Journal of Graph Algorithms and Applications
http://jgaa.info/ vol. 29, no. 1, pp. 125–134 (2025)
DOI: 10.7155/jgaa.v29i1.3011

Broadcast Schemes of Hypercubes

Pouria Alikhanifard[1] *Mohammadhossein Fakharan*[1] *Hovhannes A. Harutyunyan*[1]

[1]Department of Computer Science and Software Engineering,
Concordia University, Canada.

Submitted: December 2024	Accepted: May 2025	Published: June 2025
Article type: Regular paper	Communicated by: Alexander Wolff	

Abstract. Broadcasting is a fundamental process in network communication, modeled as the dissemination of information across vertices in a graph. This paper investigates the broadcast problem in hypercube graphs and revisits the dimensional broadcast schemes. We propose a novel algorithm that generates all valid minimum-time broadcast schemes by using the recursive structure of hypercubes. Additionally, we compute the total number of valid minimum-time broadcast schemes in hypercubes, which is also the number of all spanning binomial trees. We also give an enumeration of all valid minimum-time broadcast schemes.

1 Introduction

Broadcasting is a fundamental process for information dissemination in connected networks, which are commonly modeled as undirected graphs $G = (V, E)$, where V represents the vertex set and E represents the edge set of the graph G. Classical broadcasting is organized in discrete time units, beginning with an initial vertex, referred to as the originator, holding the message. During each time unit, every informed vertex (sender) transmits the message to one of its uninformed neighbors (receiver). This process continues until all vertices in the graph are informed. A *broadcast scheme* is a collection of all edges, each labeled with the time unit during which it is used in the broadcast. The objective is to find a broadcast scheme that achieves the minimum possible broadcast time.

The study of the minimum broadcast graphs (mbgs), graphs with the smallest possible broadcast time unit and number of edges, has a rich history, originating with Farley et al. [11] in 1979. In their work, they introduced the concept of mbgs and identified the first infinite family of such graphs, consisting of hypercube graphs with $n = 2^k$ vertices, where k is a positive integer. Subsequently, Park and Chwa [6] proposed another family of mbgs for graphs on 2^k vertices. Knödel graphs are shown to be mbgs on $n = 2^k$ [24] and $n = 2^k - 2$ vertices by Khachatrian and Haroutunian [23] and Dinneen et al. [7], independently. The reader is referred to foundational works [2, 4, 5, 8], as well as to subsequent developments and technical insights [13–17, 23, 27]. In addition, we recommend surveys [12, 18, 19, 21, 22] for a complete overview of the topic.

Determining the broadcast time $b(u)$ for an arbitrary originator u in an arbitrary graph G has been proved to be NP-complete in [28]. The problem remains NP-complete even for 3-regular planar graphs [25], or when restricted to graphs with the feedback vertex set number one [29]. Recently, the problem was proved to be NP-complete for cactus and graphs with pathwidth 2 [1]. Ravi [26] also investigated the minimum broadcast time problem under the classical model to derive an $\mathcal{O}(\log^2 n/ \log\log n)$-approximation algorithm for the minimum broadcast time problem of a graph on n vertices. This approximation has improved to $\mathcal{O}(\log n)$ [3]. The current best approximation ratio of $\mathcal{O}(\log n/ \log\log n)$ for broadcasting was introduced by Elkin and Kortsarz [9, 10].

E-mail addresses: pouria.alikhanifard@concordia.ca (Pouria Alikhanifard) mohammadhossein.fakharan@mail.concordia.ca (Mohammadhossein Fakharan) haruty@cs.concordia.ca (Hovhannes A. Harutyunyan)

In this paper, we consider broadcasting in the hypercube graphs. In [11], the authors demonstrated the existence of specific broadcasting called dimensional broadcast schemes in the hypercube H_k. These schemes require that if $(\alpha_1, \ldots, \alpha_k)$ is a permutation of dimensions $(1, 2, \ldots, k)$ of hypercube H_k, every informed vertex at time unit i must inform its neighbor along dimension α_i. In this work, we extend these schemes and propose an algorithm that generates all possible minimum-time broadcast schemes for hypercubes. We also, give an enumeration of all valid minimum-time broadcast schemes.

2 Notations

Definition 1. The *hypercube* of dimension k, denoted by H_k, is a simple graph with vertices representing 2^k binary strings of length $k \geq 1$ such that adjacent vertices have binary strings differing in exactly one-bit position, called a *dimension*, numbered $1, \ldots, k$. An edge lies *along dimension* i if the differing bit is in position i.
Also, H_k is defined recursively as follows (for any $k \in \mathbb{N}$):

- H_0 is a single vertex with no edges.
- H_k consists of two copies of H_{k-1} with edges joining corresponding vertices with a new dimension k, for any $k \geq 1$.

