3^t] = L_2^{t-2} + 2L_3^{t-3} + 3L_4^{t-4} + \cdots + (i-1)L_i^{t-i} + \cdots , \tag{21}$$

$$D[L_4^t] = L_3^{t-3} + 3L_4^{t-4} + 6L_5^{t-5} + \cdots + \binom{i-1}{2} L_i^{t-i} + \cdots , \tag{22}$$

and more generally

$$D[L_k^t] = \sum_{i \geq k-1} \binom{i-1}{k-2} L_i^{t-i}. \tag{23}$$

Theorem 7 $T_1^t(\mathbf{B}_3) = 2T_1^{t-1}(\mathbf{B}_3) + T_1^{t-3}(\mathbf{B}_3) - 2T_1^{t-4}(\mathbf{B}_3) - T_1^{t-5}(\mathbf{B}_3) - 2.$

Proof: Truncating equation (20) at level 3 gives

$$T_1^t = T_1^{t-1} + T_1^{t-2} + 1 + L_3^{t-2}. \tag{24}$$

Applying the difference operator to (24) we get $T_1^t = T_1^{t-1} + D[T_1^t] = 2T_1^{t-1} - T_1^{t-3} + D[L_3^{t-2}]$. By (21), $D[L_3^{t-2}] = L_2^{t-4} + 2L_3^{t-5} = T_1^{t-5} + 2L_3^{t-5}$, so we get

$$T_1^t = 2T_1^{t-1} - T_1^{t-3} + T_1^{t-5} + 2L_3^{t-5}. \tag{25}$$

Substituting $t-3$ for t in equation (24) gives $L_3^{t-5} = T_1^{t-3} - (T_1^{t-4} + T_1^{t-5} + 1)$ and so (25) becomes $T_1^t = 2T_1^{t-1} + T_1^{t-3} - 2T_1^{t-4} - T_1^{t-5} - 2$. □

Corollary 5 $T_1^t(\mathbf{B}_3) \sim 1.913^t$.

It is interesting to note that $T_1^t(\mathbf{B}_3) = T_1^t(\mathbf{A}_4)$ (compare Theorems 7 and 4) even though the protocols are different. The originator and nodes of types L_1 and L_2 behave the same in the two protocols. In Protocol $\mathbf{A}_4$, each level 3 node informs a type $R_{2,1}$ node and then informs level 4 nodes until the end of the protocol. Each level 4 node informs one level 3 node and then becomes idle. In Protocol $\mathbf{B}_3$, each level 3 node informs a type $R_{2,1}$ node and a type $R_{2,2}$ node and then becomes idle. The type $R_{2,1}$ nodes behave the same in the two protocols. To see that the two protocols inform the same level 1 nodes during each round, we will compare the parts of the protocols that are different. Figure 3 shows parts of the broadcast trees rooted at a level 3 node $\delta_1\delta_2\delta_3$. In both protocols, node $\delta_1\delta_2\delta_3$ informs the type $R_{2,1}$ node $\delta_2\delta_3$ during round $t+1$. Node $\delta_2\delta_3$ behaves the same in both protocols, so it is not shown. In the figure, communications that are in Protocol $\mathbf{A}_4$ are shown in normal typeface and communications that are in Protocol $\mathbf{B}_3$ are shown in bold typeface. Notice that the two protocols inform different level 3 nodes, but the same level 2 nodes are informed. In both protocols, node $\delta_3\alpha_1$ will inform the new level 1 node α_1 during round $t+5$, node $\delta_3\alpha_2$ will inform the new level 1 node α_2 during round $t+6$, and so on.

The proofs of the next two theorems appear in the appendix.

