PROBLEMS RELATED TO BROADCASTING IN GRAPHS

Hayk Grigoryan

A thesis
in
The Department
of
Computer Science and Software Engineering

Presented in Partial Fulfillment of the Requirements
For the Degree of Doctor of Philosophy (Computer Science)
Concordia University
Montréal, Québec, Canada

September 2013

CONCORDIA UNIVERSITY
School of Graduate Studies

This is to certify that the thesis prepared

By: **Mr. Hayk Grigoryan**

Entitled: **Problems related to broadcasting in graphs**

and submitted in partial fulfillment of the requirements for the degree of

Doctor of Philosophy (Computer Science)

complies with the regulations of the University and meets the accepted standards with respect to originality and quality.

Signed by the final examining commitee:

______________________________ Chair
Dr. C. Wang

______________________________ External Examiner
Dr. A. Pelc

______________________________ External to Program
Dr. A. Iovita

______________________________ Examiner
Dr. V. Chvatal

______________________________ Examiner
Dr. J. Opatrny

______________________________ Thesis Supervisor
Dr. H. Harutyunyan

Approved by ______________________________
Dr. V. Haarslev, Graduate Program Director

September 6, 2013 ______________________________

Dr. C. Trueman, Interim Dean
Faculty of Engineering and Computer Science

Abstract

Problems related to broadcasting in graphs

Hayk Grigoryan, Ph.D.
Concordia University, 2013

The data transmission delays become the bottleneck on modern high speed interconnection networks utilized by high performance computing or enterprise data centers. This motivates the study directed towards finding more efficient interconnection topologies as well as more efficient algorithms for information exchange between the nodes of the given network.

Broadcasting is the process of distributing a message from a node, called the *originator*, to all other nodes of a communication network. Broadcasting is used as a basic communication primitive by many higher level network operations, which involve a set of nodes in distributed systems. Therefore, it is one the most important operations, which can determine the total efficiency of a given distributed system.

We study interconnection networks via modeling them as graphs. The results described in this work can be used for efficient message routing algorithms in switch based interconnection networks as well as in the choice of the interconnection topologies of such networks.

This thesis is divided into six chapters. Chapter 1 gives a general introduction to the research area and literature overview. Chapter 2 studies the family of graphs for which the broadcast time is equal to the diameter. Chapter 3 studies the routing and broadcasting problem in the Knödel graph. Chapter 4 studies the possible vertex degrees and the possible connections between vertices of different degrees in a broadcast graph. Using this, a new lower bound is obtained on broadcast function $B(n)$. Chapter 5 presents some miscellaneous results. Chapter 6 summarizes the thesis.

Acknowledgments

I would like to express my gratitude to my supervisor, Dr. Hovhannes Harutyunyan, for his excellent guidance, patience and invaluable encouragements.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Broadcasting is the process of distributing a message from a node, called the *originator*, to all other nodes of a communication network. The process of this transmission is called *message broadcasting*. Message broadcasting is one of the basic *collective operations* in parallel computing [32]. It is used in the implementation of other higher level collective operations such as: *accumulation* (or *gather* in MPI terms) where the data from all nodes of the network is collected in the single node, *scatter* during which the set of data(message) is divided into pieces and each piece is sent to different node, or global reduction operations(for example *sum*, *maximum*, *logical and/or*, etc.). Despite the importance, the effective implementation of collective operations is still not a resolved problem [106]. The criteria for estimating the efficiency are many. It can be the time required for broadcasting the message, the interconnection network construction cost, fault tolerance etc.

1.1 Definitions of the main problems in the research area

The real interconnection networks can be modeled in many different ways. By making different assumptions on the communication lines of the network, on the ability of nodes to simultaneously send/receive multiple messages and on many other network characteristics, we can get different models. One of the most common assumptions on the communications lines are:

1. Each call requires one unit of time.

2. Each call involves only one informed node and one of its adjacent nodes.

3. Each node can participate in at most one call per unit of time.

Under these assumptions, the model we are getting is realistic and at the same time, simple enough to be studied mathematically. This model in literature is called the *telephone model* or *classical model*. Many results obtained in this model directly or after small modifications usually are also applicable in the real world networks. For example, the first assumption about ignoring the message length may be seen as too artificial. However, we can still represent the real world networks with this model where the message transmission time usually depends on the length of the message. One approach to this, is to assume that large messages are divided into smaller ones, so that all the message transmission times can be considered as equal.

A network can be modeled as a connected graph $G = (V, E)$, where V is the set of all nodes and E is the set of all communication lines.

The *broadcast time* $b(s, G)$ or just $b(s)$ of a vertex s in a connected graph G is defined as the minimum time required to inform all the vertices of G from originator s.

We can define the *message broadcasting* from an originator s in a graph $G = (V, E)$ more formally. The message broadcasting is a sequence of vertex sets $\{s\} = S_0 \subset S_1 \subset ... \subset S_k = V$, where each S_i represents the set of informed vertices after the $i-$th time unit for all $i = 1, ..., k$. We also require that all the vertices of set $S_i \backslash S_{i-1}$ are connected by disjoint edges with set S_{i-1}, so they all can receive the message from S_{i-1} in one time unit via these connecting edges. The minimal k will be the *broadcast time* $b(s, G)$ of vertex s in G.

The broadcast time $b(G)$ of a graph $G = (V, E)$ is defined as:

$$b(G) = max\{b(s) \mid s \in V\}.$$

The minimum broadcast time in a graph G, i.e. the broadcast time of the optimally chosen vertex in G, is denoted by $b_{min}(G)$. The formal definition is the following:

$$b_{min}(G) = min\{b(v) \mid v \in V\}.$$

The set of calls used to distribute the message from originator s to all other vertices is called a *broadcast scheme* for vertex s. The broadcast scheme for s is a spanning tree rooted at s and all the communication lines are labeled with the transmission time. Each communication line is used exactly once and the message is always transmitted from a parent to a child. We will refer to this tree as *broadcast tree*.

The *broadcast center* of the given graph $G = (V, E)$, denoted $BC(G)$, is defined as:

$$BC(G) = \{v \mid v \in V, b(v, G) = b_{min}(G)\}.$$

Thus, $BC(G)$ is the set of vertices in G, with the smallest broadcast time.

We observe that in each time unit, an informed vertex may send the broadcast message to at most one of its uninformed neighbours. So after each time unit, the number of informed vertices can at most double. Therefore, to complete the broadcasting from a single vertex, in any graph G with n vertices we need at least $\lceil \log_2 n \rceil$ time units, thus

$$b(G) \geq \lceil \log_2 n \rceil .$$

A graph G on n vertices, with $b(G) = \lceil \log_2 n \rceil$ is called a *broadcast graph*. In other words, from each vertex of a broadcast graph, it is possible to complete the broadcasting in the theoretically minimum possible time. A broadcast graph with the minimum possible number of edges is called a *minimum broadcast graph (mbg)*. More formally, a graph $G = (V, E)$ with $|V| = n$ is an *mbg* if and only if $b(G) = \lceil \log_2 n \rceil$ and for each spanning subgraph $G' = (V, E')$, where $E' \subset E$, we have $b(G') > \lceil \log_2 n \rceil$.

An *mbg* represents the cheapest possible architecture to build a network, in which broadcasting can be accomplished in theoretically minimum possible time. The *broadcast function* $B(n)$ is defined as the number of edges in an *mbg* on n vertices.

There are two main directions of the research about broadcasting in graphs. Both directions received considerable attention in the literature. The first one is about finding various network related properties of a given graph. The second one is about designing graphs with certain properties. In this thesis we focus on two problems in the classical model of broadcasting:

1. **Broadcast Time problem**: Given a graph $G = (V, E)$ and a vertex $s \in V$, determine $b(s, G)$.

2. **Minimum Broadcast Graph problem (MBG)**: For any given n, construct a minimum broadcast graph on n vertices.

These two problems represent two different approaches to the broadcasting in graphs. The solution of the first problem for a given graph G, in some sense, tells as how good topology G is for a construction of an interconnection network where the broadcasting is expected to be a frequent operation. The second problem addresses the issue of optimizing the cost of the network construction between n nodes. Under the constraint that broadcasting should be completed in the theoretically fastest possible time, we are trying to construct a graph (interconnection topology) on n vertices (nodes) by using the smallest possible number of edges (links).

According to [63], the broadcast time problem was introduced in 1977 by Slater, Cockayne and Hedetniemi. Some initial research results on this problem are presented in [26, 104]. The minimum broadcast graph problem was introduced in [27]. Large sources of information about broadcasting and related problems are survey articles [33, 63, 67] and book [68].

1.2 Known results about the broadcast time problem

The NP-hardness of the broadcast time problem for general graphs is mentioned in [37] and proved in [104]. The proof used a reduction of the *three-dimensional matching (3DM)* problem to the broadcast time problem. In [88] it is shown that the broadcast time problem remains NP-hard even for regular planar graphs of degree 3. As in the case of other NP-hard problems, the minimum broadcast time problem is addressed by many different approximation algorithms and heuristics with reasonably good proximities and running times.

Only for very few graph families, an algorithm for the broadcast time problem with a polynomial time complexity is known. An algorithm with linear time complexity for the exact solution of the broadcast time problem in trees is presented by Slater, Cocakyne and Hedetniemi in [104]. Their algorithm finds in linear time the broadcast center of the given tree, and using it, determines the broadcast times of all vertices of the tree. In [95] Proskurowski suggested another linear time algorithm for the same problem, which without finding the broadcast center, determines the broadcast time of a vertex in a tree. A linear time algorithm is also known for the unicyclic graphs (connected graphs with only one cycle) [57, 58]. Algorithms for the exact solution of the broadcast time problem for a few other tree-like graph families are presented in [85].

1.2.1 Approximation algorithms for the broadcast time problem

The first approximation results for the broadcast time problem were an $O(\frac{\log^2 n}{\log\log n})$ approximation algorithm presented by Ravi [96] and an $O(\sqrt{n})$ additive approximation algorithm presented in [75]. The algorithm by Ravi is based on calculating the *poise* of a graph. The *poise* of the tree is defined as the sum of its diameter and the maximum degree. The poise of a graph G, denoted $P(G)$, is defined as the minimum poise of all its spanning tress. Determining the $P(G)$ is another NP-hard problem, but Ravi gives $O(\log n)$ approximation algorithm for it and shows the $b(G) = O(\frac{\log n}{\log\log n} \cdot P(G))$ relation between broadcast time of a graph and its poise. This yields the mentioned approximation algorithm.

Two different algorithms with $O(\log n)$ approximation ratio were presented in [4] and in [23]. The first algorithm uses the linear programming, while the second one is based only on a combinatorial approach. The best known approximation algorithm for broadcast time problem is presented in [24] and has a sub-logarithmic approximation ratio $O(\frac{\log n}{\log\log n})$.

There are several results on the lower bound of the possible approximation ratio as well. By using a reduction from the $E3-SAT$ problem, Schindelhauer in [102] proved that there does not exist a polynomial time approximation scheme ($PTAS$) for the broadcast time problem unless $P = NP$. In particular the author proved $57/56 - \epsilon$ inapproximability of the broadcast time problem for arbitrary $\epsilon > 0$. This result was improved in [23]. It was shown that it is NP-hard to approximate the solution of the broadcast time problem within a factor of $(3 - \epsilon)$ for arbitrary $\epsilon > 0$.

1.2.2 Heuristics for the broadcast time problem

We observe, that in each time unit of an optimal broadcast process, the edges used to send the message, form a *maximum matching* between sets of informed and uninformed vertices. For a graph $G = (V, E)$, let $b(S, G)$ denote the time needed to complete the broadcasting from set $S \in V$ of informed vertices in G. Using this notation, the following recurrence relation for $B(S, G)$ is presented in [101]

$$b(S, G) = 1 + min\{b(S \cup M(S), G) \mid M \text{ is a maximum}$$
$$\text{matching in } G \text{ between sets } S \text{ and } V - S\}.$$

As claimed in [101], by using this relation, a backtracking algorithm for the broadcast time problem with exponential running time is implemented in [100]. The complexity analysis of the algorithm was not given by the authors. Various rules used in the implementation to decrease the run time of the algorithm, make it very hard to perform such analysis. In the same article [101], three heuristics for the broadcast time problem are given. All of them are based on the relation above, but instead of choosing an optimal matching in each round, they choose the matching according to some well defined rule. In [75] it is shown that on some graphs all these three heuristics may yield $\Omega(\sqrt{n})$ time worse broadcast time than the optimal.

The comparison of some early suggested heuristics behavior on partial meshes (grids) can be found in [36]. Simulation results suggest that the best results are achieved by the heuristics presented in [6] and [60]. The heuristic in [6] is a based on a matching and has a complexity $O(Rmn \log n)$, where R is the broadcast time returned by the heuristic, n and m respectively are the number of vertices and edges of the graph. The heuristic presented in [60] archives similar or on some graph models from $ns - 2$ network simulator better results with significantly smaller complexity $O(Rm)$.

Additional results about heuristics can be found in [49, 62, 65, 97].

1.3 Known results about the minimum broadcast graph problem

There has been significant research on the problem of finding minimum broadcast graphs. Despite the considerable effort, an *mbg* is known only for very few n. Only three non isomorphic infinite graph families are known as minimum broadcast graphs. These families are the hypercube H_k, the recursive circulant graph $G(2^k, 4)$ introduced in [90] and the Knödel graph $W_{k,2^k}$ introduced in [73]. The hypercube and the recursive circulant graph give an mbg only for $n = 2^k (k \geq 1)$ [27, 73, 90], while the Knödel graph is an mbg also for $n = 2^k - 2 (k \geq 2)$[21, 70]. A detailed description and comparison of these three graph families can be found in [30].

1.3.1 Broadcast function

The *broadcast function*, denoted $B(n)$, is defined as the number of edges in an n vertex minimum broadcast graph. To determine the value of $B(n)$, we need to find a solution of the corresponding MBG problem, which means that determining $B(n)$ is a difficult problem.

The value of the broadcast function, obtained by the above mentioned three graph families is:

$$B(2^k) = k \cdot 2^{k-1},$$

$$B(2^k - 2) = (k-1) \cdot (2^{k-1} - 1).$$

For some small values of n, specially constructed minimum broadcast graphs are also known. Figure 1 illustrates minimum broadcast graphs for $2 \leq n \leq 15$ from [27]. All the currently known values of $B(n)$ and the corresponding references are presented in Table 1.

Since it is extremely difficult to find the exact values of the broadcast function, a considerable effort has been made on finding tight upper and lower bounds on $B(n)$. Usually to get an upper bound on $B(n)$ we should construct a broadcast graph on n vertices and pick the number of its edges. For a lower bound we should use the graph-theoretic properties of the minimum broadcast graphs, e.g. the minimum possible vertex degree and derive from them a lower bound on its number of edges.

n	$B(n)$	Ref.
1	0	[27]
2	1	[27]
3	2	[27]
4	4	[27]
5	5	[27]
6	6	[27]
7	8	[27]
9	10	[27]
10	12	[27]
11	13	[27]
12	15	[27]
13	18	[27]
14	21	[27]
15	24	[27]
17	22	[89]
18	23	[9, 109]
19	25	[9, 109]
20	26	[84]
21	28	[84]
22	31	[84]
23	33 or 34	[11, 84]
24	35 or 36	[5, 9]
25	$38, 39$ or 40	[5, 9]
26	42	[99, 110]
27	44	[99]
28	48	[99]
29	52	[99]
30	60	[9]
31	65	[9]
58	121	[99]
59	124	[99]
60	130	[99]
61	136	[99]
63	162	[78]
127	389	[47]
1023	4650	[103]
4095	22680	[103]
2^p	$p2^{p-1}$	[27, 73, 90]
$2^p - 2$	$(p-1)(2^{p-1}-1)$	[21, 70]

Table 1: Known values of broadcast function $B(n)$.

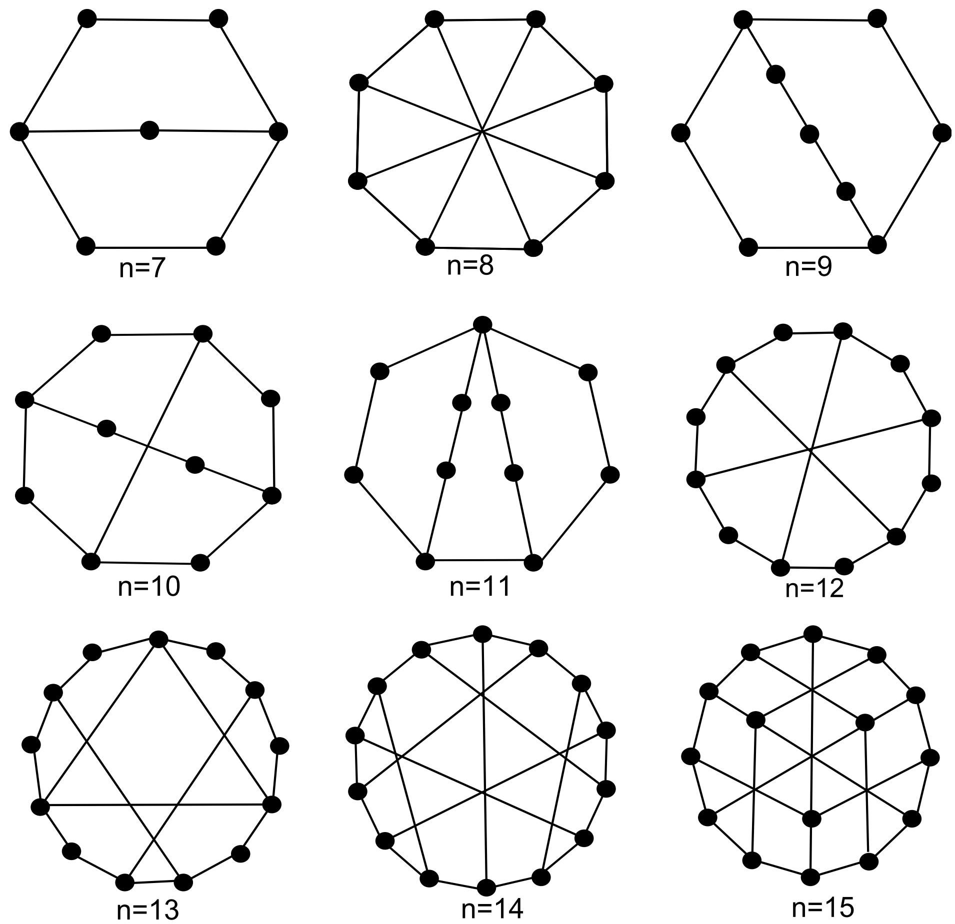

Figure 1: Minimum broadcast graphs for $7 \leq n \leq 15$.

Below is a list of some general properties of the broadcast function:

- $B(n) \leq \frac{1}{2}n \cdot \lfloor \log_2 n \rfloor$. This upper bound is obtained in [25] by a recursive construction of a broadcast graph on n vertices. The construction combines two or three smaller broadcast graphs on approximately $n/2$ or $n/3$ vertices. This construction was generalized in [12] where the combination of 5, 6 or 7 smaller broadcast graphs was used. These $5-way$, $6-way$ and $7-way$ split methods later generalized to $k-way$ split method described in [8].

- $B(n) \geq \frac{n}{2} \cdot (\lfloor \log_2 n \rfloor - \log_2(1 + 2^{\lfloor \log_2 n \rfloor} - n))$. This lower bound is obtained in [39]. Let p be the index of the leftmost 0 bit in the binary representation $(\alpha_{m-1}\alpha_{m-2}...\alpha_1\alpha_0)$ of $n-1$. In [74] the $B(n) \geq \frac{n}{2} \cdot (m-p-1)$ bound is obtained. This bound later improved in [52] to $B(n) \geq \frac{n}{2} \cdot (m-p-1+\beta)$ where $\beta = 0$ if $p = 0$ or if $\alpha_0 = \alpha_1 = ... = \alpha_{p-1} = 0$, otherwise $\beta = 1$. In [103] it has been shown that except few special forms of the binary representation of $n-1$, the $B(n) \geq \frac{n}{2} \cdot (m-p+\beta)$ bound holds.

- $B(n) \in \Theta(nL(n))$, where $L(n)$ is the number of leading 1's in the binary representation of $n-1$. This result is obtained in [40].

- $B(n) \leq n(m-k+1) - 2^{m-k} - \frac{1}{2}(m-k)(3m+k-3) + 2k$ for $n = 2^m - 2^k - r$, $0 \leq k \leq m-2, 0 \leq r \leq 2^k - 1$. This result is obtained in [51] where an *ad-hoc* construction method is described. The construction for arbitrary n produces a broadcast graph with the above mentioned number of edges. Considering the fact that $B(n) \in \Theta(nL(n))$, it is worth to be mentioned that this upper bound on $B(n)$ is $O(n)$ for n in the range $2^m + 1 \leq n \leq 2^m + 7 \cdot 2^{m-3}$ for arbitrary $m \geq 3$.

- $B(n)$ is non-decreasing for all n from interval $2^{m-1}+1 \leq n \leq 2^{m-1}+2^{m-3}$. This result is obtained in [53] and partially addressed a long-standing conjecture that $B(n)$ is monotone for n in the range $2^m + 1 \leq n \leq 2^{m+1}$ for all $m \geq 0$. This conjecture is mentioned in [9, 34].

We will speak more about the broadcast function in Chapter 4 which is dedicated to the problem of finding tight lower bounds on $B(n)$.

1.3.2 Broadcast graph construction methods

Recall that during the discussion of the properties of *broadcast function*, we mentioned methods for constructing broadcast graphs such as *recursive*, *ad-hoc* and $k-$*way split* methods. These methods allow to construct a broadcast graph for arbitrary n and give an upper bound on $B(n)$. In this section we will continue this discussion.

In [70], the *compounding* method of broadcast graph construction was presented. The method is based on combining smaller broadcast graphs which have certain type of vertex cover. In [8] authors generalize the method by introducing the concept of *soled h-cover*. Although it is hard to use the compounding for general n, many known values for $B(n)$ are obtained through this method.

A construction based on combining hypercubes is presented in [40]. Authors used generalized Fibonacci numbers for describing the broadcast schemes in constructed broadcast graphs. Thus referring their construction as the *Fibonacci method*.

In [108] the concepts of *center node* and *official broadcasting* was introduced. Using these concepts, the *generalized doubling* method was presented to construct a broadcast graph on $2n$ vertices using a broadcast graph on n vertices for some special forms of n.

Construction, which has improved most of the known upper bounds for $B(n)$, is presented in [51]. It is based on compounding Knödel graphs or hypercubes and after merging one or several vertices to get sparse broadcast graphs for most values of n.

Several other methods are presented in [1, 9, 39, 40, 47, 107, 109].

The reason why so many construction methods exist is that most of them give good results only for very special forms of n. Known upper bounds on $B(n)$ are obtained by combining different methods for broadcast graph construction. This makes it impossible to say that any single construction method is the best.

We will speak more about broadcast graph construction methods in Chapters 2 and 3.

1.4 Some well studied graph families

In this section we list some graph families with well studied properties related to networks. Such properties are the diameter, the maximum degree, the number of edges, broadcast time etc. Most of them are well known graphs from the graph theory. Others are specially introduced as good topologies for interconnection networks. The main properties of discussed graph families are summarized in Table 2. For some of these graphs, the exact value of broadcast time is not known. In such cases the best known lower and upper bounds in the form of an interval are given in Table 2.

- **The path graph P_n (see Figure 2):**

$$P_n = (\{v_1, v_2, ..., v_n\}, \{(v_i, v_{i+1}) \mid 1 \le i \le n-1\}).$$

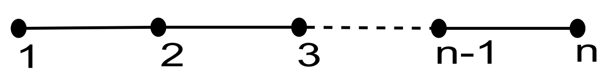

Figure 2: The path P_n.

- **The cycle(ring) graph C_n (see Figure 3):**

$$C_n = (\{v_1, v_2, ..., v_n\}, \{(v_i, v_{i+1}) \mid 1 \leq i \leq n-1\} \cup \{(v_1, v_n)\}).$$

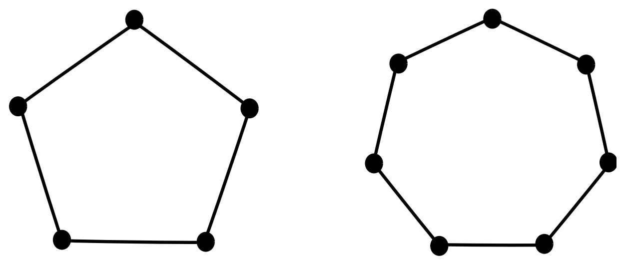

Figure 3: The C_5 and C_7 graphs.

- **The complete graph K_n (see Figure 4):**

$$K_n = (\{v_1, v_2, ..., v_n\}, \{(v_i, v_j) \mid 1 \leq i \leq n, 1 \leq j \leq n, i \neq j\}.$$

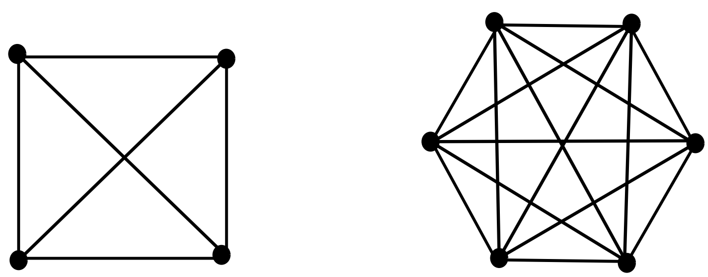

Figure 4: The K_4 and K_6 graphs.

