Pavol Jozef Šafárik University in Košice
Faculty of Science
Institute of Computer Science

COMMUNICATION IN RADIO NETWORKS

PHD THESIS

František Galčík

thesis advisor:
doc. RNDr. Gabriel Semanišin, PhD.

Košice May 2009

Contents

List of Figures

Acknowledgements

First of all, I would like to thank my advisor Gabriel Semanišin for supporting me during my PhD studies, introducing me to scientific world, useful advices, and for giving me so much freedom of research interests.

My acknowledgments go to Rastislav Královič for learning me the basics of distributed computing and algorithms, introducing the research area of radio networks to me during my master studies at Comenius University, and for his support during my PhD studies.

Especially, I would like to thank Leszek Gąsieniec who took care of me during my stay in Liverpool, where I learned what are science and research. My thanks are for his support, enthusiasm, and innumerable discussions full of inspiring ideas and suggestions.

I also wish to thank all people I met, because each discussion brings something new and changes (slightly or significantly) the way of thinking. I wish to mention all my colleagues at P.J. Šafárik University in Košice, former schoolmates and teachers at Comenius University in Bratislava, all people I met at the University of Liverpool, and researchers Andrzej Lingas, Igor Potapov, Dariusz Kowalski, and Andrzej Pelc.

Lastly, and most importantly, I would like to thank my family for their love, unlimited support, and patience during all my life.

Abstract

The thesis is devoted to complexity and algorithmical aspects of communication in multi-hop radio networks modelled by graph-based models. Primary focus is on scenarios (topologies, models, and communication tasks) that have not been in the center of attention of the research community. We investigate how restricted topology of radio networks can help to design more efficient communication algorithms and protocols in both centralized and distributed settings. Time, the number of rounds required to accomplish a communication task, is considered as the main efficiency criterion. However, the majority of proposed algorithms are designed in such a way that minimizes the number of transmissions per node, i.e., the energy consumption is considered as another efficiency criterion.

We deal with intensively investigated centralized broadcasting in the case when an underlying reachability graph belongs to a certain graph class (e.g., class of planar graphs, class of k-degenerate graphs, etc.). In the fully distributed setting, we consider radio broadcasting in networks with the grid topology. Although, we deal with special topologies, some results are directly related to broadcasting in radio networks with arbitrary topology. Moreover, in the fully distributed setting with nodes capable of collision detection, we explore the concept of encoding information into collisions. We focus on more complex communication tasks like maximum finding and computation of some network parameters as well.

The standard graph model of radio networks adopted by the algorithmic community is often criticized due to its simplifications that do not reflect communication environment of some real-world radio networks properly. In order to treat this problem, we propose a new graph-based model of radio networks that generalizes the standard graph model. The new model is based on the concept of interference reachability graphs that allow to model radio networks with long-range interference, i.e., radio networks where interference range of a node exceeds its transmission range. We study how presence of long-range interference influences time efficiency of centralized broadcasting. As a result, we design several algorithms generating radio broadcasting schedules for radio networks of arbitrary topology with long-range interference.

Abstrakt

Práca sa venuje zložitostným a algoritmickým aspektom komunikácie v multi-hop rádiových sieťach modelovaných grafovými modelmi. Hlavná pozornosť je upriamená na komunikačné scenáre (topológie, modely a komunikačné úlohy), ktoré doposiaľ neboli stredobodom záujmu výskumnej komunity. V práci študujeme, do akej miery predpoklad obmedzenej topológie rádiovej siete umožňuje vytvoriť efektívnejšie komunikačné algoritmy a protokoly. Skúmame tak centralizované, ako aj plne distribuované komunikačné prostredia. Ako hlavné kritérium efektívnosti uvažujeme čas - počet komunikačných kôl potrebných na realizáciu komunikačnej úlohy. Väčšina z prezentovaných algoritmov je však navrhnutá takým spôsobom, aby okrem času minimalizovala aj počet vysielaní realizovaných uzlami siete. Teda popri čase vykonávania úlohy je ďalším uvažovaným kritériom efektívnosti energetická náročnosť komunikačného protokolu.

V tejto dizertačnej práci sa zaoberáme intenzívne skúmaným problémom centralizovaného broadcastingu v prípade, že graf dosiahnuteľnosti prislúchajúci rádiovej sieti patrí do nejakej konkrétnej triedy grafov (napr. triedy planárnych grafov, triedy k-degenerovaných grafov, atď.). V prípade plne distribuovaného komunikačného prostredia uvažujeme taktiež broadcasting v rádiových sieťach, ktorých topológiou je mriežka. Hoci sa venujeme prevažne špeciálnym topológiám sietí, niektoré z našich výsledkov sú priamo aplikovateľné aj v sieťach s ľubovoľnou topológiou. Okrem toho, uvážiac plne distribuovaný model s uzlami siete vybavenými schopnosťou detekcie kolízií, skúmame koncepciu kódovania informácie do kolízií spôsobených interferenciou. V tomto modely sa tiež zaoberáme komplexnejšími komunikačnými úlohami ako sú výpočet maximálnej hodnoty, či výpočet niektorých parametrov siete.

Štandardný grafový model rádiových sietí, ktorý bol prijatý algoritmickou komunitou, je často kritizovaný z dôvodu zjednodušení, ktoré nedostatočným spôsobom reflektujú komunikačné prostredie niektorých reálnych rádiových sietí. V snahe riešiť tento problém sme navrhli nový grafový model rádiových sietí, ktorý zovšeobecňuje štandardný grafový model. Tento nový model je založený na tzv. interferenčných grafoch dosiahnuteľnosti, ktoré umožňujú modelovať rádiové siete s interferenciou väčšieho dosahu, t.j. rádiové siete, v ktorých interferenčný rozsah uzla presahuje jeho vysielací rozsah. Skúmame, ako prítomnosť zvýšenej interferencie ovplyvňuje možnosť časovo efektívnej realizácie centralizovaného broadcastingu. Výsledkom nášho výskumu je okrem iného aj niekoľko algoritmov generujúcich časovo efektívne broadcastovacie rozvrhy pre ľubovoľné rádiové siete so zvýšenou interferenciou.

