subproof2**Proof.** TOPROVE 1 ■

On the Complexity of Telephone Broadcasting
From Cacti to Bounded Pathwidth Graphs

Aida Aminian, Shahin Kamali, Seyed-Mohammad Seyed-Javadi, and Sumedha

York University, Toronto, Canada
{aminian, kamalis, smjavadi}@yorku.ca, sumedhab@my.yorku.ca

Abstract

In Telephone Broadcasting, the goal is to disseminate a message from a given source vertex of an input graph to all other vertices in the minimum number of rounds, where at each round, an informed vertex can send the message to at most one of its uninformed neighbors. For general graphs of n vertices, the problem is NP-complete, and the best existing algorithm has an approximation factor of $\mathcal{O}(\log n/\log\log n)$. The existence of a constant factor approximation for the general graphs is still unknown.

In this paper, we study the problem in two simple families of sparse graphs, namely, cacti and graphs of bounded pathwidth. There have been several efforts to understand the complexity of the problem in cactus graphs, mostly establishing the presence of polynomial-time solutions for restricted families of cactus graphs (e.g., [3, 5, 12, 13, 16, 17]). Despite these efforts, the complexity of the problem in arbitrary cactus graphs remained open. We settle this question by establishing the NP-completeness of telephone broadcasting in cactus graphs. For that, we show the problem is NP-complete in a simple subfamily of cactus graphs, which we call snowflake graphs. These graphs not only are cacti but also have pathwidth 2. These results establish that, despite being polynomial-time solvable in trees, the problem becomes NP-complete in very simple extensions of trees.

On the positive side, we present constant-factor approximation algorithms for the studied families of graphs, namely, an algorithm with an approximation factor of 2 for cactus graphs and an approximation factor of $\mathcal{O}(1)$ for graphs of bounded pathwidth.

1 Introduction

The Telephone Broadcasting problem involves disseminating a message from a single given source vertex to all other vertices in a network through a series of *telephone calls*. The network is often modeled as an undirected and unweighted graph of n vertices. Communication takes place in synchronous rounds. Initially, only the source is informed. During each round, any informed vertex can transmit the message to at most one of its uninformed neighbors via a “call”. The goal is to minimize the number of rounds required to inform the entire network. Hedetniemi et al. [18] identifies Telephone Broadcasting as a fundamental primitive in distributed computing and communication theory, forming the basis for many advanced tasks in these fields.

Slater et al. [24] established the NP-completeness of Telephone Broadcasting. Nonetheless, efficient algorithms have been developed for specific classes of graphs. In particular, Fraigniaud and Mitjana [8] demonstrated that the problem is solvable in polynomial time for trees. There are polynomial algorithms for several other graph families; see, e.g., [4, 10, 20, 25].

In this paper, we study cactus graphs, which are graphs in which any two cycles share at most one vertex. Cacti are a natural generalization of trees and ring graphs, providing a flexible model for applications such as wireless sensor networks, particularly when tree structures are too restrictive [1].

There have been several studies for broadcasting in specific families of cactus graphs [3, 5, 12, 13, 16, 17]. For instance, for unicyclic graphs, a simple subset of cactus graphs containing exactly one cycle, the problem is solvable in linear time [12]. Similarly, [17] proved that the chain of rings, which consists of cycles connected sequentially by a single vertex, has an optimal algorithm that runs in linear time. In k-restricted cactus graphs, where each vertex belongs to at most k cycles, for a fixed constant k, Čevnik and Žerovnik [3] proposed algorithms that compute the optimal broadcast scheme in $\mathcal{O}(n)$ time. Despite all these efforts, the complexity of TELEPHONE BROADCASTING for arbitrary cactus graphs has remained open, a question that we resolve in this paper.

TELEPHONE BROADCASTING is NP-hard for general graphs [9], and even for restricted graphs. In particular, Tale recently showed that the problem remains NP-hard for graphs of pathwidth of 3 [26], a result that naturally extends to graphs with higher pathwidths. However, the complexity of the problem for graphs of pathwidth 2 has remained an open question, which we address in this paper.

Elkin and Kortsarz [6] proved that approximating TELEPHONE BROADCASTING within a factor of $3-\epsilon$ for any $\epsilon > 0$ is NP-hard. Kortsarz and Peleg [19] showed that TELEPHONE BROADCASTING in general graphs has an approximation ratio of $\mathcal{O}(\log n/\log\log n)$. It is possible that there is a constant factor approximation for general graphs. However, constant-factor approximation exists only for restricted graph classes such as unit disk graphs [23] and certain sub-families of cactus graphs such as k-cycle graphs [2].

1.1 Contribution

This paper investigates the complexity and approximation algorithms for the TELEPHONE BROADCASTING problem for cacti and graphs of bounded pathwidth. The key contributions are summarized as follows:

- We present a simple polynomial-time algorithm, named CACTUS BROADCASTER, and prove it has an approximation factor of 2 for broadcasting in cactus graphs (Theorem 3.2). Constant-factor approximations are known for certain subfamilies of cactus graphs (e.g., k-cycle graphs [2]), and our result extends this to arbitrary cactus graphs. CACTUS BROADCASTER is reminiscent of the tree algorithm of [8] and leverages the separatability of cactus graphs. In our analysis of CACTUS BROADCASTER, we use the *k-broadcasting* model of broadcasting [11] as a reference point, where each vertex can inform up to two neighbors in a single round.

- Our main contribution is to establish the NP-completeness of TELEPHONE BROADCASTING problem in "snowflake graphs", which are a subclass of cactus graphs and also have pathwidth at most 2, therefore resolving the complexity of the problem in these graph families (Theorem 4.15). For a formal definition of snowflake graphs, refer to Definition 2.3. This hardness result is achieved through a series of reductions that start from $3,4$-SAT, a variant of the satisfiability problem that is NP-complete [27].

- We show the existence of a constant-factor approximation for graphs of bounded pathwidth. For that, we show the algorithm of Elkin and Kortsarz [7] has an approximation factor of $\mathcal{O}(4^w)$ for any graph of constant pathwidth w (Theorem 5.2). Note that this result does not rely on having a path decomposition certifying a bounded pathwidth. Constant-factor approximation algorithms are known for certain families of graphs of bounded path-width such as k-path graphs, which admit a 2-approximation algorithm [14], and our result extends this to any graph of bounded pathwidth.

1.2 Paper Structure

In Section 2, we present the preliminaries. In Section 3, we propose CACTUS BROADCASTER, which gives a 2-approximation for cactus graphs. In Section 4, we establish the NP-completeness of the problem in snowflake graphs. In Section 5, we present a constant-factor approximation algorithm for graphs of bounded pathwidth. We conclude in Section 6.

2 Preliminaries

For a positive integer n, we use notation $[n]$ to denote $\{1, 2, \dots, n\}$. Also, we use $[i, j]$ to refer to denote $\{i, i+1, \dots, j\}$. For a graph G, we use $G \setminus \{v\}$ to refer to the subgraph of G induced by all vertices of G except v.

Definition 2.1 ([18]). *An instance (G, s) of the* TELEPHONE BROADCASTING *problem is defined by a connected, undirected, and unweighted graph $G = (V, E)$ and a vertex $s \in V$, where s is the only informed vertex. The broadcasting protocol is synchronous and occurs in discrete rounds. In each round, an informed vertex can inform at most one of its uninformed neighbors. The goal is to broadcast the message as quickly as possible so that all vertices in V get informed in the minimum number of rounds.*

A broadcast scheme describes the ordering at which each vertex informs its neighbors. One can describe a broadcast scheme S with a *broadcast tree*, which is a spanning tree of G rooted at source s; if a vertex u is informed through vertex v, then u will be a child of v in the broadcast tree. Given that the optimal broadcast scheme of trees can be computed in linear time, a broadcast tree can fully describe the broadcast scheme. We use $\boldsymbol{br}^*(G, s)$ to refer to the number of rounds in the optimal broadcast scheme.