- **A tree graph T_n :** A tree T_n is an acyclic connected graph on n vertices.
- **The binomial tree B_d (see Figure 5):** We will define the binomial tree recursively. The binomial tree of order 0 (B_0) is a single vertex. The binomial tree of order k (B_k) has a root vertex of degree k whose children are roots of binomial trees of order $k-1, k-2, ..., 0$ (in this order).
- **The hypercube graph H_d (see Figure 6):** $H_d = (V, E)$ where

$$V = \{(\alpha_1 \alpha_2 ... \alpha_d) \mid \alpha_i \in \{0, 1\}\},$$

$$E = \{((\beta_1 \beta_2 ... \beta_d), (\gamma_1 \gamma_2 ... \gamma_d)) \mid |\beta_1 - \gamma_1| + |\beta_2 - \gamma_2| + ... + |\beta_d - \gamma_d| = 1\}.$$

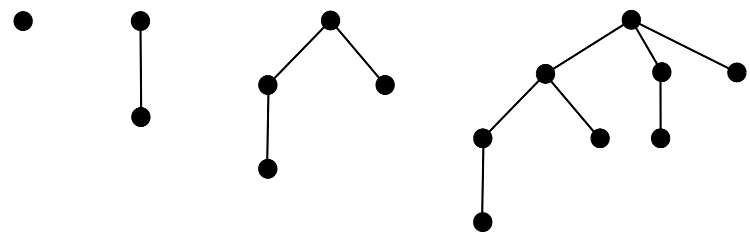

Figure 5: The B_0, B_1, B_2 and B_3 graphs.

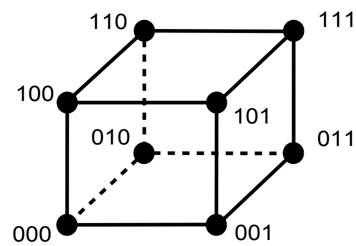

Figure 6: The H_3 graph.

- **The d-torus graph** $T(n_1, ..., n_d)$ **(see Figure 7):**

$$T(n_1, ..., n_d) = C_{n_1} \times ... \times C_{n_d}.$$

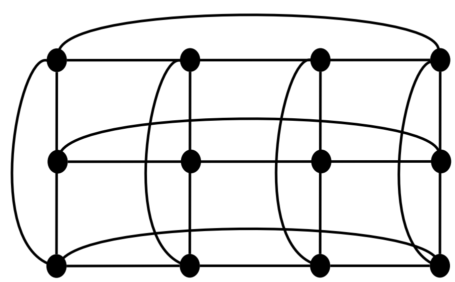

Figure 7: The $T(3, 4)$ graph.

- **The d-grid graph** $GD(n_1, ..., n_d)$ **(see Figure 8):**

$$GD(n_1, ..., n_d) = P_{n_1} \times ... \times P_{n_d}.$$

- **The cube connected cycles graph** CCC_d**:** $CCC_d = (V, E)$ where

$$V = \{(i, \alpha) \mid i = 0, 1, ..., d-1, \alpha \in \{0,1\}^d\},$$

$$E = \{((i, (\alpha_1\alpha_2...\alpha_d)), ((i+1) \bmod d, (\alpha_1\alpha_2...\alpha_d)))\} \cup$$

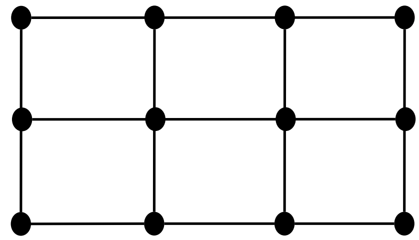

Figure 8: The $GD(3,4)$ graph.

$$\{(((i, \alpha_1...\alpha_j...\alpha_d)), (i, (\alpha_1...\bar{\alpha_j}...\alpha_d))) \mid j = 1, 2, ..., d\}.$$

We can construct CCC_d by replacing the vertices of the hypercube H_d with the cycles of length d. The edges of the hypercube are adjusted such that each vertex on a cycle will have two neighbours on the same cycle and one neighbour in some other cycle.

- **The binary de Bruijn graph** DB_d **(see Figure 9):** $DB_d = (V, E)$ where

$$V = \{(\alpha_1\alpha_2...\alpha_d) \mid \alpha_i \in \{0, 1\}\},$$

$$E = \{((\beta\alpha_2...\alpha_d), (\alpha_1\alpha_2...\alpha_{d-1}\gamma)) \mid \alpha_i, \beta, \gamma \in \{0, 1\}\}.$$

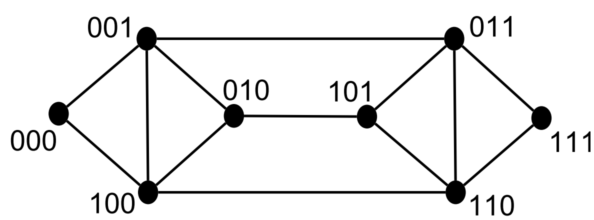

Figure 9: The DB_3 graph.

- **The star graph** S_d **(see Figure 10):** The star graph is a member of a class of graphs called Cayley graphs [3, 87]. The $d-star$ graph is the Cayley graph on the group S_d consisting of all permutations on d symbols, and the set of $d-1$ generators

$$g = \{(2, 1, 3, ..., d-1, d), (3, 2, 1, ..., d), ...,$$

$$(i, 2, ..., i-1, 1, i+1, ..., d), ..., (d, 2, 3, ..., 1)\}.$$

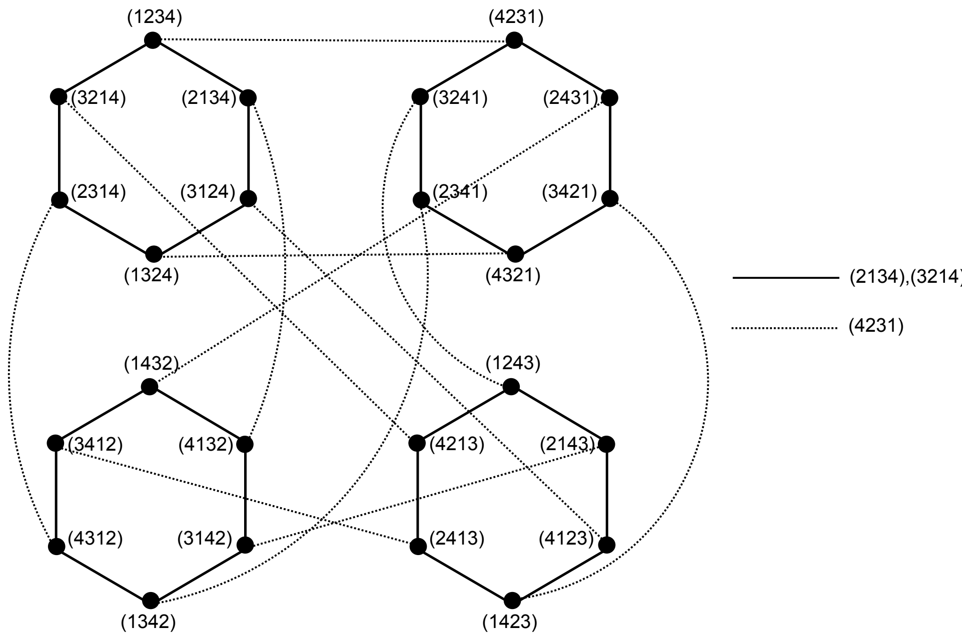

Figure 10: The S_4 graph.

- **The shuffle-exchange graph SE_d (see Figure 11)** : $SE_d = (V, E)$ where

$$V = \{(\alpha_1\alpha_2...\alpha_d) \mid \alpha_i \in 0, 1\},$$

$$E = \{((\alpha_1\alpha_2...\alpha_d), (\alpha_1\alpha_2...\bar{\alpha_d}))\} \cup \{((\alpha_1\alpha_2...\alpha_d), (\alpha_d\alpha_1...\alpha_{d-1}))\}.$$

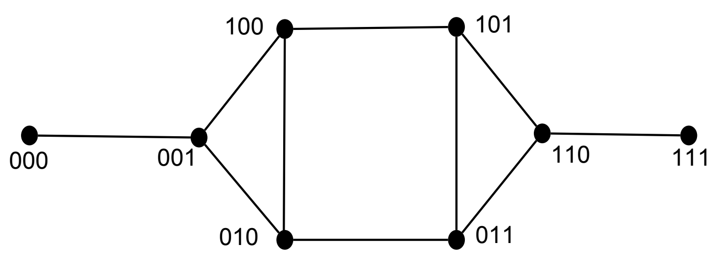

Figure 11: The SE_3 graph.

- **The butterfly graph BF_d** : $BF_d = (V, E)$ where

$$V = \{(i, \alpha) \mid i = 0, 1, ..., d-1, \alpha \in \{0, 1\}^d\},$$

$$E = \{((i, (\alpha_1\alpha_2...\alpha_d)), ((i+1) \bmod d, (\alpha_1\alpha_2...\alpha_d)))\} \cup$$

$$\{(((i, \alpha_1...\alpha_j...\alpha_d)), ((i+1) \bmod d, (\alpha_1...\bar{\alpha_j}...\alpha_d))) \mid j = 1, 2, ..., d\}.$$

- **The Knödel graph $W_{\Delta,n}$ (see Figure 12):** The Knödel graph is only defined for an even number of vertices. $W_{\Delta,n} = (V, E)$ where

$$V = \{(i, j) \mid i = 1, 2\ j = 0, 1, ..., n/2 - 1\},$$

$$E = \{((1, j), (2, (j + 2^k - 1) \bmod (n/2))) \mid j = 1, ..., n/2; k = 0, 1, ..., \Delta - 1\}.$$

We will speak more about the Knödel graph and its properties in Chapter 3. It is dedicated to the routing and broadcasting problem in the Knödel graph.

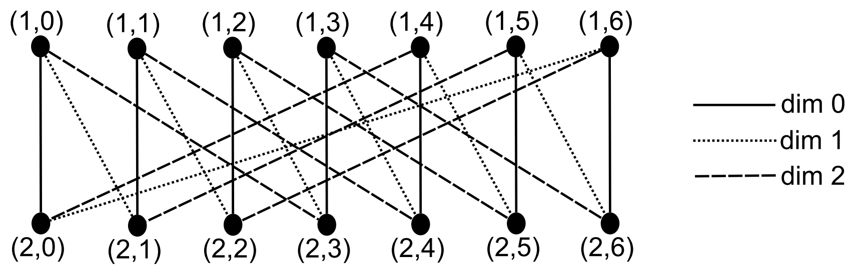

Figure 12: The $W_{3,14}$ graph.

- **The recursive circulant graph $RC(n, d)$ (see Figure 13):** The $RC(n, d)$ graph introduced in [90] and is only defined for $d \geq 2$. $RC(n, d) = (V, E)$ where

$$V = \{0, 1, ..., n - 1\},$$

$$E = \{(u, u + d^i \bmod n) \mid i = 0, 1, 2, ..., \lceil \log_d n \rceil - 1\}.$$

1.5 Other models of broadcasting

The model of broadcasting discussed in previous sections is called the *telephone model* or *classical model*. It is the most studied model, but there are some other models too. To have a complete introduction to the research area, we will shortly describe some of these models in the following.

1.5.1 k-broadcasting

The *k-broadcasting* is a variation of the classical broadcasting model where an informed vertex in a single round can simultaneously inform up to k neighbours. The

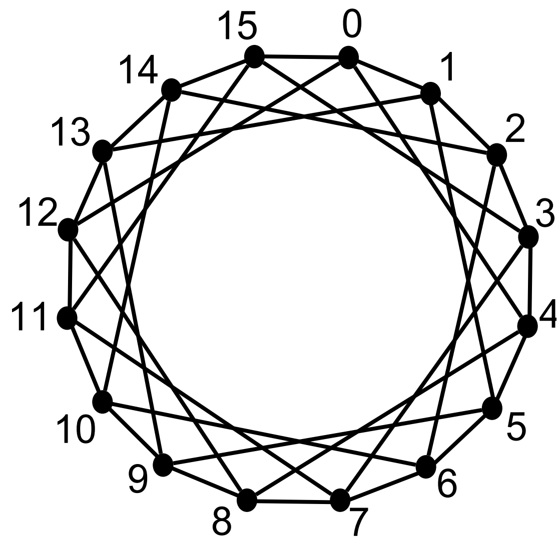

Figure 13: The $RC(16, 4)$ graph.

k-broadcasting in trees is studied in [52, 56, 76, 77, 95, 104]. Results for general graphs are presented in [40, 52, 74, 79, 80].

1.5.2 Open-path model

The *open-path* model was introduced by Farley in [26]. In this model in each time unit an informed vertex u may send the message to an uninformed vertex v via a path of arbitrary length. This is a generalization of the *classical model* where in each time unit only adjacent vertices can receive the message from an informed vertex. The paths used in each time unit, must be vertex disjoint. For this reason, the *open-path* model is sometimes called *vertex disjoint path mode broadcasting.*

1.5.3 Open-line model

The *open-line* model is also introduced in [26]. The model is similar to the *open-path* model, except that in each time unit the paths used for sending the message, must be edge disjoint. This model is also called *edge disjoint path mode broadcasting.*

1.5.4 Broadcasting with universal lists

The *broadcasting with universal lists* is a model of broadcasting where the order in which an informed vertex informs its neighbours is predefined and does not depend on the broadcast originator. The *broadcast scheme* in this model consists of a function, which assigns a single ordered list of neighbours to each vertex in a graph. This list is called the *universal list*. Regardless of the broadcast originator, each informed vertex informs its neighbours in the order defined by its universal list.

This model has two versions: *adaptive* and *nonadaptive*. In the adaptive version, each informed vertex tracks the vertices from which it receives the message. During sending the message, it skips them from its universal list. In the nonadaptive version an informed vertex does not know from which neighbour the message comes from. It sends the message to all of its neighbours.

The adaptive model of broadcasting with universal lists was introduced in [98] where the broadcasting in trees under this model was studied. In [19, 20], authors introduced the nonadaptive model of broadcasting with universal lists. They studied both adaptive and nonadaptive models in trees, rings, grids, toruses and complete graphs. An upper bound on nonadaptive broadcast time for two dimensional tori is presented in [61], where this broadcast model was studied under the name *orderly broadcasting*. In [71] the nonadaptive version of this model was studied in paths, complete k-ary trees, grids, complete graphs, and hypercubes. The nonadaptive broadcasting in trees was recently studied in [55].

1.5.5 Broadcasting in unknown networks or messy broadcasting

The *messy broadcasting* model was introduced in [2]. The main difference of this model, from the classical model, is that here the protocols of nodes are not coordinated. In this model, a node knows nothing about the topology of the network or the broadcast originator. The behavior of a node only depends on its all or some neighbours. Depending on the amount of information available for each node about its neighbours, 3 sub models were considered. Below, these sub models are listed in the decreasing order of information available in each node.

- **Model M_1:** Each node knows the state of its all neighbours, i.e which are informed and which are not. Using this information, in each time unit, an informed node sends the message to one of its uninformed neighbours.

- **Model M_2:** Each node keeps a list of all neighbours from which it received a message or sent a message. In each time unit, it sends the message to a neighbour not present in the list.

- **Model M_3:** Each node keeps a list of all neighbours to which it sent a message. In each time unit it sends the message to a neighbour not present in the list.

We note that in M_2 and M_3 models, a node may send a message to a node which is already informed. This is a "price" paid for the simplicity of the broadcast protocols.

The exact values for the worst-case messy broadcast time of various graphs such as complete graphs, paths, cycles, and complete d-ary trees for all three sub models are presented in [50]. More recent results are presented in [48]. The exact values of the worst-case messy broadcast time in M_1 and M_2 models and bounds in M_3 model are given for the hypercube. In [16, 45] multidimensional directed tori and complete bipartite graphs are studied. The average-case messy broadcast time of stars (claws), paths, cycles, complete d-ary trees and hypercubes are studied in [81].

The same model under the name *broadcasting in unknown networks* is studied in [38]. Very similar model with a limited knowledge of the network topology is studied in [86], where a vertex knows the topology of the network only within knowledge radius r from it.

1.5.6 Fault-tolerant broadcasting

The fault-tolerant communication is a huge area of research. It is assumed that the nodes and/or links of the network are not reliable. A faulty link may stop transmit messages and a faulty node may stop to send or receive messages. Number of results are presented specially for broadcasting and gossiping in faulty networks.

The *fault-tolerant broadcasting* model was introduced in [82], where the *k-tolerant broadcast function* $B_k(n)$ is defined. $B_k(n)$ is the minimum number of links in a network supporting k-tolerant broadcasting from any originator in theoretical smallest possible time. This research was continued in [13].

In [1], the *minimal k-fault tolerant broadcast graphs* were studied. Authors generalize the minimum broadcast graph problem. They study the networks where up to k links may fail, but any originator still must be able to complete the broadcasting in optimal time.

The survey article [91] provides detailed overview of the fault-tolerant communication problem in the context of broadcasting and gossiping. More recent overview of the known results can be found in [68].

1.5.7 Radio broadcasting

In the *radio broadcasting* model, an informed vertex in each time unit can send the message to all its neighbours simultaneously. Note that a vertex cannot send the message to a strict subset of its neighbours. A vertex is considered to be informed if it receives the messages from precisely one neighbour in a certain time unit. The intuition behind this constraint is that a message received from more than one neighbour in the same time unit gets corrupted. There is a considerable amount of literature regarding this model. See e.g. [15, 17, 18, 92, 93, 105].

Graph	Vert.	Edges	Diam.	Deg.	$b(G)$
$P_n(n \geq 3)$	n	$n-1$	n	2	$n-1$
$C_n(n \geq 3)$	n	n	$\left\lfloor \frac{n}{2} \right\rfloor$	2	$\left\lceil \frac{n}{2} \right\rceil$
K_n	n	$\frac{n(n-1)}{2}$	1	$n-1$	$\lceil \log_2 n \rceil$
H_d	2^d	$d \cdot 2^{d-1}$	d	d	d
$T(n_1, ..., n_d)$	$\prod_{i=1}^d n_i$	$d \prod_{i=1}^d n_i$	$\sum_{i=1}^d \left\lfloor \frac{n_i}{2} \right\rfloor$	$2d$	$\sum_{i=1}^d \left\lfloor \frac{n_i}{2} \right\rfloor \leq D \leq \sum_{i=1}^d \left\lfloor \frac{n_i}{2} \right\rfloor + max\{0, \sum_{i=1}^d n_i \text{mod} 2 - 1\}$ [33]
$GD(n_1, ..., n_d)$	$\prod_{i=1}^d n_i$	$\prod_{i=1}^d (n_i - 1)$	$\sum_{i=1}^d n_i - d$	$2d$	$\sum_{i=1}^d n_i - d$ [28]
CCC_d	$d \cdot 2^d$	$3d \cdot 2^{d-1}$	$\left\lfloor \frac{5d}{2} \right\rfloor - 2$	3	$\left\lfloor \frac{5d}{2} \right\rfloor - 2$ [83]
DB_d	2^d	2^{d+1}	d	4	$1.4404d \leq D(DB_d) \leq \frac{3}{2}(d+1)$ [94] [10]
S_d	$n!$	$\frac{n-1}{2} n!$	$\left\lfloor \frac{3(n-1)}{2} \right\rfloor$	$d-1$	$\lceil \log_2 n! \rceil \leq D(S_d) \leq \lceil \log_2 n! \rceil + \left\lceil \frac{7d}{4} \right\rceil + \lceil \log_2 d \rceil$
SE_d	2^d	$3 \cdot 2^{d-1}$	$2d-1$	3	$2d-1$ [66]
BF_d	$d \cdot 2^d$	$d \cdot 2^{d+1}$	$\left\lfloor \frac{3d}{2} \right\rfloor$	4	$1.7417d \leq D(BF_d) \leq 2d-1$ [72]
$W_{d,2^d}$	2^d	$d \cdot 2^{d-1}$	$\left\lceil \frac{d+2}{2} \right\rceil$	d	d
$W_{\Delta,n}$	n	$\frac{1}{2}\Delta n$	See [44] or Chapter 3	Δ	$2\left\lfloor \frac{1}{2} \left\lceil \frac{n-2}{2^{\Delta}-2} \right\rceil \right\rfloor + 1 \leq b(W_{\Delta,n}) \leq \left\lceil \frac{n-2}{2^{\Delta}-2} \right\rceil + \Delta - 1$ [41], Chapter 3
$RC(2^d, 4)$	2^d	$d \cdot 2^{d-1}$	$\left\lceil \frac{3d-1}{4} \right\rceil$	d	d

Table 2: The properties of the discussed graph families.

Chapter 2

Diametral Broadcast Graphs

This chapter studies the family of graphs for which the broadcast time is equal to the diameter. The *diametral broadcast graph (dbg) problem* is to answer the question whether for a given n and d a graph on n vertices can be constructed whose diameter and broadcast time are equal to d. Several dbg constructions are presented, which solve the dbg problem for all possible values of n and d. We also define the *diametral broadcast function* $DB(n, d)$ as the minimum possible number of edges in a dbg on n vertices and diameter d. We describe all the trees on n vertices with diametral broadcast time. Using these trees, we give the exact value of $DB(n, d)$ when tree based dbg construction is possible. For general case we give an upper bound on $DB(n, d)$. One of the presented dbg constructions produces for any n a broadcast graph, which is a subgraph of hypercube. Also note that in all constructions every vertex receives the message from the originator via shortest path.

2.1 Introduction

Recall that for the broadcast time of any graph G we have:

$$b(G) \geq \lceil \log_2 n \rceil .$$

Another obvious lower bound on the broadcast time is the diameter of the graph $D(G)$. We have:

$$b(G) \geq D(G).$$

Also recall that a graph G with $b(G) = \lceil \log n \rceil$ is called a *broadcast graph*. A broadcast graph with the minimum possible number of edges is called *minimum broadcast graph (mbg)*.

The construction of the graphs with $b(G) = \lceil \log_2 n \rceil$ (i.e. broadcast graphs) is a well studied problem in literature. See e.g. [8, 9, 12, 21, 25, 27, 39, 40, 47, 51, 53, 54, 70, 73]. However, the graphs with $b(G) = D(G)$ have not yet been studied.

In this chapter, we introduce the problem of existence of graphs with broadcast time equal to their diameter.

For a connected graph G on n vertices, the broadcast time can be any value from the range

$$\lceil \log_2 n \rceil \leq b(G) \leq n - 1.$$

The question is now whether for any fixed n and fixed d where $\lceil \log_2 n \rceil \leq d \leq n - 1$, we can construct a graph G such that

$$b(G) = D(G) = d.$$

We refer to this problem as *diametral broadcast graph(dbg)* problem. For convenience, we consider a form of the *dbg* problem where a diameter d and number of vertices n from the range $d + 1 \leq n \leq 2^d$ are given and a graph G on n vertices such that $b(G) = D(G) = d$ is to be constructed.

We are interested in construction of diametral broadcast graphs with as few edges as possible. In analogy to the broadcast function $B(n)$, we define the *diametral broadcast function* $DB(n, d)$ as the minimum possible number of edges in a dbg on n vertices and with diameter d.

We describe all the *diametral broadcast trees* and present how for a given d, we can construct all the trees with $b(T) = D(T) = d$. We describe the values of n and d for which a *dbg* construction is possible using only trees. Since the trees are the sparsest connected graphs, this gives us the exact value of $DB(n, d)$ for these particular values of n and d.

To cover the rest of values of n and d which were not covered by the tree based dbg construction, we present two other dbg constructions. We use hypercubes and binomial trees as construction blocks. Graphs constructed with these methods provide a solution for the dbg problem as well as upper bounds on $DB(n, d)$.

In the following, $V(G)$ and $E(G)$ will denote the set of vertices and the set of edges in a graph G.

We recall the recursive definition of the m dimensional *binomial tree* B_m. B_0 is a single vertex (root). For $m \geq 1$, B_m is obtained from two copies of B_{m-1} by connecting their roots and setting one of the roots as the root of B_m. Recall that $|V(B_m)| = 2^m$, $D(B_m) = 2m - 1$ and $b(r) = m$ where r is the root of B_m.

We define a *binomial subtree* as a tree created from binomial tree by removing some of its vertices such that the longest path from the root to a leaf remains intact. The dimension of a binomial subtree will be the dimension of the binomial tree from which it is created. For example, the minimal m dimensional binomial subtree is a path of length m.

2.2 Diametral broadcast graphs from trees

In this section we describe the class of trees with diametral broadcast time, i.e. trees T with $b(T) = D(T)$.

When $n \geq 2$ is fixed, the diameter of a tree T_n on n vertices may be any value in the range $2 \leq D(T_n) \leq n - 1$. From the results presented in [52, 69, 77], it follows that a broadcast time of a tree on n vertices must asymptotically be at least $\log_{(1+\sqrt{5})/2} n \approx 1.44042 \cdot \log_2 n$. Therefore, we have:

$$2 \leq D(T_n) \leq n - 1,$$

$$1.44042 \cdot \log_2 n \leq b(T_n) \leq n - 1.$$

From these inequalities it follows that when the diameter d is from the range

$$\log_2 n \leq d \leq 1.44042 \cdot \log_2 n,$$

trees cannot be a *diametral broadcast graph.*

Lemma 1. *All trees with $b(T) = D(T)$ have unique diametral path of length $D(T) = d$.*

Proof. The proof is by contradiction. Suppose we have tree T containing two paths of length d. Note that if the paths intersect in two or more vertices that are not connected by an edge, then we will have a cycle, which is not possible. It follows that there are only two possibilities: (a) the paths are disjoint (see Fig. 14a) or (b) the paths intersect either in a single vertex or the intersection is a path (see Fig. 14b).

In case (a), since the tree is connected, then there is a path connecting the two disjoint paths, which will crate a path in the tree with length greater than diameter d, thus contradiction.