Chapter 1

Introduction

Nowadays, new technologies in computers and telecommunications cause that computer and communication networks are part and parcel of the human being. Communication networks are collections of information processing, autonomous nodes that are interconnected for the purpose of data communication. The nodes communicate via sending messages (packets). There are various taxonomies of communication networks. Some of them are the following:

- based on distance among nodes - LAN (local area network), WAN (wide area networks), etc.
- based on hops among nodes - single-hop networks (each pair of nodes is connected by a communication link) or multi-hop networks (remote nodes communicate via intermediate nodes)
- based on transmission technology - nodes are connected by cooper wire, optic fiber, or wireless medium
- based on method of sending and receiving information - point-to-point networks (certain pairs of nodes are connected by a communication link), broadcast networks (all nodes share a single communication channel), etc.
- based on network topology - bus network, star network, hierarchical network, etc.

Intuitively, particular features of a communication network (i.e., used transmission technology, method of sending information, etc.) can significantly influence design and effectiveness of algorithms realizing a communication task. For instance, the fact that a message (signal) comes over a communication link can be less relevant than what happens if two or more messages arrive simultaneously at a node. Indeed, there are communication networks for which simultaneous arrival of messages causes that some or all arrived messages are discarded. It can be caused by features of transmission medium or limited buffering capabilities of nodes. In that case, (possibly expensive) retransmission of discarded messages could be necessary.

In recent years, wireless and mobile communication have seen explosive growth both in terms of the number of services provided and the types of technologies that have become available. Most of wireless solutions use radio waves as a transmission medium. Radio communication has been moved from typically military applications to the civil sphere (cellular phones, wireless local area networks, sensor networks, etc.). Success of radio communication lies in two its key features. Radio networks allow mobility of nodes (users) and building of low-cost communication infrastructure.

1.1 Radio Networks

This thesis is devoted to the area of radio networks. Radio networks differ from other communication networks in the way how the nodes send and receive messages. We consider a *radio network* to be a collection of receiver-transmitter nodes (devices, stations). Depending on transmission power and geographic characteristics of the surrounding region, a message (signal) which is transmitted by a node can reach only a given subset of other nodes. This subset of nodes is referred to as a *transmission range of node.* Communication in radio networks has broadcast character, but only in a limited distance. Particularly, a message transmitted by a node is always sent to all nodes which are located within its transmission range. On the other hand, if at least two messages arrive simultaneously at a node, an interference of radio waves causes that message content is destroyed, i.e., no message is received.

The network is *synchronous*, i.e., the nodes have access to a clock. The clocks tick at the same rate and it is assumed that they start at the same time, but with possibly different initial values. A clock cycle is called a *step*, a *time slot*, or a *round.* This type of synchronization defines locally synchronous model. In the case that initial clock values are the same a globally synchronous model is defined. As mentioned above, time is assumed to be slotted and all transmissions are realized at time slot boundaries. In a given time slot, a node can transmit or listen to radio channel (not simultaneously). As noted in [58], synchronization of local clocks can be achieved by interfacing a global positioning systems. Under current technology, GPS systems provide time information accurate to 100 nanoseconds. It allows nodes to detect time slot boundaries and thus to synchronize. Note that asynchronous communication in radio networks was studied in [13, 63].

All nodes operate on the same radio frequency. This assumption seems to be a significant restriction, but it is easy to see, that constantly more radio frequencies can improve effectiveness at most by a constant factor.

Another assumption that significantly determines effectiveness and design of communication algorithms is an assumption concerning *initial knowledge* of the nodes. In the literature, it is mostly assumed that initial knowledge of a node contains at least its unique integer identifier (ID) or a label. The range of identifiers is assumed to be $1, \ldots, N$, where $N = O(n)$ and n is the number of nodes forming a radio network. Some works (e.g., [35, 38]) deal with the assumption of large labels, i.e., the upper bound of identifiers is $N = O(n^p)$, for a constant $p > 1$. A network without unique identifiers of the nodes is called an *anonymous network.* In *ad-hoc (or unknown) radio networks*, initial knowledge of a node is limited only to its own identifier. The authors of other works assume that initial knowledge of nodes is extended by information about the network in their surroundings. For instance, a node knows identifiers of nodes located within its transmission range [2, 45] or in a defined distance from it [23]. Assuming a further extension of initial knowledge of nodes, one can define a *known topology radio network.* In such a radio network, each node is aware of the whole network topology. For known topology radio networks, algorithms are often referred to as *centralized*, since a schedule of transmissions can be computed by the nodes in advance. Communication tasks in known topology radio networks were intensively investigated in [26, 27, 36, 49].

Two types of models of radio networks prevail in the literature. The first one is a *graph model.* The network is modelled by a directed graph called a *reachability graph* $G = (V, E)$. The vertex set of G consists of the network nodes and two vertices $u, v \in V$ are connected by an edge $e = (u, v)$ if and only if the transmission of the node u can reach the node v. In such a case the node u is called a *neighbor* of the node v. If the transmission power of all nodes is the same, then the reachability graph is symmetric, i.e., a symmetric radio network can be modelled by an undirected graph. The second type of model is a *geometric model.* Each node of a radio network is represented as a point in k-dimensional Euclidian space. Each point has a region associated with it. If a point which corresponds to a node v lies in a region associated to a point corresponding to a node u, then any transmission of the node u can reach the node v. Again, u is called a neighbor of the node v. Of course, the most interesting case is 2-dimensional Euclidian space in which a region associated to a point is a circle centered at this point. Radius of the circle corresponds to the power of transmitter.