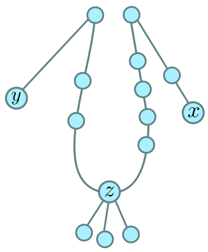
Definition 2.2. Cactus graphs *are connected graphs in which any two simple cycles have at most one vertex in common.*

Definition 2.3. *A tree T is said to be a* reduced caterpillar *if there are three special nodes x, y, and z in T such that every node in T is either located on the path between x and y or is connected to z.*

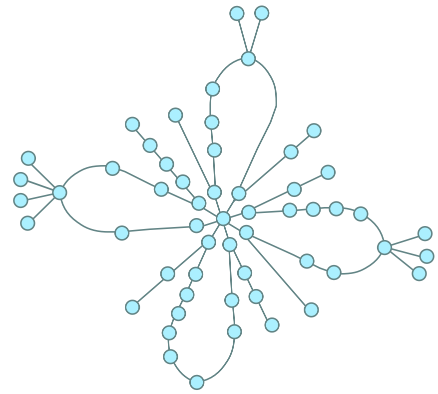
A graph G is said to be a snowflake, if and only if it has a center vertex c such that $G \setminus \{c\}$ is a set of disjoint reduced caterpillars such that c is connected to exactly two vertices in any of these caterpillars, none being special vertices.

An example of snowflake graphs and one of its corresponding reduced caterpillar components is shown in Figure 1. Informally, a snowflake graph is formed by a set of cycles that have a common center c; moreover, each cycle has a special vertex (z vertices). Any vertex in G is either i) a part of one of the cycles or ii) a part of a "dangling path" connected to a neighbor of c or iii) finds a special vertex as its sole neighbor.

Definition 2.4 ([22]). *A path decomposition D of a given graph $G = (V, E)$ is a sequence $\langle B_1, B_2, \dots, B_k \rangle$, where each B_i is called a bag and contains a subset of V, such that every vertex $v \in V$ appears in at least one bag and, for every edge $(u, v) \in E$, there exists a bag B_i containing*



(a) An example of a reduced caterpillar, with the special vertices x, y, and z highlighted



(b) An example of a snowflake graph

Figure 1: An illustration of reduced caterpillar and snowflake graphs

both u and v. Furthermore, if a vertex v appears in B_i and in B_j, it must appear in any B_k where $k \in [i, j]$.

The width of the path decomposition D is the maximum cardinality of its bags minus 1. Now, G is said to have pathwidth w if it has a path decomposition of width at most w.

Observation 2.1. *Snowflake graphs have a pathwidth of at most 2.*

Proof. TOPROVE 2 □

3 Cactus Broadcaster: A 2-Approximation for Cactus Graphs

In this section, we present our 2-approximation algorithm for cactus graphs. We use ideas from *k-broadcasting* model with parameter k [11], where a vertex can inform up to k of its neighbors in a single round via a *super call*. It is easy to see that if one can complete k-broadcasting in m rounds, then it is possible to complete broadcasting (in the classic setting) within km rounds. This can be achieved by "simulating" a super-call with up to k regular calls (see Lemma 3.1). For our algorithm, we use *k-broadcasting* with $k = 2$. In particular, we design an algorithm for 2-*broadcasting* in a cactus graph G and show that if it completes within m rounds, then any broadcast scheme for classic broadcasting in G takes at least m rounds (see Lemma 3.4). Therefore, if we simulate every super call with two regular calls (in arbitrary order) using Lemma 3.1, the broadcasting completes within $2m$ rounds, and thus, we achieve an approximation factor of 2.

3.1 *k-broadcasting* Model

In the *k-broadcasting* model, an informed vertex can simultaneously inform up to k neighbors in a single round, a process we refer to as a **super call**. Creating networks that allow fast broadcasting under this model has been studied in previous work [15]. We will present a method to convert a broadcasting schema in the *k-broadcasting* model into the classic model (without super calls). The final number of rounds in the classic model will be at most k times the number of rounds in the *k-broadcasting* model. This model applies not only to cactus graphs but also to every arbitrary graph.

Lemma 3.1. *Let S_k be a broadcast schema for graph G in the k-broadcasting model. It is possible to convert S_k to a broadcast scheme S in linear time for graph G in the classic model such that*

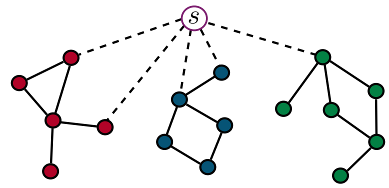


Figure 2: Two possible types for broadcasting to connected components after deleting s in `single-br`(G, s) are shown. The green vertices form a single-neighbor component. The red and blue components are double-neighbor components.

broadcasting in S completes within k times the number of rounds as broadcasting in S_k, i.e., $\boldsymbol{br}(S) \leq k \cdot \boldsymbol{br}(S_k)$.

Proof. TOPROVE 3 $\square$

3.2 Cactus Broadcaster Algorithm

This section presents Cactus Broadcaster, a 2-*broadcasting* algorithm that yields our 2-approximation algorithm for the Telephone Broadcasting problem in cactus graphs. Cactus Broadcaster is a mutually recursive algorithm with two main methods, namely, `single-br`(G, s) and `double-br`(G, s_1, s_2). The `single-br`(G, s) method is used for 2-*broadcasting* a connected subgraph G starting from a source vertex s. The `double-br`(G, s_1, s_2) method is used for 2-*broadcasting* a connected subgraph G using two sources s_1 and s_2 under an assumption that both s_1 and s_2 have the message at time 0 and there is a unique path between them in G.

Broadcasting from a single source. We explain how `single-br`(G, s) operates. Observe that removing the source vertex s partitions G into one or more disjoint connected components. By the definition of cactus graphs, each of these connected components contains at most two neighbors of s (see Observation 3.1). We refer to a connected component with one neighbor (respectively two neighbors) of s as a *single-neighbor* (respectively *double-neighbor*) component (see Figure 2).

Observation 3.1. *Let H be a connected induced subgraph of a cactus graph G. Any vertex $v \in G$ that is not in H finds at most 2 neighbors in H.*

Proof. TOPROVE 4 $\square$

`single-br`(G, s) computes the broadcast time of each single-neighbor component C recursively by computing `single-br`(C, u), where u is the single neighbor of s in C. Similarly, it computes the broadcast time of each double-neighbor component C by computing `double-br`(C, u, v), where u, v are neighbors of s in C. After calculating all these broadcasting times, `single-br`(G, s) sorts these values in non-increasing order and informs the neighbors of s in this order. For that, it uses a regular call for single-neighbor components and a super-call for double-neighbor components. In other words, `single-br`(G, s), informs the component with the largest broadcasting time in the first round, the component with the second largest broadcast time next, and so on. This is reminiscent of broadcasting in trees [8], except that super-calls are used for double components. Let b_i denote the broadcast time of the i'th component C_i in this order.

The total broadcasting time for `single-br(G,s)` is then computed as $\max_i(i + b_i)$, which will be the output of `single-br(G,s)`.

Broadcasting from two sources. Next, we describe `double-br`(G, s_1, s_2). Let P be the unique path between s_1 and s_2. In any broadcast tree T of `double-br`, some vertices on P are informed through s_1 and some through s_2. Thus, exactly one edge $e \in P$ is excluded from the tree. After removing such e, the graph G will be partitioned into two disjoint connected components G_1^e and G_2^e, which are respectively informed through s_1 and s_2 (See Figure 3). The method `double-br` works by removing the edge e^* with a minimum value of $\max\{\texttt{single-br}(G_1^e, s_1), \texttt{single-br}(G_2^e, s_2)\}$ over all $e \in P$.

Next, we explain how `double-br` finds e^*. An exhaustive approach that tries all edges $e \in P$ and calls `single-br` twice per edge may take exponential time.

To fix this, `double-br` works in two phases:

- In Phase 1, the method pre-computes the broadcast time for the connected subgraphs of G after removing vertices of P from G. By Observation 3.1, every resulting connected component after removing P has at most two neighbors in P. For any single-neighbor component C with vertex s connected to P, `single-br`(C, s) is computed recursively. Similarly, for any double-neighbor component with vertices s_1, s_2 connected to P, `double-br`(C, s_1, s_2) is computed recursively.