We use both definitions in the proofs. Since hypercubes are vertex and edge transitive [20], with no loss of generality, throughout this paper, vertex 00...00 is considered to be the originator. For simplicity, we refer to it as 0 for the rest of the paper.

Definition 2. A *binomial tree* B_k of dimension k on 2^k vertices is defined recursively as follows:
B_0 is a single vertex with no edges.
B_k consists of two copies of B_{k-1} connecting their roots by an edge and considering one of them as the root of B_k (see Figure 1).

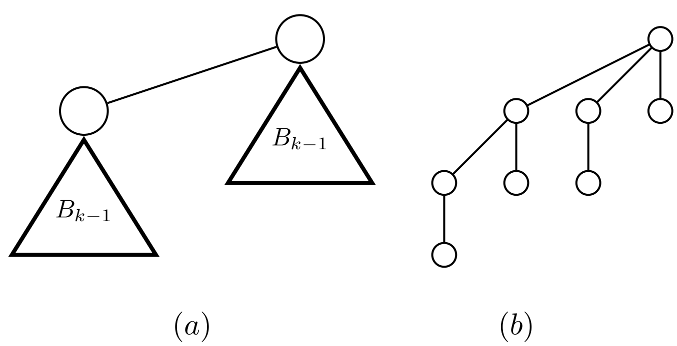

Figure 1: a) Construction of B_k from two copies of B_{k-1}. b) An example of binomial tree B_3.

Definition 3. Let G be a graph on n vertices and v be the broadcast originator in graph G. $b(G, v)$ denotes the minimum number of time units required to broadcast from originator v in graph G. The *broadcast time* of graph G, $b(G) = \max\{b(G, v) \mid v \in V(G)\}$ indicates the maximum number of time units required from any originator to broadcast in graph G.

Let u be the root of a binomial tree B_k. Then B_k has 2^k vertices and $b(B_k, u) = k$. Note that if every informed vertex is active (i.e., sends the message) during each time unit of the process, then the resulting broadcast scheme after time unit t forms a binomial tree B_t.

Note that for every graph G on n vertices, we have $b(G) \geq \lceil \log n \rceil$, since the number of informed vertices can at most double during each time unit. If $b(G) = \lceil \log n \rceil$, then G is called a *broadcast graph*. It means that for every vertex v, we have $b(G, v) = \lceil \log n \rceil$. For a hypercube H_k, if all informed vertices use dimension t in time unit t, for $1 \leq t \leq k$, then every vertex will be informed after time unit k. Hence,

$b(H_k) = k$. A broadcast graph G on n vertices is called *minimum broadcast graph* (*mbg*) if there does not exist any other broadcast graph G' on n vertices such that $|E(G')| < |E(G)|$. Since $deg(v) \geq k$ for any broadcast graph on 2^k vertices, H_k is an mbg.

The distance between two vertices u and v in a graph G is the shortest path length between them, denoted by $d(u, v)$.

3 Main Results

In this section, we provide our algorithm to generate valid minimum-time broadcast schemes in hypercubes. Furthermore, we prove its correctness and enumerate all algorithm-generated broadcast schemes. Finally, we show that the algorithm gives all valid minimum-time broadcast schemes in hypercubes.
We use the next two combinatorial lemmas to prove the main theorem.

Lemma 3.1. *For integers t and k such that $2 \leq k \leq t$ the following holds:*

$$\binom{k-2}{k-2} + \binom{k-1}{k-2} + \binom{k}{k-2} + \cdots + \binom{t-2}{k-2} = \binom{t-1}{k-1}.$$

Proof: We provide a combinatorial proof. RHS represents the total number of ways to select $k-1$ elements from a set of $t-1$ elements. We evaluate this by examining cases based on the choices involving specific elements from the set:

- There are $\binom{t-2}{k-2}$ ways to select the subset such that the last element is chosen.
- In $\binom{t-3}{k-2}$ ways, the second-to-last element is selected while the last element is excluded.
- Similarly, $\binom{t-4}{k-2}$ counts the cases where the third-to-last element is chosen, with the last two elements excluded.