In case (b), it is clear that the paths can intersect only in a "symmetric" way, that is before and after the "common" part, the paths must have equal lengths, i.e. $i = l$ and $j = s$, otherwise, we will have a path in the tree with length greater than diameter d. In this case, we have a vertex (e.g. v_0) and two other vertices (v_d and u_d) at distance d. From observation that at round i at most one vertex (either v_i or u_i) at distance i from originator v_0 can be informed, it follows that $b(T) \geq d + 1$, a contradiction. □

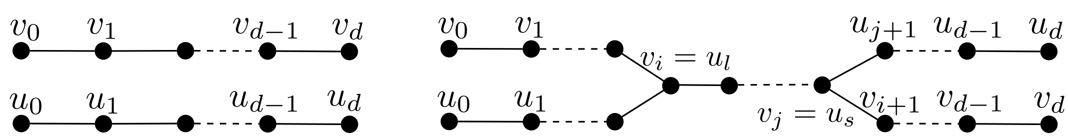

Figure 14: Possible configurations of two paths of the length d in a tree: (a) disjoint paths, (b) intersecting paths.

Theorem 2. *A tree T is a diametral broadcast graph, i.e. $b(T) = D(T) = d$, if and only if T contains a diametral path of length d where each inner vertex v_i ($i = 1, ..., d-1$) on the diametral path $v_0 ... v_d$ is a root of a subtree of a binomial tree of dimension $min\{i, d-i\} - 1$.*

Figure 15 illustrates the maximum tree with $D(T) = d = 10$.

Proof. If part: We will show that in the tree described above, d rounds are enough to broadcast from any originator.

The broadcast strategy from any originator will be to inform the closest vertex on the unique diametral path. After receiving the message, each vertex on the diameter will inform its uninformed neighbour(s) on the diametral path and then will continue broadcasting in its subtree.

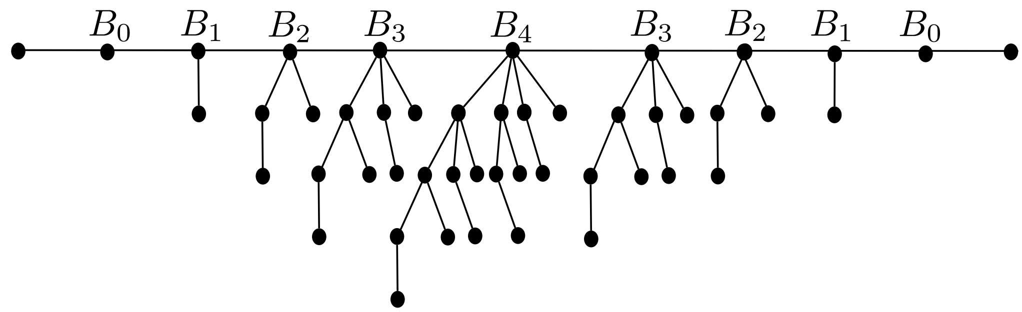

Figure 15: Tree with the maximum number of vertices and $b(T) = D(T) = 10$.

We observe that if the originator is v_0 or v_d, then we will need precisely d rounds to broadcast in T. Now let's assume that the originator u is in the subtree at vertex v_i ($i = 1, ..., d-1$). We note that since the tree dimension is $min\{i, d-i\} - 1$, it will take at most $min\{i, d-i\} - 1$ rounds to inform v_i from u. After receiving the message, v_i will need at most $max\{i, d-i\} + 1$ rounds to complete the broadcasting in T. Therefore,

$$b(u) \leq min\{i, d-i\} - 1 + max\{i, d-i\} + 1 = d.$$

Only if part: From Lemma 1 it follows that each tree with $b(T) = d$, contains unique path $P = v_0 v_1 ... v_d$ of length d. For end vertices of P we have:

$$b(v_0, T) = b(v_d, T) = d.$$

We have to show that from this it follows that T is the type of tree described in the theorem.

When broadcasting from v_0 (or v_d), each vertex after receiving the message should at first pass it to the next vertex on path P, otherwise after d rounds vertex v_d of P will not be informed. In the following rounds, each vertex v_i can inform additional vertices outside of the path P. It can be observed that for each inner vertex v_i ($i = 1, ..., d-1$) of P, the expression $min\{i, d-i\} - 1$ is the number of remaining rounds by which broadcast process must be finished. Since T is a tree, each vertex v_i is a root of a tree (possibly empty) outside of the path P. We complete the proof by noting that all vertices of this tree can be informed in $min\{i, d-i\} - 1$ rounds if and only if it is a connected rooted subtree of a binomial tree of dimension $min\{i, d-i\} - 1$ with the root v_i . $\square$

The family of diametral broadcast trees allows us to state the following theorem.

Theorem 3. *For given n and d such that $d+1 \leq n \leq \sqrt{2} \cdot 2^{\frac{d}{2}}$ and d is odd, or $d+1 \leq n \leq \frac{3}{2} \cdot 2^{\frac{d}{2}}$ and d is even, there exists a diametral broadcast graph on n vertices in the class of diametral broadcast trees and $DB(n,d) = n-1$.*

Proof. A dbg must be a connected graph, hence $DB(n,d) \geq n-1$. So, any diametral broadcast tree, in fact, is a minimum diametral broadcast graph. We will show the values of n and d for which a diametral broadcast tree T_d exists. For these special n and d we will have

$$DB(n,d) = n-1.$$

For determining the maximum possible number of vertices in T_d we need to consider the parity of d.

Suppose d is odd. In this case the maximum diametral broadcast tree described in Theorem 2 is actually a $\frac{d+1}{2}$ dimensional binomial tree. Thus, for odd d we have

$$|V(T_d)| \leq 2^{\frac{d+1}{2}} = \sqrt{2} \cdot 2^{\frac{d}{2}}.$$

Suppose d is even. We note that removing the middle vertex of the diameter, with the attached tree, from diametral broadcast tree described in Theorem 2, we will get $\frac{d}{2}$ dimensional binomial tree. By observing that the removed tree is a binomial tree of dimension $\frac{d-2}{2}$ we will get that for even d,

$$|V(T_d)| \leq 2^{\frac{d}{2}} + 2^{\frac{d-2}{2}} = \frac{3}{2} \cdot 2^{\frac{d}{2}}.$$

From these bounds on $|V(T_d)|$ follows that for any given n and d satisfying the theorem condition we can construct a diametral broadcast tree by picking the maximal tree T_d (on $\sqrt{2} \cdot 2^{\frac{d}{2}}$ or $\frac{3}{2} \cdot 2^{\frac{d}{2}}$ vertices) and, by keeping the diametral path intact, remove necessary number of vertices from subtrees till we get exact n vertices. $\square$

As we see, Theorem 3 does not cover most of the possible values of d and n. This motivates us to look at graph families other than trees for a complete solution of the *dbg* problem.

2.3 Diametral broadcast graphs from hypercube and binomial trees

In this section we present a dbg construction for all $2^{\frac{d}{2}} < n \leq 2^{d-1}$. The new construction uses a hypercube with binomial subtrees attached to its vertices.

For given positive integers m and r let us define a graph $G_{m,r}$ as follows:

1. $G_{m,r}$ consists of an m dimensional hypercube with binomial subtrees attached to its vertices such that each hypercube vertex is a root of a binomial subtree (possibly of dimension 0, i.e containing single root vertex).

2. Two of the binomial subtrees with the largest dimensions are attached to the opposite ends of a diametral path in the hypercube. The other subtrees are attached arbitrary.

3. $G_{m,r}$ graph has two different subtypes $G'_{m,r}$ and $G''_{m,r}$. $G'_{m,r}$ contains at least two binomial subtrees of maximal dimension r (i.e. all other subtrees have at most dimension r). $G''_{m,r}$ contains precisely one subtree of maximal dimension r and at least one of dimension $r-1$ (i.e. all other subtrees have at most $r-1$ dimension).

For example, the two types of $G_{2,2}$ graph with maximal number of vertices are presented in Figure 16. In the first type, the maximal subtree dimensions are $r = 2$, in the second $r = 2$ and $r - 1 = 1$. In Figure 17, $G_{2,2}$ graphs with minimal number of vertices are presented.

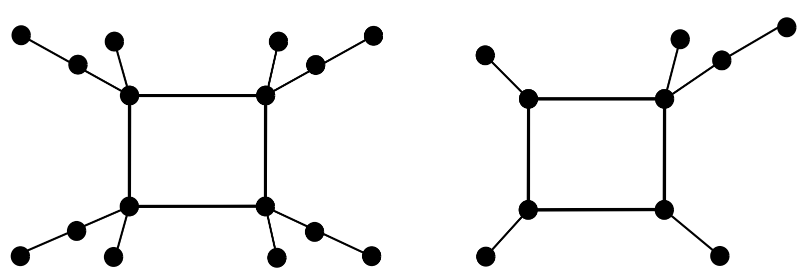

Figure 16: Maximal $G'_{2,2}$ with $d = 6$ and $G''_{2,2}$ with $d = 5$.

The following lemma proves that the $G_{m,r}$ graphs are diametral broadcast graphs.

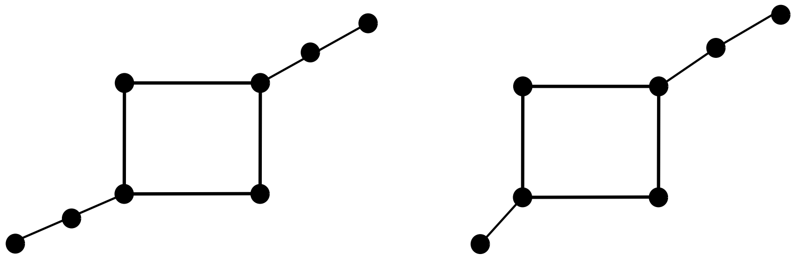

Figure 17: Minimal $G'_{2,2}$ with $d = 6$ and $G''_{2,2}$ with $d = 5$.

Lemma 4. *$G_{m,r}$ is a diametral broadcast graph with*

$$D(G'_{m,r}) = b(G'_{m,r}) = m + 2r,$$

$$D(G''_{m,r}) = b(G''_{m,r}) = m + 2r - 1.$$

Proof. We will show that (a) $D(G'_{m,r}) = m + 2r$, $D(G''_{m,r}) = m + 2r - 1$ and (b) $b(G'_{m,r}) = m + 2r$, $b(G''_{m,r}) = m + 2r - 1$.

(a) In $G'_{m,r}$ one of the diametral paths of m dimensional hypercube diameters will have two subtrees of dimension r attached to its end vertices. This will create a path of length $r + m + r = m + 2r$, which is obviously the longest possible in G, therefore

$$D(G'_{m,r}) = m + 2r.$$

Similarly, in $G''_{m,r}$ one of the hypercube diameters will have subtrees of dimension r and $r - 1$ attached to its end vertices. This will create a path of maximal length $r + m + (r - 1) = m + 2r - 1$, therefore

$$D(G''_{m,r}) = m + 2r - 1.$$

(b) To broadcast from any originator u in $G'_{m,r}$, it first informs the root of its binomial subtree which is a hypercube vertex. This can be done in at most r rounds. In the following m rounds, all the vertices of hypercube will be informed. This means that in at most $m + r$ rounds, we will have all the subtree roots informed. In the next r rounds, every hypercube vertex will inform the corresponding binomial subtree and complete broadcasting. Therefore

$$b(G'_{m,r}) = m + 2r.$$

The proof for $G''_{m,r}$ is similar to the above, except, that if we initially need r rounds to inform a hypercube vertex, because $G''_{m,r}$ has only one r dimensional binomial subtree, once all the hypercube vertices are informed, we will need only $r-1$ additional rounds to complete the broadcasting in binomial subtrees. Therefore in this case

$$b(G''_{m,r}) = m + 2r - 1.$$

$\square$

Theorem 5. *For given n and d such that $2^{\frac{d}{2}} < n \leq 2^{d-1}$, there exists a diametral broadcast graph $G'_{m,r}$ (or $G''_{m-1,r+1}$) where $m = 2\lceil \log_2 n \rceil - d$, $r = d - \lceil \log_2 n \rceil$ and*

$$DB(n,d) \leq (2\lceil \log_2 n \rceil - d - 2) \cdot 2^{2\lceil \log_2 n \rceil - d - 1} + n \leq \frac{1}{2} n (\lceil \log_2 n \rceil - 1).$$

Proof. In the following, for given n and d we will show how to determine the values of m and r to construct a $G_{m,r}$ graph with minimal number of edges. The number of edges in the constructed graph will provide an upper bound on $DB(n,d)$.

For given diameter d, graph $G'_{m,r}$ may contain a hypercube of dimension at most $m = d-2$. In this case we will have a $G'_{d-2,1}$ graph. The $|V(G'_{d-2,1})|$ will be maximal if all attached subtrees have dimension 1. This will give

$$|V(G'_{d-2,1})| = 2^{d-2} + 2^{d-2} = 2^{d-1}.$$

This means that when $n \leq 2^{d-1}$ we can construct a $G'_{m,r}$ dbg graph on n vertices.

Note that $G_{m,r}$ contains 2^m hypercube vertices and $n - 2^m$ tree vertices. There are $T_1, T_2, ..., T_{2^m}$ trees rooted at 2^m hypercube vertices, with $T_i = (V_i, E_i), i = 1, 2, ..., 2^m$. $|E_i| = |V_i| - 1$, thus $\sum_{i=1}^{2^m} |E_i| = \sum_{i=1}^{2^m} |V_i| - 2^m = n - 2^m$. Thus, $G_{m,r}$ contains $m \cdot 2^{m-1}$ hypercube edges and $n - 2^m$ tree edges. It follows that $E(G_{m,r}) = m \cdot 2^{m-1} + n - 2^m = (m-2) \cdot 2^{m-1} + n$. $|E(G_{m,r})|$ grows much faster with the size of hypercube m than with r. So in order to construct a sparse dbg, we should pick a $G_{m,r}$ with the smallest possible m.

For given n and d, in order to solve the dbg problem with $G'_{m,r}$ graph, we must have $n = |V(G'_{m,r})| \leq 2^{m+r}$ and $d = D(G'_{m,r}) = m + 2r$. After substituting $r = \frac{d-m}{2}$

from the second equality into the first inequality we will get

$$n \leq 2^{\frac{m+d}{2}} \Rightarrow m \geq 2\lceil \log_2 n \rceil - d.$$

We need $m \geq 0$ to be as small as possible, but, since r cannot be a fractional number, we also need $d - m$ to be even. Picking $m = 2\lceil \log_2 n \rceil - d$ will guarantee that $d - m = 2d - 2\lceil \log_2 n \rceil$ is always even. The lower bound $n > 2^{\frac{d}{2}}$ from the theorem condition guarantees that $m = 2\lceil \log_2 n \rceil - d > 0$. For the number of edges in $G'_{m,r}$ we have $|E(G'_{m,r})| = (m-2) \cdot 2^{m-1} + n$ where $m = 2\lceil \log_2 n \rceil - d$. This gives

$$DB(n,d) \leq |E(G'_{m,r})| = (2\lceil \log_2 n \rceil - d - 2) \cdot 2^{2\lceil \log_2 n \rceil - d - 1} + n.$$

By using the fact that $n \leq 2^{d-1} \Rightarrow d \geq \lceil \log_2 n \rceil + 1$, we can present a less tight, but simpler expression for the upper bound on $DB(n,d)$.

$$DB(n,d) \leq (2\lceil \log_2 n \rceil - (\lceil \log_2 n \rceil + 1) - 2) \cdot 2^{2\lceil \log_2 n \rceil - (\lceil \log_2 n \rceil + 1) - 1} + n =$$

$$(\lceil \log_2 n \rceil - 3) \cdot 2^{\lceil \log_2 n \rceil - 2} + n \leq$$

$$(\lceil \log_2 n \rceil - 3) \cdot 2^{\log_2 n - 1} + n =$$

$$\frac{1}{2} n(\lceil \log_2 n \rceil - 1).$$

We note that $G'_{m,r}$ contains at least $2^m + 2r$ vertices. This means that for given n, after choosing values of m and r, we may have a case when $n < 2^m + 2r$. This will not allow us to use $G'_{m,r}$ graph to solve the dbg problem. Instead, in this special case, we will use $G''_{m-1,r+1}$ graph. $G''_{m-1,r+1}$ is a dbg with diameter

$$D(G''_{m-1,r+1}) = (m-1) + (r+1) + r = m + 2r = D(G'_{m,r}).$$

Also $|E(G''_{m-1,r+1})| \leq |E(G'_{m,r})|$, so the claimed upper bound on $DB(n,d)$ remains valid for this case. For instance, when $d = 6$ and $n = 17$, according to the construction method, we pick $m = 2\lceil \log_2 n \rceil - d = 4$ and $r = \frac{d-m}{2} = 1$ and try to use $G'_{4,1}$ graph. However, it is not a valid dbg, since $G'_{4,1}$ contains at least 18 vertices. Instead, we pick $G''_{3,2}$ graph (see Figure 18). $\square$

The idea of using binomial trees attached to a hypercube is not new. For example

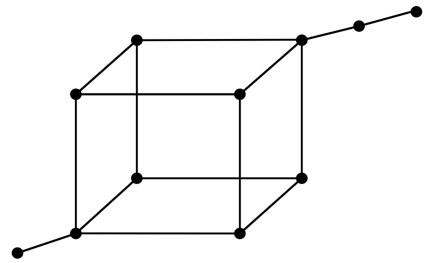

Figure 18: A *dbg* for $d = 6, n = 17$ based on $G''_{3,2}$ graph.

it is used in [53] and [40] to construct broadcast graphs.

2.4 Diametral broadcast graphs from subgraphs of hypercube

We recall that in the dbg problem, for a given diameter d, the number of vertices n can be any value from the range $d + 1 \leq n \leq 2^d$. The constructions from previous sections give us a solution for the dbg problem only for $d + 1 \leq n \leq 2^{d-1}$. In this section we present a hypercube based construction for all $2^{d-1} + 1 \leq n \leq 2^d$.

Graphs on $|V(G)| = n \geq 2^{d-1} + 1$ vertices and $b(G) = d$ are in fact broadcast graphs, meaning that to construct a dbg on n vertices for $n \geq 2^{d-1} + 1$ one would have to construct a broadcast graph with the additional condition that $b(G) = D(G)$. Thus, we cannot expect to have a trivial construction.

In this section we will construct broadcast graphs which will be subgraphs of a hypercube, i.e. obtained from a hypercube by removing some vertices and their adjacent edges.

Number of broadcast graph construction methods are known e.g. [25, 39, 40, 51, 70], but neither of them produces subgraphs of a hypercube. The new construction method will not provide improvement on the number of the edges over previously known broadcast graphs, but it will construct graphs with one important property of being hypercube subgraphs.

We denote the diametral broadcast graphs to be constructed by HS_d (d dimensional hypercube subgraph).

Theorem 6. *For given n and d such that $2^{d-1} < n \leq 2^d$ there exists a diametral* *broadcast graph and*

$$DB(n,d) \leq \frac{1}{2}(n \lceil \log_2 n \rceil - (\lceil \log_2 n \rceil - \lceil \log_2 x \rceil)x) \leq \frac{1}{2} n \lceil \log_2 n \rceil,$$

where $x = 2^d - n$.

Proof. We will (a) describe the construction of HS_d graph and will show that (b) $b(HS_d) = d$, (c) $D(HS_d) = d$, (d) $E(HS_d) \leq \frac{1}{2}(n \lceil \log_2 n \rceil - (2^d - n)) \leq \frac{1}{2} n \lceil \log_2 n \rceil$.

(a) Assume $n = 2^d - x$, where $0 \leq x < 2^{d-1}$. We pick the d dimensional hypercube H_d. Our goal is to remove x vertices from H_d such that $1 \leq x < 2^{d-1}$ and the remaining graph is a broadcast graph with diameter d. This will give a dbg for $2^{d-1} < n < 2^d$. For $n = 2^d$, we trivially will pick the H_d as a dbg.

The number of vertices to be removed can be presented as $x = 2^{\alpha_1} + 2^{\alpha_2} + ... + 2^{\alpha_k}$ where $d - 1 > \alpha_1 > \alpha_2 > ... > \alpha_k \geq 0$.

We will use a recursive approach in our construction. The recursion will be on k, i.e. on the number of terms in the above representation of x.

We require one additional condition on the removed vertices in order for the recursion to work. That is, all the x removed vertices must be contained in a certain $\lceil \log_2 x \rceil$ dimensional sub-hypercube of H_d.

The base case is when $k = 1$, i.e. we need to remove from a H_d hypercube 2^m vertices ($m < d$). We note that H_d can be presented as a $d-m$ dimensional hypercube where each vertex itself is a m dimensional hypercube. We will remove one of these m dimensional sub-hypercubes and the remaining graph will be a broadcast graph of diameter d.

For $k > 1$, the algorithm recursively reduces the problem of removing $x = 2^{\alpha_1} + 2^{\alpha_2} + ... + 2^{\alpha_k}$ vertices from H_d to the problems of removing 2^{α_1} and $2^{\alpha_2} + ... + 2^{\alpha_k}$ vertices from H_{d-1} hypercube. Below is the description of the recursive step.

We present H_d as H'_{d-1} and H''_{d-1} hypercubes with one-to-one mapped vertices (see Figure 19). We remove H'_{α_1} sub-hypercube containing 2^{α_1} from H'_{d-1} and recursively remove $x - 2^{\alpha_1}$ vertices from the sub-hypercube in H''_{d-1} which is the image of H'_{α_1}. We can always do this, since we know that $x - 2^{\alpha_1} < 2^{\alpha_1}$. Also we observe that all the removed vertices will be from the H_{α_1+1} sub-hypercube of H_d created by combining of H'_{α_1} and H''_{α_1}. This important observation will allow us to recursively remove $x - 2^{\alpha_1}$ vertices from predefined H''_{α_1} sub-hypercube of H''_{d-1}.

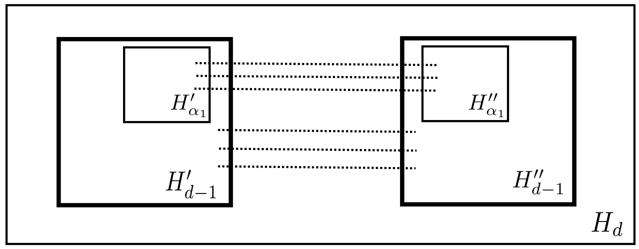

Figure 19: The recursive step in the construction of HS_d.

Figure 20 illustrates the recursive algorithm to construct *dbg*-s for $d = 4$ and $9 \leq n \leq 16$.

(b) Now we will show that $b(HS_d) = d$. According to the recursive assumption, $b(H''_{d-1}) = d - 1$. From the broadcast properties of a hypercube we know that $b(H'_{d-1}) = d - 1$. Using this we have to show that $b(HS_d) = d$.

Case 1: The originator belongs to H'_{d-1}. In the first round it will inform its corresponding vertex in H''_{d-1}. This can be done since each vertex from H'_{d-1} is connected to a vertex from H''_{d-1}. In the following $d - 1$ rounds, both informed vertices will complete the broadcasting in their sub-hypercubes. Hence $b(HS_d) = d$.

Case 2: The originator belongs to H''_{d-1}. Since we have removed more vertices from H'_{d-1} than from H''_{d-1}, we can claim that each vertex from H'_{d-1} is connected to a vertex from H''_{d-1} while the opposite is not true. The originator from H''_{d-1} may not have a neighbour belonging to H'_{d-1} in HS_d graph and we cannot use the same broadcast strategy as in the previous case. Instead, in first $d-1$ rounds the originator will inform all the vertices of H''_{d-1} and in the last round all these informed vertices will inform their corresponding vertices in H'_{d-1}. This can be done because, according to the recursive construction algorithm, $b(H''_{d-1}) = d - 1$. So $b(HS_d) = d$ in this case as well.

(c) Since $\alpha_1 < d - 1$, we can assume that H'_{α_1} constitutes at most half of H'_{d-1}. The other half with its image will create a H_{d-1} dimensional hypercube in the final graph on $2^d - x$ vertices. Combined with at least one not removed vertex from H''_{d-1}, this will guarantee that the resulted graph will have a diameter at least d. Since it also has a broadcast time d, it follows that diameter is at most d, hence $D(HS_d) = d$.

