

Improved Upper Bound on m -Time-Relaxed k -Broadcasting Communication Networks ¹

A. Averbuch I. Peeri Y. Roditty

School of Computer Sciences

Tel Aviv University

Ramat Aviv, Tel Aviv 69978

Israel

e-mail: jr@post.tau.ac.il

Abstract

Broadcasting is a process in which an individual has an item of information which needs to be transmitted to all of the members in a network (which is viewed as a connected graph). k -broadcasting is a variant of broadcasting in which each processor can transmit the message to up to k of its neighbors in one time unit. Another variant of broadcasting is the concept of m -time-relaxed broadcasting, where we allow adding m time units to the optimal broadcast time, in exchange for fewer edges. The combination of m -time-relaxed and k -broadcasting is studied in this paper.

The result shown here is an improvement in the upper bound obtained by Harutyunyan and Listman in [9] that constructs a communication network model, namely, a connected graph, which admits the m -time-relaxed k -broadcasting.

In addition, we generalize (for $k > 1$) and prove a conjecture of Shastri [18] concerning the monotonicity of such m -time-relaxed models with respect to n (the number of vertices in this graph).

I. INTRODUCTION

Broadcasting is a one to all information dissemination process. More simply, broadcasting is a process in which one individual, or originator, has an item of information, or message, which needs to be communicated, or transmitted, to all members (processors) of a network. Broadcasting is subject to the following rules:

- 1) A processor may send a message to an adjacent processor only.
- 2) At a given time each processor will be in one of the following states:
 - a) receive a message;

- b) send a message to some neighbors;
- c) be idle.

More formally, we can view the communication network as a finite, connected, undirected graph on n vertices, where the set of vertices are considered as processors and each edge which connects two vertices assumed to be a direct communication link between these vertices. Then, we define broadcasting from a vertex v (*the originator*) as transmitting a message from v to every vertex in $V \setminus \{v\}$ using the above rules. This problem, that was introduced in [19], is a variation of the gossiping problem [10].

For basic graph theoretical definitions one may see [6] or [21].

We define the *broadcast number* of $v \in V$, $G = (V, E)$, denoted by $b(v)$, as the minimum time required to broadcast one message from v . The *broadcasting time* of G is defined as: $b(G) = \max\{b(v) \mid v \in V(G)\}$. Let $b(n)$ denote the minimal message broadcast time $b(G)$ over all graphs G with n vertices. A graph G is said to be *minimal broadcast graph* if $b(G) = b(n)$.

The problem of broadcasting in a general graph, namely, the problem of determining $b(v)$ for an arbitrary vertex in an arbitrary graph, was proved by Johnson (see [19]) to be NP-complete. On the other hand, in a tree with equal weights it happens to be linear [19]. Recently, a generalization of broadcasting in trees was obtained [1], where, positive weights were assigned to the edges or the vertices of the tree. For another results concerning broadcasting in various topologies, as, complete graphs, hypercubes, two, three and d -dimensional grids, one may see [3],[4],[11], [13], [17] and [20].

The notion of *m -Relaxed Broadcast Graphs (m -RBG)*, as appeared in [18], is a generalization of *1-RBG* that appeared in [5], and was motivated by exploring the sparsest possible graphs in which broadcasting can be accomplished in slightly more than the optimal time of $\lceil \log_2 n \rceil$.

Denote by $B^m(n)$ the number of edges in the sparsest possible graph on n vertices in which broadcasting can be accomplished in $\lceil \log_2 n \rceil + m$ steps (or time units). Such a graph is called *m -Relaxed Minimum Broadcast Graph (m -RMBG)*.

k -broadcasting, introduced in [7], is a generalization of the above notion of broadcasting, in which a processor can send the message to k (or fewer) neighbors in one time unit.

The *k -broadcast time of vertex v* is defined in a similar way as the broadcast time $b(v)$, enabling k -broadcasting, and is denoted: $b_k(v)$. The lower bound on the k -broadcast time is $b_k(v) \geq \lceil \log_{k+1} n \rceil$ since at each time unit the number of informed vertices is multiplied by at most $k + 1$. A *k -broadcast graph* is a graph G on n vertices where $b_k(G) = \lceil \log_{k+1} n \rceil$, that is a graph where k -broadcasting from any vertex is accomplished within at most $\lceil \log_{k+1} n \rceil$ time units.

The *k -broadcast function*, $B_k(n)$, is the minimum number of edges in any k -broadcast graph on n vertices. A *minimum k -broadcast graph* is a k -broadcast graph on n vertices having $B_k(n)$ edges.

Denoting the exact number of consecutive leading k 's in the $(k + 1)$ -ary representation of $n - 1$ by $L_k(n)$, Gringl and Peleg [5] proved that:

Theorem I.1: For all $n \geq 1$, $\frac{k}{2}(L_k(n) - 1)n < B_k(n) < [k(L_k(n) + 1) + 1]n$, i.e., $B_k(n) = \Theta(knL_k(n))$.

Asymptotically, Gringl and Peleg's construction produces the best-known upper bounds on $B_k(n)$ for most values of n .

Lazard [14] studied minimum k -broadcast graphs and, in particular, gave some values of $B_2(n)$, $B_3(n)$ and $B_4(n)$ for small values of n . More generally, Lazard showed that for $n \leq k + 1$, $B_k(n) = \frac{1}{2}n(n - 1)$. He also showed that $B_k(k + 2) = k + 1$ and that $B_k([k + 1]^m) = \frac{1}{2}km(k + 1)^m$ for $m \geq 1$.

Konig and Lazard [12] generalized some results on 1-broadcasting, found minimum k -broadcast graphs for all n in the range $k + 3 \leq n \leq 2k + 3$, and gave a more precise statement of Gringl and Peleg's lower bound on $B_k(n)$:

Theorem I.2:

$$B_k(n) \geq \frac{nk}{2}(m - p - 1).$$

where m is the length of $(k + 1)$ -ary representation of $n - 1$ and p is the index of the leftmost digit in that representation which is not equal to k .

In addition, they showed that $B_k(n) \leq \frac{nk}{2} \lceil \log_{k+1} n \rceil$ for $n \geq k$.

In 2000, Harutyunyan and Liestman [7] continued the search for minimum k -broadcast graphs and improved bounds on $B_k(n)$. They presented a new minimum 3-broadcast graph on 11 vertices for which $B_3(11) = 18$, improving a previous upper bound of 25 found by Lee and Ventura [15].

They also improved the lower bound achieved by Konig and Lazard [12], Theorem I.2. Using the $(k + 1)$ -ary representation of $n - 1$, $n - 1 = (\gamma_{m-1}\gamma_{m-2}\dots\gamma_0)_{k+1}$, where $0 \leq \gamma_i \leq k$ for $0 \leq i \leq m - 1$ and $\gamma_{m-1} \neq 0$. Denote $p = \max\{i | \gamma_i \neq k\}$. Define $\beta = \gamma_p$ if $p = 0$ or if $\gamma_0 = \gamma_1 = \dots = \gamma_{p-1} = 0$. Otherwise, $\beta = \gamma_p + 1$.

Theorem I.3:

When n is not a power of $k + 1$, $B_k(n) \geq \frac{nk}{2}(m - p - 1) + \frac{n}{2}\beta$, where m , p and β are as described above.

In addition, they investigated Moore graphs and proved the following Theorem.