This process continues until only the first $k-1$ elements are chosen which is exactly one way to choose them. The summation of these terms (LHS) thus equates to the RHS. □

Lemma 3.2. *For positive integers n and k such that $k < n$ the following holds:*

$$\binom{k-1}{k-1}(n-k) + \binom{k}{k-1}(n-k-1) + \binom{k+1}{k-1}(n-k-2) + \cdots + \binom{n-2}{k-1} = \binom{n}{k+1}.$$

Proof: As a combinatorial proof, observe that the first term on the left-hand side counts the number of ways to choose $k+1$ elements from the set $\{1, \ldots, n\}$ such that the second-largest element is k. Similarly, the second term counts the number of such subsets where the second-largest element is $k+1$, and so on, up to the last term, which corresponds to the case where the second-largest element is $n-1$. Summing over all possible values of the second-largest element yields the total number of ways to choose $k+1$ elements from n, as given by the right-hand side. □

We present our algorithm, whose core idea is based on the recursive definition of hypercubes. Let H_n be a hypercube of dimension n. When the originator selects an arbitrary dimension i in the first time unit, the hypercube H_n is effectively divided into two smaller hypercubes H_{n-1} by deleting all i-th dimensional edges, each containing an informed vertex. These informed vertices can then independently broadcast within their respective H_{n-1} subhypercubes. In the second time unit, the informed vertices in the two H_{n-1} subhypercubes are free to use different, arbitrary dimensions for broadcasting, with the restriction that neither they nor their descendants can reuse dimension i in subsequent time units. This process continues, utilizing the recursive structure of H_{n-1} in each time unit to advance the broadcasting process. The algorithm operates as follows: Initially, the originator has a set of all dimensions, referred to as the *allowed* set. When an informed vertex u uses arbitrary dimension j from its own allowed set to inform a neighbor v, vertex u removes dimension j from its own allowed set and transfers the remaining dimensions to v. This ensures that each vertex broadcasts without violating the dimensional constraints dictated by the structure of hypercube.

Algorithm 1: Hypercube Graph Broadcasting with Allowed Dimensions

Input : Hypercube $G = (V, E)$ of dimension n, originator o, message M
Result: All vertices in G receive message M

1 Initialize all vertices $v \in V$ as uninformed;
2 Mark o as informed;
3 Initialize $t \leftarrow 1$;
4 Initialize allowed dimensions for o: $allowed(o) \leftarrow \{1, 2, \ldots, n\}$;
5 **while** $t \leq n$ **do**
6     $t \leftarrow t + 1$;
7     $curr\_informed \leftarrow \{v \in V \mid \text{vertex } v \text{ is marked as informed}\}$;
8   **foreach** *vertex* v *in* $curr\_informed$ **do**
9   Arbitrarily select a dimension $i \in allowed(v)$;
10   Remove i from $allowed(v)$;
11   Determine neighbor u of v along dimension i;
12         $M(u) \leftarrow M$ `// Node` u `receives the message`;
13         $allowed(u) \leftarrow allowed(v)$ `// Pass remaining allowed dimensions`;
14   Mark u as informed;

Since each call at time unit t uses a dimension, say i, it informs two vertices belonging to two different smaller hypercubes. Moreover, none of the subsequent broadcasts initiated from these vertices reuse dimension i, ensuring that their descendants remain in two disjoint sub-hypercubes and do not overlap. As a result, the broadcast processes are entirely separated, and the number of informed vertices before time unit t is exactly 2^{t-1}. Therefore, the `for` loop in line 8 iterates exactly 2^{t-1} times for each t. Consequently, this results in a total of $\sum_{i=1}^{n} 2^{i-1} = 2^n - 1$ iterations. Hence, the time complexity of the algorithm is $\Theta(2^n)$.

The recursive structure of hypercubes is instrumental in our approach. In the following theorem, we utilize this structure to prove the correctness of the proposed algorithm.

Theorem 3.3. *Let H_n be a hypercube of dimension n. Then any broadcast scheme generated by Algorithm 1 is a minimum time broadcast scheme in H_n.*

Proof: We prove the validity by induction on n. The base case $n = 1$ is straightforward. Suppose that $n \geq 1$. Without loss of generality, assume that vertex $0 = (0, \ldots, 0)$ uses the first dimension in time unit 1 to inform vertex $u = (1, 0, \ldots, 0)$. By the recursive structure of hypercubes, H_n consists of two copies, H^0_{n-1} and H^1_{n-1}, which contain vertices 0 and u, respectively. Thus, in each H^i_{n-1} for $i = 0, 1$, there is an initially informed vertex.
According to Algorithm 1, no vertex will use dimension 1 again. By the induction hypothesis, any broadcast scheme in Algorithm 1 for H^0_{n-1} and H^1_{n-1} is valid and completes in $n - 1$ time units. Consequently, combining any two schemes for H^0_{n-1} and H^1_{n-1} coming from Algorithm 1 with the use of the first dimension in time unit 1 forms a valid minimum-time broadcast scheme for H_n. □