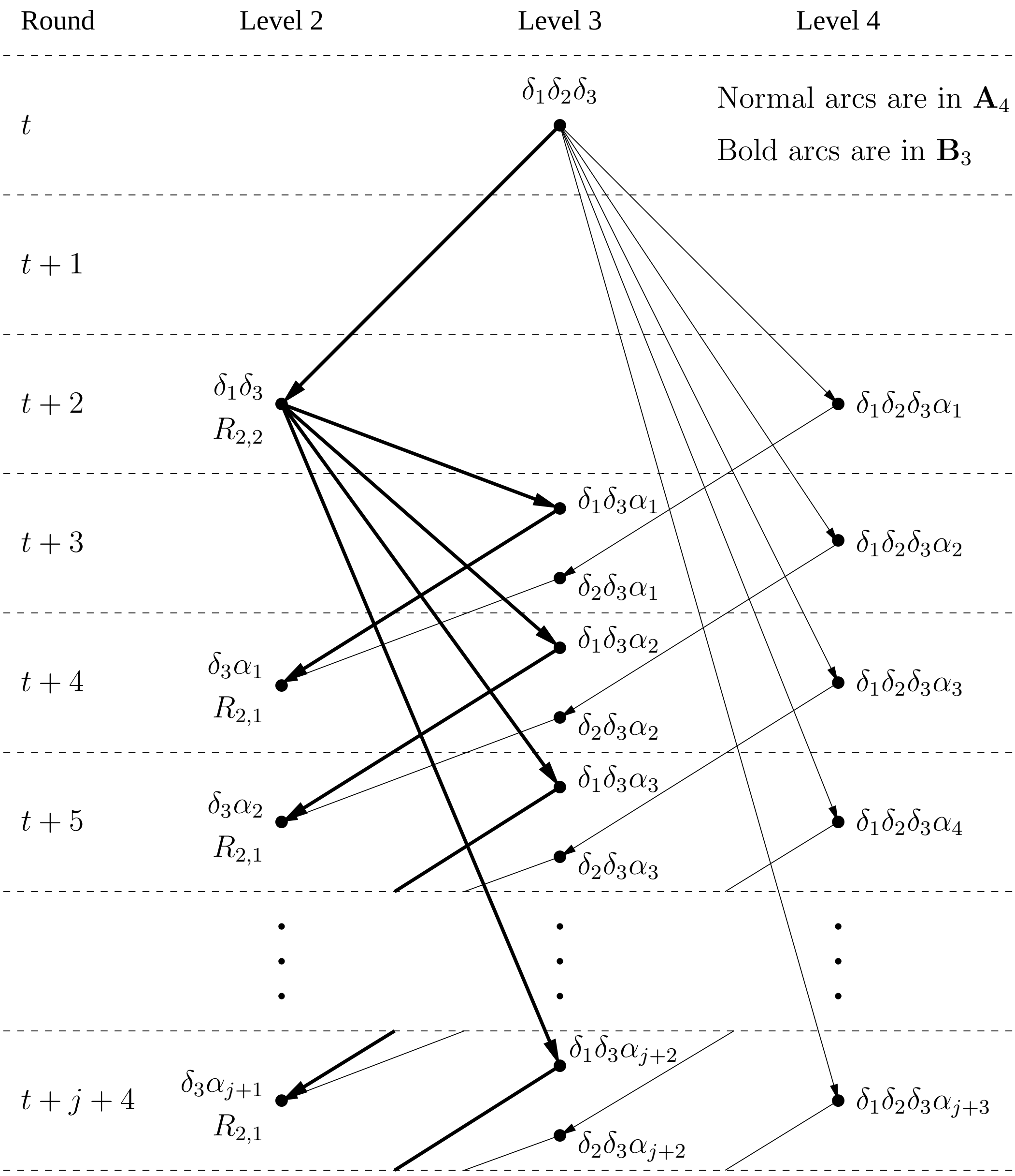

Figure 3: Differences between Protocols $\mathbf{A}_4$ and $\mathbf{B}_3$.

Theorem 8 $T_1^t(\mathbf{B}_4) = 3T_1^{t-1}(\mathbf{B}_4) - 2T_1^{t-2}(\mathbf{B}_4) + T_1^{t-3}(\mathbf{B}_4) - 5T_1^{t-5}(\mathbf{B}_4) + T_1^{t-6}(\mathbf{B}_4) + 3T_1^{t-8}(\mathbf{B}_4) + T_1^{t-9}(\mathbf{B}_4) + 3.$

Corollary 6 $T_1^t(\mathbf{B}_4) \sim 1.9867^t$.