Theorem I.4:

$$B_\Delta(\Delta^2 + 1) = \frac{\Delta(\Delta^2 + 1)}{2} \text{ if and only if there exists a } (\Delta, 2) \text{ Moore graph.}$$

In the same paper, they gave two constructions of k -broadcast graphs based on $(k + 1)$ -nomial trees. The first construction is similar to that of Gringl and Peleg [5] only in more detail and they give an exact value for the number of edges in these graphs. For some small n and k , the more precisely stated version of Gringl and Peleg's bound is better than the upper bounds reported by Lee and Ventura

[15]. The second construction deletes edges which are not used during the k -broadcast scheme they described, and therefore yields a better upper bound. For most large n and k , their improved bound was then the best upper bound.

In another paper Harutyunyan and Liestman [8] continued the investigation of k -broadcasting, focusing on k -broadcasting in trees. They described the construction of optimal trees and asymptotically estimated the number of vertices in these trees.

The next broadcasting concept, introduced in [5] (but only for $m = 1$), combines both concepts defined above. Namely m -time-relaxed k -broadcasting, which is k -broadcasting within at most $\lceil \log_{k+1} n \rceil + m$, $m \geq 0$, time units. The minimum number of edges in an m -relaxed k -broadcast graph on n vertices is denoted $B_k^m(n)$. A *minimum m -relaxed k -broadcast graph* (m -RkBG) is one with $B_k^m(n)$ edges.

Harutyunyan and Listman [9] described two methods to construct m -time-relaxed k -broadcast graphs and providing upper bounds on $B_k^m(n)$:

Theorem I.5:

$$B_k^m(n) \leq \min\left\{n + \frac{1}{2}(k+1)^{\lceil \log_{k+1} n \rceil - m} (k \lceil \log_{k+1} n \rceil - km - 2), (n-1) + \lfloor n/m \rfloor\right\}, \quad 1 \leq m \leq \lceil \log_{k+1} n \rceil$$

The case of $k = 1$, $m \geq 1$ was dealt with by Shastri [18], and a better bound was obtained recently by Averbuch, Roditty, Shoham in [2].

In this paper the bound of Theorem I.5 is improved for $k \geq 1$, namely,

Theorem I.6:

$$(i) \quad B_k^m(n) \leq n \left(1 + \frac{1}{(k+1)^{m-1}}\right) - \Theta\left(\left(\frac{n}{(k+1)^{m-1}}\right)^{\frac{2}{3}}\right), \quad n > (k+1)^{m+1}, n = (k+1)^t$$

$$(ii) \quad B_k^m(n) \leq A_{m,n_1} - \lfloor \frac{(k+1)^t - n}{(k+1)^{t-c} - 1} \rfloor \left((k+1)^{t-c-m+1} - k(t-c) - k(m-1) + 1 \right) - ((k+1)^t - n)$$

$$, n > (k+1)^{m+1}, (k+1)^{t-1} < n < (k+1)^t$$

A_{m,n_1} is the bound in (i) with $n_1 = (k+1)^t$ and $c = \lfloor \frac{2}{3}(t-m+1) \rfloor$.

$$(iii) \quad B_k^m(n) = n - 1, \quad n \leq (k+1)^{m+1}$$

Furthermore, Shastri [18] conjectured that $B_1^m(n) < B_1^m(n+1)$. In the following Theorem we prove

the generalized version of that conjecture, namely,

Theorem I.7: $B_k^m(n) < B_k^m(n + 1)$, for all $n, k \geq 1$.

This paper is organized as follows: Section II-A presents some necessary definitions and preliminary results. In Section II-B and II-C we presents the proof of Theorem I.6 and I.7, respectively.

II. m -TIME-RELAXED k -BROADCASTING GRAPHS

A. Preliminary Results

Construction of $(k + 1)$ -Nomial Trees

The basic underlying structure used in this paper is the $(k + 1)$ -nomial tree. This is a generalization of the *Binomial-Tree* introduced in [16]. This tree enables k -broadcasting from the root in $\lceil \log_{k+1} n \rceil$ time units where $n = (k + 1)^t$.

The tree is constructed as follows:

Definition II.1: The $(k + 1)^s$ -nomial tree on $(k + 1)^s$ vertices, denoted by T_{k+1}^s , is constructed recursively. The tree T_{k+1}^0 is a single vertex which is the root. T_{k+1}^s is constructed by connecting the roots of k copies of T_{k+1}^{s-1} to the root of another copy of T_{k+1}^{s-1} . This vertex is the root of T_{k+1}^s .

Each vertex in $V(T_{k+1}^s)$ is denoted by v_y^x where,

x - is a string $(x_1x_2\dots x_i)$ where $1 \leq i \leq s$ and $1 \leq x_j \leq k, 1 \leq j \leq i$.

$|x|$ denotes the distance of the vertex from the root, therefore this label is not unique.

y - is a $(k + 1)$ -ary string of length s .

That is, a numeric string representing an integer in base $k + 1$. This label is unique.

The root is denoted $v_{(0\dots 0)}$.

It has ks children which are denoted:

$$v_{(0\dots 01)}^{(1)}, \dots, v_{(0\dots 0k)}^{(1)}, v_{(0\dots 010)}^{(2)}, \dots, v_{(0\dots 0k0)}^{(2)}, \dots, v_{(10\dots 0)}^{(s)}, \dots, v_{(k0\dots 0)}^{(s)}.$$

Regarding the labelling of their children, only the first $k(s - 1)$ vertices, labelled $v_{(0\dots 01)}^{(1)}, \dots, v_{(0\dots 0k)}^{(1)}, v_{(0\dots 010)}^{(2)}, \dots, v_{(0\dots 0k0)}^{(2)}, \dots, v_{(10\dots 0)}^{(s-1)}, \dots, v_{(k0\dots 0)}^{(s-1)}$, have children.

Their labelling is as follows:

The children of the vertices labelled: $v_{(0\dots 01)}^{(1)}, \dots, v_{(0\dots 0k)}^{(1)}$, are:

$$v_{(0\dots 011)}^{(1,1)}, \dots, v_{(0\dots 0k1)}^{(1,1)}, \dots, v_{(10\dots 01)}^{(1,s-1)}, \dots, v_{(k0\dots 01)}^{(1,s-1)}, \dots, v_{(0\dots 01k)}^{(1,1)}, \dots, v_{(0\dots 0kk)}^{(1,1)}, \dots, v_{(10\dots 0k)}^{(1,s-1)}, \dots, v_{(k0\dots 0k)}^{(1,s-1)}$$

The children of the vertices labelled: $v_{(0\dots010)}^{(2)}, \dots, v_{(0\dots0k0)}^{(2)}$, are:
 $v_{(0\dots0110)}^{(2,1)}, \dots, v_{(0\dots0k10)}^{(2,1)}, \dots, v_{(10\dots010)}^{(2,s-2)}, \dots, v_{(k0\dots010)}^{(2,s-2)}, \dots, v_{(0\dots01k0)}^{(2,1)}, \dots, v_{(0\dots0kk0)}^{(2,1)}, \dots, v_{(10\dots0k0)}^{(2,s-2)}, \dots, v_{(k0\dots0k0)}^{(2,s-2)}$
and so on, so that the leaves are labelled:
 $v_{(10\dots0)}^{(s)}, \dots, v_{(k0\dots0)}^{(s)}, v_{(10\dots0k)}^{(1,s-1)}, \dots, v_{(k0\dots0k)}^{(1,s-1)}, \dots, v_{(1k0\dots0)}^{(s-1,1)}, \dots, v_{(kk0\dots0)}^{(s-1,1)}, \dots, v_{(1\dots1)}^{(1,\dots,1)}, \dots, v_{(1\dots1k)}^{(1,\dots,1)}, \dots, v_{(k\dots k1)}^{(1,\dots,1)}, \dots, v_{(k\dots k)}^{(1,\dots,1)}$

whereas the superscript label $(1, \dots, 1)$ is of length s .