Remark 3.1. After each time unit, when a vertex informs its neighbor of dimension i, all its descendant vertices in this branch are prevented from using dimension i in next time units. Thus, it appears that by following Algorithm 1, many possible valid minimum-time broadcast schemes are ignored.
To illustrate this with a simple observation, consider the following scenario. Without loss of generality, suppose that the first dimension is used in time unit 1. Assume that two vertices, $0a_2 \cdots a_{n-2}11$ and $1a_2 \cdots a_{n-2}10$, are informed. Imagine that in one of the time units, instead of informing vertices $0a_2 \cdots a_{n-2}10$ and $1a_2 \cdots a_{n-2}11$, respectively, these vertices inform $1a_2 \cdots a_{n-2}11$ and $0a_2 \cdots a_{n-2}10$ by using dimension 1, respectively, as shown in Figure 2. At first glance, it appears that this broadcast scheme can be completed in the minimum number of time units, yet Algorithm 1 does not generate this broadcast scheme. However, the next theorem, surprisingly, shows that not only can this scenario not occur, but also that Algorithm 1 gives us all valid minimum-time broadcast schemes.

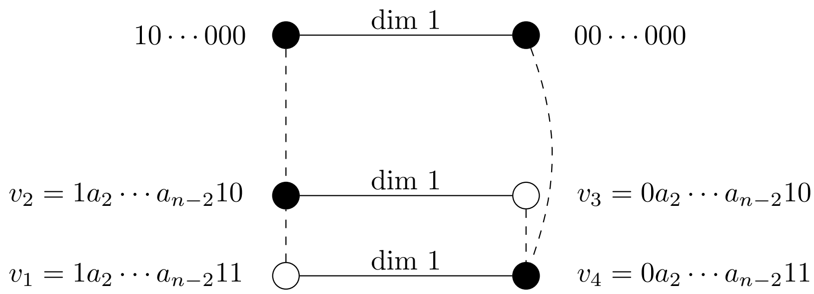

Figure 2: Informed vertices v_2 and v_4 use dimension 1 and inform vertices v_3 and v_1, respectively, instead of informing v_1 and v_3, respectively.

In the next theorem, we show that the previous remark is not correct, and Algorithm 1 gives us all valid minimum-time broadcast schemes.

Theorem 3.4. *Let H_n be a hypercube of dimension n. All n-time unit valid broadcast schemes of H_n are generated by Algorithm 1.*

Proof: Note that if B_n is a binomial spanning subtree of H_n rooted at vertex u, then starting the broadcast from the originator u gives a valid minimum-time broadcast scheme. Conversely, since H_n has 2^n vertices, every informed vertex must be active in each time unit of any valid minimum-time broadcast scheme (consisting of n time units). Hence, as discussed in Section 2, every valid minimum-time broadcast scheme corresponds to a spanning binomial subtree of H_n of dimension n, and vice versa. Note that by Theorem 3.3, all broadcast schemes produced by Algorithm 1 are valid. We show that the number of binomial trees of H_n is equal to the total number of broadcast schemes produced by Algorithm 1.
First, we analyze Algorithm 1. Since in each time unit, one dimension is removed from the allowed set of each informed vertex, in time unit t, each informed vertex has $n - t + 1$ available dimensions to send the message. Thus, because there are 2^{t-1} informed vertices before time unit t, the broadcast scheme of Algorithm 1 has $(n - t + 1)^{2^{t-1}}$ ways to extend the scheme in time unit t. Consequently, the total number of broadcast schemes produced by Algorithm 1 is

$$n(n-1)^{2^1}(n-2)^{2^2}\cdots 3^{2^{n-3}}2^{2^{n-2}} = \prod_{i=0}^{n-1}(n-i)^{2^i}.$$