Theorem 9 $T_1^t(\mathbf{B}_5) = 4T_1^{t-1}(\mathbf{B}_5) - 5T_1^{t-2}(\mathbf{B}_5) + 3T_1^{t-3}(\mathbf{B}_5) - T_1^{t-4}(\mathbf{B}_5) - T_1^{t-5}(\mathbf{B}_5) - 6T_1^{t-6}(\mathbf{B}_5) + 7T_1^{t-7}(\mathbf{B}_5) - T_1^{t-8}(\mathbf{B}_5) + 4T_1^{t-9}(\mathbf{B}_5) + 7T_1^{t-10}(\mathbf{B}_5) - 4T_1^{t-11}(\mathbf{B}_5) - 2T_1^{t-12}(\mathbf{B}_5) - 4T_1^{t-13}(\mathbf{B}_5) - T_1^{t-14}(\mathbf{B}_5) - 4.$

Corollary 7 $T_1^t(\mathbf{B}_5) \sim 1.9989^t$.

4 Conclusions

The following table shows the numbers of informed level 1 nodes for several protocols. These numbers were obtained using programs based on the recurrence relations in this paper. The numbers for the truncated protocols can also be obtained using the theorems in this paper. The protocols in Table 2 are ordered left to right according to increasing number of informed level 1 nodes. An entry shown in bold font indicates the first round during which a protocol is better than the protocol on its left.

Round	$\mathbf{A}_2 = \mathbf{B}_2$	$\mathbf{A}_3$	$\mathbf{A}_4 = \mathbf{B}_3$	$\mathbf{A}$	$\mathbf{B}_4$	$\mathbf{B}$
1	1	1	1	1	1	1
2	2	2	2	2	2	2
3	4	4	4	4	4	4
4	7	7	7	7	7	7
5	12	12	12	12	12	12
6	20	**21**	21	21	21	21
7	33	37	37	37	37	37
8	54	66	66	66	66	66
9	88	119	**120**	120	120	120
10	143	216	221	221	**222**	222
11	232	394	411	411	416	416
12	376	721	771	**772**	788	788
13	609	1322	1455	1461	1507	1507
14	986	2427	2757	2780	2905	2905
15	1596	4459	5240	5316	5634	**5635**
20	17710	93723	132662	142644	163510	164203
25	196417	1972659	3392169	4013545	4958328	5039922
30	2178308	41523767	86856182	115996781	152476127	158120581

Table 2: Level 1 Nodes Informed.

It is interesting to examine the last row of Table 2 which shows the numbers of informed nodes after 30 rounds. Protocol $\mathbf{A}_3$ nearly doubles the number of informed nodes compared to Protocol $\mathbf{A}_2$, and Protocol $\mathbf{A}_4$ more than doubles it again. Protocol $\mathbf{B}$ is so much better than Protocol $\mathbf{A}$ that even the truncated Protocol $\mathbf{B}_4$ outperforms the untruncated Protocol $\mathbf{A}$. We know from Corollary 4 that Protocol $\mathbf{B}$ is asymptotically optimal. The last two columns suggest that Protocol $\mathbf{B}_4$ is almost as good as the untruncated Protocol $\mathbf{B}$. To examine this further, we used programs based on the recurrence relations to determine lower bounds on the rates that the truncated protocols inform level 1 nodes. More precisely, the number of level 1 nodes informed by each truncated Protocol $\mathbf{A}_k$ is proportional to a_k^t where a_k is the largest root of the associated polynomial of $T_1^t(\mathbf{A}_k)$. Similarly, the performance of Protocol $\mathbf{B}_k$ is proportional to b_k^t where b_k is the largest root of associated polynomial of $T_1^t(\mathbf{B}_k)$. The results are shown in Figure 4. The lower curve shows the sequence $\{a_k\}$, $k = 3, 4, 5, \ldots$ and the upper curve shows the sequence $\{b_k\}$, $k = 3, 4, 5, \ldots$. (We have omitted the value $a_2 = b_2 = \frac{1+\sqrt{5}}{2} \approx 1.618$ for Protocol $\mathbf{A}_2$ = Protocol $\mathbf{B}_2$ to reduce the range of the vertical scale of the graph.) The graph shows that the sequence $\{b_k\}$ converges very quickly with increasing k towards the optimal value 2 (shown as a horizontal line at the top of the graph). The sequence $\{a_k\}$ converges more slowly, but it is clear that it is also approaching the optimal value.