- Next, we describe Phase 2. At each iteration of Phase 2, we fix an edge $e = (u, v) \in P$ and aim to compute `single-br`(G_1^e, s_1) and `single-br`(G_2^e, s_2), where $u \in G_1^e$ and $v \in G_2^e$. We explain how to efficiently compute `single-br`(G_1^e, s_1). Finding `single-br`(G_2^e, s_2) is done similarly. Let $\langle u(= u_1), u_2, \ldots, u_k(= s_1)\rangle$ be the sequence of vertices in the unique path from u to s_1. Let H_i^e be the connected component of G that contains u_i after removing the edge (u_i, u_{i+1}) from G_1^e. Note that $H_k^e = G_1^e$. We compute $b_i =$`single-br`(H_i^e, u_i) using the values computed in Phase 1 as follows in an iterative manner from $i = 1$ to $i = k$.

  Consider all connected components of H_i^e after removing u_i. Note that, for $i > 1$, H_{i-1}^e is one of these components. Consider the set B of broadcast times for all these components, with the neighbor(s) of u_i as the source(s) of broadcast. All these values are computed in Phase 1 except for b_{i-1}, which is computed in the previous iteration. `double-br` sorts B in non-increasing order of broadcast times and informs neighbors of u_i accordingly (similar to `single-br`). As a result, in the i'th iteration, the broadcast time b_i is computed as $\max_{j\in[B]}(j + B[j])$. At iteration $i = k$, we compute b_k, which is indeed `single-br`(G_1^e, s_1) (see Observation 3.2).

Figure 3 provides an illustration.

Theorem 3.2. Cactus Broadcaster *runs a polynomial-time for* Telephone Broadcasting *and achieves an approximation factor of* 2 *on cactus graphs.*

Proof Overview. We provide an overview of the proof before a formal argument. First, we show that Cactus Broadcaster runs in $\mathcal{O}(n^3 \log n)$. This can be established using induction on the size of the input graph, based on the recursive nature of the algorithm. Details can be found in Lemma 3.3.

Second, we show that Cactus Broadcaster has an approximation factor of 2. Intuitively, `single-br` with super-calls runs no longer than an optimal broadcast scheme without super-calls; this is because every super call only expedites informing one of the two sources of broadcasting in double-neighbor components. As mentioned earlier (Lemma 3.1), a broadcast scheme with super calls can be simulated with a scheme (without super-calls)

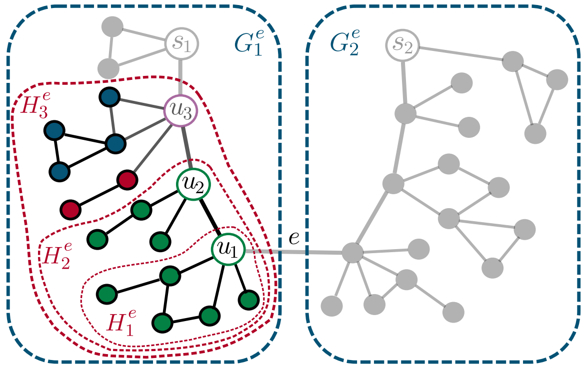


Figure 3: An illustration of Phase 2 of the `double-br` method. Different components in iteration $i = 3$ of Phase 2 (for u_3) are highlighted in different colors.

that completes no later than twice the number of rounds. On the other hand, the selection of edge e^* by `double-br` ensures that $\max(\texttt{single-br}(G_1^{e^*}, s_1), \texttt{single-br}(G_2^{e^*}, s_2)) \leq \max(\texttt{single-br}(G_1^{e^+}, s_1), \texttt{single-br}(G_2^{e^+}, s_2))$, where e^+ is the edge that is absent in the optimal broadcast tree of G. A formal proof follows from an inductive argument on the input size (see Lemma 3.4).

Observation 3.2. *Let $\langle u_1(= u), u_2, \ldots, u_k(= s_1)\rangle$ be the vertices on the path from u_1 (an endpoint of the removed edge e) and the source s_1. Then, for any $i \in [k]$, the broadcast time b_i for (H_i^e, u_i) in `double-br` equals to `single-br` applied to H_i, that is, $b_i =$ `single-br`(H_i^e, u_i).*

Proof. TOPROVE 5 ☐

Lemma 3.3. Cactus Broadcaster *terminates in $\mathcal{O}(n^3 \log n)$ time.*

Proof. TOPROVE 6 ☐

Lemma 3.4. *Consider an instance (G, s) of the broadcasting problem in a cactus graph G. Let $\textbf{br}(G, s)$ denote the number of rounds that it takes for* Cactus Broadcaster *to complete k-broadcasting in G with $k = 2$, and $\textbf{br}^*(G, s)$ denote the optimal number of rounds for broadcasting in the classic model. Then we have $\textbf{br}(G, s) \leq \textbf{br}^*(G, s)$.*

Proof. TOPROVE 7 ☐

From the above results, we can establish the proof of Theorem 3.2 as follows.

Proof. TOPROVE 8 ☐

The approximation factor given by Theorem 3.2 is tight. Consider a graph G formed by t triangles that all share an endpoint s. The broadcasting time of Cactus Broadcaster on (G, s) is $2t$ as its broadcast tree will be a star with $2t$ leaves, while the optimal broadcast tree completes broadcasting in $t + 1$ rounds (s will have degree t). Thus, the approximation factor tends to 2 for large t.

4 Hardness Proof of Snowflake Graphs

The NP-hardness proof of Telephone Broadcasting in snowflake graphs proceeds through successive reductions. First, we reduce 3, 4-SAT, which is known to be NP-hard [27] to a new problem that we call TWIN INTERVAL SELECTION (Lemma 4.5), which itself reduces to another new problem DOME SELECTION WITH PREFIX RESTRICTIONS (Lemma 4.12). Finally, we present a reduction from DOSEPR to TELEPHONE BROADGUESS, which establishes our main result (Theorem 4.15).

4.1 Hardness of Twin Interval Selection

The first step in reducing $3,4$-SAT to TELEPHONE BROADGUESS is to establish a reduction from $3,4$-SAT to TWIN INTERVAL SELECTION (TWIS). For that, we first explain how to construct a TWIS instance from a given $3,4$-SAT instance and demonstrate the equivalence of solutions between the two problems.

Definition 4.1 ([27]). *The* $3,4$-SAT *problem is a special type of the classic* 3-*satsifiabiity (*3-SAT*) problem. The input is a boolean CNF formula $\phi = (X, C)$, where X is the set of variables and C is the set of clauses such that every clause $C_i \in C$ contains* exactly *three literals $\ell^i_k \in C_i$ for $k \in [3]$, and each variable $x \in X$ appears in at most four clauses. The decision problem asks whether there is a satisfying assignment of ϕ.*

The $3,4$-SAT problem is known to be NP-complete [27]. In what follows, we call a pair of intervals $(I_i, \overline{I_i})$ for $i \in [n]$ a *twin interval*, assuming that I_i and $\overline{I_i}$ are non-crossing and have equal lengths with endpoints in $[m]$ for some positive m. The endpoints of I_i are less than the endpoints of $\overline{I_i}$. We refer to I_i and $\overline{I}_i$ the *left* and the *right* interval of the twin, respectively.

Definition 4.2. *An instance of the* TWIN INTERVAL SELECTION (TWIS) *is formed by a tuple (I, r, m), where I is a set of* twin *interval pairs and r is a* restriction function *with domain $[m]$, where m is referred to the* horizon *of the instance. We have $I = \{(I_1, \overline{I_1}), \dots, (I_n, \overline{I_n})\}$, where for each $i \in [m]$, $(I_i, \bar{I}_i)$ are non-crossing intervals with endpoints in $[m]$.*

The objective is to select exactly one of I_i and $\overline{I}_i$, while respecting the restriction imposed by the restriction function as follows. The restriction function $r : [m] \to [n]$ requires that for any $t \in [m]$, the number of selected intervals that contain t be at most $r(t)$. The decision problem asks whether there is a valid selection that satisfies these restrictions.

For a solution S and $t \in [m]$, let $\Gamma_S(t)$ be the number of selected intervals in S containing t.