Now, we show that the number of binomial trees of H_n, denoted by $f(n)$, also equals $\prod_{i=0}^{n-1}(n-i)^{2^i}$. To compute the distances of all vertices from the originator 0, we employ a breadth-first search (BFS) algorithm. It is worth mentioning that the distance between two vertices is equal to the number of different bits in their binary representation. We classify vertices based on this distance, denoting all vertices at a distance d from 0 as level d vertices.
We demonstrate, by using strong induction on d, that in every valid minimum-time broadcast scheme, all level $d+1$ vertices must be informed just by level d vertices, $1 \le d \le n-1$. This means that every broadcast scheme of H_n is a shortest-path broadcast scheme. In other words, every vertex receives the message via a shortest path from the originator. Note that every informed vertex cannot be idle in all time units. If not, then the number of informed vertices after time unit n will be less than 2^n. So, the originator, vertex 0, is responsible for informing all its neighbors which shows the correctness of induction's base case for $d = 1$. Now, assume as the inductive hypothesis that for $1 \le l \le d$, all level l vertices are informed just by level $l - 1$ vertices. We aim to show that this property holds for $\binom{n}{d+1}$ vertices of level $d + 1$. It is worth mentioning that once a level d vertex u is informed in time unit t, it is obligated to inform $n - t$ of its neighbors at the level $d + 1$ because there is no idle vertex as mentioned above and u cannot inform any level $d - 1$ vertices by IH.
Next, we use a second induction on d to show that the number of level d vertices which become informed during time unit t is $\binom{t-1}{d-1}$. This is because this count matches the number of informed vertices at level $d - 1$ by the end of time unit $t - 1$, which by IH is

$$\binom{d-2}{d-2}+\binom{d-1}{d-2}+\binom{d}{d-2}+\cdots+\binom{t-2}{d-2}.$$

It is equal to $\binom{t-1}{d-1}$ by Lemma 3.1. Hence, all level d vertices must be able to inform

$$\binom{d-1}{d-1}(n-d)+\binom{d}{d-1}(n-d-1)+\binom{d+1}{d-1}(n-d-2)+\cdots+\binom{n-2}{d-1}$$

level $d+1$ vertices by time n. But, by Lemma 3.2, this sum equals $\binom{n}{d+1}$ which means that all level $d+1$ vertices must be informed just by level d vertices.
Now, assume that without loss of generality, the originator vertex $0 = 00\cdots0$ informs vertex $u = 100\cdots0$ by using dimension 1 in the first time unit. Let $V_0 = \{0a_2\cdots a_n \mid a_i \in \{0,1\}, 2 \le i \le n\}$ and $V_1 = \{1a_2\cdots a_n \mid a_i \in \{0,1\}, 2 \le i \le n\}$. We claim that no vertex can use dimension 1 in other time units.
If $v \in V_1$ is a vertex such that $d(u,v) = d$ and uses dimension 1 in time unit $t > 1$ to inform $v' \in V_0$, then $d(0,v) = d+1$ and $d(0,v') = d$. This would mean a level $d+1$ vertex informs a level d vertex, as illustrated in Figure 3, a contradiction. If $v \in V_0$ uses dimension 1 in time unit $t > 1$ to inform a vertex in V_1, then since no vertex of V_1 can send back to V_0, implying that vertex v cannot contribute to informing the maximum number of vertices within V_0, and hence the broadcast within V_0 cannot be completed, a contradiction. Therefore, the claim is proved.

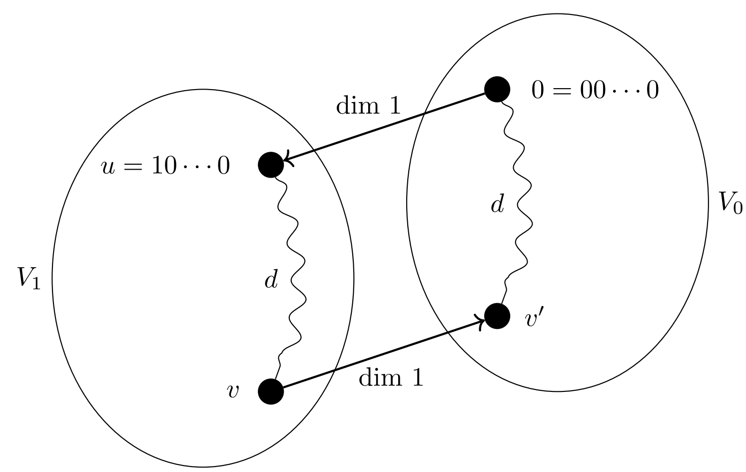

Figure 3: A vertex $v \in V_1$, with $d(u,v) = d$, uses dimension 1 in time unit $t > 1$ to inform $v' \in V_0$.

Since there is a one-to-one correspondence between valid minimum-time broadcast schemes and binomial trees, and the originator has n dimensions to use in the first time unit, one can see that $f(n) = nf^2(n-1)$, with $f(1) = 1$, where $f(n)$ is the number of all spanning binomial trees of H_n. By induction on n, we show that $f(n) = \prod_{i=0}^{n-1}(n-i)^{2^i}$. Assuming by IH that this holds for $n-1$, we have

$$f(n) = n\left(\prod_{i=0}^{n-2}(n-1-i)^{2^i}\right)^2 = n\prod_{i=0}^{n-2}(n-(i+1))^{2^{i+1}} = \prod_{j=0}^{n-1}(n-j)^{2^j}.$$

Therefore, the number of binomial trees of H_n is equal to the total number of broadcast schemes produced by Algorithm 1. □

As an immediate consequence of the previous proof, we have the following corollary.