Obviously, the graph model is more general than the geometric model. Given a set of points, it is easy to construct a reachability graph on the set of these points such that there is a directed edge from u to v if and only if the point v lies in the region associated to the point u. On the other hand, more effective algorithms were designed (e.g., in [24] or [29]) for radio networks modelled by the geometric model. As for the applicability, the geometric model is more appropriate for radio networks in flat regions without large obstacles, where a transmission reaches the same distance in every direction. In regions with large obstacles (mountains, buildings, etc.), where reach of a transmitter depends on the direction, the graph model is more appropriate.

In what follows, we focus only on radio networks modelled by the graph model.

1.2 Communication in Radio Networks

As previously noted, communication in radio networks is synchronous. At the beginning of a round (time slot), each node decides whether it will operate as a *receiver* or as a *transmitter*. Note that it is assumed that all computations concerning an action in the given round are not taken into consideration (an action can be determined at the early beginning of a round or at the end of the previous round).

If a node operates as a transmitter, it transmits a message. The transmitted message reaches all nodes within its transmission range in the same round.

If a node operates as a receiver, there are three possible cases.

(1) If a node is not located within transmission range of a transmitting node, i.e., none of its neighbors transmits in the given round, it hears a *background noise* and no message is received.

(2) If a node has exactly one transmitting neighbor, it receives the transmitted message.

(3) If a node has more than one transmitting neighbor, i.e., at least two messages arrive simultaneously in this round, a *collision* occurs and it hears an *interference noise*. The interference causes that no message is received.

The nodes of a radio network are always able to recognize the second case, i.e., the case when exactly one message arrives. Generally, it is assumed that the background noise and the interference noise are indistinguishable for the nodes. It means that a node is not aware whether none or at least two of its neighbors transmit in a given round.

If a node can distinguish the background noise (the first case) and the interference noise (the third case), we say that it is equipped with a *collision detection capability*. Under the assumption of collision detection capability, adopting a notation from [10, 59], we say that a node, acting as a receiver in a given round, hears a *μ-signal*, if at least one of its neighbors operates as a transmitter and sends a message in this round. Otherwise, we say that it hears a *λ-signal*.

Simultaneous transmission of a message to all neighbors of a node seems to be a feature which brings speed-up for communication tasks disseminating information. On the other hand, collisions cause that design of time-efficient algorithms is difficult. If a neighbor u of a node v has to deliver a message to v, rules of an algorithm should guarantee that there will be a round in which no other neighbor of v transmits. Similarly, in the case when more neighbors of the node v possess the same message requiring delivery to v, they have to decide which of them informs v. In many cases, simultaneous transmissions and collisions are the reason of communication slowdown.

1.3 Communication Scenarios

In this section, we present various assumptions concerning the process of communication in radio networks. Their combinations result in many communication scenarios used in the literature. They precise a considered model of a radio network and thus significantly influence design of effective communication algorithms. In the literature, the most considered assumptions are the following.

- *randomization in communication process* - Randomized algorithms run faster that deterministic, require little initial knowledge, and are easier to implement. On the other hand, an accomplishment of a task (totally or in given efficiency bounds) is not guaranteed, but is achieved with high probability. Note that there are settings and communication tasks (e.g., broadcasting in anonymous networks without collision detection) for which deterministic algorithms do not exist, however, randomization provides a solution working with high probability.

- *centralized / distributed control* - Centralized algorithms assume existence of a central monitor that is aware of network topology and controls transmissions of the nodes. Central monitor can compute a schedule of transmissions in advance. Assuming known topology radio networks (possibly with globally synchronized clocks), some centralized algorithms can be implemented in distributed way. In each round, each node acts according to a schedule of the central monitor, which can be computed locally utilizing an algorithm of the central monitor. If full topology information is not available, designed algorithms have to rely only on local initial knowledge.

- *adaptivity* - Nonadaptive algorithms have all transmissions scheduled in advance. In adaptive algorithms, a node may schedule future transmissions on-line according to its previous history. Since the transmissions can be scheduled in centralized algorithms or in known topology radio networks beforehand, adaptivity can help more significantly in distributed settings. In this setting, the nodes have only limited knowledge about the network. Note that assumptions concerning adaptivity are closely related to memory and computational requirements of real-world devices.

- *collision detection capability* - If the nodes transmit in an appropriate way, it is possible to obtain extra information using collision detection mechanism. Thus, we can design more efficient communication algorithms. In order to produce radio devices as simple as possible, the assumption about collision detection capability cannot be always fulfilled. Hence, most of works assume a lack of collision detection capability.

- *fault-tolerance* - It is usually assumed that the communication environment is fault-free. Clearly, this assumption is not realistic, because probability of a fault increases with increasing size and complexity of the communication network. Fault-tolerant communication algorithms have to complete their tasks even in the case when at most a certain number of components fails. In the following, we shall consider only fault-free communication environment. Note that fault-tolerant algorithms for radio networks were discussed in [43, 51].

- *locally / globally synchronized clocks* - Mostly, if the clocks are not globally synchronous, a node cannot participate in a communication algorithm before successful receiving of a message. It follows that there is initially a limited possibility of parallelization. Indeed, a sort of broadcasting has to be completed before. On the other hand, if the nodes know a round in which a communication algorithm starts, they can simultaneously retrieve additional information about the network and so to speed-up execution of the algorithm. If the nodes are allowed to transmit before receiving a message sent by another node during previous execution of the algorithm, we say that *spontaneous transmissions* are allowed.