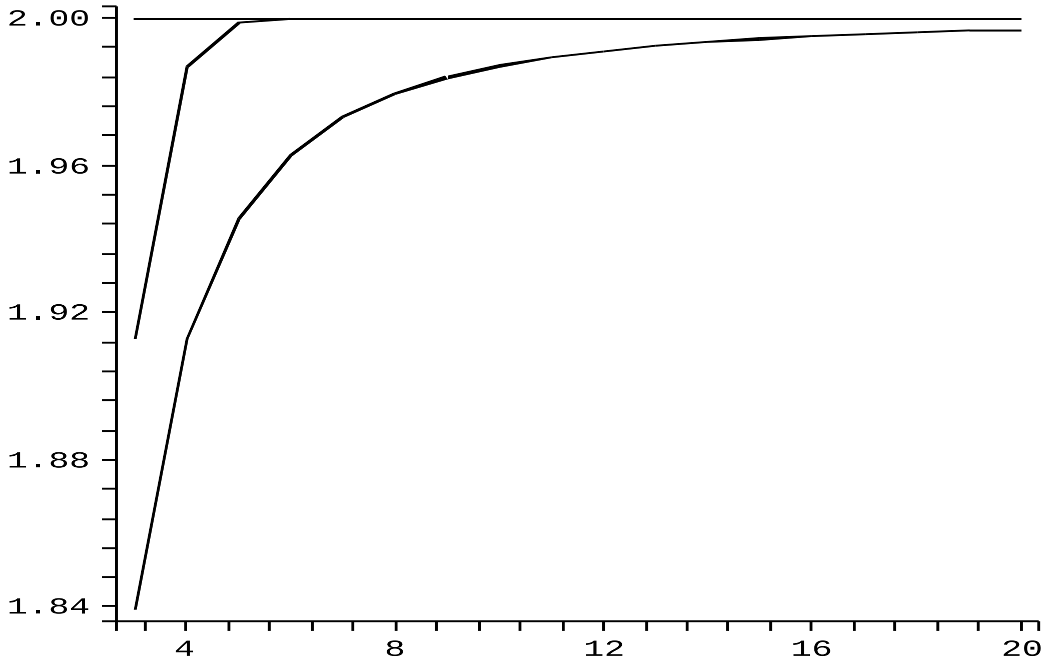

Figure 4: Asymptotic Convergence of Largest Roots a_k and b_k for $k \geq 3$.

An alternative approach to solving the recurrence relations in this paper is to use the matrix approach described in [8, 9]. We have applied this approach to the protocols in this paper and obtained the same polynomials for the truncated protocols.

We note that the recurrence relations that we have presented in this paper apply to k-neighbourhood broadcasting for any $k \geq 1$. It is possible to extend our analysis to determine expressions for the truncated protocols for $k > 1$, but the derivations might be quite long.

Finally, we re-iterate that improvement of the lower bound for neighbourhood broadcasting or a proof that no protocol can inform the neighbours of the originator faster than Protocol $\mathbf{B}$ are open problems.

References

[1] J.-C. Bermond, A. Ferreira, and J.G. Peters. Partial broadcasting in hypercubes. *International Workshop on Interconnection Networks (IWIN)*, Luminy, France, July 1991.

[2] J.-C. Bermond, A. Ferreira, S. Pérennes, and J.G. Peters. Neighbourhood broadcasting in hypercubes. Manuscript, August 1998.

[3] M. Cosnard and A. Ferreira. On the real power of loosely coupled parallel architectures. *Parallel Processing Letters* 1: 103–111, 1991.

[4] S. Even and B. Monien. On the number of rounds necessary to disseminate information. *Proc. 1^{st} ACM Symp. on Parallel Algorithms and Architectures*, Santa Fe, New Mexico: 318–327, June 1989.

[5] P. Fraigniaud and E. Lazard. Methods and problems of communication in usual networks. *Discrete Applied Mathematics* 53: 79–133, 1994.

[6] G. Fertin and A. Raspaud. k-neighborhood broadcasting. *Proc. International Colloquium on Structural Information and Communication Complexity (SIROOCO 8)*, Vall de Núria, Spain, June 2001. *Proceedings in Informatics* 11, Carleton Scientific (2001), 133–146.

[7] G. Fertin and A. Raspaud. Neighborhood communications in networks. *Proc. Euroconference on Combinatorics, Graph Theory and Applications (COMB01)*, Barcelona, Spain, September 2001. *Electronic Notes on Discrete Mathematics* 10, Elsevier (2001).