Corollary 3.5. *The number of binomial subtrees rooted from vertex 0 as well as all valid minimum-time broadcast schemes of a hypercube H_n is*

$$n(n-1)^{2^1}(n-2)^{2^2}\cdots 3^{2^{n-3}}2^{2^{n-2}} = \prod_{i=0}^{n-1}(n-i)^{2^i}.$$

As highlighted in Corollary 3.5, the number of valid minimum-time broadcast schemes for a hypercube H_n is in $\Omega(2^{2^{n-2}})$. This exponential growth underscores the inherent difficulty of identifying all valid minimum-time broadcast schemes, even for a highly structured graph such as a hypercube. When considering general graphs, which often lack the regularity and symmetry of hypercubes, the complexity of the problem becomes even more pronounced.

4 Enumerating All Broadcast Schemes of Algorithm 1

Algorithm 1 not only provides a valid minimum-time broadcast scheme but also enables the generation of all possible minimum-time broadcasting schemes (discussed in Theorem 3.4). Since Line 7 of the algorithm selects an arbitrary number from the set of allowed vertices, we can systematically enumerate all broadcast schemes generated by this algorithm. Furthermore, the algorithm facilitates iteration through these schemes and allows unique identification by assigning a distinct number to each scheme. For instance, a hypercube of dimension 3 has 12 possible broadcasting schemes. In this section, we discuss how to determine the details of a scheme corresponding to a given broadcast number (ranging from 0 to number of possible schemes -1) and vice versa.

For an informed vertex u, let $H^{u,t}$ denote the largest sub-hypercube containing u as its only informed vertex after time unit t. When u uses dimension i to send the message to vertex v at time unit t, $H^{u,t-1}$ is divided into two smaller hypercubes, $H^{u,t}$ and $H^{v,t}$. Because dimension i is removed from the allowed sets of both u and v, the hypercubes $H^{u,t}$ and $H^{v,t}$ become distinct and separated. We call $H^{u,t}$ the first hypercube containing u (the sender) and $H^{v,t}$ the second hypercube containing v (the receiver).

Let H_n be a hypercube for $n \geq 3$. Suppose that $0 \leq s \leq f(n) - 1$ is an arbitrary number, where $f(n) = \prod_{i=0}^{n-1}(n-i)^{2^i}$ is the number of valid minimum-time broadcast schemes of H_n by Corollary 3.5. We show that s uniquely corresponds to a specific broadcast scheme.

Define the cluster size $c(n) = f(n)/n$, which represents the total number of minimum-time broadcast schemes after the first time unit, where the originator has sent the message to one of its neighbors. For simplicity, we denote it by c. Now, consider the representation of s in base c, $s = (a_1, a_2)_c$, which contains at most two numbers denoted by a_1 and a_2 because $a_1 = \lfloor \frac{s}{c} \rfloor < n \leq c$ for $n \geq 3$. Obviously, $a_1 + 1$ equals the first dimension that the originator u uses in the first time unit to inform v. We show that remainder a_2 represents two broadcast schemes of the two smaller hypercubes H_{n-1}. It is worth mentioning that $c' = \sqrt{c}$ equals the number of minimum-time broadcast schemes in H_{n-1}. Similarly, the representation of the remainder a_2 in base c' has at most two numbers, denoted $a_2 = (b_1, b_2)_{c'}$. Now, the $(b_1 + 1)$-th scheme of the first hypercube and the $(b_2 + 1)$-th scheme of the second hypercube, together with the edge uv, form a binomial tree, which constitutes a broadcast scheme for H_n.

To prove its correctness, note that if s and s' are two different numbers, then their representations in the same base (c or c') are different. Therefore, the resulting broadcast schemes are distinct.

For the inverse, suppose that B is a minimum-time broadcast scheme of H_n. Assume the originator u uses dimension i to inform vertex v in the first time unit, and the broadcast scheme in $H^{u,1}$ is the b_1-th broadcast scheme, while the broadcast scheme in $H^{v,1}$ is the b_2-th broadcast scheme. Then $s = (i-1, a_2)_c$, where $a_2 = (b_1 - 1, b_2 - 1)_{c'}$, is the enumeration of B in Algorithm 1. Again, it is easy to verify that two different schemes yield two different numbers.