1.4 Communication Tasks

In order to carry out complex communication tasks in radio networks (or in other communication networks), we have to design communication algorithms heading towards desired goals and preserving acceptable efficiency costs. There are communication tasks that appear more often than others and are utilized usually as subroutines of more complex tasks. Such communication tasks are called *communication primitives.* Since effectiveness of designed algorithms for these primitives has a great impact on effectiveness of more complex tasks, their study makes many challenges. In the rest, we present some communication primitives studied in the context of radio communication.

1.4.1 Broadcasting

The study of communication in radio networks was initiated in the context of broadcasting. Broadcasting is typically used as an initialization subroutine or a subroutine distributing computed information to all nodes in the network. The goal of broadcasting is to disseminate information (a *source message*) from one distinguished node, called a *source*, to all other nodes in the network. In multi-hop radio networks, remote nodes are informed via intermediate nodes. Also, a broadcasting algorithm can be naturally described by a *radio broadcasting schedule.* It prescribes for each round, which nodes transmit the source message. Clearly, only a node that already knows the source message can transmit. The broadcasting task is completed when all nodes become informed. Note that it is not required that the source is notified after the broadcasting is completed. This is not true for an *acknowledged radio broadcasting*, which is completed in a round such that all nodes of the network are informed and the source is aware of this fact. Obviously, if the reachability graph of an unknown radio network is not strongly connected, it is not possible to complete acknowledged radio broadcasting. If the broadcasting time is upper-bounded by an expression and the source is aware of all parameters occurring in this expression, the source knows (can compute) the round in which the task is completed at the latest.

1.4.2 Gossiping

Initially, each node possesses a message. The goal of *gossiping* is to distribute all messages to all network nodes. In a n-node radio network, gossiping can be seen as n simultaneous broadcastings or as an exchange of messages between all pairs of nodes. Gossiping problem was mostly considered in models with globally synchronous clocks, where the nodes know a round in which the gossiping starts. Many works assume that it is possible to include whole history of received messages in each message transmitted during the work of a gossiping algorithm. This not very realistic assumption is avoided in works focusing on $b(n)$-gossiping. In $b(n)$-gossiping, each transmitted message can contain at most $b(n)$ single messages or $b(n)$ bits of auxiliary information, where b is an integer function and n is the number of network nodes. The 1-gossiping is referred to as a gossiping with unit messages. Note that a survey of some results concerning the gossiping task can be found in [70].

1.4.3 Multipoint-to-multipoint Multicast

In order to support group communication in radio networks, *multipoint-to-multipoint multicast* was investigated. The goal of multipoint-to-multipoint multicast (M2M multicast) is to support an exchange of messages within a fixed group of nodes. It can be seen as a generalization of gossiping which concerns only a certain subset of nodes called *participants*.

1.4.4 Maximum Finding

Let us assume that each node possesses a value (an integer or real value in appropriate coding). The goal of maximum finding is to compute the maximal possessed value in one distinguished node called the *initiator*. The process of computation is started by the initiator in a round unknown to other network nodes.

Maximum finding algorithms can be utilized in the following real-world situation. Consider a collection of sensors measuring a physical quantity and communicating via low-power radio. There is one distinguished node (for instance, a node connected to the control center) that needs to know the maximum among measured values in order to perform an operation (e.g., to indicate an alert). One can utilize maximum finding algorithm for a selection of a node which satisfies desired properties (e.g., it possesses the largest number of messages, it has the largest identifier, etc.).

1.4.5 Wake-up Problem

At the beginning, all nodes are asleep, but some nodes wake up spontaneously in arbitrary time slots. The main objective is to activate (wake up) all network nodes. A node becomes activated if and only if it wakes up spontaneously or it hears a message. Note that a node hears a message in a given round if and only if exactly one of its neighbors transmits in this round. All activated nodes transmit according to a protocol in order to wake up all other network nodes.

The wake-up problem is closely related to other synchronization-type primitives like leader election and synchronization of local clocks. Since the wake-up problem that assumes exactly one spontaneously woken-up node is equal to the broadcasting problem, the wake-up problem can be seen as a generalization of the broadcasting problem.

1.5 Graph Model and Related Terminology

Throughout this thesis, we focus on radio networks modelled by graphs. Graph is one of the most natural and simplest ways to describe a communication network. Therefore, the graph model was adopted by distributed algorithms community and it prevails in most of works coming from this community. Let V be a set of network nodes. We can associate with each node $v \in V$ a set of nodes $T_R(v)$ which can receive transmission from the node v directly. We refer to $T_R(v)$ as a *transmission range* of the node v. A graph $G = (V, E)$ modelling a radio network is called a *reachability graph.* The vertex set of G is the set of network nodes. Two vertices $u, v \in V$ are connected by an edge $e = (u, v)$ if and only $v \in T_R(u)$, i.e., $(u, v) \in E \iff v \in T_R(v)$. The node u is called a *neighbor* of the node v. Unless stated otherwise, throughout this thesis, we assume that an underlying reachability graph is symmetric. This corresponds to the case when transmission power of all nodes is the same.

Thanks to modelling a network by a graph, we can describe communication network by graph parameters of an underlying graph. We adopt the standard graph terminology.

Definition 1.5.1. *Let $G = (V, E)$ be a graph. We denote the smallest integer d, s.t., there is a path along d edges connecting a node u to a node v, as $d(u, v)$. Eccentricity of a node u, denoted as $ecc(u)$, is defined as $max\{d(u, v)|v \in V\}$. We refer to $max\{ecc(v)|v \in V\}$ as a diameter D of the graph G.*

We shall denote a set of all neighbors of a node v as $N(v)$. Degree of a node v, denoted as $deg(v)$, is the number of its neighbors, i.e., $deg(v) = |N(v)|$. Finally, we refer to $\Delta = max\{deg(v)|v \in V\}$ as the maximum degree of the graph.

Algorithms and analysis of algorithms require often to consider all nodes that are in the same distance from a distinguished node. For a given node s, we denote as $L_i = \{v|d(s,v) = i\}$ a set of nodes that are in distance i from the node s. The set L_i is referred to as an i-th layer of the graph with respect to the node s.