In order to simplify the notation, we shall denote the tree T_{k+1}^s by T^s .

k -Broadcasting in $(k + 1)$ -Nomial Trees (Algorithm Tree)

The following algorithm, which is the k -broadcast scheme in the optimal $(k + 1)^s$ -nomial tree, shows that $b(v_{(0\dots0)}) \leq \lceil \log_{k+1} n \rceil$, and since $b(v_{(0\dots0)}) \geq \lceil \log_{k+1} n \rceil$, it follows

$$b(v_{(0\dots0)}) = \lceil \log_{k+1} n \rceil.$$

- 1) At $t = 1$, the root, $v_{(0\dots0)}$, broadcasts to the vertices $v_{(0\dots01)}^{(1)}, \dots, v_{(0\dots0k)}^{(1)}$.
- 2) At $t = 2$, the root broadcasts to vertices $v_{(0\dots010)}^{(2)}, \dots, v_{(0\dots0k0)}^{(2)}$ and the vertices $v_{(0\dots01)}^{(1)}, \dots, v_{(0\dots0k)}^{(1)}$ broadcast to their children $v_{(0\dots011)}^{(1,1)}, \dots, v_{(0\dots0k1)}^{(1,1)}, \dots, v_{(0\dots01k)}^{(1,1)}, \dots, v_{(0\dots0kk)}^{(1,1)}$, respectively.
- 3) Each vertex $v_{(y)}^{(x_1, x_2, \dots, x_{i-1})}$, $1 \leq i \leq s$, $0 \leq y \leq (k + 1)^s - 1$ that receives the message at time t' , will broadcast to k uninformed neighbors, labelled $v_{(y)}^{(x_1, x_2, \dots, x_{i-1})}$ at time $t' + 1$, to neighbors labelled $v_{(y)}^{(x_1, x_2, \dots, x_{i-1}2)}$ at time $t' + 2$ and so on.

Larger subtrees are called before smaller one. This is ensured by the labelling of the vertices.

When broadcasting from any vertex (not necessarily the root), the k -broadcast time is at most $\log_{k+1} n + s$. That is, the maximum time needed to broadcast from any vertex to the root, which is at most s , and then from the root to the whole tree $\log_{k+1} n$.

B. Proof of Theorem I.6

The structure used in the proof of Theorem I.6 is a m -Time-Relaxed Minimal k -Broadcast Graph (m -RkBG), for $k > 1$, since, as noted before, the case of $k = 1$ was dealt with already in [2].

To the sequel, $\lceil \log_{k+1} n \rceil$ is denoted by t .

Proof:

The proof is done by construction of m -RkBG in each of the cases and demonstrating the appropriate broadcast scheme in each case.

In order to simplify the notation, we shall denote the cube $Q_{(k+1)^s}$ by Q^s .

Construction

Case 1: $n > (k + 1)^{m+1}$, $n = (k + 1)^t$

The m - $RkBG$ is constructed as follows:

Let c be a positive integer (c will be determined later according to the optimal construction). Take $(k + 1)^c$ copies of the T^{t-c} tree denoted $T^{(i)}$, $0 \leq i \leq (k + 1)^c - 1$. Their vertices are labelled as explained in Section II-A, using the base 10 notation, with an additional index preceding the vertex label which denotes the tree. Therefore, their originators are: $v_{(0)(0)}$, $v_{(1)(0)}, \dots, v_{((k+1)^c)(0)}$.

Create a Q^c - cube using their originators.

The value of c will be determined .

Now additional edges need to be added to the trees:

Decompose each $T^{(i)}$ into vertex disjoint isomorphic copies of a T^{m-1} tree, (assuming $(m - 1 \leq t - c)$).

There are exactly $\frac{(k+1)^{t-c}}{(k+1)^{m-1}}$ such copies.

Each originator $v_{(i)(0)}$ is joined to one of the originators in each T^{m-1} subtree which is at distance at least 2 from $v_{(i)(0)}$.

Hence, by this process there are $\frac{(k+1)^{t-c}}{(k+1)^{m-1}} - [k(t - c) - k(m - 1)] - 1$ additional edges added to each tree.

Now since,

$(k + 1)^c$ - is the number of trees , $[(k + 1)^{t-c} - 1]$ - is the number of edges in each tree and $\frac{c}{2}(k + 1)^c k$ - is the number of edges in a Q^c - cube we obtain the following upper bound,

$$\begin{aligned} B_k^m(n) &\leq (k + 1)^c \left[(k + 1)^{t-c} - 1 + \frac{(k+1)^{t-c}}{(k+1)^{m-1}} - [k(t - c) - k(m - 1)] - 1 \right] + \frac{c}{2}(k + 1)^c k \\ &= n \left(1 + \frac{1}{(k + 1)^{m-1}} \right) - (k + 1)^c \left[k(t - m + 1) + 2 - \frac{3}{2}ck \right] \end{aligned} \quad (1)$$

Whence, the derivative of the right side of (1), with respect to c , is:

$$-(k + 1)^c \left[\ln(k + 1)[k(t - m + 1) + 2 - \frac{3}{2}ck] - \frac{3}{2}k \right]$$

Therefore, the minimum is achieved when:

$$c = \frac{2}{3}(t - m + 1) + \frac{4}{3k} - \frac{1}{\ln(k+1)}$$

It is easily observed that the term $\frac{4}{3k} - \frac{1}{\ln(k+1)}$ converges to 0.1, therefore it follows that:

$$c = \lfloor \frac{2}{3}(t - m + 1) \rfloor \quad (2)$$

Thus, substituting the value of c obtained in (2) into (1) yields (i) of Theorem I.6, as required.

Case 2: $n > (k+1)^{m+1}$, $(k+1)^{t-1} < n < (k+1)^t$

The m -RkBG is constructed as follows:

Take $(k+1)^c$ copies of the T^{t-c} tree (labelled $T^{(i)}$, $0 \leq i \leq (k+1)^c - 1$) and create a Q^c -cube using their originators in the same way as Case 1, with $n = (k+1)^t$.

Now delete vertices as needed using the following strategy:

First delete whole trees $T^{(i)}$ without their originators $v_{(i)(0)}$, as needed, as long as $(k+1)^t - n \geq |T^{(i)}| - 1$.

Next delete vertices from the same tree, starting with the leaves.

Hence, the total number of edges removed from the bound obtained in (1) is at least:

$$\begin{aligned} & \lfloor \frac{(k+1)^t - n}{(k+1)^{t-c} - 1} \rfloor \left[(k+1)^{t-c} - 1 + \frac{(k+1)^{t-c}}{(k+1)^{m-1}} - (k(t-c) - k(m-1) + 1) \right] \\ & \quad + (k+1)^t - n - [(k+1)^{t-c} - 1] \lfloor \frac{(k+1)^t - n}{(k+1)^{t-c} - 1} \rfloor \\ & = \lfloor \frac{(k+1)^t - n}{(k+1)^{t-c} - 1} \rfloor \left[\frac{(k+1)^{t-c}}{(k+1)^{m-1}} - (k(t-c) - k(m-1) + 1) \right] + (k+1)^t - n \quad (3) \end{aligned}$$

where, $\lfloor \frac{(k+1)^t - n}{(k+1)^{t-c} - 1} \rfloor \left((k+1)^{t-c} - 1 + \frac{(k+1)^{t-c}}{(k+1)^{m-1}} - (k(t-c) - k(m-1) + 1) \right)$ is the total number of edges deleted with the deletion of whole trees, and

$(k+1)^t - n - [(k+1)^{t-c} - 1] \lfloor \frac{(k+1)^t - n}{(k+1)^{t-c} - 1} \rfloor$ is the number of edges deleted due to the remaining vertices left to be deleted.