Construction. Suppose we are given an instance $\phi = (X, C)$ of $3,4$-SAT. We construct an instance $\mathcal{I}_\phi$ of TWIS as follows. For each variable $x_i \in X$, form a pair of $(X_i, \overline{X_i})$ of twin intervals, each of length 7, where $X_i = [16i - 15, 16i - 8]$ and $\overline{X}_i = [16i - 7, 16i]$. We refer to X_i (respectively $\overline{X_i}$) as the *principal* interval of literal x_i (respectively $\neg x_i$). The length 7 of these intervals allows for placing up to 4 non-crossing unit intervals, each of length 1, within the principal interval; these intervals will be used for clause gadgets, as we will explain. For each clause C_j, we define three interval twin pairs $(I^j_k, \overline{I^j_k})$, each associated with one of the literals of $\ell^j_k \in C_j$ for each $k \in [3]$. Suppose ℓ^j_k is the p'th literal where its corresponding variable $x_i \in X$ appears ($p \in [4]$). If ℓ^j_k is a positive (respectively negative) literal of variable x_i, then define I^j_k as a unit interval starting at $16i - 7 + 2(p-1)$ (respectively, starting at $16i - 15 + 2(p-1)$). Intuitively, I^j_k is defined as one of the four-unit intervals within the principal interval of $\neg x$. Meanwhile, $\overline{I^j_k}$ is $[16n + 2j, 16n + 2j + 1]$ for all $k \in [3]$. Finally, to define the restriction function, we let $r(t) = 1$ for all $t \le 16n$ and $r(t) = 2$, otherwise. Figure 4 illustrates an example of the above reduction.

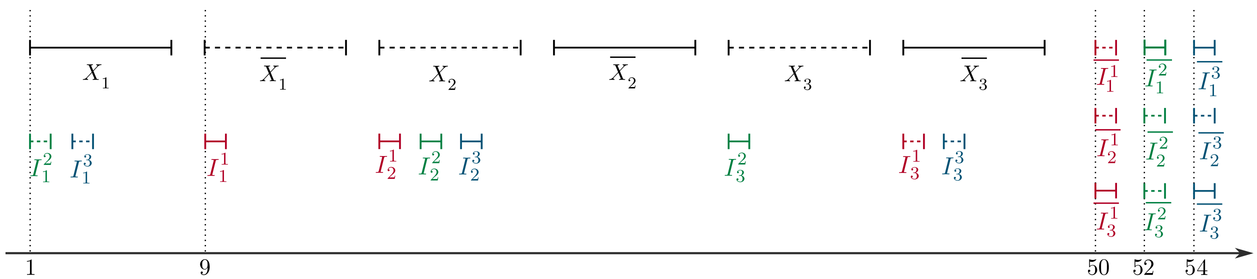


Figure 4: Construction of the instance $\mathcal{I}_\phi$ of the TwIS problem from the 3, 4-SAT instance $\phi = (x_1 \vee \neg x_2 \vee x_3) \wedge (\neg x_1 \vee \neg x_2 \vee \neg x_3) \wedge (\neg x_1 \vee \neg x_2 \vee x_3)$. The restriction function $r(t)$ is defined as $r(t) = 1$ for $t \leq 48$ and $r(t) = 2$ otherwise. A satisfying assignment for ϕ is $\sigma(x_1) = 1$ and $\sigma(x_2) = \sigma(x_3) = 0$; the selected intervals in the corresponding solution for $\mathcal{I}$ are shown in solid color.

Correctness. First, we provide an intuitive explanation of why this reduction works. We argue that a satisfying assignment to instance ϕ of the 3,4-SAT problem bijects to a satisfying interval selection for the instance $\mathcal{I}_\phi$ of TwIS. The bijection requires that for any literal ℓ that is true in a satisfying assignment of ϕ, the principal interval of ℓ is selected in the solution for $\mathcal{I}_\phi$.

The restrictions imposed by r imply that whenever a principal interval of a literal ℓ is selected, the unit intervals associated with (up to 4) occurrences of $\neg\ell$ cannot be selected (because $r(t) = 1$ in their intersecting points). Thus, whenever the principal interval of a literal ℓ is selected (i.e., its twin, which is the principal interval of $\neg\ell$ is not selected), one can select the unit intervals associated with any occurrence of ℓ in clauses where it appears; this is because they intersect with the principal interval of $\neg\ell$, which is not selected. Selecting these unit intervals means their twin unit intervals This yields an equivalent between a satisfying assignment in the SAT formula and the TwIS. We further note that the number of twin intervals in $\mathcal{I}_\phi$ is polynomial in $|X|$ and $|C|$. We can conclude the following lemma, which establishes the hardness of TwIS in the strong sense based on the above reduction.

To establish the main result of this section, we first establish the two sides of the reduction in the following lemmas.

Lemma 4.3. *If the answer to the instance $\mathcal{I}_\phi$ of* TwIS *is yes, then the instance ϕ of* 3, 4-SAT *is satisfiable.*

Proof. TOPROVE 9 □

Lemma 4.4. *If an instance $\phi = (X, C)$ of the* 3, 4-SAT *problem is satisfiable, then the answer to the instance $\mathcal{I}_\phi$ of* TwIS *is yes.*

Proof. TOPROVE 10 □

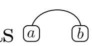
Lemma 4.5. *Answering instances (I, r, m) of the* TwIS *problem is NP-hard even if the m is polynomial in $|I|$.*

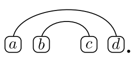
Proof. TOPROVE 11 □

4.2 Hardness of Dome Selection with Prefix Restrictions

We present a polynomial-time reduction from TwIS to a new problem that we refer to as Dome Selection with Prefix Restrictions (DoSePr). Before defining the DoSePr problem, we first define the notion of a *dome* as follows.

Definition 4.6. *A* dome *is formed by four positive integers (a, b, c, d) such that $a \leq b < c \leq d$ and $b - a = d - c$. We refer to (a, d) (respectively (b, c)) as an* arc *with endpoints a and d (respectively b and c). We refer to (a, d) and (b, c) the* outer arc *and* inner arc *of the dome, respectively. When $a = b$ and $c = d$, we call the dome a* singleton dome *and otherwise call it a* regular dome*.*

In our figures, a singleton dome (a, b) is shown as 

and regular dome (a, b, c, d) is shown as 

.

Definition 4.7. *An instance (D, m) of the* Dome Selection with Prefix Restrictions *(*DoSePr*) is defined by a multiset $D = \{D_1, \ldots, D_n\}$ of domes and a positive integer m, where $D_i = (a_i, b_i, c_i, d_i)$ s.t. $d_i \leq m$ for some integer m that we refer to as* horizon*. The decision problem asks whether there is a multiset S of arcs with exactly one arc from each dome D_i, such that, for any $t \in [m]$, it holds that $\mathcal{N}_S(t) \leq t$, where $\mathcal{N}_S(t)$ denotes the number of arc endpoints in S with value at most t, counting each endpoint as many times as it appears across different arcs in S.*

Example 1. *Consider an instance of the* DoSePr *with domes $D = \{(2, 3, 4, 5), (3, 4), (4, 5, 9, 10), (7, 8)\}$ and $m = 10$. A valid solution for this instance is $S = \{(2, 5), (3, 4), (5, 9), (7, 8)\}$. For example, when $t = 5$, we have $\mathcal{N}_S(t) = 5$, which is no more than t. In particular, selected endpoints $\leq t$ are $\{2, 5, 3, 4, 5\}$. Note that there are two endpoints with a value of 5, which belong to two different arcs.*

Next, we explain how the TwIS problem reduces to the DoSePr problem.

Construction. Given an arbitrary instance $\mathcal{I} = (I, r, m')$ of TwIS, where $|I| = n'$ for some positive n', we construct the corresponding instance $\mathcal{D} = (D, m)$ of DoSePr as follows. For each twin pair of intervals $(I_i, \overline{I_i}) \in I$, where $I_i = (a', b')$ and $\overline{I_i} = (c', d')$, we construct a regular dome $D_i = (a_i, b_i, c_i, d_i)$ where $a_i = 6n'a', b_i = 6n'b' + 3n', c_i = 6n'c'$, and $d_i = 6n'd' + 3n'$. It is easy to verify that D_i is indeed a dome, that is $b_i - a_i = d_i - c_i$ (see Observation 4.1).