For instance, in Figure 4, let vertex A be the originator, and define the dimensions as follows: $A \to B$ corresponds to dimension 1, $A \to D$ to dimension 2, and $A \to E$ to dimension 3. After sending along $A \to B$ in dimension 1, this dimension is removed from the set of allowed dimensions, dividing the hypercube H_3 into two separate smaller hypercubes H_2. As discussed earlier, the first hypercube is designated as the one containing the sender A. Thus, $ADHE$ forms the first hypercube, and $BCGF$ forms the second. The broadcasting scheme used within these smaller hypercubes determines the final number corresponding to the complete broadcasting scheme.

There are two possible schemes for H_2. Using the same logic, if sending the message occurs along $A \to D$, corresponding to the smaller-numbered dimension, this is designated as $scheme_1$. Similarly, sending the message along $A \to E$ is assigned as $scheme_2$.

Now, consider the scenario where the first hypercube follows $scheme_2$ ($A \to E$), and the second hypercube follows $scheme_1$ ($B \to C$). Combining these two schemes yields the binary number (10). The binary (10) corresponds to decimal 2, which represents this particular broadcasting scheme for the smaller hypercubes. Finally, we incorporate the initial step, $A \to B$, which occurred along the first dimension. In base 4, this corresponds to (02), resulting in the final scheme number being 2.

Thus, the following scheme corresponds to

$$\begin{aligned} &t_1 : A \to B, \\ &t_2 : A \to E,\ B \to C, \\ &t_3 : A \to D,\ B \to F,\ E \to H,\ C \to G. \end{aligned} \tag{4.1}$$

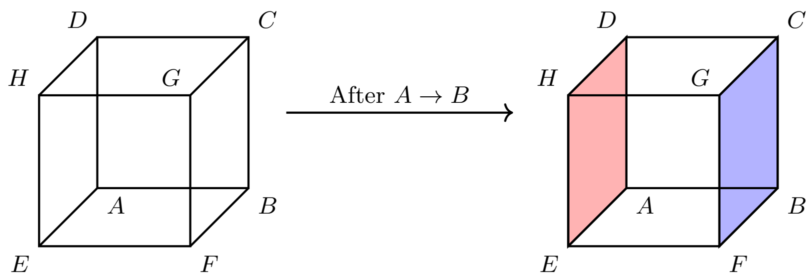

Figure 4: Transformation of the hypercube to two smaller hypercubes after the first transmission.

Conversely, if we are given scheme number 2 and need to determine the corresponding broadcast scheme, we proceed as follows:

First, we express the number into base 4, which is equal to $(02)_4$. The first digit (0) indicates the action selected by the originator, meaning the originator chose its first dimension, $A \to B$.

Next, the numbers corresponding to the two hypercubes are represented as $(10)_2$. This indicates that the first hypercube has chosen its second broadcast scheme, $A \to E$, because we defined the order of the dimensions earlier. The second hypercube has selected its first broadcast scheme, $B \to C$. Finally, the remaining scheme leads us directly to Relations (4.1).

5 Conclusion

In this paper, we studied the minimum-time broadcast schemes in hypercube graphs. We introduced an algorithm that generates all valid optimal broadcast schemes. Our analysis revealed the enumeration and compact encoding/decoding of every possible scheme for minimum-time broadcasting in hypercubes. While constructing a naïve minimum-time broadcast scheme is straightforward, our main contribution lies in the complete characterization of all such schemes and the demonstration that no other variants exist beyond those generated by our algorithm.

References

[1] A. Aminian, S. Kamali, S. M. Seyed-Javadi, and Sumedha. On the complexity of telephone broadcasting: From cacti to bounded pathwidth graphs. *International Colloquium on Automata, Languages, and Programming (ICALP)*, 2025.

[2] A. Averbuch, R. Shabtai, and Y. Roditty. Efficient construction of broadcast graphs. *Discrete Applied Mathematics*, 171:9–14, 2014. doi:10.1016/j.dam.2014.01.025.

[3] A. Bar-Noy, S. Guha, J. Naor, and B. Schieber. Multicasting in heterogeneous networks. In *Proceedings of the 30th annual ACM symposium on Theory of computing (STOC)*, pages 448–453, 1998. doi:10.1145/276698.276857.

[4] J. C. Bermond, P. Fraigniaud, and J. G. Peters. Antepenultimate broadcasting. *Networks*, 26(3):125–137, 1995. doi:10.1002/net.3230260302.

[5] J. C. Bermond, P. Hell, A. L. Liestman, and J. G. Peters. Sparse broadcast graphs. *Discrete Applied Mathematics*, 36(2):97–130, 1992. doi:10.1016/0166-218X(92)90226-Z.