[8] M. Flammini and S. Pérennes. On the optimality of general lower bounds for broadcasting and gossiping. *SIAM Journal on Discrete Mathematics* 14: 267–282, 2001.

[9] M. Flammini and S. Pérennes. Lower bounds on the broadcasting and gossiping time for restricted protocols. *SIAM Journal on Discrete Mathematics* 17: 521–540, 2004.

[10] S. Fujita. Neighborhood information dissemination in the star graph. *IEEE Transactions on Computers* 49: 1366–1370, 2000.

[11] S. Fujita. Time-efficient multicast to local vertices in star interconnection networks under the single-port model. *IEICE Transactions on Information and Systems* E87-D: 315–321, 2004.

[12] S. Fujita. Optimal neighbourhood broadcast in star graphs. *Journal of Interconnection Networks* 4: 419–428, 2003.

[13] S. Fujita, S. Pérennes, and J.G. Peters. Neighbourhood gossiping in hypercubes. *Parallel Processing Letters* 8: 189–195, 1998.

[14] S.M. Hedetniemi, S.T. Hedetniemi, and A.L. Liestman. A survey of gossiping and broadcasting in communication networks. *Networks* 18: 319–349, 1986.

[15] J. Hromovič, R. Klasing, A. Pelc, P. Ružička, and W. Unger. Dissemination of Information in Communication Networks: Broadcasting, Gossiping, Leader Election, and Fault-Tolerance. *Texts in Theoretical Computer Science*, Springer, 2005.

[16] D.D. Kouvatsos and I.M. Mkwawa. Neighbourhood broadcasting schemes for Cayley graphs with background traffic. *Proc.* 4^{th} *EPSRC/BCS PG Symposium on the Convergence of Telecommunications, Networking and Broadcasting (PG Net 2003)*, M. Merabti (Ed.), Liverpool: 143–148, June 2003.

[17] D.W. Krumme. Fast gossiping for the hypercube. *SIAM Journal of Computing* 21: 365–380, 1992.

[18] D.W. Krumme, G. Cybenko, and K.N. Venkataraman. Gossiping in minimal time. *SIAM Journal of Computing* 21: 111–139, 1992.

[19] I.M. Mkwawa and D.D. Kouvatsos. An optimal neighbourhood broadcasting scheme for star interconnection networks. *Journal of Interconnection Networks* 4: 103–112, 2003.

[20] K. Qiu and S.K. Das. A novel neighbourhood broadcasting algorithm on star graphs. *Proc.* 9^{th} *International Conference on Parallel and Distributed Systems (ICPADS'02)*, Taiwan: 37–41, 2002.

Appendix: Proofs of Theorems 8 and 9

Theorem 8 $T_1^t(\mathbf{B}_4) = 3T_1^{t-1}(\mathbf{B}_4) - 2T_1^{t-2}(\mathbf{B}_4) + T_1^{t-3}(\mathbf{B}_4) - 5T_1^{t-5}(\mathbf{B}_4)$
$+T_1^{t-6}(\mathbf{B}_4) + 3T_1^{t-8}(\mathbf{B}_4) + T_1^{t-9}(\mathbf{B}_4) + 3.$

Proof: In this case, equation (20) becomes

$$T_1^t = T_1^{t-1} + T_1^{t-2} + 1 + L_3^{t-2} + L_4^{t-3} \tag{26}$$

and by difference

$$T_1^t = T_1^{t-1} + D[T_1^t] = 2T_1^{t-1} - T_1^{t-3} + D[L_3^{t-2}] + D[L_4^{t-3}]. \tag{27}$$

Truncating (21) and (22) at level 4 and using the value $L_3^{t-3} + L_4^{t-4} = T_1^{t-1} - T_1^{t-2} - T_1^{t-3} - 1$ deduced from (26) (with $t-1$ substituted for t) gives:

$$D[L_3^t] = T_1^{t-3} + 2L_3^{t-3} + 3L_4^{t-4} = 2T_1^{t-1} - 2T_1^{t-2} - T_1^{t-3} - 2 + L_4^{t-4}; \tag{28}$$

$$D[L_4^t] = L_3^{t-3} + 3L_4^{t-4} = T_1^{t-1} - T_1^{t-2} - T_1^{t-3} - 1 + 2L_4^{t-4}. \tag{29}$$