Notice that the actual number of deleted edges may be greater than the one subtracted since it is difficult to determine the exact values by using only parameters.

Case 3: $n \leq (k + 1)^{m+1}$

The m -RkBG is a T^m tree, and has $n - 1$ edges. To conform with the vertex labelling in Case 1 and Case 2, the vertices are labelled $v_{(0)(i)}$, $0 \leq i \leq (k + 1)^m - 1$.

In the following examples we demonstrate the above construction:

$$k = 3$$

$$m = 1$$

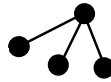
$$n = 26$$

$$t = \lceil \log_{k+1} n \rceil = \lceil \log_4 26 \rceil = 3$$

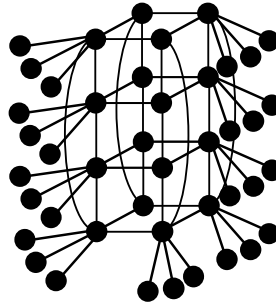
$$c = \lfloor \frac{2}{3}(t - m + 1) \rfloor = \lfloor \frac{2}{3}(3) \rfloor = 2$$

Since $16 < 26 < 64$ this is an example of Case 2.

Take 16 copies of T^1 -



Create a Q^2 - cube with their originators -



(For clarity, the figure shows a Q^2 - cube without internal edges and with 9 copies instead of 16 copies of the T^1 - tree.)

There is no need to add edges to the trees since their depth is less than 2.

Before deleting vertices, there are now $|V| = 64$, $|E| = 96$, which is the case for $n = 64$.

Now delete vertices as needed - first, delete 12 whole trees without their originators.

We have now $|V| = 28$, $|E| = 60$.

Next delete 2 vertices from the same tree -

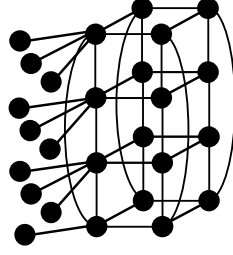


Fig. 1. 1-Time-Relaxed Minimal 3-Broadcast Graph for $n = 26$

Finally, $|V| = 26, |E| = 58$.

Broadcast Scheme in the various cases:

Case 1,2

Case i: The originator is $v \notin \{v_{(i)(0)} | 0 \leq i \leq (k+1)^c - 1\}$

Assume $v \in T^j, v \neq v_{(j)(0)}$.

1) v broadcasts to $v_{(j)(0)}$.

This takes at most m time units due to the additional edges in each tree.

2) $v_{(j)(0)}$ k -broadcasts to all of the vertices in the Q^c -cube using the first vertex label.

This takes another c time units.

3) $v_{(i)(0)}, 0 \leq i \leq (k+1)^c - 1$, k -broadcasts to all of the vertices in each $T^{(i)}$ using Algorithm Tree.

This takes at most $t - c$ time units.

Therefore, in total, broadcasting is accomplished within at most $t + m$ time units.

Case ii: The originator is $v \in \{v_{(i)(0)} | 0 \leq i \leq (k+1)^c - 1\}$

Assume $v = v_{(j)(0)}$.

1) $v_{(j)(0)}$ k -broadcasts to all of the vertices in the Q^c -cube using the first vertex label.

This takes c time units.

2) $v_{(i)(0)}, 0 \leq i \leq (k+1)^c - 1$, k -broadcasts to all of the vertices in each $T^{(i)}$ using Algorithm Tree.

This takes at most $t - c$ time units.

Therefore, in total, broadcasting is accomplished within at most t time units.

Case 3

Case i: The originator is $v \neq v_{(0)(0)}$

1) v broadcasts to $v_{(0)(0)}$.

Since the depth of the tree is m this takes at most m time units.

2) $v_{(0)(0)}$ k -broadcasts to the vertices in T^m using Algorithm Tree.

This is completed within at most t additional time units.

Therefore, in total, broadcasting is accomplished within at most $t + m$ time units.

Case ii: The originator is $v = v_{(0)(0)}$

1) $v_{(0)(0)}$ k -broadcasts to the vertices in T^m using Algorithm Tree.

This is completed within at most t additional time units.

Therefore, in total, broadcasting is accomplished within at most t time units.

Concluding Remark: The first method of Harutyunyan and Liestman [9] uses a constant r , $r = \lceil \log_{k+1} n \rceil - m$, to determine the ratio between the Q^c -cube and the $(k + 1)$ -nomial trees and uses conventional $(k + 1)$ -nomial trees (without additional edges). Since we determine the ratio by calculating the optimal construction, our results in Theorem I.6, are an improvement for all n , $n > (k + 1)^{m+2}$ and are equal for n in the range: $(k + 1)^{m-1} \leq n \leq (k + 1)^{m+2}$.

Regarding the second method, which achieves a better bound for small values of m , our results are an improvement for all n , k , when $m > 1$. For $m = 1$, it is difficult to accurately compare between the results. Specific results are: 1) $k = 2$, $2 \leq n \leq 4000$, 60% of the results are improved; 2) $k = 3$, $4 \leq n \leq 5000$, only 40% of the results are improved.

For further comparisons see the Appendix.

C. Proof of Theorem I.7

The proof of Theorem I.7, namely that $B_k^m(n)$ is a monotone function, is based on the results of Theorem I.6.

Define:

- 1) $A = \{n | n > (k + 1)^m, n = (k + 1)^t\}$
- 2) $B = \{n | n > (k + 1)^m, (k + 1)^{t-1} < n < (k + 1)^t\}$
- 3) $C = \{n | n \leq (k + 1)^m\}$

Obviously $A \cup B \cup C = N$, where N is the set of positive integers.

The required monotonicity of $B_k^m(n)$ is proved through the following cases:

Case 1: $n, n + 1 \in C$

In this case:

$$\begin{aligned} n &\leq (k+1)^m - 1, \\ n+1 &\leq (k+1)^m \end{aligned}$$

By Theorem I.6, $B_k^m(n) = n - 1$ and $B_k^m(n+1) = n$ so that obviously $B_k^m(n)$ is monotone increasing.

Case 2: $n \in C, n+1 \in B$

In this case it follows that $n = (k+1)^m$

Then by Theorem I.6, $B_k^m(n) = n - 1$.

Since $B_k^m(n) \geq n - 1$ for any n it follows that $B_k^m(n+1) \geq n$ which gives the required monotonicity.

Case 3: $n, n+1 \in B$

In this case:

$$\begin{aligned} n &> (k+1)^m, (k+1)^{t-1} < n < (k+1)^t, \\ n+1 &> (k+1)^m, (k+1)^{t-1} < n+1 < (k+1)^t \end{aligned}$$

To the sequel we denote:

$$\begin{aligned} p &= [(k+1)^{t-c-m+1} - (k(t-c) - k(m-1) + 1)] \\ \alpha_n &= \lfloor \frac{(k+1)^t - n}{(k+1)^{t-c-1}} \rfloor p + (k+1)^t - n \\ \alpha_{n+1} &= \lfloor \frac{(k+1)^t - (n+1)}{(k+1)^{t-c-1}} \rfloor p + (k+1)^t - (n+1) \end{aligned}$$

By Theorem I.6,

$$B_k^m(n) \leq A_{m,n_1} - \alpha_n \text{ and } B_k^m(n+1) \leq A_{m,n_1} - \alpha_{n+1}$$

The difference $\alpha_n - \alpha_{n+1}$ is:

$$p \left(\lfloor \frac{(k+1)^t - n}{(k+1)^{t-c-1}} \rfloor - \lfloor \frac{(k+1)^t - (n+1)}{(k+1)^{t-c-1}} \rfloor \right) + 1$$

Since $p \geq 0$ and $0 \leq \lfloor \frac{(k+1)^t - n}{(k+1)^{t-c-1}} \rfloor - \lfloor \frac{(k+1)^t - (n+1)}{(k+1)^{t-c-1}} \rfloor \leq 1$, it follows that, $\alpha_n - \alpha_{n+1} > 0$ which proves $B_k^m(n) < B_k^m(n+1)$.