We define m to be a large enough horizon. In particular, we let $m = 164(n'm')^2$. For any $t \in [m]$ that is a multiple of $6n'$ and $t \leq 3n'(2m' + 1)$, we further add $d(t)$ some singleton domes that all start at t and end at m. In other words, we add $d(t)$ identical singleton domes (t, m). Next, we explain how $d(t)$ is defined. For convenience, we let $d(t) = 0$ if t is not a multiple of $6n'$ or $t > 3n'(2m' + 1)$. Suppose t is indeed a multiple of $6n'$. The idea is to define $d(t)$ in a way to project the requirements imposed by the restriction function r in $\mathcal{I}$ to the requirement $\mathcal{N}_S(t) \leq t$ in DoSePr.

Let $\mathfrak{D}_i^t$ be the set of regular domes with exactly i endpoints before or at point t, and define $c(t) = 2|\mathfrak{D}_4^t| + |\mathfrak{D}_3^t| + |\mathfrak{D}_2^t|$.

Example 2. *For $t = 18$, dome $(1, 4, 5, 8) \in \mathfrak{D}_4^t$, dome $(11, 14, 16, 19) \in \mathfrak{D}_3^t$, dome $(9, 10, 26, 27) \in \mathfrak{D}_2^t$, dome $(17, 20, 21, 24) \in \mathfrak{D}_1^t$, and dome $(25, 26, 27, 28) \in \mathfrak{D}_0^t$. Figure 5 provides an illustration.*

Intuitively, this definition of c implies that there are $c(t)$ arc endpoints with value at most t in any valid solution S (a set of arcs), regardless of the choices made to form S. This is because:

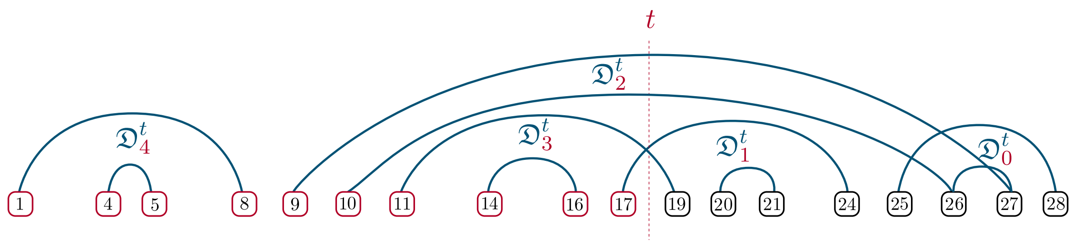


Figure 5: An example of different types of domes based on their positions relative to $t = 18$

- (i) all arc endpoints of domes in $\mathfrak{D}_4^t$ are at most t, and two of them (associated with one arc) contribute to $c(t)$.
- (ii) three arc endpoints of domes in $\mathfrak{D}_3^t$ are at most t, and any such dome contributes at least 1 to $c(t)$.
- (iii) the left endpoints of both arcs of domes in $\mathfrak{D}_2^t$, and since one arc is in S, the dome contributes exactly 1 to $c(t)$.

Finally, we let

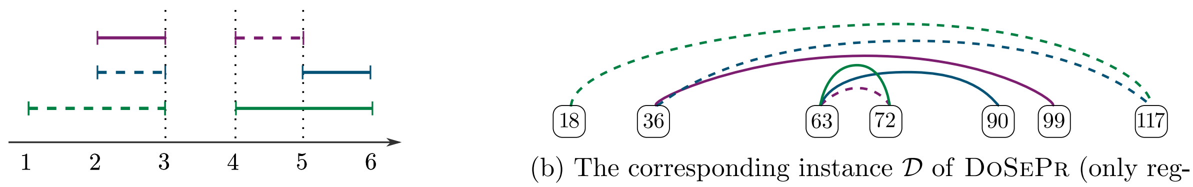
$$d(t) = t - c(t) - \sum_{j<t} d(j) - r(t/(6n')).$$

The scaling argument that is applied when forming the DoSePr instance from TwIS instance ensures that $d(t)$ is indeed non-negative for any $t \in [m]$ (see Observatio 4.2 for details). This completes our construction of the DoSePr instance. In a nutshell, we have added one regular dome per twin interval in the TwIS instance, and for any $t \in [m]$, we added extra identical singleton domes to capture the requirements imposed by the restriction function in the TwIS instance.

Example 3. *Consider an instance $\mathcal{I} = (I, r, m')$ of the* TwIS *problem with $I = \{((1,3)(4,6)), ((2,3),(5,6)), ((2,3),(4,5))\}$ and $m' = 6$. Suppose $r(1) = r(2) = r(6) = 3$, $r(3) = r(4) = 1$, and $r(5) = 2$. The corresponding instance $\mathcal{D} = (D, m)$ of* DoSePr *has regular domes $D = \{(18,63,72,117),(36,63,90,117),(36,63,72,99)\}$ and $m = 164(n'm')^2 = 164 \cdot (3 \cdot 6)^2 = 53,136$ (see Figure 6). In addition, $3n'(2m'+1) = 117$ for any $t \in [1,117]$ that is a multiple of $6n' = 18$, $d(t)$ singleton domes (t, m) are added. Some examples of $d(t)$ are shown as follows.*

$$\begin{aligned}
&d(18) = 18 - 0 - 0 - 3 = 15 && c(18) = 0\\
&d(36) = 36 - 0 - 15 - 3 = 18 && c(36) = 0\\
&d(54) = 54 - 0 - (18+15) - 1 = 20 && c(54) = 0\\
&d(72) = 72 - 3 - (15+18+20) - 1 = 15 && c(72) = 3
\end{aligned}$$

Correctness. First, we provide some intuitions about the correctness of the reduction. We show that any valid solution S for instance $\mathcal{D}$ of DoSePr bijects to a valid solution S' of $\mathcal{I}$. Note that S contains exactly one arc, the inner or the outer arc, of each regular dome D_i of $\mathcal{D}$ (in addition to singleton domes, which are contained in any valid solution for $\mathcal{D}$). Now, S' selects the left interval I_i when the outer arc of D_i is selected in S and the right interval $\overline{I_i}$ when the inner arc



(a) The instance $\mathcal{I}$ of TwIS

(b) The corresponding instance $\mathcal{D}$ of DoSePr (only regular domes are depicted.)

Figure 6: An illustration of the construction of an instance $\mathcal{D}$ of DoSePr from an instance $\mathcal{I}$ of TwIS, as explained in Example 3. Solid lines show the selected intervals in a valid solution of $\mathcal{I}$ and the selected arcs in the corresponding solution in $\mathcal{D}$.

of D_i is selected in S (see Figure 6). We let $\Gamma_{S'}(t')$ denote the number of intervals selected by S' that include $t' \in [m']$. We show that S is a valid solution for $\mathcal{D}$ if and only if S' is a valid solution to $\mathcal{I}$.

Suppose S is a valid solution for DoSePr. Then we must have $\mathcal{N}_S(t) \leq t$ for any $t \in [m]$, where $\mathcal{N}_S(t)$ is the number of arc endpoints in S with value at most t. As mentioned earlier, $c(t)$ endpoints will be present in any valid solution regardless of the choices to form S. Moreover, S must contain all endpoints of singleton domes in $\mathcal{D}$ for $j \leq t$. That is, any valid solution for $\mathcal{D}$, including S, must contain $c(t) + \sum_{j \leq t} d(j)$ arc endpoints. In addition to these fixed points, there will be more arcs in S, which are contributed by the domes in $\mathfrak{D}_1^t$ and $\mathfrak{D}_3^t$ as follows. Let D_i be a regular dome in $\mathcal{D}$, and let $(I_i, \overline{I}_i)$ be its corresponding twin intervals in $\mathcal{I}$.