We are interested in the worst case analysis of designed communication algorithms. Since $ecc(v) \leq D \leq 2 \cdot ecc(v)$ for an arbitrary node v, the eccentricity of a distinguished node (a source or an initiator) and the diameter of the graph can be interchanged without affecting asymptotical complexity.

1.6 Complexity Measures

Effectiveness of designed algorithms is typically bounded by a function of some parameters of an underlying reachability graph, e.g., the diameter D, the number of nodes n, the eccentricity $ecc(s)$ of a distinguished node s, or the maximum degree Δ.

Most of works considers the time that is required to complete a prescribed communication task as a measure of effectiveness of designed algorithms. In the synchronous model of radio networks, time corresponds to the number of rounds. Each communication task defines two distinguished rounds: the first one is considered as a start round of the task and the second one as a completion round. Main reason why this measure is so important is the assumption that the topology of a radio network remains unchanged during work of an algorithm. This assumption is realistic, if designed algorithms are fast, i.e., they accomplish the task in (short) time which is negligible comparing to a period of topology changes. Note that we shall denote the minimal number of rounds required to complete a communication task by an optimal schedule with respect to a source/initiator s and a reachability graph G as $mintime(s,G)$.

Another considered efficiency measure is the number of transmissions during the work of an algorithm. Very often the nodes of radio networks are powered by batteries and thus there is close relationship between lifetime of nodes and the number of transmissions. It raises the issue of design of fast algorithms using low number of transmissions. Energy consumption can be considered in a global view as a sum of all transmissions realized by nodes, or in a local view when we limit the number of transmissions per node during the whole work of algorithm. The local view results in k-shot transmission schedules when each network node is allowed to transmit at most k-times.

Naturally, we can combine both previously mentioned measures in a combined efficiency measure.

1.7 Summary of Results

In Chapter 2, we study the broadcasting problem in known topology radio networks. We focus on the broadcasting problem in radio networks with some topology restrictions. We generalize the broadcasting algorithm presented in [36] to a general schema. The main advantage of this schema is that it is suitable for generating broadcasting schedules for arbitrary reachability graphs which belong to a certain graph class. Indeed, it uses another algorithm for information dissemination in bipartite reachability graphs as a sub-routine to generate parts of the broadcasting schedule. Efficiency of a generated schedule follows from the fact that this sub-routine algorithm can be specialized for bipartite graphs of some graph classes. Moreover, our designed schema simplifies proofs of some known results. Utilizing this schema, we investigate communication in radio networks with planar and k-degenerate reachability graphs. Also, we provide lower bounds matching or almost matching presented upper bounds. The main results presented in this chapter are the following:

- a schema generating radio broadcasting schedules with length $D + O(time_b(G) \cdot \log n)$, where $time_b(G)$ is the maximal length of a schedule for information dissemination in a bipartite subgraph of G generated by a given algorithm $\mathcal{A}_b$,
- a simplified algorithm for generating $D + O(\Delta \cdot \log n)$-round broadcasting schedules for arbitrary reachability graphs and $D + O(\log n)$-round schedules for planar reachability graphs,
- lower bounds $2 \cdot ecc(s)$, $3/2 \cdot D$, and $D + \Omega(\log n)$ for planar reachability graphs, and
- an algorithm generating $D + O_k(\log^2 n)$-round broadcasting schedules for k-degenerate reachability graphs.

Chapter 3 is devoted to communication in a newly proposed model of radio networks in which transmission and interference range of a node differ. This model generalizes the standard graph model of radio networks. Also, we introduce new parameters to reflect presence and structure of interference edges. In order to study communication in bipartite graphs, we define and investigate interference selective families as an useful combinatorial tool. Later, we construct other algorithms for energy and time efficient information dissemination in bipartite graphs. They are used as building blocks for algorithms generating broadcasting schedules for arbitrary interference reachability graphs. We design a general schema, similar to the schema in Chapter 2, which generates broadcasting schedules for arbitrary networks. Finally, we design another algorithm generating time and energy efficient broadcasting schedules which takes into consideration a newly defined parameter called an interference distance of a network. This algorithm generates 1-shot broadcasting schedules and it is based on clustering. The main results in Chapter 3 are the following:

- a new model of radio networks which generalizes the standard graph model and models radio networks in which transmission and interference ranges differ,
- a deterministic polynomial-time algorithm generating interference selective families with size $O((1+r(\mathcal{F}))\cdot((1+\log(\Delta_{max}/\Delta_{min})))\cdot\log|\mathcal{F}|)$, where $r(\mathcal{F})$ is an interference ratio of the collection $\mathcal{F}$,
- algorithms generating 1-shot transmission schedules for information dissemination in bipartite interference reachability graph with lengths $O(\Delta^2)$ and $O(\Delta\cdot\log n)$,
- a general schema generating broadcasting schedules in arbitrary interference reachability graphs which results in 2-shot broadcasting schedules with lengths $O(\Delta\cdot D_T+\Delta^2\cdot\log n)$ and $O(\Delta\cdot(D_T+\log^2 n))$,
- an algorithm producing 1-shot broadcasting schedules with the length $D_T+O(\Delta\cdot d_I(G)\cdot\log^4 n)$ where $d_I(G)$ is an interference distance of the graph G, and
- a lower bound $ecc_T(s)+\Omega\left(\Delta\cdot\frac{\log n}{\log\Delta}\right)$ rounds on broadcasting time in interference reachability graph with interference distance 2.