Case 4: $n \in B, n+1 \in A$

In this case:

$$n > (k+1)^m, n = (k+1)^t - 1$$

Using Theorem I.6 one can easily obtain that $B_k^m(n) = B_k^m(n+1) - a$, where a is a positive integer, a fact which proves again that $B_k^m(n) < B_k^m(n+1)$.

Case 5: $n \in A, n+1 \in B$

In this case:

$$n > (k+1)^m, n = (k+1)^t \text{ and } (k+1)^t < n+1 < (k+1)^{t+1}, t = \log_{k+1} n.$$

So that we define $\gamma(n) = \lfloor \frac{2}{3}(t-m+1) \rfloor$, for a certain n .

By Theorem I.6,

$$\begin{aligned} B_k^m(n) &\leq n(1 + \frac{1}{(k+1)^{m-1}}) - (k+1)^{\gamma(n)} [k(t-m+1) + 2 - \frac{3}{2}\gamma(n)k] \\ B_k^m(n+1) &\leq B_k^m((k+1)^{t+1}) - ((k+1)^{t+1} - (n+1)) - \\ &\quad \lfloor \frac{(k+1)^{t+1} - (n+1)}{(k+1)^{t+1-\gamma(n+1)} - 1} \rfloor [(k+1)^{t+2-\gamma(n+1)-m} - (k(t+2-\gamma(n+1)-m) + 1)] \end{aligned}$$

Denote:

$$\begin{aligned} s &= \lfloor \frac{(k+1)^{t+1} - ((k+1)^t + 1)}{(k+1)^{t+1-\gamma(n+1)} - 1} \rfloor [(k+1)^{t+2-\gamma(n+1)-m} - (k(t+2-\gamma(n+1)-m) + 1)] \\ &\quad + ((k+1)^{t+1} - ((k+1)^t + 1)) \end{aligned}$$

In order to prove $B_k^m(n) < B_k^m(n+1)$ it is sufficient to show that

$$B_k^m((k+1)^{t+1}) - B_k^m((k+1)^t) > s, \text{ provided } s \geq 0.$$

First, the difference $B_k^m((k+1)^{t+1}) - B_k^m((k+1)^t)$:

$$\begin{aligned} B_k^m((k+1)^t) &\leq (k+1)^t(1 + \frac{1}{(k+1)^{m-1}}) - (k+1)^{\gamma(n)} [k(t-m+1) + 2 - \frac{3}{2}\gamma(n)k] \\ B_k^m((k+1)^{t+1}) &\leq (k+1)^{t+1}(1 + \frac{1}{(k+1)^{m-1}}) - (k+1)^{\gamma(n+1)} [k(t+1-m+1) + 2 - \frac{3}{2}\gamma(n+1)k] \end{aligned}$$

There are two possible cases:

- 1) $\gamma(n+1) = \gamma(n)$
- 2) $\gamma(n+1) = \gamma(n) + 1$

(4)

In which case $\gamma(n)$ is the defined c from above.

When $\gamma(n+1) = \gamma(n)$,

$$B_k^m((k+1)^{t+1}) - B_k^m((k+1)^t) = k(k+1)^t \left[1 + \frac{1}{(k+1)^{m-1}}\right] - (k+1)^c k$$

When $\gamma(n+1) = \gamma(n) + 1$,

$$B_k^m((k+1)^{t+1}) - B_k^m((k+1)^t) = k(k+1)^t \left[1 + \frac{1}{(k+1)^{m-1}}\right] - (k+1)^c \left[k \left[k(t-m) + \frac{k}{2} + \frac{3}{2} - \frac{3}{2}ck\right]\right]$$

Regarding s , we again have the possibilities as in (4):

When $\gamma(n+1) = \gamma(n)$,

$$s = \lfloor \frac{k(k+1)^t - 1}{(k+1)^{t+1-c-1}} \rfloor [(k+1)^{t+2-c-m} - (k(t+2-c-m) + 1)] + k(k+1)^t - 1$$

When $\gamma(n+1) = \gamma(n) + 1$,

$$s = \lfloor \frac{k(k+1)^t - 1}{(k+1)^{t-c-1}} \rfloor [(k+1)^{t-c-m+1} - (k(t-c-m+1) + 1)] + k(k+1)^t - 1$$

What's left is to compare between $B_k^m((k+1)^{t+1}) - B_k^m((k+1)^t)$ and s in the two cases.

When $\gamma(n+1) = \gamma(n)$,

$$B_k^m((k+1)^{t+1}) - B_k^m((k+1)^t) - s = \lfloor \frac{k(k+1)^t - 1}{(k+1)^{t+1-c-1}} \rfloor [k(t+1-c-m)] + 1$$

In several parts of s , $\lfloor \frac{k(k+1)^t - 1}{(k+1)^{t+1-c-1}} \rfloor$ is substituted by $k(k+1)^{c-1}$.

Since $\lfloor \frac{k(k+1)^t - 1}{(k+1)^{t+1-c-1}} \rfloor \geq 0$ and $[k(t+1-c-m)] \geq 0$,

$$B_k^m((k+1)^{t+1}) - B_k^m((k+1)^t) - s > 0$$

which proves $B_k^m(n) < B_k^m(n+1)$.

When $\gamma(n+1) = \gamma(n) + 1$,

$$B_k^m((k+1)^{t+1}) - B_k^m((k+1)^t) - s = \frac{k}{2}(k+1)^c [k(c+1) - 1] + 1$$

In s , $\lfloor \frac{k(k+1)^t - 1}{(k+1)^{t-c-1}} \rfloor$ is substituted by $k(k+1)^c$.

Since $[k(c+1) - 1] \geq 0$ for $k \geq 1$, it follows that

$$B_k^m((k+1)^{t+1}) - B_k^m((k+1)^t) - s > 0$$

which proves $B_k^m(n) < B_k^m(n+1)$.

REFERENCES

- [1] A. Averbuch, Y. Roditty and B. Shoham, "Computation of Broadcasting Multiple Messages in a Positive Weighted Tree", *J. of Combin. Math. Combin. Compu.* 35(2000),161-184.
- [2] A. Averbuch, Y. Rodity, B. Shoham, "Construction of Minimum Time-Relaxed Broadcasting Communication Networks", *J. of Combin. Math. Combin. Compu.* 43(2002),123-134.