- Suppose $D_i \in \mathfrak{D}_1^t$. Now, if S contains the inner arc of D_i, the contribution of D_i to $\mathcal{N}_S(t)$ would be 0. This is equivalent to including the right interval $\overline{I_i}$ in S', and thus the twin interval $(I_i, \overline{I_i})$ does not contribute $\Gamma_{S'}(t')$ for $t' = t/6n$. Informally, the contribution of the dome D_i to the left-hand side of inequality $N_S(t) \leq t$ and the contribution of $(I_i, \overline{I_i})$ to the left-hand side of the inequality $\Gamma_{S'}(t') \leq r(t')$ will be both 0. Similarly, if S contains the outer arc of D_i, the contribution of D_i to $\mathcal{N}_S(t)$ would be 1. This is equivalent to including the left interval $\overline{I_i}$ in the TwIS instance, and $(I_i, \overline{I_i})$ contribute 1 item to $\Gamma_{S'}(t')$. Informally, the contribution of the dome D_i to the left-hand side of inequality $\mathcal{N}_S(t) \leq t$ and the contribution of $(I_i, \overline{I_i})$ to the left-hand side of $\Gamma_{S'}(t/(6n')) \leq r(t')$ will be both 1.
- Suppose $D_i \in \mathfrak{D}_3^t$. Assume S contains the inner (respectively the outer) arc of D_i. In this case, in addition to the fixed 1 unit of the contribution of D_i to $\mathcal{N}_S(t)$ (captured in $c(t)$), it further contributes 1 (respectively 0) unit to $\mathcal{N}_S(t)$. This is equivalent to including the right interval $\overline{I}_i$ (respectively the left interval I_i) in S'. In this case, the number of selected intervals in S' that intersect t is increased by 1 (respectively 0). Intuitively, both left-hand sides of $\mathcal{N}_S(t) \leq t$ and $\Gamma_{S'}(t/(6n')) \leq r(t')$ increase by 1 (respectively 0).

We note that $n = |D|$ is polynomial in both n' and m', and thus $m = 164(n'm')^2$ is polynomial in n. Therefore, we can conclude the following lemma, which establishes the NP-hardness of DoSePr. We start the formal proof with the following observation showing that the regular domes in $\mathcal{D}$ are indeed valid.

Observation 4.1. *For any $i \in [n']$, any constructed $D_i = (a_i, b_i, c_i, d_i)$ in the reduction is a valid dome, i.e., $b_i - a_i = d_i - c_i$.*

Proof. TOPROVE 12 □

Next, we observe that the number $d(t)$ of extra singleton domes added for any $t \in [m]$ is non-negative.

Observation 4.2. *In the construction of $\mathcal{D}$ from $\mathcal{I}$, for any $t \in [m]$, we have $d(t) \geq 0$.*

Proof. TOPROVE 13 $\square$

For each dome $D_i \in D$, let the boolean variable x_i indicate whether the outer arc of D_i is selected in S or not. Specifically, $x_i = 1$ means the outer arc is selected, while $\neg x_i = 1$ means the inner arc is selected.

The following two lemma express the values of $\mathcal{N}_S(t)$ and $\Gamma_{S'}(t)$ in a way that allows us to relate a solution S for the DoSePr instance $\mathcal{D}$ to its associated solution S' in the TwIS instance $\mathcal{I}$. These lemmas will later help us in proving the correctness of the reduction.

Lemma 4.8. *Let $\mathcal{D} = (D, m)$ be the* DoSePr *problem constructed from an instance $\mathcal{I}$ of the* TwIS *problem, and let S be a multiset of arcs that contains exactly one arc from each dome $D_i \in D$. For any $t \in [m-1]$, the number of arc endpoints in S with value at most t is exactly $\mathcal{N}_S(t) = \sum_{i \in \mathfrak{D}_1^t} x_i + \sum_{i \in \mathfrak{D}_3^t} \neg x_i + |\mathfrak{D}_2^t| + |\mathfrak{D}_3^t| + 2|\mathfrak{D}_4^t| + \sum_{j \leq t} d(j)$.*

Proof. TOPROVE 14 $\square$

Lemma 4.9. *Let $\mathcal{D} = (D, m)$ be the* DoSePr *problem constructed from an instance $\mathcal{I} = (I, r, m')$ of the* TwIS *problem, and let S' be a selection of intervals from $\mathcal{I}$ that contains exactly one interval from each twin interval $(I_i, \overline{I_i}) \in I$. For any $t' \in [m-1]$, we have $\Gamma_{S'}(t') = \sum_{i \in \mathfrak{D}_3^t} \neg x_i + \sum_{i \in \mathfrak{D}_1^t} x_i$, where $t = 6n't'$.*

Proof. TOPROVE 15 $\square$

Now, we are ready to prove the correctness of the reduction.

Lemma 4.10. *If the answer to the instance $\mathcal{D} = (D, m)$ of* DoSePr *is yes, then the answer to its corresponding instance $\mathcal{I} = (I, r, m')$ of* TwIS *is also yes.*

Proof. TOPROVE 16 $\square$

Lemma 4.11. *If the answer to the instance $\mathcal{I} = (I, r, m')$ of* TwIS *is yes, then the answer to its corresponding instance $\mathcal{D} = (D, m)$ of* DoSePr *is also yes.*

Proof. TOPROVE 17 $\square$

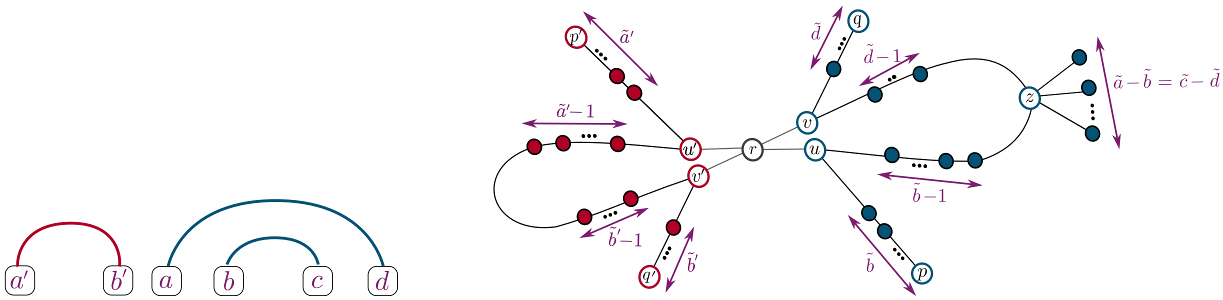
From the above lemmas, we can conclude the main result of this section.

Lemma 4.12. *Answering instances (D, m) of the* DoSePr *problem is NP-hard even if m is polynomial in $|D|$.*

Proof. TOPROVE 18 $\square$

4.3 Hardness of Telephone Broadguess in Snowflake Graphs

We refer to the decision variant of the broadcasting problem as the Telephone Broadguess problem with instances (G, s, ρ), where G is the input graph, s is the source, and ρ is a positive integer. The decision question asks whether it is possible to complete broadcasting in G from s within ρ rounds. This section presents our final reduction, from DoSePr to Telephone Broadguess in snowflake graphs.



(a) An instance $\mathcal{D}$ of the DoSePr instance

(b) The corresponding instance $\mathcal{T}=(G,r,\rho)$ of the Telephone Broadguess constructed from $\mathcal{D}$.

Figure 7: An illustration of the construction of the Telephone Broadcasting instance corresponding to a DoSePr instance

Construction. Given an instance $\mathcal{D}=(D,m)$ of DoSePr we construct an instance $\mathcal{T}=(G,r,\rho=2m)$ of Telephone Broadguess, where G is a snowflake graph with center r. For a given $x\in[\rho]$, let $\tilde{x}=\rho-x$. We construct the graph G by creating a reduced caterpillar (see Definition 2.3) for each dome in $\mathcal{D}$ as follows.

- The reduced caterpillar of a singleton dome (a,b) is a path of length $2\tilde{a}+2\tilde{b}$ with endpoints p and q (here, p,q,z are special vertices in the reduced caterpillar, where z is any arbitrary vertex).
- The reduced caterpillar of a regular dome (a,b,c,d) is formed by a path of length $2\tilde{b}+2\tilde{d}+1$ with endpoints p and q, which are two special vertices of the reduced caterpillar. The other special vertex z of the reduced caterpillar is the vertex at distance $2\tilde{b}$ of p (and thus distance $2\tilde{d}$ of q). In addition to the vertices on the path between p and q, the reduced caterpillar includes $\tilde{a}-\tilde{b}$ (which equals $\tilde{c}-\tilde{d}$) other vertices which find z as their sole neighbor.