(d) We recall that all the x removed vertices are contained within a $\lceil \log_2 x \rceil$ dimensional hypercube. This means that each of them has at least $d - \lceil \log_2 x \rceil$ intact neighbours remaining in HS_d. Hence the sum of vertex degrees in HS_d is upper bounded by $n \lceil \log_2 n \rceil - (d - \lceil \log_2 x \rceil)x$. Based on this, we can claim the following upper bound

$$E(HS_d) \leq \frac{1}{2}(n \lceil \log_2 n \rceil - (\lceil \log_2 n \rceil - \lceil \log_2 x \rceil)x) \leq \frac{1}{2} n \lceil \log_2 n \rceil .$$

□

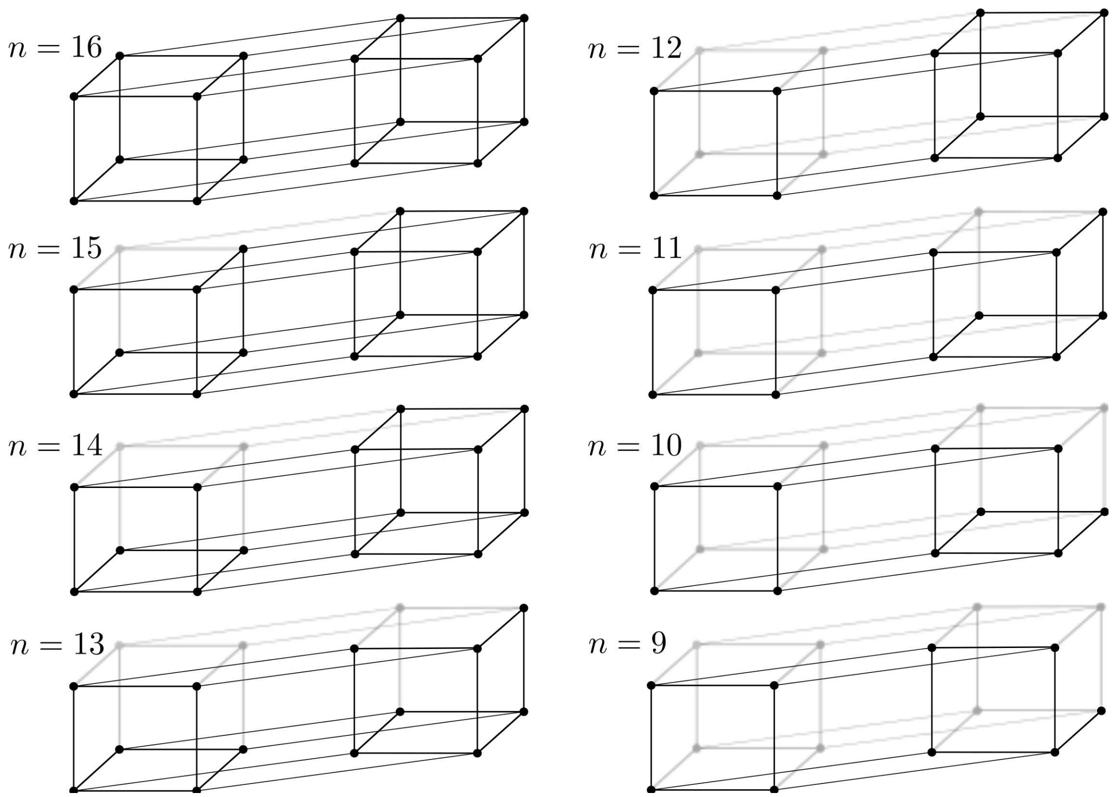

Figure 20: The HS_4 graph for $9 \leq n \leq 16$. The removed vertices end edges from H_4 hypercube are coloured gray.

2.5 Summary and discussion

To solve the *dbg* problem for all possible values of n and d, we presented three different constructions. The summary of the results on $DB(n, d)$ diametral broadcast function obtained via these constructions are presented in Table 3.

The first construction was based on trees and was providing the exact value of $DB(n, d)$. The limitation was that tree based construction was possible only for few values of n and d.

The second construction was based on a hypercube and binomial subtrees attached to it. The main challenge in the construction method was for any given values of n and d to optimally choose the hypercube and attached binomial subtrees dimensions such that the resulting graph will be a *dbg* with as few edges as possible.

In the last construction a *dbg* was obtained by removing certain vertices with adjacent edges from the hypercube. The main challenge was that our goal was to construct a *dbg* for values of n and d such that $2^{d-1} < n \leq 2^d$. The graphs with $|V(G)| = n > 2^{d-1}$ vertices and $b(G) = d = \lceil \log_2 n \rceil$ are actually broadcast graphs. This means that for this case the *dbg* problem becomes a harder version of the broadcast graph construction problem and we could not expect to have a simple solution for it. The constructed HS_d broadcast graphs have number of edges upper bounded by $\frac{1}{2}n \lceil \log_2 n \rceil$. Combined with the fact that the presented construction is relatively simpler than previously known ones and that produced broadcast graphs have a much simpler structure, that is, they are hypercube subgraphs, we can claim that the presented *dbg* construction method is a useful ad hoc construction of broadcast graphs in general.

Although we have not proved specially, we note that for all constructions and for all originators every vertex receives the broadcast message from the originator via a shortest path. For trees this property follows from the uniqueness of the path between any two vertices. For hypercubes the property follows from the observation that a vertex at some distance p from the originator gets informed via a path of length exactly p, which is obviously a shortest possible. Since $G_{m,r}$ is constructed using only trees and a hypercube it satisfies the property as well. Finally for HS_d graph the property follows from the fact that HS_d is a hypercube subgraph. This can be formally proven using recursion.

Range of n	Upper bound on $DB(n,d)$	Theorem
$(d, \sqrt{2} \cdot 2^{\frac{d}{2}}]$, d is odd $(d, \frac{3}{2} \cdot 2^{\frac{d}{2}}]$, d is even	$n-1$ (is optimal)	3
$(2^{\frac{d}{2}}, 2^{d-1}]$	$\frac{1}{2}n(\lceil \log_2 n \rceil - 1)$	5
$(2^{d-1}, 2^{d}]$	$\frac{1}{2}n \lceil \log_2 n \rceil$	6

Table 3: Summary of the presented results on $DB(n,d)$.

Chapter 3

Broadcasting and Routing in the Knödel graph

The Knödel graph $W_{\Delta,n}$ is a graph of even order and degree Δ where $2 \leq \Delta \leq \lfloor \log_2 n \rfloor$. We study the routing and broadcasting problem in $W_{\Delta,n}$. We give a tight bound on the distance between any two vertices in $W_{\Delta,n}$. We show that for almost all vertex pairs with labels $(1, x_1)$ or $(2, x_1)$ and $(1, x_2)$ or $(2, x_2)$

$$2 \left\lfloor \frac{|x_2 - x_1|}{2^{\Delta-1} - 1} \right\rfloor + 1 \leq dist(x_1, x_2) \leq 2 \left\lfloor \frac{|x_2 - x_1|}{2^{\Delta-1} - 1} \right\rfloor + 3,$$

where $dist(x_1, x_2)$ is the distance between them. Note that the presented expression uses only the second component of a vertex label, i.e. the partition of a vertex is not relevant. Using some of the results on distances, we present tight bounds on $b(W_{\Delta,n})$ for all even n and $2 \leq \Delta \leq \lfloor \log_2 n \rfloor$. We show that

$$2 \left\lfloor \frac{1}{2} \left\lceil \frac{n-2}{2^{\Delta} - 2} \right\rceil \right\rfloor + 1 \leq b(W_{\Delta,n}) \leq \left\lceil \frac{n-2}{2^{\Delta} - 2} \right\rceil + \Delta - 1.$$

The proofs are constructive and allow to construct a short path between any pair of vertices and to perform quick broadcasting from any vertex in $W_{\Delta,n}$.

3.1 Introduction

The Knödel graph $W_{\Delta,n}$ is a regular graph of even order and degree Δ where $2 \leq \Delta \leq \lfloor \log_2 n \rfloor$. It was introduced by Knödel for $\Delta = \lfloor \log_2 n \rfloor$ in 1975 and was used in an optimal gossiping algorithm [73]. For smaller Δ, the Knödel graph is formally defined in 2001 [35].

Multiple definitions are known for the Knödel graph. We use the following definition from [35], which explicitly presents the Knödel graph as a bipartite graph.

Definition 7. *The Knödel graph on an even number of vertices n and of degree Δ where $2 \leq \Delta \leq \lfloor \log_2 n \rfloor$ is defined as $W_{\Delta,n} = (V, E)$ where*

$$V = \{(i, j) \mid i = 1, 2;\ j = 0, ..., n/2 - 1\},$$

$$E = \{((1, j), (2, (j + 2^k - 1) \bmod (n/2))) \mid \\ j = 1, ..., n/2; k = 0, 1, ..., \Delta - 1\}.$$

We say that an edge $((1, j'), (2, j'')) \in E$ is r-dimensional if $j' = (j'' + 2^r - 1) \bmod (n/2)$ where $r = 0, 1, ..., \Delta-1$. In this case, $(1, j')$ and $(2, j'')$ are called r-dimensional neighbours. Also, we say that the edge is modular when $j' + 2^r - 1 > n/2$.

Usually the partition in which a vertex occurs is not relevant, so we just use x to refer to either vertex $(1, x)$ or vertex $(2, x)$.

The Knödel graph was widely studied as an interconnection network topology and has good properties in terms of broadcasting and gossiping. The Knödel graph $W_{\Delta,2^\Delta}$ is one of the three non-isomorphic infinite graph families known to be minimum broadcast and gossip graphs (graphs that have the smallest possible broadcast and gossip times and the minimum possible number of edges). The other two families are the well known hypercube [27] and the recursive circulant graph [90]. The Knödel graph $W_{\Delta-1,2^\Delta-2}$ is a minimum broadcast and gossip graph also for $n = 2^\Delta - 2 (\Delta \geq 2)$ [21, 70]. One of the advantages of the Knödel graph, as a network topology, is that it achieves the smallest diameter among known minimum broadcast and gossip graphs for $n = 2^\Delta (\Delta \geq 1)$. All the minimum broadcast graph families — k-dimensional hypercube, $C(4, 2^k)$-recursive circulant graph and $W_{k,2^k}$ Knödel graph — have the same degree k, but have diameters equal to k, $\lceil \frac{3k-1}{4} \rceil$ and $\lceil \frac{k+2}{2} \rceil$ respectively. A detailed description of some graph theoretic and communication properties of these three graph families and their comparison can be found in [30].

As shown in [7], the edges of the Knödel graph can be grouped into dimensions which are similar to hypercube dimensions. This allows to use these dimensions in a similar manner as in hypercube for broadcasting and gossiping. Unlike the hypercube, which is only defined for $n = 2^k$, the Knödel graph is defined for any even number of vertices. Properties such as small diameter, vertex transitivity as a Cayley graph [64], high vertex and edge connectivity, dimensionality, embedding properties [30] make the Knödel graph a good candidate as a network topology and good architecture for parallel computing. $W_{\lfloor \log_2 n \rfloor, n}$ guarantees the minimum time for broadcasting and gossiping. So, it is a broadcast and gossip graph [7, 31, 35]. Moreover, $W_{\lfloor \log_2 n \rfloor, n}$ is used to construct sparse broadcast graphs of a bigger size by interconnecting several smaller copies or by adding and deleting vertices [8, 22, 46, 47, 51, 53, 54, 70].

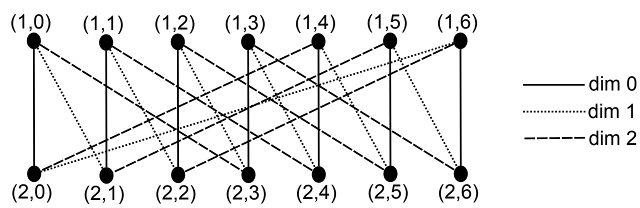

Figure 21: The $W_{3,14}$ graph and its 0, 1 and 2-dimensional edges.

Despite being highly a symmetric and widely studied graph, the diameter of the Knödel graph $D(W_{\Delta,n})$ is known only for $n = 2^{\Delta}$. In [31], it was proved that $D(W_{\Delta,2^{\Delta}}) = \lceil \frac{\Delta+2}{2} \rceil$. The nontrivial proof of this result is algebraic and the actual diametral path is not presented. The problem of finding the shortest path between any pair of vertices in the Knödel graph $W_{\Delta,2^{\Delta}}$ is studied in [59], where an 2-approximation algorithm with the logarithmic time complexity is presented.

Most properties of the Knödel graph are known only for $W_{\Delta,2^{\Delta}}$ and $W_{\Delta-1,2^{\Delta}-2}$. In this paper we present a tight upper bound on the diameter of the Knödel graph $D(W_{\Delta,n})$ for all even n and $2 \leq \Delta \leq \lfloor \log_2 n \rfloor$. We show that the presented bound may differ from the actual diameter by at most 2 for almost all Δ. Our proof is constructive and provides a near optimal diametral path in $W_{\Delta,n}$.

The distance between vertices u and v is denoted by $dist(u, v)$. Using these notations and the vertex transitivity of the Knödel graph, we can state that

$$D(W_{\Delta,n}) = max\{dist(0, x) | 0 \leq x < n/2\}.$$

We actually give a tight upper bound on $dist(0, x)$ for all $0 \le x < n/2$.

We study the broadcast time problem in the Knödel graph. The broadcast time of the Knödel graph is known only for $W_{\Delta,2^\Delta}$ and for $W_{\Delta-1,2^{\Delta-1}}$. It is shown that $b(W_{\Delta,2^\Delta}) = \Delta(\Delta \ge 1)$ [27, 73, 90] and that $b(W_{\Delta-1,2^{\Delta-1}}) = \Delta(\Delta \ge 2)$ [21, 70].

3.2 Paths in the Knödel graph

In this section we construct three different paths between two vertices in the Knödel graph $W_{\Delta,n}$. These paths have certain properties and are used in the next section to prove the upper bound on the diameter of $W_{\Delta,n}$.

Before presenting our formal statements, let us get better understanding of the Knödel graph and the set of vertices which can be reached from vertex 0 using only 0 and $(\Delta-1)$-dimensional edges. Note that we can "move" in two different directions from vertex $0 = (1, 0)$ or $0 = (2, 0)$ of $W_{\Delta,n}$. Figure 22 illustrates the discussed paths. We can choose the path

$$(1, 0) \to (2, 2^{\Delta-1} - 1) \to (1, 2^{\Delta-1} - 1) \to (2, 2(2^{\Delta-1} - 1)) \to ...$$

or we can move in the opposite direction following the path

$$(1, 0) \to (2, 0) \to (1, n/2 - (2^{\Delta-1} - 1)) \to (2, n/2 - (2^{\Delta-1} - 1)) \to ....$$

Every second edge in these paths is 0-dimensional. The $(\Delta-1)$-dimensional edges are used to move "forward" by $2^{\Delta-1} - 1$ vertices, while the 0-dimensional edges are only to change the partition. These two paths will eventually intersect or overlap somewhere near vertex $\lceil n/4 \rceil$. Excluding vertex 0, we have only $n/2 - 1$ vertices in each partition. The $(\Delta-1)$-dimensional edges will split $W_{\Delta,n}$ into $\left\lceil \frac{n/2-1}{2^{\Delta-1}-1} \right\rceil$ segments, each having length $2^{\Delta-1} - 1$, except the one containing vertex $\lceil n/4 \rceil$. We can perform only $\left\lfloor \frac{1}{2} \left\lceil \frac{n/2-1}{2^{\Delta-1}-1} \right\rceil \right\rfloor = \left\lfloor \frac{1}{2} \left\lceil \frac{n-2}{2^\Delta-2} \right\rceil \right\rfloor$ $(\Delta-1)$-dimensional passes in each of these two paths before they intersect. Therefore, we will never use more than $\left\lfloor \frac{1}{2} \left\lceil \frac{n-2}{2^\Delta-2} \right\rceil \right\rfloor$ $(\Delta-1)$-dimensional passes to reach a vertex in $W_{\Delta,n}$.

Our first lemma constructs a path between vertex 0 and some vertex y which is relatively close to our destination vertex x. Vertex y will have a special form making such construction straightforward. Recall that x refers to $(1, x)$ or $(2, x)$, and y refers

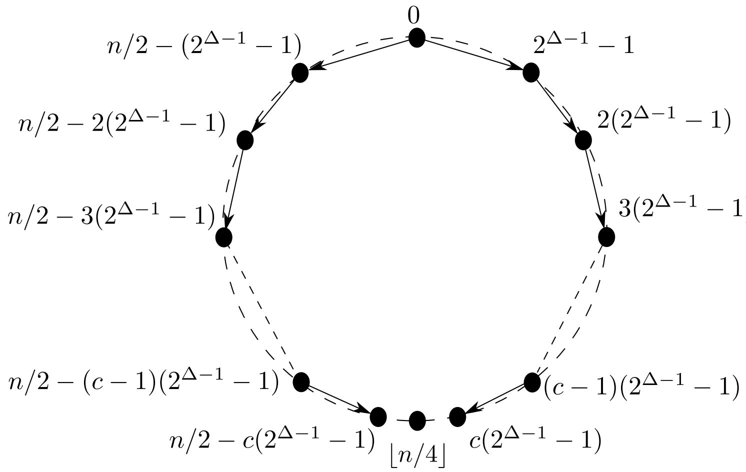

Figure 22: Schematic illustration of the paths. Note that $c = \left\lfloor \frac{1}{2} \left\lceil \frac{n-2}{2^{\Delta}-2} \right\rceil \right\rfloor$.

to $(1, y)$ or $(2, y)$.

Lemma 8. *For any vertex x of $W_{\Delta,n}$, by using at most $2 \left\lfloor \frac{x}{2^{\Delta-1}-1} \right\rfloor + 1$ edges when $x \leq \lfloor n/4 \rfloor$ or by using at most $2 \left\lfloor \frac{n/2-x}{2^{\Delta-1}-1} \right\rfloor + 1$ edges when $x > \lfloor n/4 \rfloor$ we can construct a path from vertex 0 to reach some vertex y such that $|x-y| \leq 2^{\Delta-1}-1$.*

Proof. Our goal is to reach some vertex y of form $y = c(2^{\Delta-1}-1)$ or $y = n/2 - c(2^{\Delta-1}-1)$ such that $|x-y| \leq 2^{\Delta-1}-1$. We use only 0 and $(\Delta-1)$-dimensional edges and one of two paths described above and illustrated in Figure 22. We consider two cases. In the first case we cover the values of x that can be reached by moving in "clockwise" direction from vertex 0. For the remaining values of x, we use the path from Figure 22 moving to the opposite direction.

Case 1: $x \leq \lfloor n/4 \rfloor$. By alternating between 0 and $(\Delta-1)$-dimensional edges, we can reach a vertex y of form $y = c'(2^{\Delta-1}-1)$ and closest to x from vertex $0 = (2, 0)$. We will need at most $2c'+1$ edges for that. The path to reach $y = (1, y)$ will be

$$(2,0) \to (1,0) \to (2, 2^{\Delta-1}-1) \to (1, 2^{\Delta-1}-1) \to (2, 2(2^{\Delta-1}-1)) \to ...$$

$$\to (2, c'(2^{\Delta-1}-1)) \to (1, c'(2^{\Delta-1}-1)) = y.$$

It is clear that $c' = \left\lfloor \frac{x}{2^{\Delta-1}-1} \right\rfloor$, hence the bound on the length of constructed path follows. From the form of y follows that $|x-y| \leq 2^{\Delta-1}-1$. Figure 23 shows the described path from $(2,0)$ to $y = 6 = (1,6)$.

Case 2: $x > \lfloor n/4 \rfloor$. This case is similar to case 1 except in order to construct shorter path to y of form $y = n/2 - c'(2^{\Delta-1} - 1)$, we are moving from vertex $0 = (1, 0)$ in anticlockwise direction. The path for $y = (2, y)$ will be

$$(1,0) \to (2,0) \to (1, n/2 - (2^{\Delta-1} - 1) \to (2, n/2 - (2^{\Delta-1} - 1)) \to ...$$

$$\to (1, n/2 - (c' - 1)(2^{\Delta-1} - 1)) \to (2, n/2 - (c' - 1)(2^{\Delta-1} - 1)) = y$$

and will have length at most $2c' + 1 = 2\left\lfloor \frac{n/2-x}{2^{\Delta-1}-1} \right\rfloor + 1$. Obviously we will have $|x - y| \leq 2^{\Delta-1} - 1$ as well. □

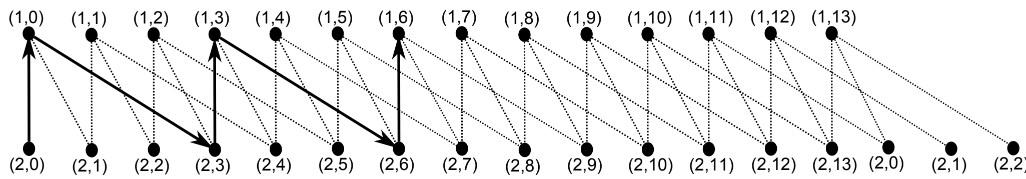

Figure 23: A path between vertices $(2, 0)$ and $(1, 6)$ in $W_{3,28}$ graph. To simplify the figure, we repeat vertices $(2, 0), (2, 1)$ and $(2, 2)$.

The following lemma constructs a path between two vertices of $W_{\Delta,n}$ that are relatively close to each other. More precisely, when the difference of their labels is upper bounded by $2^{\Delta-1} - 1$. We construct a path between two vertices x_1 and x_2 which is not necessarily a shortest path between them. To reach the given vertex with label $x_2 > x_1$ from vertex labeled x_1, we first use a large dimensional edge to "jump over" vertex x_2 and reach some vertex $y \geq x_2$, such that $y - x_2$ is the smallest. After that, we start moving from y in backward direction till we reach x_2 from right. This backward steps are performed in a greedy way. At each step, we are using the largest dimensional edge to reach some new vertex y' such that $y' - x_2$ is minimal and y' is on the right side of x_2 i.e. $y' \geq x_2$.

Lemma 9 (Existence of a special path)**.** *For any two vertices of $W_{\Delta,n}$ labeled x_1 and x_2, if $|x_2 - x_1| \leq 2^{\Delta-1} - 1$, then there exists a special path between x_1 and x_2 of length at most $2\Delta - 3$. This path contains one "direct" d-dimensional edge where $d \leq \Delta - 1$, some 0-dimensional edges and some edges having dimensions between 1 and $d - 1$ pointing in "backward" direction. The number of these backward edges is at most $\Delta - 2$.*

Proof. Without loss of generality, we assume that $x_1 = 0$ and $x_2 > x_1$. In order to construct the described path, we use an edge to get from vertex 0 to some vertex

y closest to x_2 such that $y > x_2$ and y is directly connected to 0. This will be our "direct" d-dimensional edge. After reaching vertex y, we start to move in "backward" direction towards x_2. Once started moving in backward direction, the distance from y to x_2 which is upper bounded by $2^{\Delta-2}$, will be cut at least by half with each backward edge. Therefore we need at most $\Delta - 2$ backward edges. Combined with the 0-dimensional edges between these backward edges, this will give a path of length $2(\Delta - 2)$. By adding the initial edge, we get the $2\Delta - 3$ upper bound on the length of the constructed path.

Figure 24 shows the described path between vertices $x_1 = (1, 0)$ and $x_2 = (2, 5)$. In the illustrated example $y = 7$, $d = 4$, the "direct" edge is $((1, 0), (2, 7))$ and the "backward" edges are $((2, 7), (1, 6))$ and $((2, 6), (1, 5))$.

The reason we chose this particular path between x_1 to x_2 is that the backward passes can be performed in the path constructed by Lemma 8. This will be crucial in the proof of the main theorem.

□

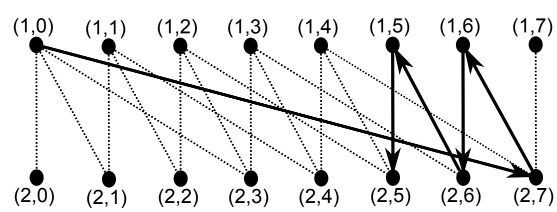

Figure 24: A path between vertices $(1, 0)$ and $(2, 5)$ in a section of a Knödel graph of degree 5.

Our last lemma deals with the problem of finding the shortest path in a particular section of the Knödel graph.

Lemma 10 (Shortest path approximation)**.** *For any two vertices of $W_{\Delta,n}$ labeled x_1 and x_2, if $|x_2 - x_1| \leq 2^d - 1$ for some $d \leq \Delta - 1$, then there exist a path between x_1 and x_2 of length at most $3\lceil d/4 \rceil + 4$.*

Proof. Without loss of generality, we assume that $x_1 = 0$ and $x_2 > x_1$. Our goal is to construct a short path from vertex 0 to vertex $x_2 = x \leq 2^d - 1$. The proof is based on a recursive construction of a path between vertices 0 and x having length at most $3\lceil d/4 \rceil + 4$. The recursion will be on d.

The base case is when $d \leq 3$. This case is illustrated in Figure 25, from which we observe that we can reach any vertex x where $0 \leq x \leq 2^d - 1 = 7$ with a path of length at most 4.

For $d > 3$, using at most three edges, we can cut the distance between 0 and x by a factor of 16. Figure 26 presents a schematic illustration of this. We divide the initial interval of length $2^d - 1$ into eight smaller intervals $A_1, A_2, ..., A_8$, each having length at most $\lceil (2^d - 1)/8 \rceil$, where $A_i = [(i-1)m, im)$, $i = 1, ..., 8$ and $m = 2^{d-3}$.

It is not difficult to see that all these intervals, except A_6, have both their end vertices reachable from 0 by using at most three edges. For A_6, using at most 3 edges we can reach its middle vertex $11m/2 - 1$ and the end vertex $6m$. The paths, which use at most 3 edges, are illustrated in Figure 26. This means that when $x \in A_i$ for all $1 \leq i \leq 8$, using at most three edges, we will be within distance $m/2$ from x. After relabeling the vertices, we will get the same problem of finding a path between vertices 0 and x, but the new x will be at least 16 times smaller.

It will take at most $\lceil \log_{16}(2^d - 1) \rceil$ recursive steps to reach the base case, and we will use at most three edges in each step. By combining this with at most 4 edges used for the base case, we will get that

$$dist(0, x) \leq 3 \lceil \log_{16}(2^d - 1) \rceil + 4 \leq 3 \lceil d/4 \rceil + 4.$$

□

We note that each recursive step in Lemma 10 involves only constant number of operations. Therefore the described path can be constructed by an algorithm of complexity $O(\log n)$.

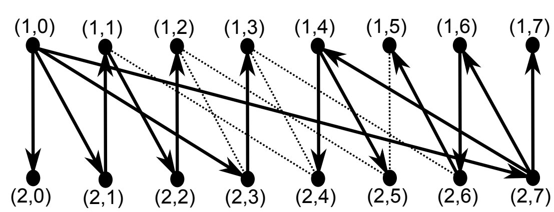

Figure 25: Paths from vertex 0 to all other vertices $x \leq 7$ in a section of a Knödel graph.