- [3] P. Chinn, S. Hedetniemi and S. Mitchell, "Multiple message broadcasting in complete graphs." *Proceedings of the 10th SE Conference on Combinatorics, Graph Theory and Computing. Utilitas Math. Winnipeg (1979)*, 251-260.
- [4] E.J. Cockayne and A. Thomason, "Optimal multi-message broadcasting in complete graphs." *Proceedings of the 11th SE Conference on Combinatorics, Graph Theory and Computing. Utilitas Math. Winnipeg (1980)*, 181-199.
- [5] M. Grigni and D. Peleg, "Tight bounds on minimum broadcast networks", *SIAM J. Disc. Math.* 4(1991), 207-222.
- [6] F. Harary, "Graph Theory", *Addison-Wesley*, 1969.
- [7] H.A. Harutyunyan and A.L. Liestman, "Improved upper and lower bounds for k -broadcasting", *Networks* 37 (2001) 94-101.
- [8] H.A. Harutyunyan and A.L. Liestman, " k -Broadcasting in trees", *Networks* 38 (2001), 163-168.
- [9] H.A. Harutyunyan and A.L. Liestman, "On the monotonicity of the broadcast function", *Discrete Mathematics* 262 (2003) 149-157.
- [10] S.M. Hedetniemi, S.T. Hedetniemi and A.L. Liestman. "A Survey of Gossiping and Broadcasting in Communication Networks", *Networks* 18 (1988), 319-349.
- [11] P. Hell and K. Seyffarth. "Broadcasting in Planar Graphs", *Australas. J. Combin* 17 (1998), 309-318.
- [12] J.-C. Konig and E. Lazard, "Multiple k -broadcast graphs", *Discrete Applied Mathematics* 53 (1994), 199-209.
- [13] Oh-Heum Kwon and Kyung-Yong Chwa, "Multiple messages broadcasting in communication networks." *Networks* 26(1995), 253-261.
- [14] E. Lazard, "Broadcast in DMA-bound bounded degree graphs", *Discrete Applied Mathematics*, 37/38 (1992), 387-400.
- [15] S. Lee and J.A. Ventura, "An Algorithm for Constructing Minimal c -Broadcast Networks", *Networks* 38 (2001).
- [16] A. Proskurowski, "Minimum broadcast trees", *IEEE Trans. on Computers*, 5(1981),363-366.
- [17] Y. Roditty and B. Shoham, "On Broadcasting Multiple Messages in a d -dimensional Grid." *Discrete Applied Mathematics*, 75(1997), 277-284.
- [18] Aditya Shastri, "Time-relaxed broadcasting in communication networks." *Discrete Applied Mathematics*, 83 (1998), 263-278.
- [19] P.J. Slater, E.J. Cockayne and S.T. Hedetniemi, "Information Dissemination in Trees", *SIAM J. Comput.* 10 (1981) No. 4, 692-701.
- [20] Frances L. Van Scoy and Jeffrey A. Brooks, "Broadcasting multiple messages in a grid." *Discrete Applied Mathematics* 53(1994), 321-336.
- [21] D. West, Introduction to Graph Theory, *Simon @ Schuster A Viacom Company, Upper Saddle River, NJ 07458, (1996)*.

APPENDIX

The following section shows the comparison between the results achieved in this paper and the results obtained in [9].

This will be shown through specific results in the range: $k = 2$, $2 \leq n \leq 4000$, $1 \leq m \leq 4$ and $k = 3$, $2 \leq n \leq 5000$, $1 \leq m \leq 4$ and graphs for those ranges.

We shall use the abbreviation HL in the sequel to denote the results obtained by the authors of [9].