For any reduced caterpillar C constructed above (from either a singleton or a regular dome), we label the vertex at distances $\tilde{a}$ form p as the *first gate* of C, denoted as u, and the one at distance $\tilde{b}$ from q as the *second gate of* C and denote it with v. We form the equivalent instance of the Telephone Broadguess problem by forming a graph G with a center vertex r (source) that is connected to all gates of all reduced caterpillars formed by the domes of D. By Definition 2.3, G will be a snowflake graph with center r. Figure 7 provides an illustration of this reduction.

Correctness. Before proving the correctness of our reduction, we prove the following lemmas for broadcasting in the reduced caterpillars associated with singleton and regular domes, respectively.

Lemma 4.13. *Consider a reduced caterpillar C of a singleton dome $D=(a,b)$ with gates u and v. One can complete broadcasting in C within ρ rounds if and only if u is informed no later than time a and v is informed no later than time b.*

Proof. TOPROVE 19 ☐

Lemma 4.14. *Consider a reduced caterpillar C of a regular dome $D=(a,b,c,d)$ with gates u and v. One can complete broadcasting in C within ρ rounds if and only if one of the following happens:*

- *(i) u is informed by time b and v is informed by time c.*

- *(ii) u is informed by time a and v is informed by time d.*

Proof. TOPROVE 20 ☐

We are now ready to prove the main result of this section.

Theorem 4.15. Telephone Broadcasting *problem is NP-complete for snowflake graphs.*

Proof. TOPROVE 21 ☐

5 Constant-Factor Approximation for Bounded Pathwidth

Graphs of pathwidth 1 are caterpillars [21], which are special types of trees, for which the Telephone Broadcasting problem is solvable in linear time [8]. On the other hand, for graphs with a pathwidth larger than 1, the Telephone Broadcasting problem is NP-hard, as established by our result for the hardness of the problem in snowflake graphs (Theorem 4.15). Recall that snowflake graphs have pathwidth 2 (Observation 2.1).

In this section, we establish the existence of a constant-factor approximation for Telephone Broadcasting on graphs with bounded pathwidth. Recall that we use $\boldsymbol{br}^*(G, s)$ to denote the optimal broadcasting time for an instance (G, s). We will demonstrate that the algorithm of Elkin and Kortsarz [7], which has an approximation factor of $\mathcal{O}\left(\frac{\log n}{\log \boldsymbol{br}^*(G,s)}\right)$ for any graph G, achieves a constant factor approximation for graphs of bounded pathwidth. For general graphs, this algorithm has an approximation factor of $\mathcal{O}(\log n/\log\log n)$ (because $\boldsymbol{br}^*(G, s) \geq \log n$), which is the best known approximation factor.

For any graph G of pathwidth w, we will show that $\boldsymbol{br}^*(G, s) = \Omega(n^{4^{-(w+1)}})$, which establishes that the algorithm of Elkin and Kortsarz [7] has a constant factor approximation for graphs of bounded pathwidth w.

To find a lower bound for $\boldsymbol{br}^*(G, s)$ where G is a graph of constant pathwidth, we repeatedly remove a vertex from G and show that the broadcasting in the remainder of G is not much slower compared to G. For that, we will use the following lemma, which holds for any instance of Telephone Broadcasting (but we only use it for graphs of bounded pathwidth).

Lemma 5.1. *Consider any instance (G, s) of* Telephone Broadcasting*, and let v be any arbitrary vertex in G. Let $H_1, \dots, H_m$ be the connected components resulting from removing v from G ($m \geq 1$). For any $i \in [m]$, choose an arbitrary vertex $s_i \in H_i$. Then, the following inequality holds: $\sum_{i\in[m]} \boldsymbol{br}^*(H_i, s_i) \leq \boldsymbol{br}^*(G, s)(2\boldsymbol{br}^*(G, s) + 1)$.*

Proof. TOPROVE 22 ☐

For completeness, we show that Lemma 5.1 is asymptotically tight.

Observation 5.1. *There are instances of the* Telephone Broadcasting *problem (G, s) and vertex $v \in G$ for which Lemma 5.1 is asymptotically tight.*

Proof. TOPROVE 23 ☐

The main result of this section can be stated as follows.

Theorem 5.2. *There is a polynomial-time algorithm for* Telephone Broadcasting *in graphs with constant pathwidth w, achieving an approximation ratio of $\mathcal{O}(4^w)$.*

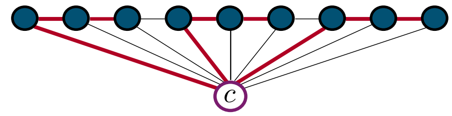


Figure 8: An illustration of Observation 5.1

Proof Overview. As mentioned above, it suffices to prove a lower bound of $\Omega(n^{4^{-(w+1)}})$ for $\boldsymbol{br}^*(G,s)$. For that, we assume G is a "w-path" in the sense that any two vertices that share a bag are neighbors. If G is not a w-path, we will add the missing edges to get a w-path without increasing the pathwidth. Clearly, the addition of new edges does not decrease the broadcast time of G, and since we are looking for a lower bound for the broadcast time, it suffices to focus on w-paths.

We assume a *standard path decomposition* of the input graph G, in which no bag is a subset of another. In such decompositions, we define the notion of *span* of a vertex v as the number of bags where v appears. The span of a decomposition is then the maximum span over the spans of all its vertices.

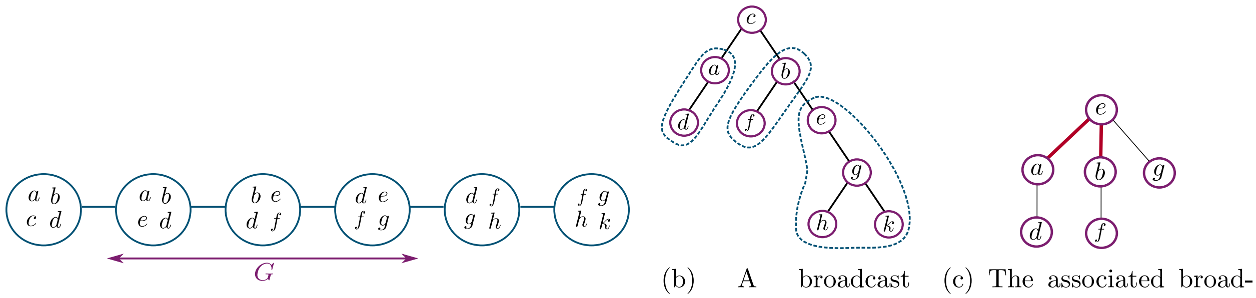
The proof of lemma is established by induction over the size of the input graph G. For that, we consider two cases. First, if the span of the decomposition is "small", we argue that the diameter of G will be "large", and then the desired lower bound for $\boldsymbol{br}^*(G,s)$ holds. On the other hand, when the span is "large", we consider the vertex v_m that has the maximum span and consider the graph G_{v_m} induced by vertices that share a bag with v_m. Note that G_{v_m} is a smaller graph compared to G, and we can show that broadcasting in G_{v_m}, starting from any vertex, cannot be much slower than broadcasting in G, starting from s (Lemma 5.3). Moreover, we will use Lemma 5.1 to show that if we extract v_m from G_{v_m} to get a set $\{H_1,\dots,H_q\}$ of disjoint connected components, total broadcast time in these component components (starting from arbitrary sources) is not much slower in the absence of v_m. On the other hand, since v_m is removed from G_{v_m}, any of these components H_i has a pathwidth that is at least one unit less than G_{v_m}, and we can use an inductive argument to achieve a lower bound on the broadcast times of any H_i (Lemma 5.4). In summary, by the induction hypothesis, total broadcast time in H_i's gives a lower bound for broadcasting in G_{v_m} which itself gives a lower bound for broadcasting in G.

Now for the formal proof, we prove a lower bound for the broadcast time of any connected graph G with n vertices and pathwidth $w \in \mathcal{O}(1)$. In particular, for any $s \in G$, we will show $\boldsymbol{br}^*(G,s) \in \Omega(n^{4^{-w}})$. Consider a fixed path decomposition of G; we assume the bags in the deposition are arranged from left to right. For now, suppose s is in the left-most bag (this assumption will be relaxed later in Lemma 5.4).