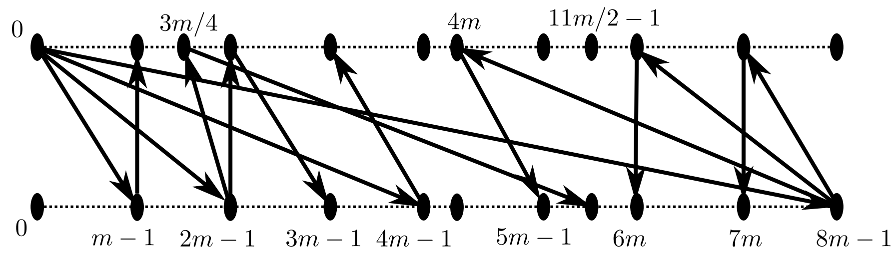

Figure 26: Illustration of the recursive step. $m = 2^{d-3}$

Lemma 10 can be used to construct a short path between any two vertices of $W_{\Delta,n}$ for the case when $\Delta = \lfloor \log_2 n \rfloor$. The length of the constructed path will be at most $3\lceil(\Delta-1)/4\rceil + 4$. It follows that when $\Delta = \lfloor \log_2 n \rfloor$, then

$$D(W_{\lfloor \log_2 n \rfloor,n}) \leq 3\lceil(\lfloor \log_2 n \rfloor - 1)/4\rceil + 4.$$

3.3 Upper bound on distance

In this section, using the lemmas from Section 2, we construct a path between vertices 0 and x for any vertex x in $W_{\Delta,n}$. The maximum length of such a path will be an upper bound on the diameter of $W_{\Delta,n}$. Without loss of generality, we assume that $0 = (2,0)$ and $x \leq \lfloor n/4 \rfloor$. For the case $x > \lfloor n/4 \rfloor$, we can replace x with $n/2 - x$ and all the following statements will remain true.

Our first upper bound on $dist(0,x)$ in $W_{\Delta,n}$ will trivially follow from Lemma 8 and Lemma 10.

Theorem 11. *For any vertex x of $W_{\Delta,n}$,*

$$dist(0,x) \leq 2\left\lfloor \frac{x}{2^{\Delta-1}-1} \right\rfloor + 3\lceil(\Delta-1)/4\rceil + 5.$$

Proof. According to Lemma 8, for any vertex x in $W_{\Delta,n}$, we need at most $2\left\lfloor \frac{x}{2^{\Delta-1}-1} \right\rfloor + 1$ edges to reach from vertex 0 to a vertex y of form $y = c(2^{\Delta-1}-1)$ such that $|x-y| \leq 2^{\Delta-1}-1$. Now we can apply Lemma 10 and claim that $dist(x,y) \leq 3\lceil d/4\rceil + 4$ where $d \leq \Delta - 1$. Thus, we have that

$$dist(0,x) \leq dist(0,y) + dist(y,x) \leq 2\left\lfloor \frac{x}{2^{\Delta-1}-1} \right\rfloor + 3\lceil(\Delta-1)/4\rceil + 5.$$

$\square$

Theorem 11 combines the paths described in Lemmas 8 and 10 in the most trivial way. With the slight modification of the path described in Lemma 8 and combining it with paths from Lemmas 9 and 10 we can significantly improve the presented upper bound on on the distance.

Theorem 12 (Main). *For a vertex x of $W_{\Delta,n}$, if*

$$\left\lfloor \frac{x}{2^{\Delta-1}-1} \right\rfloor \geq \Delta - 2$$

then

$$dist(0,x) \leq 2\left\lfloor \frac{x}{2^{\Delta-1}-1} \right\rfloor + 3,$$

otherwise

$$dist(0,x) \leq 2\left\lfloor \frac{x}{2^{\Delta-1}-1} \right\rfloor + 3\left\lceil (\Delta - 2 - \left\lfloor \frac{x}{2^{\Delta-1}-1} \right\rfloor)/4 \right\rceil + 7 \leq$$

$$\frac{3}{4}\Delta + \frac{5}{4}\left\lfloor \frac{x}{2^{\Delta-1}-1} \right\rfloor + \frac{17}{2}.$$

Proof. Case 1: $\left\lfloor \frac{x}{2^{\Delta-1}-1} \right\rfloor \geq \Delta - 2$. From Lemma 8 we recall that $\left\lfloor \frac{x}{2^{\Delta-1}-1} \right\rfloor$ is the maximum number of $(\Delta-1)$-dimensional edges necessary to reach a vertex of form $y = c(2^{\Delta-1}-1)$ or $y = n/2 - c(2^{\Delta-1}-1)$ closest to our destination vertex x. Recall that $\Delta - 2$ is the maximum number of "backward" edges used in the path from Lemma 9. We observe that when $\left\lfloor \frac{x}{2^{\Delta-1}-1} \right\rfloor \geq \Delta - 2$ then all the "backward" passes can be performed by modifying the path described in Lemma 8 used to reach vertex y. We just need to replace some of the 0-dimensional passes from Lemma 8 used only for switching the graph partition with the corresponding "backward" passes from Lemma 9. As a result of this modification, instead of reaching y, with $2\left\lfloor \frac{x}{2^{\Delta-1}-1} \right\rfloor + 1$ edges we will reach some vertex y' such that $|x - y'| = 2^{\Delta} - 1$. Using one $(\Delta-1)$-dimensional and one 0-dimensional edge we can perform the final pass and reach x with a path of length at most $2\left\lfloor \frac{x}{2^{\Delta-1}-1} \right\rfloor + 3$.

Case 2: $\left\lfloor \frac{x}{2^{\Delta-1}-1} \right\rfloor < \Delta - 2$. In this case we will be able to perform only some of the "backward" passes from Lemma 9 by modifying the path from Lemma 8. More precisely, out of $\Delta - 2$ "backward" passes we will be able to perform only $\left\lfloor \frac{x}{2^{\Delta-1}-1} \right\rfloor$

in the modified path. We note that each “backward” pass in Lemma 9 cuts the distance to x by half. This means that performing $\left\lfloor \frac{x}{2^{\Delta-1}-1} \right\rfloor$ “backward” passes in the path constructed by Lemma 8 of length $2\left\lfloor \frac{x}{2^{\Delta-1}-1} \right\rfloor + 3$ we will be within distance $2^{\Delta-2-\lfloor x/(2^{\Delta-1}-1)\rfloor}$ from x compared to $2^{\Delta-2}$ without performing these “backward” passes. Now we can use Lemma 10 with $d = \Delta - 2 - \left\lfloor \frac{x}{2^{\Delta-1}-1} \right\rfloor$ and claim that by using $3\left\lceil (\Delta - 2 - \left\lfloor \frac{x}{2^{\Delta-1}-1} \right\rfloor)/4 \right\rceil + 4$ additional edges we will be able to reach x. Thus

$$dist(0,x) \leq 2\left\lfloor \frac{x}{2^{\Delta-1}-1} \right\rfloor + 3\left\lceil (\Delta - 2 - \left\lfloor \frac{x}{2^{\Delta-1}-1} \right\rfloor)/4 \right\rceil + 7 \leq$$

$$\frac{3}{4}\Delta + \frac{5}{4}\left\lfloor \frac{x}{2^{\Delta-1}-1} \right\rfloor + \frac{17}{2}.$$

□

3.4 Tightness of the bound on distance

In this section we analyze the tightness of the upper bound on the distance from Theorem 12. To do that we will first present a lower bound on $dist(0,x)$ in the Knödel graph $W_{\Delta,n}$. Without loss of generality, we again assume that $x \leq \lfloor n/4 \rfloor$.

Theorem 13 (Lower bound)**.** *For a vertex x of $W_{\Delta,n}$,*

$$dist(0,x) \geq 2\left\lfloor \frac{x}{2^{\Delta-1}-1} \right\rfloor + 1.$$

Proof. First, note that in order to reach vertex $x = (1, c(2^{\Delta-1}-1))$ where $c = \left\lfloor \frac{x}{2^{\Delta-1}-1} \right\rfloor$ from vertex $(2,0)$, we cannot construct a path shorter than the one described in Lemma 8 and illustrated in Figure 22. This path contains exactly $c+1$ 0-dimensional edges used for changing the graph partition and c $(\Delta-1)$-dimensional edges used for moving towards x in the fastest possible way. Thus, the lower bound $dist(0,x) \geq 2\left\lfloor \frac{x}{2^{\Delta-1}-1} \right\rfloor + 1$ follows. □

The following theorem shows that the presented upper bound on $dist(0,x)$ from Theorem 12 is tight, in particular, almost always it is within additive factor 2 from the actual distance.

Theorem 14. *For any $0 < \epsilon < 1$ there exists some $N(\epsilon)$ such that for all $n \geq N(\epsilon)$, $\Delta < (1-\epsilon)\lfloor \log_2 n \rfloor$ and $x > \epsilon n$ we have*

$$2\left\lfloor \frac{x}{2^{\Delta-1}-1} \right\rfloor + 1 \leq dist(0,x) \leq 2\left\lfloor \frac{x}{2^{\Delta-1}-1} \right\rfloor + 3$$

in the $W_{\Delta,n}$ graph.

Proof. From Theorem 13 it follows that the upper bound from Theorem 12 for the case when $\left\lfloor \frac{x}{2^{\Delta-1}-1} \right\rfloor \geq \Delta - 2$ may differ from actual distance by at most 2. In the following we show that under conditions of the theorem this inequality is satisfied.

After simplifying $\left\lfloor \frac{x}{2^{\Delta-1}-1} \right\rfloor \geq \Delta - 2$ inequality we get the following necessary condition for it to be true $x \geq \Delta 2^{\Delta-1}$. Using $x > \epsilon n$ and $\Delta < (1-\epsilon)\lfloor \log_2 n \rfloor$ inequalities we will get $\epsilon n \geq (1-\epsilon)\lfloor \log_2 n \rfloor 2^{(1-\epsilon)\lfloor \log_2 n \rfloor - 1}$. After further simplifications we get that if $n^\epsilon \geq \frac{1-\epsilon}{\epsilon}\lfloor \log_2 n \rfloor$ when the $\left\lfloor \frac{x}{2^{\Delta-1}-1} \right\rfloor \geq \Delta - 2$ inequality is true. We observe that for any fixed ϵ, the left side of $n^\epsilon \geq \frac{1-\epsilon}{\epsilon}\lfloor \log_2 n \rfloor$ inequality grows polynomially with n, while the right side grows only logarithmically. Hence for any ϵ there exists some $N(\epsilon)$ such that for all $n > N(\epsilon)$ this inequality will be satisfied. $\square$

Note that Theorem 12, in almost all cases, actually gives an approximation algorithm to find the distance of $W_{\Delta,n}$ with an additive factor 2.

3.5 Lower and upper bounds on the diameter of the Knödel graph

In this section we use Theorems 12 and 13 to present a tight bound on the diameter of the Knödel graph $D(W_{\Delta,n})$. We note that Theorem 12 gives two different expressions for the upper bound of a vertex distance. By picking the maximal possible value for both cases we obtain the following upper bound on the diameter of the Knödel graph.

Theorem 15 (Diameter)**.**

$$D(W_{\Delta,n}) \leq \max\{2\left\lfloor \frac{\lfloor n/4 \rfloor}{2^{\Delta-1}-1} \right\rfloor + 3, \frac{3}{4}\Delta + \frac{5}{4}\left\lfloor \frac{\lfloor n/4 \rfloor}{2^{\Delta-1}-1} \right\rfloor + \frac{17}{2}\}.$$

Proof. From Theorem 12 we have that for a vertex x of $W_{\Delta,n}$,

$$dist(0,x) \leq 2\left\lfloor \frac{x}{2^{\Delta-1}-1} \right\rfloor + 3$$

or

$$dist(0,x) \leq \frac{3}{4}\Delta + \frac{5}{4}\left\lfloor \frac{x}{2^{\Delta-1}-1} \right\rfloor + \frac{17}{2}.$$

Both expressions are monotonic on x, hence the maximal possible value for them is achieved when $x = \lfloor n/4 \rfloor$. By substituting this value of x and taking the maximum of resulting expressions we get

$$D(W_{\Delta,n}) \leq \max\{2\left\lfloor \frac{\lfloor n/4 \rfloor}{2^{\Delta-1}-1} \right\rfloor + 3, \frac{3}{4}\Delta + \frac{5}{4}\left\lfloor \frac{\lfloor n/4 \rfloor}{2^{\Delta-1}-1} \right\rfloor + \frac{17}{2}\}.$$

□

The following theorem shows that the presented upper bound is tight, in particular it is within additive factor 2, for almost all possible values of Δ.

Theorem 16 (Tightness)**.** *For any $0 < \epsilon < 1$ there exists some $N(\epsilon)$ such that for all $n \geq N(\epsilon)$ and $\Delta < (1-\epsilon)\lfloor \log_2 n \rfloor$ we have*

$$2\left\lfloor \frac{\lfloor n/4 \rfloor}{2^{\Delta-1}-1} \right\rfloor + 1 \leq D(W_{\Delta,n}) \leq 2\left\lfloor \frac{\lfloor n/4 \rfloor}{2^{\Delta-1}-1} \right\rfloor + 3.$$

Proof. From Theorem 13 it follows that the upper bound on the diameter from Theorems 12 and 15 for the case when $\left\lfloor \frac{n/4}{2^{\Delta-1}-1} \right\rfloor \geq \Delta - 2$ may differ from actual diameter by at most 2. Now, we find a sufficient condition for $\left\lfloor \frac{n/4}{2^{\Delta-1}-1} \right\rfloor \geq \Delta - 2$ to be true. By observing that $\left\lfloor \frac{\lfloor n/4 \rfloor}{2^{\Delta-1}-1} \right\rfloor \geq \frac{n/2}{2^{\Delta}} - 1$ and $\Delta - 2 \leq \Delta - 1$ we get that if $\frac{n/2}{2^{\Delta}} - 1 \geq \Delta - 1$ then the condition is satisfied. After further simplification, we get the $2\Delta 2^{\Delta} \leq n$ sufficient condition for $\left\lfloor \frac{n/4}{2^{\Delta-1}-1} \right\rfloor \geq \Delta - 2$ to be true.

It follows that for given n and Δ, where $2 \leq \Delta \leq \lfloor \log_2 n \rfloor$ such that $\Delta 2^{\Delta+1} \leq n$, we have $2\left\lfloor \frac{\lfloor n/4 \rfloor}{2^{\Delta-1}-1} \right\rfloor + 1 \leq D(W_{\Delta,n}) \leq 2\left\lfloor \frac{\lfloor n/4 \rfloor}{2^{\Delta-1}-1} \right\rfloor + 3$. Finally, we observe that for any $0 < \epsilon < 1$ and $\Delta < (1-\epsilon)\lfloor \log_2 n \rfloor$ the $\Delta 2^{\Delta+1} \leq n$ inequality is always true for sufficiently large n. □

3.6 Broadcasting in the Knödel graph

In this section we present a tight upper and lower bounds on $b(W_{\Delta,n})$ for all even n and $2 \leq \Delta \leq \lfloor \log_2 n \rfloor$. For the upper bound, we will present a broadcast algorithm in the Knödel graph. The lower will follow from the known lower bound on the diameter of $W_{\Delta,n}$ from [44].

Let $W'_{\Delta,2^\Delta}$ be a graph obtained from $W_{\Delta,2^\Delta}$ by removing all the modular edges. See Figure 27 for an illustration of $W'_{4,16}$. Note that $W'_{\Delta,2^\Delta}$ contains only half of edges of the original Knödel graph. The following lemma gives the broadcast time of vertex $(1,0)$ in $W'_{\Delta,2^\Delta}$.

Lemma 17.

$$b((1,0), W'_{\Delta,2^\Delta}) = \Delta.$$

Proof. It is clear that broadcasting from any originator must take at least Δ time units, since $W'_{\Delta,2^\Delta}$ has 2^Δ vertices. Therefore, $b((1,0), W'_{\Delta,2^\Delta}) \geq \Delta$. In the following, we present a recursive algorithm for broadcasting in $W'_{\Delta,2^\Delta}$ from originator $(1,0)$ in Δ time units. This will prove that $b((1,0), W'_{\Delta,2^\Delta}) \leq \Delta$. The recursion will be on Δ.

The base case is when $\Delta = 1$. In this case we have two vertices connected with an edge therefore, $b(W'_{1,2}) = 1$.

For $\Delta > 1$, we note that $W'_{\Delta,2^\Delta}$ can be partitioned into two $W'_{\Delta-1,2^{\Delta-1}}$ graphs as illustrated in Figure 28. The originator $(1,0)$ first will inform its $(\Delta-1)-$dimensional neighbour $(2, 2^{\Delta-1}-1)$ in $W'_{\Delta,2^\Delta}$. After this, both partitions of $W'_{\Delta,2^\Delta}$ will have an informed vertex. Each of these two informed vertices will become the new broadcast originator in its $W'_{\Delta-1,2^{\Delta-1}}$ graph. Since at each recursive step we use only one time unit and cut the graph into two equal partitions, it follows that $b(W'_{\Delta,2^\Delta}) = \Delta$. □

Figure 27 illustrates the broadcast scheme of Lemma 17 in $W'_{4,16}$. The bold edges are used for sending the message and are labeled with the time at which they were used.

In the following, we will interpret $W_{\Delta,n}$ as a "chain" of $W'_{\Delta,2^\Delta}$ graphs. The idea of the presented broadcast algorithm is to inform one or two special vertices in each of these $W'_{\Delta,2^\Delta}$ graphs as soon as it is possible. After getting informed, all these special vertices will start to broadcast in their $W'_{\Delta,2^\Delta}$ graphs in parallel as in Lemma 17.

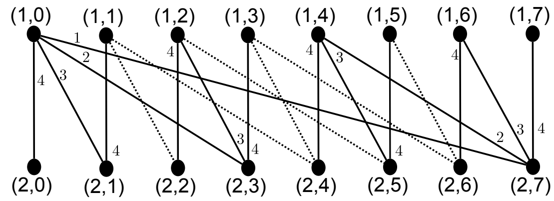

Figure 27: The $W'_{4,16}$ graph and the broadcast scheme.

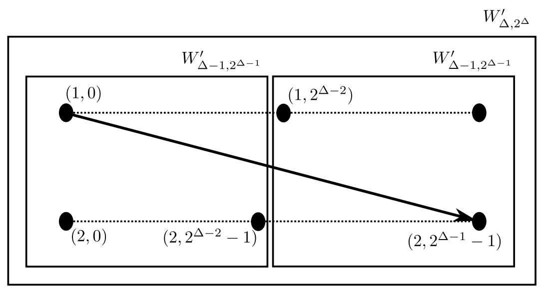

Figure 28: Recursive partitioning and broadcasting in $W'_{\Delta,2^\Delta}$.

Theorem 18 (Broadcast time)**.**

$$2\left\lfloor \frac{1}{2}\left\lceil \frac{n-2}{2^\Delta-2}\right\rceil\right\rfloor + 1 \leq b(W_{\Delta,n}) \leq \left\lceil \frac{n-2}{2^\Delta-2}\right\rceil + \Delta - 1.$$

Proof. The lower bound follows from the lower bound on $D(W_{\Delta,n})$ from Lemma 13), since obviously we will need at least $D(W_{\Delta,n})$ time units to inform a vertex at distance $D(W_{\Delta,n})$ from the broadcast originator.

To prove the upper bound, we present an algorithm for broadcasting in $W_{\Delta,n}$. The algorithm uses at most $\lceil \frac{n-2}{2^\Delta-2}\rceil + \Delta - 1$ time units.

By considering only 0 and $(\Delta-1)$-dimensional edges, the Knödel graph can be schematically illustrated as in Figure 22. Recall that the $\lceil \frac{n-2}{2^\Delta-2}\rceil$ expression represents the number of partitions in Figure 22. We note that each partition is a $W'_{\Delta,2^\Delta}$ graph. More precisely, we have $\lceil \frac{n-2}{2^\Delta-2}\rceil - 1$ partitions of the form $W'_{\Delta,2^\Delta}$ and one partition of the form $W'_{\Delta,n-(\lceil \frac{n-2}{2^\Delta-2}\rceil-1)(2^{\Delta-1}-1)-1}$.

The broadcast algorithm for $W_{\Delta,n}$ consists of three stages. In the first stage, we inform all the vertices with labels $(1,0)$ and $(2,2^{\Delta-1}-1)$ in all W' graphs except one or two farthest W' graphs from the originator. In the second stage, we use Lemma 17 to broadcast in parallel in all W' graphs. In the third stage, all the vertices of the remaining one or two W' graphs will receive the message in just 1 or 2 time units from neighbouring W' graphs and the broadcast will be complete in $W_{\Delta,n}$.

We note that the vertices of $W_{\Delta,n}$ with original labels $y = (1, c(2^{\Delta-1} - 1))$ and $y = (2, n/2 - c(2^{\Delta-1} - 1))$ where $0 \leq c \leq \left\lfloor \frac{1}{2} \left\lceil \frac{n-2}{2^{\Delta}-2} \right\rceil \right\rfloor$ after relabeling become the vertices with label $(1, 0)$ in W' partitions. Similarly, vertices $y = (2, c(2^{\Delta-1} - 1))$ and $y = (1, n/2 - c(2^{\Delta-1} - 1))$ where $0 \leq c \leq \left\lfloor \frac{1}{2} \left\lceil \frac{n-2}{2^{\Delta}-2} \right\rceil \right\rfloor$ become the vertices $(2, 2^{\Delta-1} - 1)$ in W' graphs. Therefore, we can use the paths from Figure 22 in the first stage of the broadcasting. All the vertices which need to be informed in the first stage form a "cycle" of length $\left\lceil \frac{n-2}{2^{\Delta}-2} \right\rceil$ in $W_{\Delta,n}$. Each "edge" of this cycle consists of one 0 and one $(\Delta - 1)$-dimensional edge and it takes 2 time units to send a message via such edge. It follows that we need $2 \left\lfloor \frac{1}{2} \left\lceil \frac{n-2}{2^{\Delta}-2} \right\rceil \right\rfloor$ to complete the first stage of broadcasting, i.e inform all the vertices of this "cycle" except one or two farthest ones from originator $(1, 0)$. In order to have a good upper bound on $b(W_{\Delta,n})$, we consider the parity of $\left\lceil \frac{n-2}{2^{\Delta}-2} \right\rceil$.

If $\left\lceil \frac{n-2}{2^{\Delta}-2} \right\rceil$ is odd then it will take $2 \left\lfloor \frac{1}{2} \left\lceil \frac{n-2}{2^{\Delta}-2} \right\rceil \right\rfloor - 1 = \left\lceil \frac{n-2}{2^{\Delta}-2} \right\rceil - 2$ rounds to complete the first stage. After this, all the W' partitions of the Knödel graph, except two farthest ones from the originator $(1, 0)$, will have their vertices with label $(1, 0)$ and $(2, 2^{\Delta-1} - 1)$ informed. We note that by the end of first stage, the first step of recursive broadcast algorithm from Lemma 22 will be complete. This means that we only need $\Delta - 1$ additional rounds to inform all the vertices in W' graphs. Finally, in the third stage, in just 2 time units the final two uninformed W' graphs will receive the broadcast message from the neighbouring and fully informed W'. At first, the $(\Delta - 1)$-dimensional edges will be used to inform all vertices in one of the partitions in the reaming 2 W' graphs. After this, the 0-dimensional edges will be used to inform all the vertices of the second partition. It follows that

$$b(W_{\Delta,n}) \leq (\left\lceil \frac{n-2}{2^{\Delta}-2} \right\rceil - 2) + (\Delta - 1) + 2 = \left\lceil \frac{n-2}{2^{\Delta}-2} \right\rceil + \Delta - 1.$$

If $\left\lceil \frac{n-2}{2^{\Delta}-2} \right\rceil$ is even then it will take $\left\lceil \frac{n-2}{2^{\Delta}-2} \right\rceil - 1$ rounds to complete the first stage. We note that in this case all W' graphs except one, will have two vertices with labels $(1, 0)$ and $(2, 2^{\Delta-1} - 1)$ informed. As in the previous case, we will need only $\Delta - 1$ time units tome complete the broadcasting in W' graphs according to Lemma 22. In the third stage, in just one time unit, using $(\Delta - 1)$-dimensional edges we will inform all the vertices of the remaining W' graph from neighbouring W' graphs. Hence in this case we also have

$$b(W_{\Delta,n}) \leq (\left\lceil \frac{n-2}{2^{\Delta}-2} \right\rceil - 1) + (\Delta - 1) + 1 = \left\lceil \frac{n-2}{2^{\Delta}-2} \right\rceil + \Delta - 1.$$

□

Figure 29 illustrates the broadcast algorithm of Theorem 18 in $W_{3,32}$ graph. For this case the number of partitions $\lceil \frac{n-2}{2^{\Delta}-2} \rceil = \lceil \frac{30}{6} \rceil = 5$ is odd and we deal with the first case of Theorem 18. The 0 and 2-dimensional edges divide the $W_{3,32}$ graph into 5 parts $S_1, S_2, ..., S_5$. Each part is a $W'_{3,8}$ graph. The goal of the first stage of the broadcast algorithm is to inform two special vertices in S_1, S_2 and S_5 partitions. These are the vertices $(1,0)$ and $(2,3)$ in S_1, $(1,3)$ and $(2,6)$ in S_2, $(2,0)$ and $(1,13)$ in S_5. The bold edges are used to accomplish this in 3 time units. After relabeling, these special vertices are going to have labels $(1,0)$ and $(2,3)$ in $W'_{3,8}$ partitions. During the second stage of the broadcasting, all these vertices will broadcast in parallel in S_1, S_2 and S_5 partitions as shown in Figure 30. From Lemma 17 follows that we need only 2 time units to broadcast from originators $(1,0)$ and $(2,3)$ in $W'_{3,8}$ i.e. $b(\{(1,0),(2,3)\}, W'_{3,8}) = 2$. The broadcast scheme is illustrated in Figure 29. It follows that the second stage will be complete in 2 time units. Finally, in 2 more time units, the vertices of S_2 and S_5 will inform all the vertices of S_3 and S_4. The total broadcast time will be $b(W_{3,32}) \leq 3 + 2 + 2 = 7$.