A. Graphs

Fig. 2. $k = 2, m = 1$

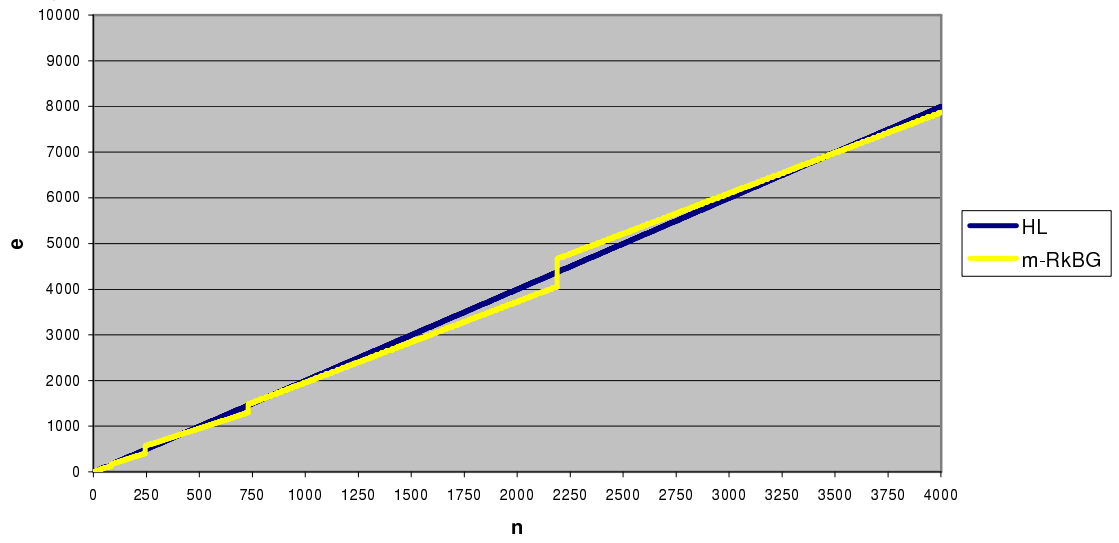


Fig. 3. $k = 2, m = 2$

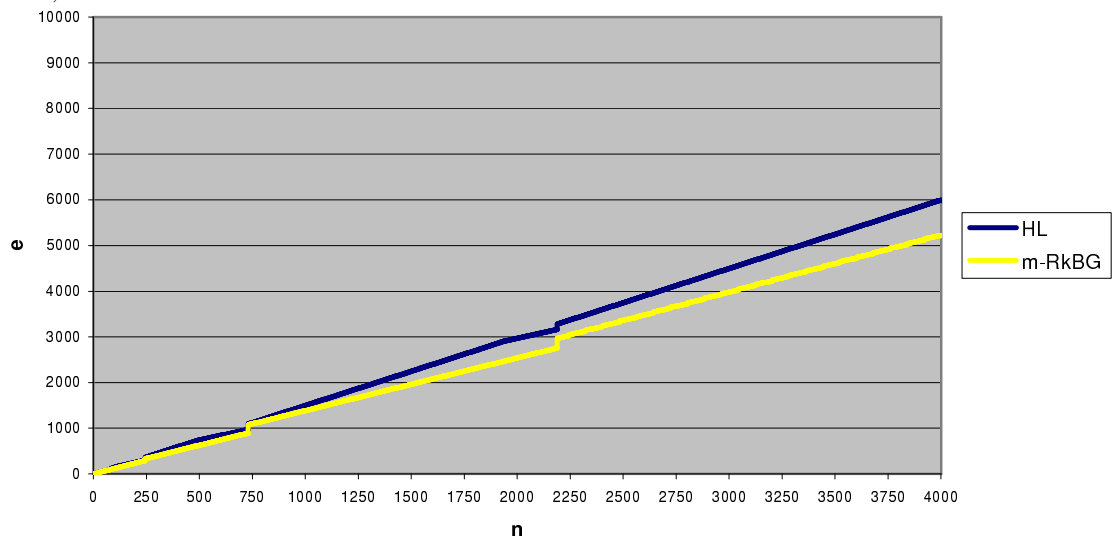


Fig. 4. $k = 2, m = 3$

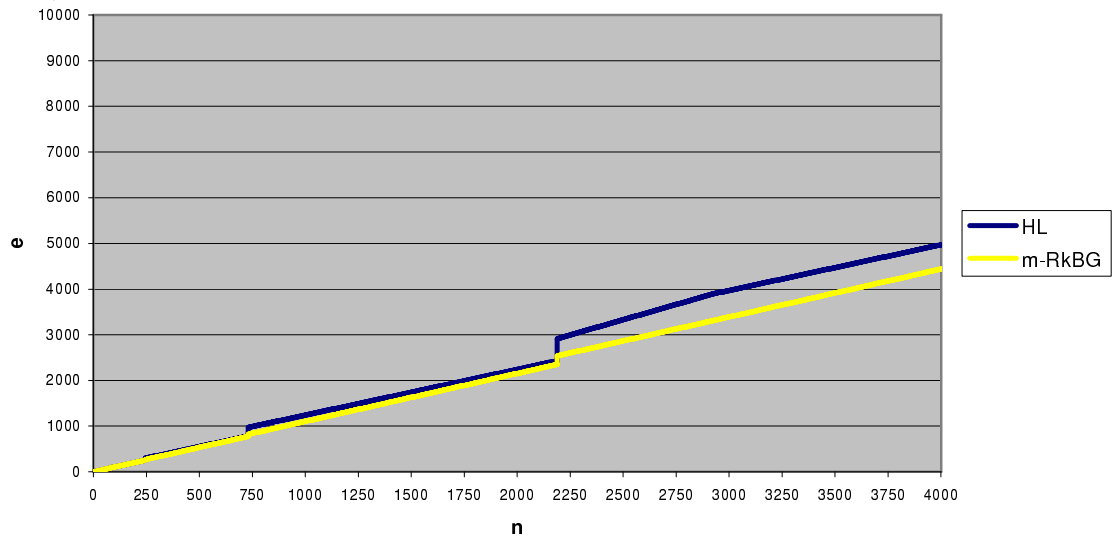


Fig. 5. $k = 2, m = 4$

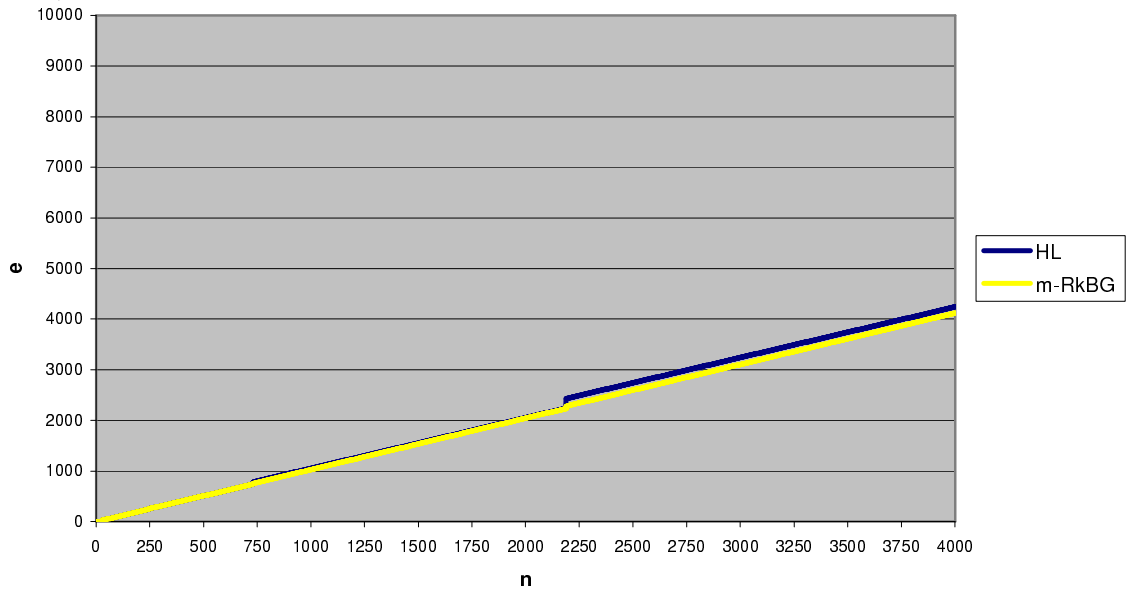


Fig. 6. $k = 3, m = 1$

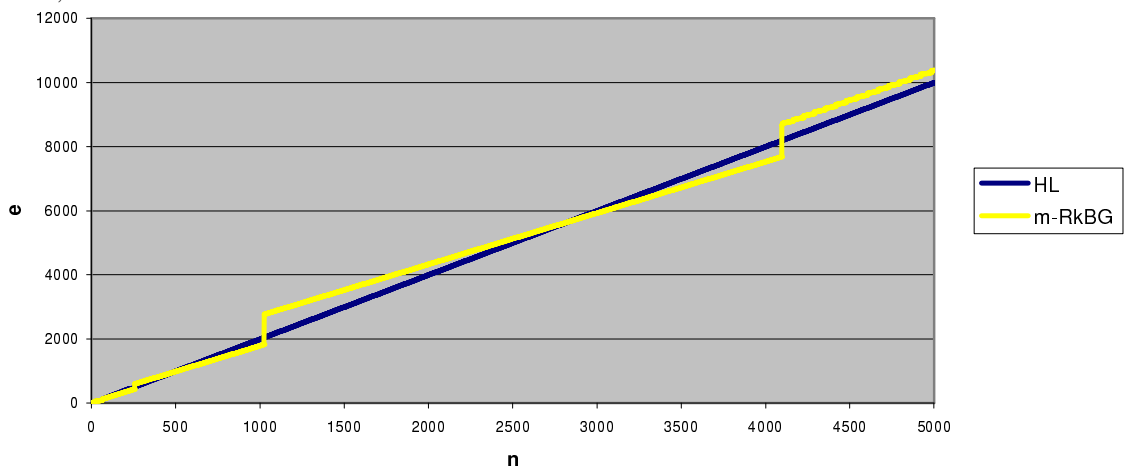


Fig. 7. $k = 3, m = 2$

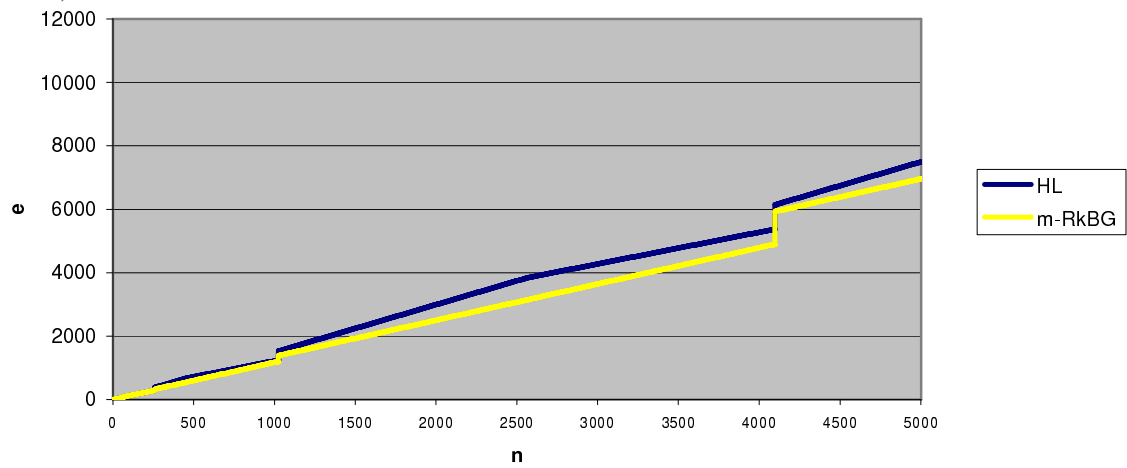


Fig. 8. $k = 3, m = 3$

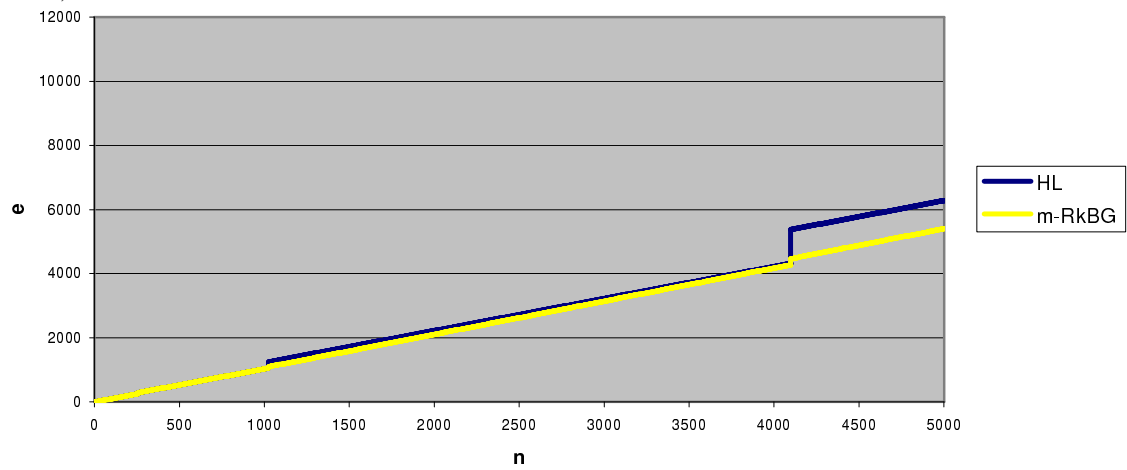
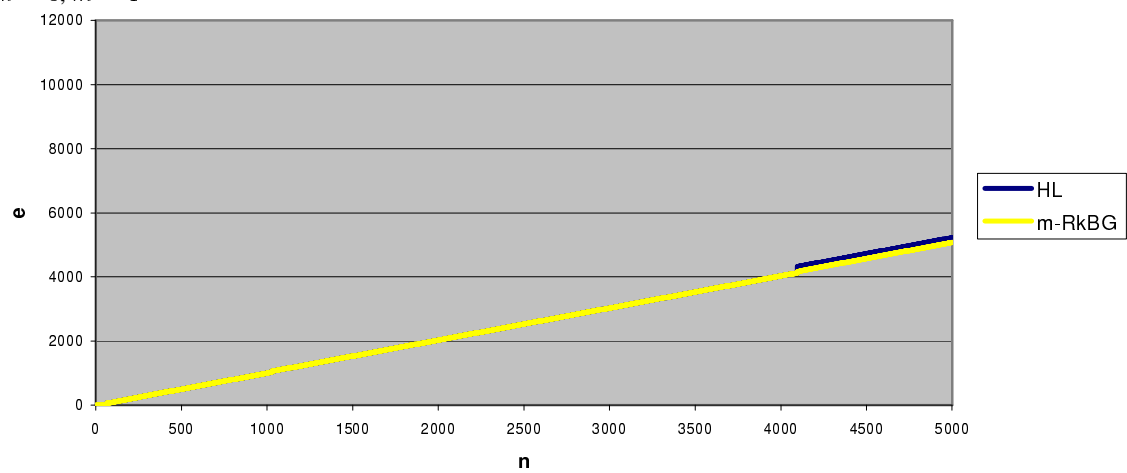


Fig. 9. $k = 3, m = 4$



B. Specific Results

TABLE I
 $k = 2, 2 \leq n \leq 4000, 1 \leq m \leq 4$

k=2		m							
		1		2		3		4	
n	t	HL	m-RkBG	HL	m-RkBG	HL	m-RkBG	HL	m-RkBG
2	1	1	1	–	–	–	–	–	–
3	1	2	2	–	–	–	–	–	–
4	2	4	4	3	3	–	–	–	–
...	2	–	–	–	–
9	2	9	9	8	8	–	–	–	–
10	3	19	19	10	10	9	9	–	–
...	3	–	–
27	3	36	36	27	27	26	26	–	–
28	4	55	49	37	37	28	28	27	27
...	4
81	4	135	126	90	90	81	81	80	80
82	5	163	164	122	103	91	91	82	82
83	5	165	165	123	104	92	92	83	83
...	5
242	5	483	404	296	287	251	251	242	242
243	5	485	405	297	288	252	252	243	243
244	6	487	571	365	334	298	265	253	253
245	6	489	572	366	335	299	266	254	254
...	6
411	6	821	822	615	525	465	444	420	420
412	6	823	823	617	526	466	445	421	421
413	6	825	824	618	527	467	446	422	422
...	6
728	6	1455	1295	971	890	782	773	737	737
729	6	1457	1296	972	891	783	774	738	738
730	7	1459	1473	1094	1073	972	820	784	751
731	7	1461	1474	1095	1074	973	821	785	752
...	7
863	7	1725	1726	1293	1230	1106	961	917	888
864	7	1727	1727	1295	1231	1107	962	918	889
865	7	1729	1728	1296	1232	1108	963	919	890
...	7
2186	7	4371	4049	3158	2753	2429	2348	2240	2231
2187	7	4373	4050	3159	2754	2430	2349	2241	2232
2188	8	4375	4660	3281	2971	2916	2539	2431	2278