We assume the path decomposition is "standard" in the sense that no bag is a subset of another bag (otherwise, one can remove the smaller bags to attain a standard decomposition). Finally, we assume that G is a "w-path" in the sense that any two vertices located in the same bag are connected in G (all edges allowed in the decomposition are present). If the input graph is not a w-path, we can make it w-path by adding all the missing edges. Clearly, the addition of new edges does not decrease the broadcast time of G, and since we are looking for a lower bound for the broadcast time, it suffices to focus on w-paths.

We now proceed with a formal proof.

Lemma 5.3. *Let (H,s) be an instance of the broadcast problem, where H is a connected w-path. Suppose we have a standard path decomposition D of width w for H where s appears in the first bag of D. Let G be an induced subgraph of H formed by vertices that appear in a consecutive set of bags in D and suppose a vertex $s' \in G$ appears in all such bags. Then, we can write*



(a) Standard path decomposition for a graph H (b) A broadcast scheme T_H for (H, s) (c) The associated broadcast scheme T_G for (G, s')

Figure 9: An illustration of Lemma 5.3

$\boldsymbol{br}^*(G, s') \leq \boldsymbol{br}^*(H, s) + 2w$.

Proof. TOPROVE 24 ☐

To establish a lower bound for broadcasting in any connected w-path graph G, we prove a slightly stronger result. Suppose we have k connected w-path graphs $H_1, H_2, \ldots, H_k$, each H_i having at least two vertices, a standard path decomposition of width w and a vertex $s_i \in H_i$ located at the left-most bag in its decomposition. We prove a lower bound for the total time for broadcasting in these graphs as follows. This is a generalization of our desired lower bound for the case of $k = 1$.

Lemma 5.4. *Consider a set of $k \geq 1$ vertex disjoint connected w-path graphs $\mathcal{H} = \{H_1, \ldots, H_k\}$. Suppose any H_i has a standard path decomposition D_i with ℓ_i bags, where a source vertex s_i is located in the leftmost bag of D_i. Let $L = \sum_{i=1}^{k} \ell_i$. Then we have $\sum_{i=1}^{k} \boldsymbol{br}^*(H_i, s_i) \geq f(L, w)$, where $f(L, w) = 27^{-w}(w!)^{-2}L^{4^{-w}}$.*

Proof. TOPROVE 25 ☐

The following lemma generalizes Lemma 5.4 (applied with $k = 1$) to the case where the source is not necessarily located in the first bag of the path decomposition.

Lemma 5.5. *Let G be any connected graph and let s be any vertex of G. Suppose G has a standard path decomposition of width w formed by L bags. Then, we have $\boldsymbol{br}^*(G, s) \geq f(L-1, w+1)$, where $f(L, w) = 27^{-w}(w!)^{-2}L^{4^{-w}}$.*

Proof. TOPROVE 26 ☐

We are now ready to prove the main result of this section.

Proof. TOPROVE 27 ☐

6 Concluding Remarks

In this paper, we resolved an open problem by proving the NP-completeness of Telephone Broadcasting for cactus graphs as well as graphs of pathwidth 2. We also established a 2-approximation algorithm for cactus graphs and a constant-factor approximation algorithm for graphs of bounded pathwidth. A possible direction for future work is to improve the approximation factor for graphs of bounded pathwidth or cactus graphs. In particular, it remains an open question whether Polynomial Time Approximation Schemes (PTASs) exist for these graph classes. A major open problem in this domain is determining whether a constant-factor approximation exists for general graphs. While progress on this question has been slow, the algorithmic ideas developed in this work may be applicable to broadcasting in other families of sparse graphs.

Acknowledgement

We acknowledge the support of the Natural Sciences and Engineering Research Council of Canada (NSERC)[funding reference number DGECR-2018-00059].

References

[1] Boaz Ben-Moshe, Amit Dvir, Michael Segal, and Arie Tamir. Centdian computation in cactus graphs. *Journal of Graph Algorithms and Applications*, 16(2):199–224, 2012.

[2] Puspal Bhabak and Hovhannes A. Harutyunyan. Constant approximation for broadcasting in k-cycle graph. In *Proceedings of first Conference of Algorithms and Discrete Applied Mathematics*, volume 8959, pages 21–32, 2015.

[3] Maja Čevnik and Janez Žerovnik. Broadcasting on cactus graphs. *Journal of Combinatorial optimization*, 33:292–316, 2017.

[4] Peter Damaschke. A linear-time optimal broadcasting algorithm in stars of cliques. *Journal of Graph Algorithms and Applications*, 28(1):385–388, 2024.

[5] Anne-Laure Ehresmann. Approximation algorithms for broadcasting in flower graphs. Master's thesis, Concordia University, 2021.

[6] Michael Elkin and Guy Kortsarz. Combinatorial logarithmic approximation algorithm for directed telephone broadcast problem. In *Proceedings of the thirty-fourth annual ACM symposium on Theory of computing*, pages 438–447, 2002.

[7] Michael Elkin and Guy Kortsarz. Sublogarithmic approximation for telephone multicast. *Journal of Computer and System Sciences*, 72(4):648–659, 2006.

[8] Fraigniaud and Mitjana. Polynomial-time algorithms for minimum-time broadcast in trees. *Theory of Computing Systems*, 35(6):641–665, 2002.

[9] M. R. Garey and David S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness.* 1979.

[10] Saber Gholami, Hovhannes A Harutyunyan, and Edward Maraachlian. Optimal broadcasting in fully connected trees. *Journal of Interconnection Networks*, 23(01):2150037, 2023.

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

[12] Hovhannes Harutyunyan and Edward Maraachlian. Linear algorithm for broadcasting in unicyclic graphs. In *Proceedings of the 13th Annual International Conference on Computing and Combinatorics*, pages 372–382, 2007.

[13] Hovhannes Harutyunyan, George Laza, and Edward Maraachlian. Broadcasting in necklace graphs. In *Proceedings of the 2nd Canadian Conference on Computer Science and Software Engineering*, pages 253–256, 2009.

[14] Hovhannes A Harutyunyan and Narek Hovhannisyan. Improved approximation for broadcasting in k-path graphs. In *International Conference on Combinatorial Optimization and Applications*, pages 111–122, 2023.

[15] Hovhannes A Harutyunyan and Arthur L Liestman. Improved upper and lower bounds for k-broadcasting. *Networks: An International Journal*, 37(2):94–101, 2001.

[16] Hovhannes A Harutyunyan and Edward Maraachlian. On broadcasting in unicyclic graphs. *Journal of combinatorial optimization*, 16:307–322, 2008.

[17] Hovhannes A Harutyunyan, Narek Hovhannisyan, and Edward Maraachlian. Broadcasting in chains of rings. In *Proceedings of the Fourteenth International Conference on Ubiquitous and Future Networks*, pages 506–511, 2023.

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

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

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

[21] Andrzej Proskurowski and Jan Arne Telle. Classes of graphs with restricted interval models. *Discret. Math. Theor. Comput. Sci.*, 3(4):167–176, 1999.

[22] Neil Robertson and Paul D Seymour. Graph minors. i. excluding a forest. *Journal of Combinatorial Theory, Series B*, 35(1):39–61, 1983.

[23] Weiping Shang, Pengjun Wan, and Xiaodong Hu. Approximation algorithms for minimum broadcast schedule problem in wireless sensor networks. *Frontiers of Mathematics in China*, 5:75–87, 2010.

[24] Peter J. Slater, Ernest J. Cockayne, and Stephen T. Hedetniemi. Information dissemination in trees. *SIAM Journal on Computing*, 10(4):692–701, 1981.

[25] Elena Stöhr. Broadcasting in the butterfly network. *Information Processing Letters*, 39(1): 41–43, 1991.

[26] Prafullkumar Tale. Double exponential lower bound for telephone broadcast. *arXiv preprint arXiv:2403.03501*, 2024.

[27] Craig A Tovey. A simplified np-complete satisfiability problem. *Discrete applied mathematics*, 8(1):85–89, 1984.