Figure 31 illustrates the broadcast algorithm of Theorem 18 in $W_{3,26}$ graph. For this case the number of partitions $\lceil \frac{n-2}{2^{\Delta}-2} \rceil = \lceil \frac{24}{6} \rceil = 4$ is even and we deal with the second case of Theorem 18. The 0 and 2-dimensional edges divide the $W_{3,32}$ graph into 4 parts S_1, S_2, S_3, S_4. Each part is a $W'_{3,8}$ graph. For this case, the goal of the first stage of the broadcast algorithm is to inform two special vertices in S_1, S_2 and S_4 partitions. The bold edges are used to accomplish this in 3 time units. As in the case of $W_{3,32}$, from Lemma 17 follows that we need only 2 time units to inform all vertices in S_1, S_2 and S_4 (see Figure 30). The broadcast scheme is illustrated in Figure 31. Finally, in just 1 time unit, the vertices of S_2 and S_4, using 2-dimensional edges, will inform all the vertices of S_3. The total broadcast time will be $b(W_{3,26}) \leq 3+2+1 = 6$.

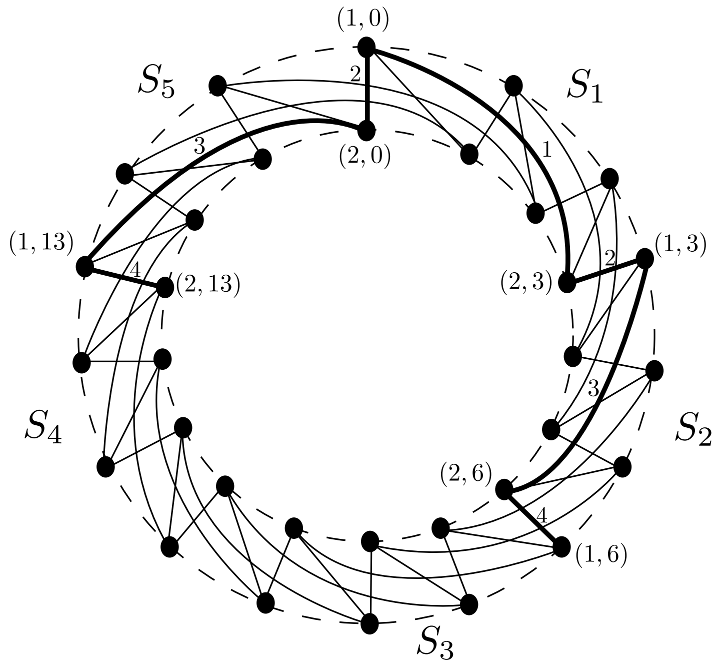

Figure 29: Broadcast scheme in $W_{3,32}$.

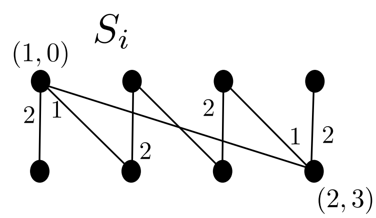

Figure 30: Two originator broadcast scheme in $W'_{3,8}$.

3.7 Summary

We addressed the routing problem in the Knödel graph $W_{\Delta,n}$ and gave a tight bound on the distance between any two vertices in $W_{\Delta,n}$. We showed that the presented bound differs from actual distance by at most 2 for almost all vertex pairs in $W_{\Delta,n}$.

We also obtained tight lower and upper bounds on the diameter of the Knödel graph $W_{\Delta,n}$ for all even n and $2 \leq \Delta \leq \lfloor \log_2 n \rfloor$. We showed that the presented bound differs from actual diameter by at most 2 for almost all Δ.

Recall that the only known results, regarding the diameter of the Knödel graph, were the exact value $D(W_{\Delta,2^\Delta}) = \lceil \frac{\Delta+2}{2} \rceil$ [31] and an 2-approximation algorithm with logarithmic time complexity for finding shortest path between any pair of vertices in $W_{\Delta,2^\Delta}$ [59]. Lemma 10 provides $D(W_{\Delta,2^\Delta}) \leq 3\lceil(\Delta-1)/4\rceil + 4$. Comparing this with

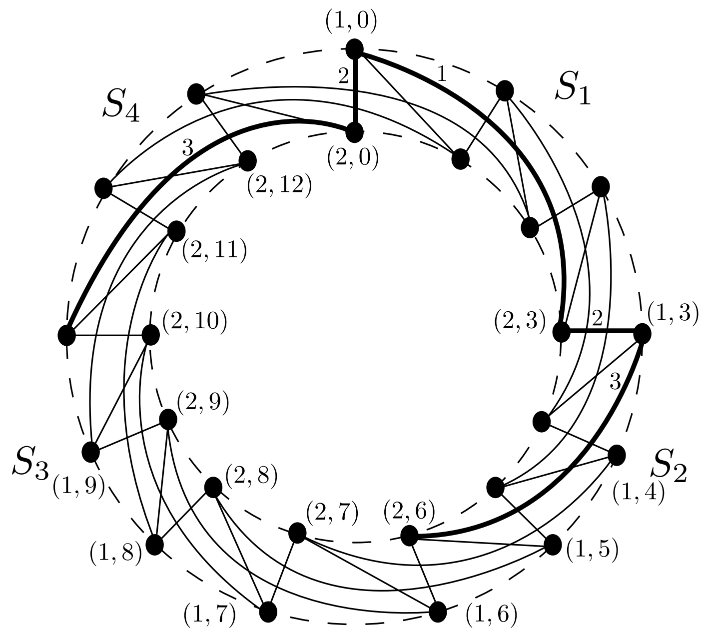

Figure 31: Broadcast scheme in $W_{3,26}$.

the exact expression above, we see that Lemma 10 provides an 3/2-approximation algorithm for the problem of finding a diametral path. This is much better than the 2-approximation algorithm presented in [59].

For the broadcast time of the Knödel graph $W_{\Delta,n}$, we showed that

$$2\left\lfloor \frac{1}{2}\left\lceil \frac{n-2}{2^{\Delta}-2}\right\rceil \right\rfloor + 1 \leq b(W_{\Delta,n}) \leq \left\lceil \frac{n-2}{2^{\Delta}-2}\right\rceil + \Delta - 1.$$

We believe that the presented lower bound, based only on $D(W_{\Delta,n})$, can be improved. Moreover, we state as a conjecture that the presented upper bound gives the exact expression for $b(W_{\Delta,n})$.

Conjecture 19.

$$b(W_{\Delta,n}) = \left\lceil \frac{n-2}{2^{\Delta}-2}\right\rceil + \Delta - 1.$$

We also note that all the proofs are constructive and allow to construct a short path between any pair of vertices and to perform quick broadcasting from any vertex in $W_{\Delta,n}$.

Chapter 4

New Lower Bounds on Broadcast Function

This chapter studies the broadcast function $B(n)$. We consider the possible vertex degrees and possible connections between vertices of different degrees in graphs with $b(G) = \lceil \log_2 n \rceil$. Using this, we present new lower bounds on $B(n)$ when $n = 2^m - 2^k$ and $n = 2^m - 2^k + 1$ $(3 \leq k < m)$. Also, we prove that $B(24) \geq 36$ for graphs with maximum vertex degree at most 4.

4.1 Introduction

We recall that $B(n)$ is defined as the number of edges in an *mbg* on n vertices.

$B(n)$ is known only for very few particular values of n. $B(n)$ is known for all $n \leq 32$ except for $n = 23, 24$ and 25. Refer to Table 1 for all currently known values of $B(n)$.

Since mbg's seem to be extremely difficult to find, a long sequence of papers presented techniques to construct broadcast graphs and to obtain upper bounds on $B(n)$ (see e.g. [8, 9, 12, 21, 25, 27, 30, 39, 40, 47, 51, 52, 53, 70, 73, 74, 90]). Most techniques combine several known mbg's and bg's on smaller sizes to create new ones of a larger size (see e.g. [8, 22, 51, 54, 70]). For this reason, it is very important to design mbg's and determine the values of $B(n)$ for small n.

However, it is extremely difficult to prove a lower bound on $B(n)$ that matches the obtained upper bound. For small n, an mbg can be found by exhaustive case analysis, but when n becomes large, the number of possible graphs grows exponentially and this technique is no longer useful.

The lower-bound proofs are based on the lower bound on vertex degree of a broadcast graph. The known minimum broadcast graphs on $n = 2^p$ and $n = 2^p - 2$ vertices are p-regular and $(p-1)$-regular graphs, respectively. In these cases there are matched lower bounds on the vertex degree. However, for other values of n, the best known broadcast graphs are not regular, and so, the upper bounds cannot match the lower bounds based only on the vertex degree.

Let v_i be the number of vertices of degree i in broadcast graph G on n vertices. We have the following expression for $B(n)$,

$$B(n) \geq \frac{1}{2}\sum_{i=1}^{n-1} i v_i.$$

Our first observation is that in an *mbg* on $n = 2^k - x$ vertices where $1 \leq x \leq 2^{k-1}$, the minimum vertex degree must be at least $k - \lfloor \log_2 x \rfloor$. A broadcast tree rooted at some vertex v of a smaller degree will contain at most $n = 2^k - x - 1$ vertices. This number is smaller than the total number of vertices. This means that not all vertices will be able receive the broadcast message by the time k from originator v. This observation gives that

$$B(2^k - x) \geq \frac{2^k - x}{2} \cdot (k - \lfloor \log_2 x \rfloor).$$

The fact that a given graph is an *mbg* determines not only the minimum possible vertex degree in it but also the possible connections between vertices of different degrees. By making more accurate observations the above mentioned bound can be improved. This approach was used in [84] to obtain lower bounds on $B(n)$ when $n = 2^m - 3, n = 2^m - 4, n = 2^m - 5$ and $n = 2^m - 6$. The following bounds are presented:

$$B(2^m - 3) \geq \left\lceil \frac{2^m - 3}{2} \cdot (m - 2 + \frac{3m - 5}{m^2 - m - 1}) \right\rceil,$$

$$B(2^m - 4) \geq \left\lceil \frac{2^m - 4}{2} \cdot (m - 2 + \frac{4}{2m + 1}) \right\rceil,$$

$$B(2^m - 5) \geq \left\lceil \frac{2^m - 5}{2} \cdot (m - 2 + \frac{2}{2m - 1}) \right\rceil,$$

$$B(2^m - 6) \geq \left\lceil \frac{2^m - 6}{2} \cdot (m - 2 + \frac{1}{m}) \right\rceil .$$

The same approach is also used in [78] to get a lower bound on $B(n)$ when $n = 2^m - 1$.

$$B(2^m - 1) \geq \left\lceil \frac{2^m - 1}{2} \cdot (m - 1 + \frac{1}{m+1}) \right\rceil .$$

We find this method of getting lower bounds on $B(n)$ promising and we will use it to find good lower bounds on $B(n)$ when $n = 2^m - 2^k$ and $n = 2^m - 2^k + 1$ $(3 \leq k < m)$. The main difficulty in the above approach is that when x increases the number of different relations between vertices of different degree increases as well and it becomes more and more difficult to deal with them and derive an improved lower bound on $B(2^k - x)$.

One of the motivations for looking on these two particular forms of n is that the smallest values for which $B(n)$ is not known are $n = 23, n = 24$ and $n = 25$. The latter two have a form $n = 2^m - 7$ and $n = 2^m - 8$ respectively. Where are known broadcast graphs on 24 and 25 vertices having 36 and 40 edges respectively [9] but whether these graphs are *mbg*'s or not is not known. Tight lower bounds on $B(24)$, $B(25)$ may help to address this problem.

4.2 Lower bound on $B(2^k - 7)$

In this section we present a new lower bound on $B(n)$ when $n = 2^k - 7$. Later, we generalize the presented result for $n = 2^m - 2^k + 1$. In our approach, we extend the technique presented by Sacle in [99].

Theorem 20.

$$B(2^k - 7) \geq \frac{n}{2} \cdot ((k - 3) + \frac{5k - 11}{(k+1)(k-2)}).$$

Proof. Recall that in an mbg on $n = 2^k - 7$ the minimum possible vertex degree is $k - 3$. Let us look at the broadcast tree rooted at a vertex u of degree $k - 3$. We observe that u must have at least one neighbour of degree at least k, at least two neighbours of degree at least $k - 1$ and at least three neighbours of degree at least $k - 2$. We also observe that a vertex cannot have all neighbours of degree $k - 3$. In other words each vertex in the graph must have at least one vertex of degree at least $k - 2$. We can write the following inequalities:

$$\sum_{i \geq k} (i-1)v_i \geq v_{k-3},$$

$$\sum_{i \geq k-1} (i-1)v_i \geq 2v_{k-3},$$

$$\sum_{i \geq k-2} (i-1)v_i \geq 3v_{k-3}.$$

For the number of edges in the graph, denoted by m, we will have

$$2m = \sum_{i \geq k-3} iv_i = n + \sum_{i \geq k-3} (i-1)v_i.$$

This implies

$$\sum_{i \geq k-3} (i-1)v_i = 2m - n.$$

After substituting this in the above three inequalities we will get

$$2m - n - (k-4)v_{k-3} - (k-3)v_{k-2} - (k-2)v_{k-1} \geq v_{k-3},$$

$$2m - n - (k-4)v_{k-3} - (k-3)v_{k-2} \geq 2v_{k-3},$$

$$2m - n - (k-4)v_{k-3} \geq 3v_{k-3}.$$

After rearrangement of the terms we will have

$$2m - n \geq (k-3)v_{k-3} + (k-3)v_{k-2} + (k-2)v_{k-1},$$

$$2m - n \geq (k-2)v_{k-3} + (k-3)v_{k-2},$$

$$2m - n \geq (k-1)v_{k-3}.$$

After subtracting v_{k-1} and v_{k-3} from the right hand sides of the first and the second inequalities respectively, we will get

$$2m - n \geq (k-3)(v_{k-3} + v_{k-2} + v_{k-1}),$$

$$2m - n \geq (k-3)(v_{k-3} + v_{k-2}),$$

$$2m - n \geq (k-1)v_{k-3}.$$

It follows that

$$v_{k-3} + v_{k-2} + v_{k-1} \leq \frac{2m-n}{k-3},$$

$$v_{k-3} + v_{k-2} \leq \frac{2m-n}{k-3},$$

$$v_{k-3} \leq \frac{2m-n}{k-1}.$$

We have that for the number of edges we also have the following expression

$$2m \geq nk - (v_{k-1} + 2v_{k-2} + 3v_{k-3}) =$$

$$= nk - (v_{k-1} + v_{k-2} + v_{k-3}) - (v_{k-2} + v_{k-3}) - v_{k-3}.$$

After the substitution of the above bounds in this inequality we will have

$$2m \geq nk - (2m-n)(\frac{2}{k-3} + \frac{1}{k-1}).$$

From which

$$m \geq \frac{n}{2} \cdot \frac{k + (\frac{2}{k-3} + \frac{1}{k-1})}{1 + (\frac{2}{k-3} + \frac{1}{k-1})} =$$

$$= \frac{n}{2} \cdot \frac{k + (\frac{2}{k-3} + \frac{1}{k-1})}{1 + (\frac{2}{k-3} + \frac{1}{k-1})}.$$

Finally, we got the following lower bound on $B(2^k - 7)$

$$B(2^k - 7) \geq \frac{n}{2} \cdot \frac{k + (\frac{1}{k-1} + \frac{2}{k-3})}{1 + (\frac{1}{k-1} + \frac{2}{k-3})} =$$

$$\frac{n}{2} \cdot ((k-3) + \frac{5k-11}{(k+1)(k-2)}).$$

□

4.3 Lower bound on $B(2^k - 2^p + 1)$

In this section we obtain a new lower bound on $B(n)$ where $n = 2^k - 2^p + 1$ based on the degree sequence restrictions of any broadcast graph on $2^k - 2^p + 1$ vertices.

Theorem 21.

$$B(2^k - 2^p + 1) \geq \frac{2^k - 2^p + 1}{2} \cdot ((k-p) + \frac{k(2p-1) - (p^2+p-1)}{k(k-1)-(p-1)}).$$

Proof. We observe that in an *mbg* on $2^k - 2^p + 1$ vertices, each vertex of degree $k-p$ must have at least one neighbour of degree at least k, two neighbours of degree at least $k-1$, three neighbours of degree at least $k-2$, ... , p neighbours of degree at least $k-p+1$. After noticing that a vertex cannot have all its neighbours of degree $k-p$ we are getting the following inequalities

$$\sum_{i \geq k} (i-1)v_i \geq v_{k-p},$$

$$\sum_{i \geq k-1} (i-1)v_i \geq 2v_{k-p},$$

$$\sum_{i \geq k-2} (i-1)v_i \geq 3v_{k-p},$$

$$\ldots$$

$$\sum_{i \geq k-p+1} (i-1)v_i \geq pv_{k-p}.$$

For the number of edges in the graph, denoted by m we will have

$$2m = \sum_{i \geq k-p} iv_i = n + \sum_{i \geq k-p} (i-1)v_i.$$

This implies that

$$\sum_{i \geq k-p} (i-1)v_i = 2m - n.$$

After substituting this in the above p inequalities and reversing their order we will get

$$2m - n - (k-p-1)v_{k-p} \geq pv_{k-p},$$

$$2m - n - (k-p-1)v_{k-p} - (k-p)v_{k-p+1} \geq (p-1)v_{k-p},$$

$$2m - n - (k-p-1)v_{k-p} - (k-p)v_{k-p+1} - (k-p+1)v_{k-p+2} \geq (p-2)v_{k-p},$$

$$\ldots$$

$$2m - n - \sum_{j=0}^{i}(k-p-1+j)v_{k-p+j} \geq (p-i)v_{k-p},$$

...

$$2m - n - \sum_{j=0}^{p-1}(k-p-1+j)v_{k-p+j} \geq v_{k-p}.$$

After rearranging the terms we will have

$$2m - n \geq (k-1)v_{k-p},$$

$$2m - n \geq (k-2)v_{k-p} + (k-p)v_{k-p+1},$$

$$2m - n \geq (k-3)v_{k-p} + (k-p)v_{k-p+1} + (k-p+1)v_{k-p+2},$$

...

$$2m - n \geq (k-p)v_{k-p} + \sum_{j=1}^{p-1}(k-p-1+j)v_{k-p+j}.$$

By replacing all the $k-2$, $k-3$, ..., $k-p+1$ coefficients on the right side of these inequalities with $k-p$ (the smallest one) we will get

$$2m - n \geq (k-1)v_{k-p},$$

$$2m - n \geq (k-p)(v_{k-p} + v_{k-p+1}),$$

$$2m - n \geq (k-p)(v_{k-p} + v_{k-p+1} + v_{k-p+2}),$$

...

$$2m - n \geq (k-p)(v_{k-p} + v_{k-p+1} + v_{k-p+2} + ... + v_{k-1}).$$

From other side we have the following trivial inequality

$$2m \geq nk - (v_{k-1} + 2v_{k-2} + 3v_{k-3} + ... + pv_{k-p}) =$$

$$= nk - (v_{k-1} + v_{k-2} + v_{k-3} + ... + v_{k-p})$$

$$-(v_{k-2} + v_{k-3} + ... + v_{k-p}) - (v_{k-3} + ... + v_{k-p}) - ... - v_{k-p}.$$

By substituting in this inequality the upper bounds obtained in the previous set of inequalities we will get

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

It follows that

$$B(2^k - 2^p + 1) \geq \frac{n}{2} \cdot \frac{k + (\frac{1}{k-1} + \frac{p-1}{k-p})}{1 + (\frac{1}{k-1} + \frac{p-1}{k-p})} =$$

$$\frac{2^k - 2^p + 1}{2} \cdot ((k-p) + \frac{k(2p-1) - (p^2+p-1)}{k(k-1) - (p-1)}).$$

□

4.4 Lower bound on $B(2^k - 2^p)$

Using the same approach as is in the case of $B(2^k - 2^p - 1)$, we get a new lower bound on $B(2^k - 2^p)$.

Theorem 22.

$$B(2^k - 2^p) \geq \frac{2^k - 2^p}{2} \cdot ((k-p) + \frac{k(2p-2) - (p^2+p-2)}{k(k-2) - (p-2)}).$$

Proof. The proof is omitted due to its similarity to the proof for $B(2^k - 2^p - 1)$. □

4.5 About the value of $B(24)$

A broadcast graph on 24 vertices and 36 edges was constructed by Bermond et al. [9]. This gives

$$B(24) \leq 36.$$

We will prove the $B(24) \geq 36$ inequality for graphs G with $\Delta(G) = 4$, i.e. for graphs with maximum vertex degree at most 4.

Let v_i denote the number of vertices of degree i, and α_{ij} denote the number of all edges between vertices of degree i and j. By our definition $\alpha_{ij} = \alpha_{ji}$.

Theorem 23. *A broadcast graph G on 24 vertices and $\Delta(G) \leq 4$, must have at least 36 edges.*

Proof. We observe that G cannot contain a vertex of degree 1, since such a vertex will be able to inform at most $17 < 24$ vertices in 5 rounds. By counting the number of edges adjacent to vertices of degree 4, we will have

$$4v_4 = \alpha_{42} + \alpha_{43} + 2\alpha_{44}.$$

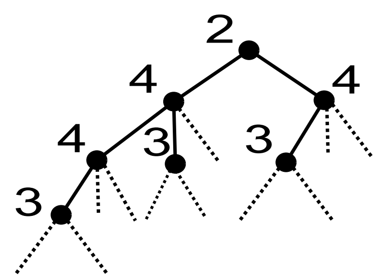

Figure 32: Subtree of a broadcast tree rooted at a vertex of degree 2.

We also observe that the broadcast tree rooted at a vertex of degree 2 must have a form shown in Figure 32. Except the root, all other vertices which may have degree 2 are omitted. The number next to each vertex indicates the minimal possible degree for that vertex. For example, a vertex with label 3 may actually have degree 4.

From the figure we observe that a vertex of degree 2 must have both its neighbours of degree 4. Therefore,

$$\alpha_{42} = 2v_2$$

From the fact that a vertex of degree 4 cannot have all its neighbours having degree 2, it follows that it has at least one adjacent edge going to vertex of degree 3 or 4. Also we note that a vertex of degree 2 must have a neighbour v (left child in Figure 32) of degree 4 having at least 2 edges going to a vertex of degree 3 or 4. Vertex v can be shared between at most 2 vertices of degree 2. It follows that there are at least $\lceil \frac{v_2}{2} \rceil$ such vertices "v", i.e. vertices of degree 4 having at least 2 edges going to a vertex of degree 3 or 4. Thus, we have that

$$\alpha_{43} + \alpha_{44} \geq v_4 + \left\lceil \frac{v_2}{2} \right\rceil.$$

From the observation that an edge between vertices of degree 4 in Figure 32 can be shared among at most 4 vertices of degree 2 we have that

$$\alpha_{44} \geq \left\lceil \frac{v_2}{4} \right\rceil.$$

Finally, by using the expressions above we will have

$$4v_4 = \alpha_{42} + \alpha_{43} + 2\alpha_{44} = \alpha_{42} + (\alpha_{43} + \alpha_{44}) + \alpha_{44} \geq$$

$$\geq 2v_2 + v_4 + \left\lceil \frac{v_2}{2} \right\rceil + \left\lceil \frac{v_2}{4} \right\rceil \geq (2 + \frac{1}{2} + \frac{1}{4})v_2 + v_4 \Rightarrow$$

$$\Rightarrow 3v_4 \geq \frac{11}{4}v_2 \Rightarrow v_4 \geq \frac{11}{12}v_2.$$

To prove that $b(24) \geq 36 = \frac{24 \cdot 3}{2}$, we must show that in any broadcast graph of on 24 vertices, the average vertex degree is at least 3. In our case, this means that in any broadcast graph G with $\Delta(G) = 4, |G| = 24$, we must show that $v_4 \geq v_2$. From $v_4 \geq \frac{11}{12}v_2$ it almost always follows that $v4 \geq v2$. The only pair of values for which it is not so is $v_2 = 12, v_4 = 11$, but this would mean that $v_3 = 24 - v_2 - v_4 = 1$, which is impossible, since in any graph the number of vertices of odd degree must be even. □

4.6 Summary

In [39] it was shown that

$$B(n) \geq \frac{n}{2} \cdot (\lfloor \log_2 n \rfloor - \log_2(1 + 2^{\lfloor \log_2 n \rfloor} - n)).$$

Let p be the index of the leftmost 0 bit in the binary representation $(\alpha_{m-1}\alpha_{m-2}...\alpha_1\alpha_0)$ of $n - 1$. In [74] the following bound was obtained

$$B(n) \geq \frac{n}{2} \cdot (m - p - 1).$$

This bound was later improved in [52] to

$$B(n) \geq \frac{n}{2} \cdot (m - p - 1 + \beta)$$

where $\beta = 0$ if $p = 0$ or if $\alpha_0 = \alpha_1 = ... = \alpha_{p-1} = 0$, otherwise $\beta = 1$.

In [103], the lower bound was further improved. It was shown that for almost all

n,

$$B(n) \geq \frac{n}{2} \cdot \frac{(D(n)+1)^2}{(D(n)+2)},$$

where $D(n) = m - p - 1 + \beta$.

As we can see, the previously known lower bounds on $B(n)$ have a rather complicated form. Therefore, it is very difficult to compare analytically our new bounds from Theorems 20 and 22 with the previous bounds. Despite this, from the numerical examples presented in Table 4, we see that our new lower bounds on $B(n)$ for $n = 2^m - 2^k$ and $n = 2^m - 2^k + 1$ ($2 \leq k < m$) are much tighter than the previous bounds.

n	k	p	New lower bound on $B(n)$	Previous lower bound on $B(n)$
5	3	2	5	5 (tight)
9	4	3	9	10 (tight)
13	4	2	18	18 (tight)
24	5	3	33	27,35
25	5	3	35	29,38
28	5	2	48	45
29	5	2	52	52
48	6	4	68	54
49	6	4	70	56
56	6	3	109	90
57	6	3	105	92
60	6	2	130	130 (tight)
61	6	2	136	136 (tight)
384	9	7	563	432
385	9	7	566	434

Table 4: Some lower bounds on $B(n)$ from Theorems 20 and 22.