Using (27), (28) with $t-2$ substituted for t, and (29) with $t-3$ substituted for t we get:

$$T_1^t = 2T_1^{t-1} + T_1^{t-3} - T_1^{t-4} - 2T_1^{t-5} - T_1^{t-6} - 3 + L_4^{t-6} + 2L_4^{t-7} \tag{30}$$

We can write equation (30) as $T_1^t = P^t + F^t(L_4)$ where

$$P^t = 2T_1^{t-1} + T_1^{t-3} - T_1^{t-4} - 2T_1^{t-5} - T_1^{t-6} - 3, \text{ and} \tag{31}$$

$$F^t(L_4) = L_4^{t-6} + 2L_4^{t-7} = T_1^t - P^t. \tag{32}$$

Using the difference operator we get

$$T_1^t = T_1^{t-1} + D[T_1^t] = T_1^{t-1} + D[P^t] + D[L_4^{t-6}] + 2D[L_4^{t-7}]. \tag{33}$$

By (29),

$$\begin{aligned} D[L_4^{t-6}] + 2D[L_4^{t-7}] &= (T_1^{t-7} + T_1^{t-8} - 3T_1^{t-9} - 2T_1^{t-10} - 3) + 2L_4^{t-10} + 4L_4^{t-11} \\ &= (T_1^{t-7} + T_1^{t-8} - 3T_1^{t-9} - 2T_1^{t-10} - 3) + 2F^{t-4}(L_4). \end{aligned} \tag{34}$$

Using (33), (34), (32) with $t-4$ substituted for t, and (31), we get

$$\begin{aligned} T_1^t &= T_1^{t-1} + (P^t - P^{t-1}) + (T_1^{t-7} + T_1^{t-8} - 3T_1^{t-9} - 2T_1^{t-10} - 3) + 2(T_1^{t-4} - P^{t-4}) \\ &= 3T_1^{t-1} - 2T_1^{t-2} + T_1^{t-3} - 5T_1^{t-5} + T_1^{t-6} + 3T_1^{t-8} + T_1^{t-9} + 3. \end{aligned}$$ □

Theorem 9 $T_1^t(\mathbf{B}_5) = 4T_1^{t-1}(\mathbf{B}_5) - 5T_1^{t-2}(\mathbf{B}_5) + 3T_1^{t-3}(\mathbf{B}_5) - T_1^{t-4}(\mathbf{B}_5) - T_1^{t-5}(\mathbf{B}_5)$
$-6T_1^{t-6}(\mathbf{B}_5) + 7T_1^{t-7}(\mathbf{B}_5) - T_1^{t-8}(\mathbf{B}_5) + 4T_1^{t-9}(\mathbf{B}_5) + 7T_1^{t-10}(\mathbf{B}_5)$
$-4T_1^{t-11}(\mathbf{B}_5) - 2T_1^{t-12}(\mathbf{B}_5) - 4T_1^{t-13}(\mathbf{B}_5) - T_1^{t-14}(\mathbf{B}_5) - 4.$

Proof: Truncating (20) at level 5 gives

$$T_1^t = T_1^{t-1} + T_1^{t-2} + 1 + L_3^{t-2} + L_4^{t-3} + L_5^{t-4}. \quad (35)$$

Using the value of $L_3^{t-3} + L_4^{t-4} + L_5^{t-5}$ deduced from (35) (with $t-1$ substituted for t) in equations (21), (22), and (23) gives:

$$D[L_3^t] = L_2^{t-2} + 2L_3^{t-3} + 3L_4^{t-4} + 4L_5^{t-5} = 2T_1^{t-1} - 2T_1^{t-2} - T_1^{t-3} - 2 + L_4^{t-4} + 2L_5^{t-5}; \quad (36)$$

$$D[L_4^t] = L_3^{t-3} + 3L_4^{t-4} + 6L_5^{t-5} = T_1^{t-1} - T_1^{t-2} - T_1^{t-3} - 1 + 2L_4^{t-4} + 5L_5^{t-5}; \quad (37)$$

$$D[L_5^t] = L_4^{t-4} + 4L_5^{t-5}. \quad (38)$$

By difference we get

$$T_1^t = T_1^{t-1} + D[T_1^t] = 2T_1^{t-1} - T_1^{t-3} + D[L_3^{t-2}] + D[L_4^{t-3}] + D[L_5^{t-4}].$$