In Chapter 4, we investigate communication in radio networks where the nodes have no topology information in advance, i.e., communication in radio networks in a fully distributed setting. We deal with non-standard and not well investigated communication tasks like maximum finding, computation of some network parameters, and computation of grid coordinates. Namely, we focus on communication in symmetric radio network with nodes capable of collision detection. We explore how encoding information into detectable collisions can help to design fast deterministic communication protocols that work even in anonymous radio networks. The concluding part of Chapter 4 is devoted to communication in radio networks such that an underlying topology is a grid graph. In this case, the nodes are not capable of collision detection and are equipped with unique identifiers. However, a network node does not know its position in the grid. We show a time-efficient protocol accomplishing the broadcasting task and design a protocol which computes coordinates of all nodes in this grid network. Chapter 4 contains the following main results:

- for anonymous symmetric radio networks with collision detection capability
  - an asymptotically optimal protocol computing eccentricity of a distinguished node s in $O(ecc(s))$ rounds,
  - an asymptotically optimal protocol computing distance of each node from a distinguished node s in $O(ecc(s))$ rounds,
  - an asymptotically optimal protocol computing the maximum Max among integer values possessed by the networks nodes during $O(ecc(s) + \log Max)$ rounds, where s is a node that initiates the computation,
- for radio networks with the grid topology
  - an asymptotically optimal radio broadcasting protocol working in $O(ecc(s) + \log N)$ rounds where N is an upper-bound on the maximal identifier in the network,
  - a protocol for computation of grid coordinates in $O(ecc(s)+\log N)$ rounds.

Our results presented in this dissertation partially appeared in the following refereed papers:

[A] F. Galčík and G. Semanišin, *Centralized broadcasting in radio networks with k-degenerate reachability graphs*, ITAT 2006 Information Technologies - Applications and Theory, Bystrá dolina, Slovakia, 26.9.-1.10.2006, (2006), pp. 41-46.

[B] F. Galčík and G. Semanišin, *Maximum finding in the symmetric radio networks with collision detection*, SOFSEM 2007: Theory and Practice of Computer Science, Harrachov, Czech Republic, January 20-26, 2007, LNCS 4362 , (2007), pp. 284-294.

[C] F. Galčík, *On radio communication in grid networks*, ITAT 2007 Information Technologies - Applications and Theory, Poľana, Slovakia, 21.9.- 27.9.2007, (2007), pp. 47-54.

[D] F. Galčík, *Centralized communication in radio networks with strong interference*, SIROCCO 2008: 15th International Colloquium on Structural Information and Communication Complexity, Villars-sur-Ollon, Switzerland, June 17-20, 2008, LNCS 5058 , (2008), pp. 277-290.

[E] F. Galčík, *A note on the lower bound of centralized radio broadcasting for planar reachability graphs*, Discrete Applied Mathematics, Volume 157, Issue 4, (2009), pp. 853-857.

[F] F. Galčík, L. Gąsieniec, and A. Lingas, *Efficient broadcasting in known topology radio networks with long-range interference*, accepted to PODC 2009: 28th Annual ACM SIGACT-SIGOPS Symposium on Principles of Distributed Computing, Canada, August 10-12, 2009.

Chapter 2

Centralized Broadcasting in Radio Networks

Centralized broadcasting refers to the broadcasting problem in the special setting when the network topology is known to all network nodes. I.e., the nodes have a labelled copy of an underlying reachability graph as a part of their initial knowledge. Due to known network topology, a schedule of transmissions that realizes the broadcasting can be precomputed by all nodes in advance. Such a schedule is called a *radio broadcasting schedule*. In this setting, the broadcasting process can be seen also as a process controlled by a central controller. As we can notice, the only difficulty arises: proper (off-line) dealing with the broadcast and interference character of the radio communication medium. From a practical point of view, the centralized broadcasting can be applied in radio networks with a stable and known network topology (e.g., in static sensor networks).

The main considered efficiency criterion (in most of the research papers) is the time of the task completion. For a given reachability graph and a source node, the goal is to design an algorithm generating a radio broadcasting schedule that uses as small number of communication rounds as possible. It is known that computation of an optimal radio broadcasting schedule for arbitrary reachability graphs is NP-hard, even if the underlying reachability graph is embedded in the plane [8, 65]. Therefore, the main challenge is design of polynomial-time algorithms generating radio broadcasting schedules with approximately optimal number of communication rounds (length of the schedule). Note that other complexity measures like energy consumption have been investigated in the context of centralized broadcasting as well.

In this chapter, we focus mainly on the broadcasting problem in radio networks in the case when an underlying reachability graph is somehow restricted, e.g., it is planar, k-degenerate, or it has a bounded maximum degree.

Intuitively, such assumptions allow to produce more efficient radio broadcasting schedules. Since, the network topology is known in advance, the nodes can use a better algorithm in the case when reachability graph fulfills given restrictions that are known in advance or are algorithmically effectively recognizable. The results show that there are graph topologies which are more broadcast friendly (allow more efficient broadcasting) than the others. When building a static radio network, we can utilize this fact, and choose locations of devices and directions of antennas in such a way that the resulting reachability graph will have a topology allowing efficient broadcasting.

2.1 Introduction

We start with a formal definition of a radio broadcasting schedule.

Definition 2.1.1. *Let $G = (V, E)$ be a directed graph and $R \subseteq V$ be a subset of nodes. A set of nodes informed by R, denoted as $I(R)$, is the set*

$$I(R) = \{v \in V \mid \textit{there exists the unique } x \in R \, \textit{such that } v \in N(x)\}.$$

For a singleton set $R = \{x\}$, $I(R) = I(\{x\}) = N(x)$.

Definition 2.1.2. *Let $G = (V, E)$ be a directed graph. A sequence of sets $\Pi = (R_1, \ldots, R_q)$ is called a radio broadcasting schedule with respect to the reachability graph G and a source $s \in V$ if and only if the following holds:*

(1) $R_i \subseteq V$, for every $i = 1, 2, \ldots, q$;

(2) $R_1 = \{s\}$;

(3) $R_{i+1} \subseteq \bigcup_{j=1}^{i} I(R_j)$, for every $i = 1, 2, \ldots, q - 1$;

(4) $V = \bigcup_{j=1}^{q} I(R_j)$.