Chapter 5

Miscellaneous Remarks

This chapter presents some basic results in our research.

5.1 Graphs with the worst possible broadcast time

This section describes all the graphs G with $b(G) = n - 1$.

In a connected graph G on n vertices, at each broadcast round, at least one new vertex receives the message. Therefore,

$$b(G) \leq n - 1.$$

To reach this bound, we must make sure that at each round exactly one new vertex gets informed.

We begin by describing all the trees with $b(T) = n - 1$.

Let S be the set of all edges between informed and uniformed vertices $S = \{(v_1, v_1'), ..., (v_k, v_k')\}$ at each broadcast round in a tree. For any $i \neq j$ we must have $v_i' \neq v_j'$, otherwise we would have a cycle in T consisting of path $(v_i, v_i')(v_i', v_j)$ and path connecting v_i and v_j in the set of informed vertices. Also note that vertices v_i where $1 \leq i \leq k$, cannot be distinct, otherwise, if for some $i \neq j$, $v_i \neq v_j$, we will have two disjoint edges (v_i, v_i'), (v_j, v_j') in S. Using these edges, we will be able to inform two new vertices v_i' and v_j' in just one round. It follows that S has a form $S = \{(v_1, v_1'), ..., (v_1, v_k')\}$ where all v_i''s are distinct vertices.

Theorem 24. *For a tree T, $b(T) = n - 1$ if and only if T has the form of a path* *connected to a "star" ($K_{1,n}$) as shown in Figure 33. We have $b(T) = b(v, T) = n-1$, where v is one of the end vertices of the path.*

Proof. Let T be a tree with $b(T) = n - 1$. There must exist a vertex $v \in V(T)$ such that $b(v, T) = n-1$. Recall that the set of edges S between informed and uninformed vertices must have a form $S = \{(v_1, v_1'), ..., (v_1, v_k')\}$. We start broadcasting from v and at each round we construct S. At each round, if $k = 1$ for the constructed S, we will get a new vertex on a path starting at v. If at some round we get $S = \{(v_1, v_1'), ..., (v_1, v_k')\}$ and $k \neq 1$, we claim that v_1 is the only neighbour for vertices $v_1', v_2', ..., v_k'$. We will prove this claim by contradiction. Without loss of generality, suppose that v_2' has a neighbour v_2'' and $v_2'' \neq v_1$. In this case, v_1 first will inform v_2' and both vertices v_1 and v_2' in the following round will be able to inform 2 new vertices v_1' and v_2''. □

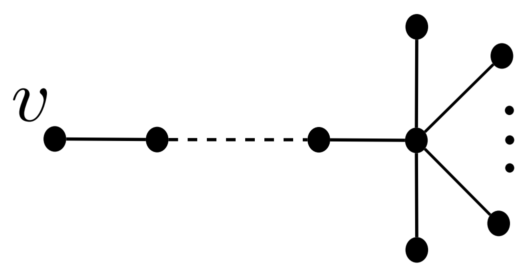

Figure 33: General form of the trees T with $b(T) = b(v) = n - 1$.

Now we describe all the cyclic graphs with $b(G) = n - 1$.

Theorem 25. *If a graph G with $b(G) = n - 1$ is not a tree then it must have a form of a path connected to a triangle as shown in Figure 34.*

Proof. If $b(G) = n - 1$, then all the spanning trees of G must have a form given by Theorem 24 and shown in Figure 33. Therefore, we can assume that G is created from a tree shown in Figure 33 by adding one or more edges. We observe that in order to have $b(G) = n - 1$ the first added edge must connect two vertices from the "star" part of the tree in Figure 33. We also observe that we cannot add other edges without getting $b(G) < n-1$. So, we are left only with the graph presented in Figure 34. □

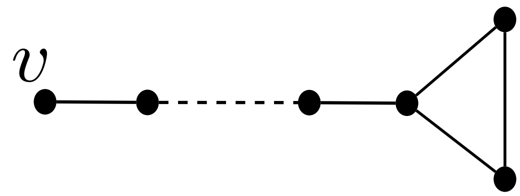

Figure 34: General form of the cyclic graphs G with $b(G) = b(v) = n - 1$.

5.2 Formulation of the broadcast time problem as an IP problem

In order to formulate the broadcast time problem as an *integer programming(IP)* problem we need more formal definition for the *broadcast scheme* in a graph.

The triple (v_i, v_j, t) denotes the fact that vertex v_i at round t informs v_j. A broadcast scheme of a graph $G = (V, E), |V| = n$ from the originator $s \in V$, denoted $BS(s, G)$, is defined as a set of triples

$$BS(s, G) = \{(v_i, v_j, t) | v_i, v_j \in V, t \in \{1, ..., n-1\}\}$$

with the following properties:

1. $(v_i, v_j, t) \in BS(s, G) \Rightarrow (v_i, v_j) \in E$.

2. $(v_i, v_j, t) \in BS(s, G) \Rightarrow v_i = s$ or for some $t' < t, \exists (v_{i'}, v_i, t') \in BS(s, G)$.

3. $(v_i', v_j, t) \in BS(s, G)$ and $(v_i'', v_j, t'') \in BS(s, G) \Rightarrow v_i' = v_i'', t = t''$.

4. $(v_i, v_j', t) \in BS(s, G)$ and $(v_i, v_j'', t) \in BS(s, G) \Rightarrow v_j' = v_j''$.

5. $|BS(s, G)| = n - 1$.

Property 1 expresses the fact that a vertex can inform only its neighbours. Property 2 expresses the fact that only an informed vertex can inform other vertices. Property 3 expresses the fact that each vertex receives the message only once, i.e we forbid unnecessary calls. Property 4 expresses the fact that at each round a vertex in graph can send the message to at most 1 vertex. Finally, the last property ensures that each vertex from $V - \{s\}$ will appear as a second component of a triple belonging to $BS(s, G)$, i.e. each vertex except the originator s will receive the broadcast message.

Now we can formulate the IP problem. Let $y_{i,j,t}$ boolean variable be 1 if $(v_i, v_j, t) \in BS(s, G))$ and 0 otherwise. The IP problem will be:

$$minimize \quad \sum_{i=1}^{n} z_i$$

$$subject \quad to$$

$$\sum_{(i',j)\in E, t'\in\{1,...,t-1\}} y_{i',j,t'} \geq y_{i,j,t} \text{ for all } i,j,t \text{ there } (i,j)\in E, t\in\{1,...,n-1\}$$

$$y_{i,j',t} + y_{i,j'',t} \leq 1 \text{ for all } j' \neq j'', (i,j'), (i,j'') \in E$$

$$\sum_{(i,j)\in E, t\in\{1,...,n-1\}} y_{i,j,t} = 1 (or \geq 1) \text{ for all } v_j \in V - \{s\}$$

$$\sum_{(i,j)\in E, t\in\{1,...,n-1\}} y_{i,j,t} = n-1$$

$$\sum_{(i,j)\in E} y_{i,j,t} \leq z_t \text{ for } t = 1,...,n-1$$

$$y_{i,j,t}, z_t \in \{0,1\} \text{ for all } i,j,t \text{ there } (i,j)\in E, t\in\{1,...,n-1\}$$

In the IP program we have $m(n-1)$ variables $y_{i,j,t}$ and $n-1$ variables z_t. In a feasible solution of the IP program, $n-1$ variables $y_{i,j,t}$ must be set to 1. We also have a $n-1$ possible choices for the values of z_t. Therefore, in total, we will have $(n-1)\cdot\binom{m(n-1)}{n-1}$ candidates for the IP's solution. It follows that the exhaustive search for the optimal solution will have a complexity $O(n\cdot\binom{mn}{n}) = O(n\cdot(\frac{emn}{n})^n) = O(n(em)^n)$.

5.3 Slightly improved general upper bound on broadcast time

In this section we prove a new upper bound on the broadcast time of any connected graph G.

Let $G = (V, E)$ be a connected graph and $u, v \in V$. $dist(u, v)$ denotes the length of the shortest path between u and v in G. Also, let $deg(G)$ denote the maximum degree of a vertex in G. The *diameter* of G is defined as

$$diam(G) = max\{dist(u, v)|u, v \in G\}.$$

The *radius* or *eccentricity* of a vertex v in G is defined as

$$rad(v, G) = max\{dist(v, x)|x \in G\}.$$

The *radius* of G is defined as

$$rad(G) = min\{rad(v, G)|v \in G\}.$$

Theorem 26. *For any connected graph G,*

$$b(G) \leq deg(G)\lfloor diam(G)/2 \rfloor + 1$$

when $diam(G)$ is odd, and

$$b(G) \leq deg(G)diam(G)/2$$

when $diam(G)$ is even.

Proof. To prove the upper bounds we describe a broadcast algorithm which finishes in $deg(G) \cdot \lfloor diam(G)/2 \rfloor + 1$ and in $deg(G) \cdot diam(G)/2$ rounds respectively.

Let $G = (V, E)$ be a connected graph with odd diameter and let $o \in V$ be the broadcast originator. The following algorithm broadcasts in G in at most $deg(G) \cdot \lfloor diam(G)/2 \rfloor + 1$ rounds.

Step 1: Find $u, v \in V$ such that $dist(u, v) = diam(G)$. In the shortest path between u and v of odd length, pick two vertices u' and v' such that $dist(u, u') = dist(v, v') = \lfloor diam(G)/2 \rfloor$ (note that u' and v' are the two middle vertices of a diametral path).

Step 2: Find in G a shortest path P which connects the originator o with one of the vertices u', v' and such that $(u', v') \in P$.

Step 3: Using path P, send the message from originator o to vertices u' and v'.

Step 4: Continue broadcasting from vertices u', v' in a "greedy" way, i.e. all informed vertices at each broadcast round pick arbitrary uninformed neighbour and send the message to it until all the vertices of G will receive the message.

The length of path P may at most be $\lfloor diam(G)/2 \rfloor + 1$, so in the third step, the algorithm will need at most $\lfloor diam(G)/2 \rfloor + 1$ rounds. All the vertices of G are within distance $\lfloor diam(G)/2 \rfloor$ from u' or v', so, in at most $(deg(G) - 1)\lfloor diam(G)/2 \rfloor$ rounds all of them will receive the message in the forth step of the algorithm. This claim follows from the observation that after $d(deg(G) - 1)$ rounds all vertices of graph G within distance d from u' or v' will receive the message. For the broadcast time of o in G we will have

$$b(o, G) \leq \lfloor diam(G)/2 \rfloor + 1 + (deg(G) - 1)\lfloor diam(G)/2 \rfloor =$$

$$deg(G)\lfloor diam(G)/2 \rfloor + 1.$$

So, we proved the theorem when $diam(G)$ is odd. However when $diam(G)$ is even, we must slightly modify the above algorithm. The following is the modified version.

Step 1: Find $u, v \in V$ such that $dist(u, v) = diam(G)$. In the shortest path between u and v of even length pick the vertex u' such that $dist(u, u') = dist(u', v) = diam(G)/2$ (u' is the middle vertex of a diameter).

Step 2: Find in G a shortest path P which connects the originator o with the vertex u'.

Step 3: Using path P, send the message from originator o to vertex u'.

Step 4: Continue broadcasting from the vertex u' in a "greedy" way, i.e. all informed vertices at each broadcast round pick arbitrary an uninformed neighbour and send the message to it until all the vertices of G will receive the message.

In the third step the algorithm will spend at most $diam(G)/2$ rounds. All vertices of G are within distance $diam(G)/2$ from u', therefore, in at most $(deg(G) - 1) \cdot diam(G)/2$ rounds all of them will receive the message. For the broadcast time of o in G we will have

$$b(o, G) \leq diam(G)/2 + (deg(G) - 1) \cdot diam(G)/2 =$$

$$deg(G) \cdot diam(G)/2.$$

This finishes the proof of the theorem. □

The previous general upper bounds on the broadcast time are presented in [68].

$$b(G) \leq (deg(G) - 1) \cdot diam(G) + 1,$$

$$b(G) \leq deg(G) \cdot rad(G).$$

Note that $\lfloor diam(G)/2 \rfloor < rad(G)$ when $diam(G)$ is odd and $diam(G)/2 \leq rad(G)$ otherwise. Thus, it follows that our new bounds from Theorem 26 are slightly tighter than the ones from [68].

5.4 Relation between minimum and maximum broadcast times in a graph

In this section we prove a relation between the broadcast time of an arbitrary chosen vertex in a graph and the broadcast time of a vertex from the *broadcast center* of the graph.

Let OPT be the minimum broadcast time of a vertex in a graph $G = (V, E)$. Recall that the *broadcast center(BC)* is the set of vertices in G with broadcast time OPT. The $dist(v, BC)$ is defined as

$$dist(v, BC) = min\{dist(v, u) | u \in BC\}.$$

Lemma 27. *For any vertex $v \in V$ we have*

$$dist(v, BC) \leq OPT - 1.$$

Proof. From the definition of the graph broadcast center we know that for all $o \in BC$ $b(o, G) = OPT$. The proof will be by contradiction.

Let us assume that there exists some vertex $v \in V$ with $dist(v, BC) > OPT - 1$. From the fact that in OPT rounds it is possible to finish broadcasting from any vertex $o \in BC$ it follows that $dist(v, BC) \leq OPT$. We need to consider only the case when $dist(v, BC) = OPT$. On a path P of length OPT connecting v with the closest vertex $o \in BC$ let $u \notin BC$ be the vertex attached to o. From the fact that $length(P) = OPT$ it follows that in order to inform the vertex v in OPT rounds, o at the first round must send the message to vertex u. But this means that vertex u also may finish broadcasting in OPT rounds by first informing o, i. e. $u \in BC$ which is a contradiction. $\square$

Theorem 28. *For any vertex $v \in V$ we have*

$$b(v, G) \leq 2 \cdot OPT - 1.$$

Proof. From Lemma 27 it follows that in at most $OPT - 1$ rounds we will be able to inform some vertex $o \in BC$ from any originator $v \in G$. Vertex u needs only OPT rounds to finish the broadcasting, therefore $b(v, G) \leq 2 \cdot OPT - 1$. $\square$

Theorem 28 provides a slight improvement over the trivial bound $b(v, G) \leq 2 \cdot OPT$.

5.5 Broadcasting from multiple optimally chosen originators

This section studies the broadcasting from multiple optimally chosen originators. In particular, we are interested in knowing how much the broadcast time may be decreased if instead of one we will be allowed to have k initially informed vertices in a graph G. Some results related to this model are presented in [14, 29].

Let OPT_k denote the broadcast time from the optimally chosen k vertices in graph G, where by "optimally" we mean a choice which minimizes the broadcast time. We prove that by having the opportunity to choose k originators instead of 1, we may decrease the broadcast time in a graph at most k times.

Theorem 29. *In any connected graph G,*

$$OPT_1 \leq k \cdot OPT_k + 1.$$

Proof. First we prove the case when $k = 2$.

Let us assume that in a connected graph $G = (V, E)$, vertices $v_1, v_2 \in V$ are the optimal originators for $2 - broadcasting$. See Figure 35. Let $P_{v_1 v_2}$ be the shortest path connecting vertices v_1, v_2. From the fact that $2-broadcasting$ can be completed in OPT_2 rounds it follows that for every vertex $u \in V$ we must have

$$min\{distance(u, v_1), distance(u, v_2)\} \leq OPT_2.$$

This is true for the vertices of path $P_{v_1v_2}$ as well, and means that $length(P) \leq 2 \cdot OPT_2$. Let us pick as a $1-broadcast$ originator the central vertex o of path $P_{v_1v_2}$ (or one of the central vertices if $P_{v_1v_2}$ has an even length). The $1-broadcasting$ algorithm will first inform vertices v_1, v_2 from o in at most $OPT_2 + 1$ rounds and by using $2-broadcasting$ algorithm from v_1, v_2 originators we will complete the broadcasting in OPT_2 additional rounds. This means that we have a $1-broadcasting$ algorithm which completes broadcasting from originator o in at most $2 \cdot OPT_2 + 1$ rounds, i.e. $b(o, G) \leq 2 \cdot OPT_2 + 1$. Also by noticing that $OPT_1 \leq b(o, G)$ we will have that

$$OPT_1 \leq 2 \cdot OPT_2 + 1.$$

This completes the proof for the case $k = 2$.

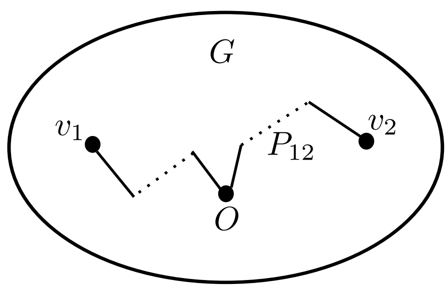

Figure 35: Illustration of the connection between OPT_1 and OPT_2.

For the case $k \geq 3$, again, at first we will inform from a specially chosen originator each of the $v_1, v_2, ..., v_k$ optimally chosen $k-broadcasting$ originators. After that, they will finish the broadcasting in the next OPT_k rounds. To complete the proof, we must show how to choose such an originator and how to inform $v_1, v_2, ..., v_k$ vertices in at most $(k-1) \cdot OPT_k + 1$ rounds.

Without loss of generality, let us assume that the distance between v_1 and v_k is the maximum among all pairs of originators. Let $P_{v_1v_k}$ be the shortest path between v_1 and v_k containing vertices $v_1, v_2, ..., v_k$. We know that as all vertices of graph G, the vertices of $P_{v_1v_k}$ must be within OPT_k distance from some originator. It follows that

$$length(P_{v_1 v_k}) \leq 2 \cdot (k-1) \cdot OPT_k.$$

Now we can pick as a broadcast originator the central vertex o of the path $P_{v_1 v_k}$ or one of the central vertices if $P_{v_1 v_k}$ has an even length. In at most $(k-1) \cdot OPT_k + 1$ rounds we will be able to inform all k originators from vertex o. In additional OPT_k rounds we will be able to complete the broadcasting from these k originators. □

To be able to judge how tight is the given lower bound on OPT_k, let us look at the path of length n. We observe that depending on the parity of n, the optimal originator will be the central (when n is odd) or one of the two central vertices (when n is even) of the path. In both cases for P_n we will have

$$OPT_1 = \left\lceil \frac{n}{2} \right\rceil.$$

Now let us pick $n = 2mk$. We will get an optimal placement of k originators in such a path by dividing the path into k segments of equal length and picking as originators the middle vertices of these segments (see Figure 36). This will give the following expression for OPT_k

$$OPT_k = m.$$

For OPT_1 in P_{2mk} graph we will have

$$OPT_1 = \left\lceil \frac{2mk}{2} \right\rceil = mk = k \cdot OPT_k.$$

By comparing this result with the upper bound from Theorem 29, we see that for P_{2mk} graph the given upper bound for OPT_1 differs from the actual value by only 1.

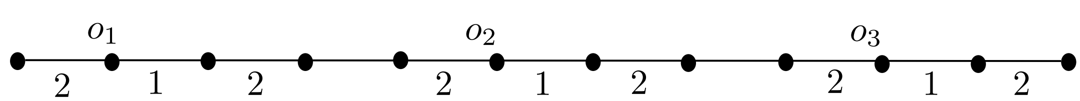

Figure 36: Optimal placement of $k = 3$ originators on P_{12} and the optimal broadcast scheme.

5.6 Distance from the broadcast center and the broadcast time

This section studies the relation between the broadcast time and the distance from the broadcast center. We show that the broadcast time of a vertex does not depend on its distance from the broadcast center.

In any graph, for the broadcast time of a vertex $v \notin BC$, the following bounds are true

$$b(BC) + 1 \leq b(v) \leq b(BC) + dist(v, BC).$$

The lower bound follows from the fact that having $b(v) = b(BC)$ would mean that $v \in BC$ which is a contradiction. From originator v in the first $dist(v, BC)$ rounds we can inform a vertex in the broadcast center and complete the broadcasting in the following $b(BC)$ rounds. Therefore, the upper bound follows.

In [104] the authors showed that the upper bound above is actually tight for the trees. They proved that for any vertex v in a tree

$$b(v) = dist(v, BC) + b(BC).$$

This equality says that by first informing via the shortest path (actually the only path) from originator v a vertex $u \in BC$ in $dist(v, BC)$ rounds and by using the broadcast scheme of u to complete the broadcasting in the next $b(BC)$ rounds, we will actually get an optimal broadcast scheme for v.

Whether the above equality holds for general graphs or not was an open question. In this section, by bringing a counterexample, we show that by adding only three edges to a tree we may have a graph where a vertex v may have arbitrary large distance from the broadcast center, but have a broadcast time $b(v) = b(BC)+1$. The constructed graph on n vertices has only $n + 2$ edges and maximum degree 3.

Theorem 30. *For any integer d, there exists a graph G having vertex v at distance d from the broadcast center and such that $b(v) = b(BC) + 1$.*

Proof. First, we will give an example of a graph which proves the theorem for $d = 2, 3$. After that, we will generalize our construction for arbitrary d.

Figure 37 shows an example of graph G for $d = 2, 3$. We observe that $BC = \{u_1, u_2, u_3, o\}$. Figure 38 shows a broadcast scheme from any of these vertices in G

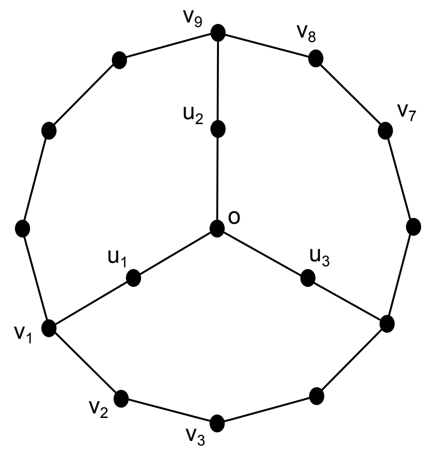

Figure 37: Illustration of the absence of the connection between broadcast time of a vertex and its distance from the broadcast center.

that completes in 5 rounds. The number on an edge, indicates the round at which the message is sent via that edge. Actually, Figure 38 illustrates the broadcast scheme only when the originator is vertex u_1 or o, but from the symmetry of the graph, it follows that for originators u_2 and u_3 we can construct similar schemes.

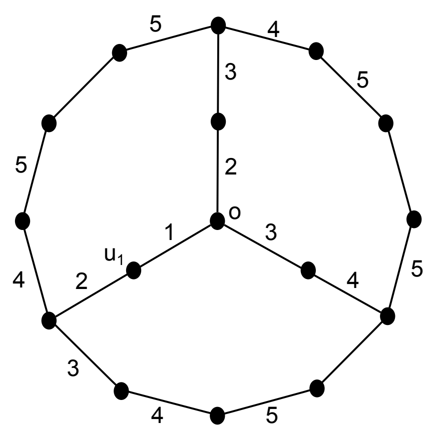

Figure 38: Broadcast scheme from the broadcast center finishing in 5 rounds.

Note that there are no other vertices which we may include in the broadcast center, since all other vertices of G have another vertex at distance 6 so they cannot complete broadcasting in 5 rounds (e.g. $dist(v_1, v_7) = 6$, $dist(v_2, v_8) = 6$, $dist(v_3, v_9) = 6$).

From Figure 37 we observe that $dist(v_2, BC) = 2$ and $dist(v_3, BC) = 3$. Figure 39 shows a broadcast scheme from vertices v_2 and v_3 which finishes in $6 = b(BC) + 1 = 5 + 1$ rounds. This completes the proof for $d = 2, 3$.

For the case $d > 3$, we generalize the graph in Figure 37 by replacing the three

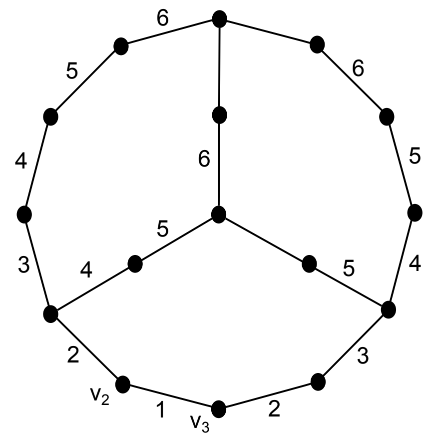

Figure 39: Broadcast scheme from the vertex v with $dist(v, BC) = 3$ finishes in $b(BC) + 1 = 6$ rounds.

paths of length 2 connecting the central vertex and the outer cycle with the paths of length $d - 1$. The new graph is presented in Figure 40. We show that in this graph, vertex v has a distance d from the broadcast center and its broadcast time differs from optimally chosen vertices only by one, i.e.

$$dist(v, BC) = d, \quad b(v) = b(BC) + 1.$$

Similar to the case $d \leq 3$, we note that in the graph presented in Figure 40, the broadcast center consists of four central vertices:

$$BC = \{u_1, u_2, u_3, o\}.$$

We also observe that by increasing the length of paths connecting the central vertex with the outer cycle by one we are increasing the broadcast time of vertices in the broadcast center by one. Therefore, from the fact that in the graph from Figure 37 the minimum broadcast time was 5 we get that for the graph from Figure 40

$$b(BC) = 5 + (d - 3) = d + 2.$$

It is clear that in the graph from Figure 40 we have $dist(v, BC) = d$. To complete the proof we need to describe a broadcast scheme of the originator v which completes in $b(BC) + 1 = d + 3$ rounds.