[6] K. Y. Chwa and J. H. Park. Recursive circulant: A new topology for multicomputer networks. In *Proceedings of ISPA Networks (ISPAN '94)*, pages 73–80, 1994. doi:10.1109/ISPAN.1994.367162.

[7] M. J. Dinneen, M. R. Fellows, and V. Faber. Algebraic constructions of efficient broadcast networks. In *International Symposium on Applied Algebra, Algebraic Algorithms, and Error-Correcting Codes (AAECC '91)*, page 152–158, 1991. doi:10.1007/3-540-54522-0_104.

[8] 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. doi:10.1016/S0166-218X(99)00043-8.

[9] M. Elkin and G. Kortsarz. Combinatorial logarithmic approximation algorithm for directed telephone broadcast problem. In *Proceedings of the 30th ACM Symposium on Theory of Computing (STOC)*, pages 438–447, 2002. doi:10.1145/509907.509972.

[10] M. Elkin and G. Kortsarz. Sublogarithmic approximation for telephone multicast. *Journal of Computer and System Sciences*, 72(4):648–659, 2006. doi:10.1016/j.jcss.2005.12.002.

[11] A. M. Farley, S. T. Hedetniemi, S. M. Mitchell, and A. Proskurowski. Minimum broadcast graphs. *Discrete Mathematics*, 25(2):189–193, 1979. doi:10.1016/0012-365X(79)90022-0.

[12] P. Fraigniaud and E. Lazard. Methods and problems of communication in usual networks. *Discrete Applied Mathematics*, 53(1–3):79–133, 1994. doi:10.1016/0166-218X(94)90180-5.

[13] L. Gargano and U. Vaccaro. On the construction of minimal broadcast networks. *Networks*, 19(6):673–689, 1989. doi:10.1002/net.3230190606.

[14] H. A. Harutyunyan and Z. Li. A new construction of broadcast graphs. *Discrete Applied Mathematics*, 280:144–155, 2020. doi:10.1016/j.dam.2018.09.015.

[15] H. A. Harutyunyan and Z. Li. Broadcast graphs using new dimensional broadcast schemes for Knödel graphs. *Discrete Applied Mathematics*, 336:56–67, 2023. doi:10.1016/j.dam.2023.03.002.

[16] H. A. Harutyunyan and A. L. Liestman. More broadcast graphs. *Discrete Applied Mathematics*, 98(1–2):81–102, 1999. doi:10.1016/S0166-218X(99)00108-0.

[17] H. A. Harutyunyan and A. L. Liestman. Upper bounds on the broadcast function using minimum dominating sets. *Discrete Mathematics*, 312(20):2992–2996, 2012. doi:10.1016/j.disc.2012.06.016.

[18] H. A. Harutyunyan, A. L. Liestman, J. G. Peters, and D. Richards. Broadcasting and gossiping. In *Handbook of Graph Theory*, pages 1477–1494. 2013.

[19] 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. doi:10.1002/net.3230180406.

[20] M. C. Heydemann, N. Marlin, and S. Perennes. Rotational Cayley graphs on transposition generated groups. *Electronic Notes in Discrete Mathematics*, 5:177–180, 2000. doi:10.1016/S1571-0653(05)80156-8.

[21] J. Hromkovic, R. Klasing, B. Monien, and R. Peine. Dissemination of information in interconnection networks (broadcasting and gossiping). In *Combinatorial Network Theory*, volume 1, pages 125–212. 1996. doi:10.1007/978-1-4757-2491-2_5.

[22] 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.

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

[24] W. Knödel. New gossips and telephones. *Discrete Mathematics*, 13:95, 1975.

[25] M. Middendorf. Minimum broadcast time is np-complete for 3-regular planar graphs and deadline 2. *Information Processing Letters*, 46:281–287, 1993. doi:10.1016/0020-0190(93)90066-I.

[26] R. Ravi. Rapid rumor ramification: Approximating the minimum broadcast time. In *Proceedings of the 35th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 202–213, 1994. doi:10.1109/SFCS.1994.365693.

[27] J. F. Saclè. Lower bounds for the size in four families of minimum broadcast graphs. *Discrete Mathematics*, 150(1–3):359–369, 1996. doi:10.1016/0012-365X(95)00200-G.

[28] P. J. Slater, E. J. Cockayne, and S. T. Hedetniemi. Information dissemination in trees. *SIAM Journal on Computing*, 10(4):692–701, 1981. doi:10.1137/0210052.

[29] P. Tale. Double exponential lower bound for telephone broadcast. *arXiv preprint arXiv:2403.03501*, 2024. doi:10.48550/arXiv.2403.03501.