2189	8	4377	4661	3282	2972	2917	2540	2432	2279
...	8
3472	8	6943	6944	5207	4575	4444	3887	3715	3586
3473	8	6945	6945	5208	4576	4445	3888	3716	3587
3474	8	6947	6946	5210	4577	4446	3889	3717	3588
...	8
3999	8	7997	7871	5997	5222	4971	4438	4242	4121
4000	8	7999	7872	5999	5223	4972	4439	4243	4122

TABLE II
 $k = 3, 2 \leq n \leq 5000, 1 \leq m \leq 4$

k=3		m							
		1		2		3		4	
n	t	HL	m-RkBG	HL	m-RkBG	HL	m-RkBG	HL	m-RkBG
2	1	1	1	–	–	–	–	–	–
3	1	2	2	–	–	–	–	–	–
4	1	3	3	–	–	–	–	–	–
5	2	7	7	4	4	–	–	–	–
...	2	–	–	–	–
16	2	18	18	15	15	–	–	–	–
17	3	33	49	19	19	16	16	–	–
...	3	–	–
32	3	63	64	34	34	31	31	–	–
33	3	65	65	35	35	32	32	–	–
...	3	–	–
64	3	96	96	66	66	63	63	–	–
65	4	129	133	96	97	67	67	64	64
66	4	131	134	98	98	68	68	65	65
...	4
77	4	153	154	109	109	79	79	76	76
78	4	155	155	110	110	80	80	77	77
79	4	157	156	111	111	81	81	78	78
...	4
255	4	479	431	287	287	257	257	254	254
256	4	480	432	288	288	258	258	255	255
257	5	513	598	384	325	289	289	259	259
258	5	515	599	386	326	290	290	260	260
...	5
485	5	969	970	709	589	517	517	487	487
486	5	971	971	710	590	518	518	488	488
487	5	973	972	711	591	519	519	489	489
...	5
1023	5	2045	1823	1247	1199	1055	1055	1025	1025
1024	5	2047	1824	1248	1200	1056	1056	1026	1026
1025	6	2049	2773	1536	1393	1249	1093	1057	1057
1026	6	2051	2774	1538	1394	1250	1094	1058	1058
...	6
2837	6	5673	5674	4117	3466	3061	2977	2869	2869
2838	6	5675	5675	4118	3467	3062	2978	2870	2870
2839	6	5677	5676	4119	3468	3063	2979	2871	2871
...	6
4095	6	8189	7679	5375	4895	4319	4271	4127	4127
4096	6	8191	7680	5376	4896	4320	4272	4128	4128
4097	7	8193	8671	6144	5926	5377	4465	4321	4165
4098	7	8195	8672	6146	5927	5378	4466	4322	4166
...	7
4999	7	9997	10383	7497	6263	6279	5403	5223	5076
5000	7	9999	10384	7499	6964	6280	5404	5224	5077