An example of such a broadcast scheme is the following: in $d + 3$ rounds vertex

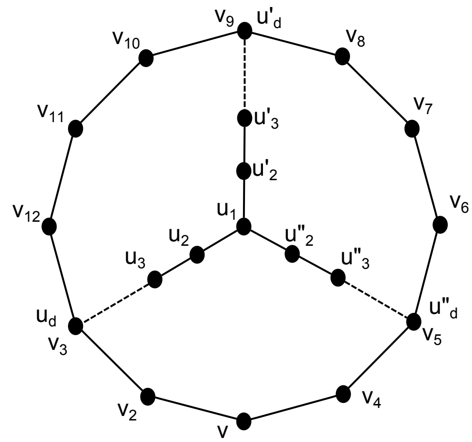

Figure 40: Illustration of the general case.

v informs all the vertices of path $vv_2v_1v_{12}v_{11}v_{10}v_9u_{d-1}u_{d-2}...u_3$ of length $d+3$ and the vertices of path $vv_4v_5v_6v_7v_8$. The vertex $v_3(u_d)$, which receives the message in the second round, after informing vertex v_{12} in the third round, will inform all the vertices of path $u_du_{d-1}...u_2u_1u_2'$ in the following d rounds. Similarly, vertex $v_5(u_d'')$, after receiving the message in the third round and sending it to v_6 in the fourth round, will inform all the vertices of path $u_d''u_{d-1}''...u_2''$ in the following $d-2$ rounds.

Note that any vertex in the graph belongs to at least one of the four paths mentioned above. Therefore, the described scheme is a valid broadcast scheme from originator v which completes in $d+3$ rounds. $\square$

Figure 41 illustrates the broadcast scheme from originator v for the case $d=5$.

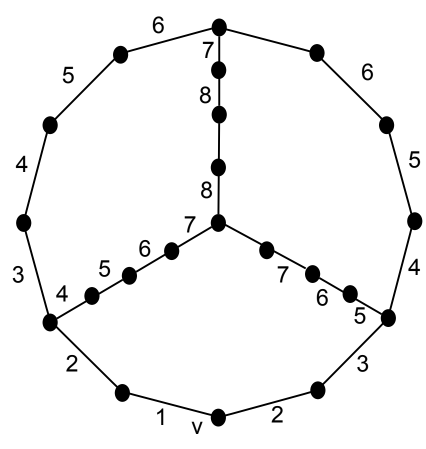

Figure 41: Illustration of the case $d=5$.

Chapter 6

Summary

In this thesis we studied several problems related to broadcasting in graphs.

We studied the family of graphs for which the broadcast time is equal to the diameter. We introduced the *diametral broadcast graph (dbg) problem* and presented several dbg constructions. These constructions solved the diametral broadcast graph problem for all possible combinations of number of vertices and diameter. In analogy to the *broadcast function*, we defined the *diametral broadcast function* $DB(n,d)$ as the minimum possible number of edges in a dbg on n vertices and diameter d. We gave the exact value of $DB(n,d)$ for certain n and d, while for general case, we presented upper bounds on $DB(n,d)$.

In the second part of this work, we studied several properties of the Knödel graph $W_{\Delta,n}$. In particular, we studied the routing and broadcasting problem in the Knödel graph. We obtained lower and upper bounds on the diameter of the Knödel graph $W_{\Delta,n}$ for all n and Δ. We showed that the presented bound differs from actual diameter by at most 2 for almost all Δ. We addressed the routing problem in the Knödel graph and gave a tight bound on the distance between any two vertices in $W_{\Delta,n}$. We showed that the presented bound differs from the actual distance by at most 2 for almost all vertex pairs in $W_{\Delta,n}$. We also addressed the broadcasting problem in $W_{\Delta,n}$. We presented a broadcast scheme and gave tight upper and lower bounds on the broadcast time of $W_{\Delta,n}$. The presented proofs are constructive and allowed to construct a short path between any pair of vertices and to perform fast broadcasting from any vertex in $W_{\Delta,n}$.

The third part of this work studied the broadcast function $B(n)$. We considered the possible vertex degrees and possible connections between vertices of different degrees in broadcast graphs. Using this, we presented new lower bounds on $B(n)$ when $n = 2^m - 2^k$ and $n = 2^m - 2^k + 1$ $(3 \leq k < m)$.

In the last part of this work, we presented several basic results we obtained during our research. In particular: we described all graphs with the worst possible broadcast time, we formulated the broadcast time problem as an IP problem, we presented slightly improved general upper bound on broadcast time, studied the relation between minimum and maximum broadcast times in a graph, studied the broadcasting from multiple optimally chosen originators, and addressed a conjecture regarding the relation of a vertex distance from the broadcast center and the broadcast time of that vertex.

Finally, we would like to mention few open questions for future research. One of them is to find a good lower bound on diametral broadcast function $DB(n, d)$ defined in Chapter 2. For given n and d such that $d + 1 \leq n \leq \sqrt{2} \cdot 2^{\frac{d}{2}}$ and d is odd, or $d + 1 \leq n \leq \frac{3}{2} \cdot 2^{\frac{d}{2}}$ we gave the exact value of $DB(n, d)$. When $2^{d-1} < n \leq 2^d$, a diametral broadcast graph, in fact, is a broadcast graph. Therefore, for this case, the known lower bounds on broadcast function $B(n)$ apply also for $DB(n, d)$. Meanwhile, for other values of n and d, we do not have a good lower bound on $DB(n, d)$.

Another open question is to prove Conjecture 19 about the broadcast time of the Knödel graph $W_{\Delta,n}$. Even if this conjecture will be hard to prove, we still believe that the presented lower bound on the broadcast time, based only on the diameter of $W_{\Delta,n}$, can be significantly improved. One possible approach to this is to use techniques similar to ones used for getting lower bounds on the broadcast time of butterfly graph BF_d and binary de Bruijn graph DB_d.

We presented some of the results of this thesis in [5, 41, 42, 43, 44].

Bibliography

[1] R. Ahlswede, L. Gargano, H.S. Haroutunian, and L.H. Khachatrian. Fault-tolerant minimum broadcast networks. *Networks*, 27(4):293–307, 1996.

[2] R. Ahlswede, H.S. Haroutunian, and L.H. Khachatrian. Messy broadcasting in networks. In *Communications and Cryptography*, volume 276 of *The Springer International Series in Engineering and Computer Science*, pages 13–24. Springer, 1994.

[3] S.B. Akers and B. Krishnamurthy. A group-theoretic model for symmetric interconnection networks. *IEEE Transactions on Computers*, 38(4):555–566, 1989.

[4] A. Bar-Noy, S. Guha, J.S. Naor, and B. Schieber. Multicasting in heterogeneous networks. In *Proceedings of the 13th ACM Symposium on Theory of Computing (STOC 1998)*, pages 448–453. ACM, 1998.

[5] G. Barsky, H. Grigoryan, and H.A. Harutyunyan. Lower bounds on broadcast function for $n = 24$ and 25. *In preparation.*

[6] R. Beier and J.F. Sibeyn. A powerful heuristic for telephone gossiping. In *The 7th International Colloquium on Structural Information and Communication Complexity (SIROCCO 2000), LAquila, Italy, 2000*, pages 17–36, 2000.

[7] J. C. Bermond, H. A. Harutyunyan, A. L. Liestman, and S. Perennes. A Note on the Dimensionality of Modified Knödel Graphs. *International Journal of Foundations of Computer Science*, 8(2):109–116, 1997.

[8] J.C. Bermond, P. Fraigniaud, and J.G. Peters. Antepenultimate broadcasting. *Networks*, 26(3):125–137, 1995.

[9] J.C. Bermond, P. Hell, A.L. Liestman, and J.G. Peters. Sparse broadcast graphs. *Discrete Applied Mathematics*, 36(2):97–130, 1992.

[10] J.C. Bermond, X. Muñoz, and A. Marchetti-Spaccamela. Induced broadcasting algorithms in iterated line digraphs. In *Euro-Par'96 Parallel Processing*, volume 1123 of *Lecture Notes in Computer Science*, pages 313–324. Springer, 1996.

[11] L. Changhong and Z. Kemin. The broadcast function value B(23) is 33 or 34. *Acta Mathematicae Applicatae Sinica (English Series)*, 16(3):329–331, 2000.

[12] S.C. Chau and A.L. Liestman. Constructing minimal broadcast networks. *Journal of Combinatorics, Information and System Sciences*, 10:110–122, 1985.

[13] S.C. Chau and A.L. Liestman. Constructing fault-tolerant minimal broadcast networks. *Journal of Combinatorics, Information and System Sciences*, 11:1–18, 1986.

[14] M.L. Chia, D. Kuo, and M.F. Tung. The multiple originator broadcasting problem in graphs. *Discrete Applied Mathematics*, 155(10):1188–1199, 2007.

[15] B.S. Chlebus, L. Gasieniec, A. Gibbons, A. Pelc, and W. Rytter. Deterministic broadcasting in ad hoc radio networks. *Distributed Computing*, 15(1):27–38, 2002.

[16] F. Comellas, H.A. Harutyunyan, and A.L. Liestman. Messy broadcasting in multidimensional directed tori. *Journal of Interconnection Networks*, 4(1):37–51, 2003.

[17] A. Dessmark and A. Pelc. Deterministic radio broadcasting at low cost. *Networks*, 39(2):88–97, 2002.

[18] A. Dessmark and A. Pelc. Broadcasting in geometric radio networks. *Journal of Discrete Algorithms*, 5(1):187–201, 2007.

[19] K. Diks and A. Pelc. Broadcasting with universal lists. In *Proceedings of the 28th Hawaii International Conference on System Sciences (HICSS 1995)*, volume 2, pages 564–573. IEEE, 1995.

[20] K. Diks and A. Pelc. Broadcasting with universal lists. *Networks*, 27(3):183–196, 1996.

[21] M. Dinneen, M. Fellows, and V. Faber. Algebraic constructions of efficient broadcast networks. In *Applied Algebra, Algebraic Algorithms and Error-Correcting Codes*, volume 539 of *Lecture Notes in Computer Science*, pages 152–158. Springer, 1991.

[22] M.J. Dinneen, J.A. Ventura, M.C. Wilson, and G. Zakeri. Compound constructions of broadcast networks. *Discrete Applied Mathematics*, 93(2):205–232, 1999.

[23] M. Elkin and G. Kortsarz. Combinatorial logarithmic approximation algorithm for directed telephone broadcast problem. In *Proceedings of the 34th ACM Symposium on Theory of Computing (STOC 2002)*, pages 438–447, New York, NY, USA, 2002. ACM.

[24] M. Elkin and G. Kortsarz. Sublogarithmic approximation for telephone multicast: path out of jungle (extended abstract). In *Proceedings of the 14th ACM-SIAM Symposium on Discrete Algorithms (SODA 2003)*, pages 76–85, Philadelphia, PA, USA, 2003. SIAM.

[25] A.M. Farley. Minimal broadcast networks. *Networks*, 9(4):313–332, 1979.

[26] A.M. Farley. Broadcast time in communication networks. *SIAM Journal on Applied Mathematics*, 39(2):385–390, 1980.

[27] A.M. Farley, S. Hedetniemi, S. Mitchell, and A. Proskurowski. Minimum broadcast graphs. *Discrete Mathematics*, 25:189–193, 1979.

[28] A.M. Farley and S.T. Hedetniemi. Broadcasting in grid graphs. In *Proceedings of the 9th Southeastern Conference on Combinatorics, Graph Theory, and Computing*, pages 275–288, Boca Raton, 1978.

[29] A.M. Farley and A. Proskurowski. Broadcasting in trees with multiple originators. *SIAM Journal on Algebraic and Discrete Methods*, 2:381, 1981.

[30] G. Fertin and A. Raspaud. A survey on Knödel graphs. *Discrete Applied Mathematics*, 137(2):173–195, 2004.

[31] G. Fertin, A. Raspaud, H. Schröder, O. Sỳkora, and I. Vrt'o. Diameter of the Knödel graph. In *Graph-Theoretic Concepts in Computer Science (WG 2000)*, volume 1928 of *Lecture Notes in Computer Science*, pages 149–160. Springer, 2000.

[32] Message Passing Interface Forum. *MPI: A Message-Passing Interface Standard, Version 2.2*. High Performance Computing Center Stuttgart (HLRS), 2009.

[33] P. Fraigniaud and E. Lazard. Methods and problems of communication in usual networks. *Discrete Applied Mathematics*, 53(1-3):79–133, 1994.

[34] P. Fraigniaud, A.L. Liestman, and D. Sotteau. Open problems. *Parallel Processing Letters*, 3(4):507–524, 1993.

[35] P. Fraigniaud and J. G. Peters. Minimum linear gossip graphs and maximal linear (δ, k)-gossip graphs. *Networks*, 38(3):150–162, 2001.

[36] P. Fraigniaud and S. Vial. Comparison of heuristics for one-to-all and all-to-all communications in partial meshes. *Parallel Processing Letters*, 9(1):9–20, 1999.

[37] M.R. Garey and D.S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman & Co., New York, NY, USA, 1979.

[38] L. Gargano, A. Pelc, S. Pérennes, and U. Vaccaro. Efficient communication in unknown networks. *Networks*, 38(1):39–49, 2001.

[39] L. Gargano and U. Vaccaro. On the construction of minimal broadcast networks. *Networks*, 19(6):673–689, 1989.

[40] M. Grigni and D. Peleg. Tight bounds on minimum broadcast networks. *SIAM Journal on Discrete Mathematics*, 4(2):207–222, 1991.

[41] H. Grigoryan and H.A. Harutyunyan. Broadcasting and routing in the Knödel graph. *In preparation.*

[42] H. Grigoryan and H.A. Harutyunyan. Diametral broadcast graphs. *Submitted for publication.*

[43] H. Grigoryan and H.A. Harutyunyan. Broadcasting in the Knödel graph. In *9th International Conference on Computer Science and Information Technologies (CSIT 2013), (to appear)*, Yerevan, Armenia, 2013.

[44] H. Grigoryan and H.A. Harutyunyan. Tight bound on the diameter of the Knödel graph. In *International Workshop On Combinatorial Algorithms (IWOCA 2013)*, Rouen, France, 2013. Springer.

[45] T.E. Hart and H.A. Harutyunyan. Improved messy broadcasting in hypercubes and complete bipartite graphs. In *Proceedings of the 33rd Southeastern Conference on Combinatorics, Graph Theory, and Computing*, pages 181–196, Boca Raton, 2002.

[46] H.A. Harutyunyan. Minimum multiple message broadcast graphs. *Networks*, 47(4):218–224, 2006.

[47] H.A. Harutyunyan. An efficient vertex addition method for broadcast networks. *Internet Mathematics*, 5(3):211–225, 2008.

[48] H.A. Harutyunyan, P. Hell, and A.L. Liestman. Messy broadcasting - decentralized broadcast schemes with limited knowledge. *Discrete Applied Mathematics*, 159(5):322–327, 2011.

[49] H.A. Harutyunyan, R. Katragadda, and C.D. Morosan. Efficient heuristic for multicasting in arbitrary networks. In *27th International Conference on Advanced Information Networking and Applications (WAINA 2009)*, pages 61–66, Bradford, 2009.

[50] H.A. Harutyunyan and A.L. Liestman. Messy broadcasting. *Parallel Processing Letters*, 8(2):149–159, 1998.

[51] H.A. Harutyunyan and A.L. Liestman. More broadcast graphs. *Discrete Applied Mathematics*, 98(1-2):81–102, 1999.

[52] H.A. Harutyunyan and A.L. Liestman. Improved upper and lower bounds for k-broadcasting. *Networks*, 37(2):94–101, 2001.

[53] H.A. Harutyunyan and A.L. Liestman. On the monotonicity of the broadcast function. *Discrete Mathematics*, 262(1-3):149–157, 2003.

[54] H.A. Harutyunyan and A.L. Liestman. Upper bounds on the broadcast function using minimum dominating sets. *Discrete Mathematics*, 312(20):2992–2996, 2012.

[55] H.A. Harutyunyan, A.L. Liestman, K. Makino, and T. C. Shermer. Nonadaptive broadcasting in trees. *Networks*, 57(2):157–168, 2011.

[56] H.A. Harutyunyan, A.L. Liestman, and B. Shao. A linear algorithm for finding the k-broadcast center of a tree. *Networks*, 53(3):287–292, 2009.

[57] H.A. Harutyunyan and E. Maraachlian. Linear algorithm for broadcasting in unicyclic graphs. In *Computing and Combinatorics*, volume 4598 of *Lecture Notes in Computer Science*, pages 372–382. Springer, 2007.

[58] H.A. Harutyunyan and E. Maraachlian. On broadcasting in unicyclic graphs. *Journal of Combinatorial Optimization*, 16(3):307–322, 2008.

[59] H.A. Harutyunyan and C.D. Morosan. On the minimum path problem in Knödel graphs. *Networks*, 50(1):86–91, 2007.

[60] H.A. Harutyunyan and B. Shao. An efficient heuristic for broadcasting in networks. *Journal of Parallel and Distributed Computing*, 66(1):68–76, 2006.

[61] H.A. Harutyunyan and P. Taslakian. Orderly broadcasting in a 2D torus. In *Proceedings of the 8th International Conference on Information Visualization (IV 2004)*, pages 370–375, London, UK, 2004.

[62] H.A. Harutyunyan and W. Wang. Broadcasting algorithm via shortest paths. In *Proceedings of the 16th International Conference on Parallel and Distributed Systems (ICPADS 2010)*, pages 299–305, Shanghai, China, 2010.

[63] S.M. Hedetniemi, S.T. Hedetniemi, and A.L. Liestman. A survey of gossiping and broadcasting in communication networks. *Networks*, 18(4):319–349, 1988.

[64] M. C. Heydemann, N. Marlin, and S. Pérennes. Complete rotations in Cayley graphs. *European Journal of Combinatorics*, 22(2):179–196, 2001.

[65] C.J. Hoelting, D.A. Schoenefeld, and R.L. Wainwright. A genetic algorithm for the minimum broadcast time problem using a global precedence vector. In

Proceedings of the 1996 ACM Symposium on Applied Computing (SAC 1996), pages 258–262. ACM, 1996.

[66] J. Hromkovič, C.D. Jeschke, and B. Monien. Note on optimal gossiping in some weak-connected graphs. *Theoretical Computer Science*, 127(2):395–402, 1994.

[67] J. Hromkovic, R. Klasing, B. Monien, and R. Peine. Dissemination of information in interconnection networks (broadcasting and gossiping). In *Combinatorial Network Theory*, volume 1 of *Applied Optimization*, pages 125–212. Springer, 1996.

[68] J. Hromkovic, R. Klasing, A. Pelc, P. Ruzicka, and W. Unger. *Dissemination of information in communication networks: broadcasting, gossiping, leader election, and fault-tolerance.* Texts in Theoretical Computer Science. An EATCS Series. Springer, 2005.

[69] L. Khachatrian and H. Haroutunian. Minimal broadcast trees. In *XIV Moscow–Minsk All Union School of Computing Networks*, pages 36–40, Minsk, Belarus, 1989.

[70] L.H. Khachatrian and O.S. Harutounian. Construction of new classes of minimal broadcast networks. In *Conference on Coding Theory*, pages 69–77, Dilijan, Armenia, 1990.

[71] J.H. Kim and K.Y. Chwa. Optimal broadcasting with universal lists based on competitive analysis. *Networks*, 45:224–231, 2005.

[72] R. Klasing, B. Monien, R. Peine, and E.A. Stöhr. Broadcasting in butterfly and DeBruijn networks. *Discrete Applied Mathematics*, 53(1):183–197, 1994.

[73] W. Knödel. New gossips and telephones. *Discrete Mathematics*, 13(1):95, 1975.

[74] J.C. Konig and E. Lazard. Minimum k-broadcast graphs. *Discrete Applied Mathematics*, 53(1-3):199–209, 1994.

[75] G. Kortsarz and D. Peleg. Approximation algorithms for minimum-time broadcast. *SIAM Journal on Discrete Mathematics*, 8(3):401–427, 1995.

[76] R. Labahn. The telephone problem for trees. *Journal of Information Processing and Cybernetics*, 22(9):475–485, 1986.

[77] R. Labahn. Extremal broadcasting problems. *Discrete Applied Mathematics*, 23(2):139–155, 1989.

[78] R. Labahn. A minimum broadcast graph on 63 vertices. *Discrete Applied Mathematics*, 53(1-3):247–250, 1994.

[79] E. Lazard. Broadcasting in DMA-bound bounded degree graphs. *Discrete Applied Mathematics*, 37:387–400, 1992.

[80] S. Lee and J.A. Ventura. An algorithm for constructing minimal c-broadcast networks. *Networks*, 38(1):6–21, 2001.

[81] C. Li, T.E. Hart, K.J. Henry, and I.A. Neufeld. Average-case messy broadcasting. *Journal of Interconnection Networks*, 9(4):487–505, 2008.

[82] A.L. Liestman. Fault-tolerant broadcast graphs. *Networks*, 15(2):159–171, 1985.

[83] A.L. Liestman and J.G. Peters. Broadcast networks of bounded degree. *SIAM Journal on Discrete Mathematics*, 1(4):531–540, 1988.

[84] M. Maheo and J.F. Sacle. Some minimum broadcast graphs. *Discrete Applied Mathematics*, 53(1-3):285, 1994.

[85] E. Maraachlian. *Optimal broadcasting in treelike graphs*. PhD thesis, Concordia University, Montreal, P.Q., Canada, 2010.

[86] G. Marco and A. Pelc. Deterministic broadcasting time with partial knowledge of the network. In *Algorithms and Computation*, volume 1969 of *Lecture Notes in Computer Science*, pages 374–385. Springer, 2000.

[87] V.E. Mendia and D. Sarkar. Optimal broadcasting on the star graph. *IEEE Transactions on Parallel and Distributed Systems*, 3(4):389–396, 1992.

[88] M. Middendorf. Minimum broadcast time is NP-complete for 3-regular planar graphs and deadline 2. *Information Processing Letters*, 46(6):281–287, 1993.

[89] S. Mitchell and S. Hedetniemi. A census of minimum broadcast graphs. *Journal of Combinatorics, Information and System Sciences*, 5:141–151, 1980.

[90] J.H. Park and K.Y. Chwa. Recursive circulant: a new topology for multicomputer networks (extended abstract). In *International Symposium on Parallel Architectures, Algorithms and Networks (ISPAN 1994)*, pages 73–80, Kanazawa, 1994.

[91] A. Pelc. Fault-tolerant broadcasting and gossiping in communication networks. *Networks*, 28(3):143–156, 1996.

[92] D. Peleg. Time-efficient broadcasting in radio networks: A review. In *Distributed Computing and Internet Technology*, volume 4882 of *Lecture Notes in Computer Science*, pages 1–18. Springer, 2007.

[93] D. Peleg and T. Radzik. Time-efficient broadcast in radio networks. In *Graphs and Algorithms in Communication Networks*, Texts in Theoretical Computer Science. An EATCS Series, pages 311–334. Springer, 2010.

[94] S. Perennes. Lower bounds on broadcasting time of de Bruijn networks. In *Euro-Par'96 Parallel Processing*, volume 1123 of *Lecture Notes in Computer Science*, pages 325–332. Springer, 1996.

[95] A. Proskurowski. Minimum broadcast trees. *IEEE Transactions on Computers*, 100(30):363–366, 1981.

[96] R. Ravi. Rapid rumor ramification: approximating the minimum broadcast time. In *Proceedings of the 35th Symposium on Foundations of Computer Science (FOCS 1994)*, pages 202–213, 1994.

[97] F.F. Rivera, O. Plata, and E.L. Zapata. Broadcasting algorithm in computer networks: accumulative depth. *IEEE Proceedings-Computers and Digital Techniques*, 137(6):427–432, 1990.

[98] A. Rosenthal and P. Scheuermann. Universal rankings for broadcasting in tree networks. In *Proceedings of the 25th Allerton Conference on Communication, Control and Computing*, pages 641–649, USA, 1987.

[99] J.F. Sacle. Lower bounds for the size in four families of minimum broadcast graphs. *Discrete Mathematics*, 150(1):359–369, 1996.

[100] P. Scheuermann and M. Edelberg. Optimal broadcasting in point-to-point computer networks. *Department of Electrical Engineering and Computer Science, Northwestern University, Evanston, IL, Technical Report*, 1981.

[101] P. Scheuermann and G. Wu. Heuristic algorithms for broadcasting in point-to-point computer networks. *IEEE Transactions on Computers*, 100(33):804–811, 1984.

[102] C. Schindelhauer. On the inapproximability of broadcasting time. In *Approximation Algorithms for Combinatorial Optimization*, volume 1913 of *Lecture Notes in Computer Science*, pages 226–237. Springer, 2000.

[103] B. Shao. *On k-broadcasting in graphs*. PhD thesis, Concordia University, Montreal, P.Q., Canada, 2006.

[104] P.J. Slater, E.J. Cockayne, and S.T. Hedetniemi. Information dissemination in trees. *SIAM Journal on Computing*, 10:692–701, 1981.

[105] I. Stojmenovic, editor. *Handbook of wireless networks and mobile computing*. John Wiley & Sons, Inc., New York, NY, USA, 2002.

[106] R. Thakur and W. Gropp. Open issues in MPI implementation. In *Advances in Computer Systems Architecture*, volume 4697 of *Lecture Notes in Computer Science*, pages 327–338. Springer, 2007.

[107] J.A. Ventura and X. Weng. A new method for constructing minimal broadcast networks. *Networks*, 23(5):481–497, 1993.

[108] M.X. Weng and J.A. Ventura. A doubling procedure for constructing minimal broadcast networks. *Telecommunication Systems*, 3(3):259–293, 1994.

[109] J. Xiao and X. Wang. A research on minimum broadcast graphs. *Chinese Journal of Computers*, 11:99–105, 1988.

[110] J. Zhou and K. Zhang. A minimum broadcast graph on 26 vertices. *Applied Mathematics Letters*, 14(8):1023–1026, 2001.