The length of the schedule Π is $q = |\Pi|$.

Observe that the property (2) of the previous definition claims that the source is the only transmitting node in the first round. The property (3) implies that only informed nodes can transmit. Finally, the property (4) claims that all nodes become informed after the broadcasting schedule is executed, i.e., every node receives a message in at least one round of the schedule.

Note that it is assumed that there is a directed path from the source s to any other network node in the graph G. If the transmission power of all nodes is the same, the resulting reachability graph is symmetric. The assumption about equal transmission powers was adopted in almost all papers. As we will see later, this assumption allows powerful use of pipelining approach in the communication process. In what follows, we shall assume that considered reachability graphs are undirected (unless stated otherwise).

Two broadcasting strategies appear in the literature. The first strategy is based on flooding. At any round, the set of network nodes can be partitioned into a set of informed nodes and a set of uninformed nodes. Each of those sets contains a subset of frontier nodes that have a neighbor in the other set. The frontier nodes form a bipartite graph where one its part consists of informed nodes and the other of uninformed nodes. Typically, only a small fraction of uninformed frontier nodes can be informed in one round due to interference. Hence, a natural subgoal is to design an algorithm which for a given bipartite graph produces a schedule of transmissions informing all nodes of the uninformed part. We shall refer to this problem as an *information dissemination in bipartite graphs (radio networks).* The key idea of the second broadcasting strategy is very fast delivery of information (source message) to close vicinity of each network node. This is done utilizing a (tree-like) communication subnetwork that allows fast spreading of information and pipelining. All known algorithms generating fast broadcasting schedules are based on combinations of those two strategies and their modifications. Transmissions that follows from information dissemination in a bipartite graph are mostly referred to as *slow transmissions.* Transmissions realizing the latter strategy are referred to as *fast transmissions.*

2.1.1 Related Work

The study of deterministic centralized broadcasting in the *packet radio network model* introduced by Chlamtac and Kutten in [8] was initiated by Chlamtac and Weinstein in [9]. The authors presented a polynomial-time algorithm producing radio broadcasting schedules with length $O(D \cdot \log^2 n/D)$ rounds. Their result was complemented by Alon *et al.* in [1]. In this work, the authors proved $\Omega(\log^2 n)$ lower bound for a family of reachability graphs with the radius 2 using a probabilistic argument. This result together with the trivial lower bound D provides a lower bound $D + \Omega(\log^2 n)$ rounds on the time of broadcasting in arbitrary radio networks. Designing an algorithm generating schedules of the length that matches this lower bound became one of important challenges in this field.

In [2], Bar-Yehuda *et al.* presented a randomized algorithm producing schedules of expected length $O(D \cdot \log n + \log^2 n)$. Hence, it was shown that a radio broadcasting schedule of the length $O(D \cdot \log n + \log^2 n)$ always exists. Recently, a deterministic algorithm producing schedules of the length $O(D \cdot \log n + \log^2 n)$ was designed by Kowalski and Pelc in [48]. Utilizing this result, deterministic schemes presented in [27, 31] gave deterministic algorithms for centralized radio broadcasting. Particularly in [31], Gaber and Mansour showed a method in which underlying reachability graph is partitioned into clusters with smaller diameter. Broadcastings in cluster graph, that forms a tree-like structure, and in each cluster are performed applying known broadcasting algorithms. Utilizing algorithm from [48] they obtained an algorithm producing a schedule of the length $O(D+\log^5 n)$. This method was improved by Elkin and Kortsarz [27]. An algorithm that produces schedules of length $O(D+\log^4 n)$ was shown. Recently, these results were further improved by Gąsieniec *et al.* [36]. The authors proposed an algorithm producing schedules of the length $D + O(\log^3 n)$ and a randomized algorithm producing schedules of expected length $D + O(\log^2 n)$. Their algorithms are based on construction of a ranked tree called a *gossiping-broadcasting spanning tree.* Finally, Kowalski and Pelc [49] presented a polynomial-time deterministic algorithm generating schedules of length $O(D + \log^2 n)$. The non-approximability result from [26] implies that schedules constructed by this algorithm are asymptotically optimal unless $NP \subset BPTIME(n^{O(\log \log n)})$. However, for large D, algorithm presented by Cicalese *et al.* in [17] with the broadcasting time $D + O(\frac{\log^3 n}{\log \log n})$ rounds is faster than $O(D + \log^2 n)$-time algorithm from [49].

Elsässer and Gąsieniec studied the broadcasting problem in random radio networks. In [28], they considered a random graph $G_p = (V, E)$ with expected average degree $d = n \cdot p$, where $n = |V|$ and p is the probability that two nodes are connected by an edge. In this setting, they showed how to perform centralized broadcasting in G_p asymptotically optimal in time $O(\ln n/\ln d + \ln d)$ rounds with probability $1 - o(1/n)$.

Time efficient broadcasting with restriction on energy consumption (the number of transmissions per node) was first investigated in unknown graphs by Berenbrink *et al.* in [3] and later in known graphs by Gąsieniec *et al.* in [33]. In particular, the authors of [33] presented an algorithm generating 1-shot (each node is allowed to transmit at most once) broadcasting schedules of lengths $O(\sqrt{n})$ and $D + O(\sqrt{n} \cdot \log n)$ for bipartite and arbitrary graphs respectively.

A detailed survey of know results concerning time-efficient broadcasting can be found in Pelc's review [61] or in the recent Peleg's review [62].