Using (36), (37), and (38) with $t-2$, $t-3$, and $t-4$ substituted for t, respectively, gives $T_1^t = Q^t + F^t(L_4, L_5)$ where

$$Q^t = 2T_1^{t-1} + T_1^{t-3} - T_1^{t-4} - 2T_1^{t-5} - T_1^{t-6} - 3 \text{ and} \quad (39)$$

$$F^t(L_4, L_5) = L_4^{t-6} + 2L_4^{t-7} + L_4^{t-8} + 2L_5^{t-7} + 5L_5^{t-8} + 4L_5^{t-9} = T_1^t - Q^t. \quad (40)$$

By difference using $t-6$, $t-7$, $t-8$ substituted for t in (37) and $t-7$, $t-8$, $t-9$ substituted for t in (38), we get

$$\begin{aligned} T_1^t &= T_1^{t-1} + (Q^t - Q^{t-1}) + (F^t(L_4, L_5) - F^{t-1}(L_4, L_5)) \\ &= T_1^{t-1} + (Q^t - Q^{t-1}) + (T_1^{t-7} - T_1^{t-8} - T_1^{t-9} - 1) \\ &\quad +2(T_1^{t-8} - T_1^{t-9} - T_1^{t-10} - 1) + (T_1^{t-9} - T_1^{t-10} - T_1^{t-11} - 1) \\ &\quad +2L_4^{t-10} + 6L_4^{t-11} + 7L_4^{t-12} + 4L_4^{t-13} + 5L_5^{t-11} + 18L_5^{t-12} + 25L_5^{t-13} + 16L_5^{t-14}. \quad (41) \end{aligned}$$

The last line of equation (41) involving terms in the L_4^j and L_5^j can be written as

$$2F^{t-4}(L_4, L_5) + 4F^{t-5}(L_4, L_5) - 2L_4^{t-11} - 3L_4^{t-12} + L_5^{t-11} - 3L_5^{t-13}. \quad (42)$$

Using (40) to deduce the values of $F^{t-4}(L_4, L_5)$ and $F^{t-5}(L_4, L_5)$ in (42) (by substituting $t-4$ and $t-5$ for t, respectively), equation (41) becomes $T_1^t = S^t + G(L_4, L_5)$ where $S^t = 3T_1^{t-1} - 2T_1^{t-2} + T_1^{t-3} - T_1^{t-5} - 7T_1^{t-6} - T_1^{t-8} + 6T_1^{t-9} + 7T_1^{t-10} + 3T_1^{t-11} + 14$ and $G(L_4, L_5) = -2L_4^{t-11} - 3L_4^{t-12} + L_5^{t-11} - 3L_5^{t-13}$.

By difference again, using $t-11$, $t-12$ substituted for t in (37) and $t-11$, $t-13$ substituted for t in (38), we get

$$\begin{aligned} T_1^t &= T_1^{t-1} + (S^t - S^{t-1}) - 2(T_1^{t-12} - T_1^{t-13} - T_1^{t-14} - 1) - 3(T_1^{t-13} - T_1^{t-14} - T_1^{t-15} - 1) \\ &\quad -3L_4^{t-15} - 6L_4^{t-16} - 3L_4^{t-17} - 6L_5^{t-16} - 15L_5^{t-17} - 12L_5^{t-18}. \quad (43) \end{aligned}$$

The second line of equation (43) involving terms in the L_4^j and L_5^j is exactly $-3F^{t-9}(L_4, L_5)$. Using (40) with $t-9$ substituted for t to get an expression for $-3F^{t-9}(L_4, L_5)$, equation (43) becomes

$$\begin{aligned} T_1^t &= 4T_1^{t-1} - 5T_1^{t-2} + 3T_1^{t-3} - T_1^{t-4} - T_1^{t-5} - 6T_1^{t-6} + 7T_1^{t-7} - T_1^{t-8} \\ &\quad +4T_1^{t-9} + 7T_1^{t-10} - 4T_1^{t-11} - 2T_1^{t-12} - 4T_1^{t-13} - T_1^{t-14} - 4. \end{aligned}$$

□