2.2 General Schema for Fast Centralized Broadcasting

Gąsieniec *et al.* [36] presented an algorithm generating radio broadcasting schedules of the length $D + O(\log^3 n)$ for arbitrary reachability graphs. Modifications of the approach introduced in their work was later used in other algorithms [17, 49, 53]. This approach can be seen (reformulated) also as a general schema. The schema expects as an input an algorithm for information dissemination in bipartite graphs that are subgraphs of a considered class of input reachability graphs. Utilizing a given algorithm, it generates fast radio broadcasting schedules. We use the schema and its properties for designing algorithms producing fast radio broadcasting schedules for radio networks with a specific graph topology (planar, k-degenerate).

The algorithm (schema) consists of two parts. At first, a sort of ranked spanning BFS tree is computed. The tree is called a *gathering-broadcasting spanning tree* (GBST). Thanks to properly defined ranks of nodes, its edges form a communication subnetwork that allows fast information dissemination. After a GBST of the network is computed, a schedule of transmissions (fast and slow transmissions) is generated.

2.2.1 Gathering-Broadcasting Spanning Tree

The crucial notion during construction of a GBST is its ranking. The standard definition of the rank of nodes in a rooted tree is adopted. This ranking was investigated and used in different contexts (see, e.g., definition of *Strahler number* of binary trees used in hydro-geology [67] or its use in computer science [68]). Given a (BFS) tree, the rank of a node v is defined as follows:

- if v is a leaf, $rank(v) = 0$,
- if v is an internal node, let r_{max} be a maximal rank among ranks of its children nodes. If v has at least 2 children with the rank r_{max}, then $rank(v) = r_{max} + 1$. Otherwise, $rank(v) = r_{max}$.

Lemma 2.2.1. *The greatest rank is assigned to the root and its value is at most $\lfloor \log n \rfloor$ where n is the number of nodes. Each simple path along tree edges from the root to any other node forms a non-increasing sequence of ranks.*

Let L_i be a set of nodes with the distance i from the root s of the tree, i.e., $L_i = \{v | d(s, v) = i\}$. Ranks of nodes provide a partition of nodes into *rank sets* $R_i = \{v | rank(v) = i\}$, where $0 \leq i \leq \log n$. Utilizing those definitions, we define *fast transmission* and *slow transmission sets*. The fast transmission set F_i^k is defined as $F_i^k = \{v | v \in L_k \cap R_i \wedge parent(v) \in R_i\}$. Further, we define $F_i = \bigcup_{k=1}^{ecc(s)} F_i^k$ and $F = \bigcup_{i=0}^{\lfloor \log n \rfloor} F_i$. I.e., F is a set of nodes that have the same ranks as its parents in the higher (closer to the root) layer. The *slow transmission set* is defined as $S_i^k = \{v | v \in L_k \cap R_i \wedge parent(v) \in R_j \wedge j > i\}$. Similarly, $S_i = \bigcup_{k=1}^{ecc(s)} S_i^k$ and $S = \bigcup_{i=0}^{\lfloor \log n \rfloor} S_i$. I.e., a node belongs to the set S in the case when its parent has different (larger) rank.

For a given arbitrary graph $G = (V, E)$, a gathering-broadcasting spanning tree is any BFS spanning tree T of the graph G such that

- T is rooted at the source node s,
- T is ranked,
- all nodes in F_i^k of T can transmit their messages to their parents simultaneously without any collision, for all $0 \leq i \leq \log n$ and $1 \leq k \leq ecc(s)$, and
- all nodes in F_i^k of T can receive messages transmitted simultaneously by their parents without any collision, for all $0 \leq i \leq \log n$ and $1 \leq k \leq ecc(s)$.

Observe that the definition of GBST implies that for any pair of nodes $u, v \in F_i^k$ it holds that

- $(u, parent(u)) \in E$, $(v, parent(v)) \in E$, and
- $(u, parent(v)) \notin E$, $(v, parent(u)) \notin E$.

In [36], the authors showed the following theorem.

Theorem 2.2.2 ([36])**.** *For a given graph $G = (V, E)$, a gathering-broadcasting spanning tree can be constructed in time $O(n^2 \cdot \log n)$.*

2.2.2 Broadcasting Schema

A GBST allows simultaneous non-conflicting transmissions for certain nodes. This is utilized in the broadcasting algorithm presented in [36]. In this section, we show generalization of this algorithm that is more suitable for radio networks whose reachability graphs belong to a graph class defined by a *hereditary property*, see [6]. A hereditary property is a graph property satisfying the condition that if a graph has a given property then all its subgraphs have this property. Typical examples of hereditary properties are planarity, to be a k-degenerate graph, to have the maximum degree smaller than Δ.

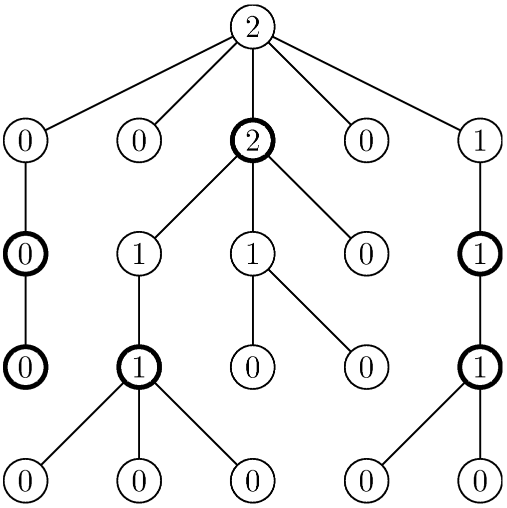

Figure 2.1: Example of a ranked BFS tree with marked nodes of the set F

As we will see later, the generalized algorithm is useful in other situations as well.

After a GBST is constructed, we define a set of slow bipartite graphs $\{G_i^k | 0 \leq i \leq \lfloor \log n \rfloor \wedge 1 \leq k \leq ecc(s)\}$. Let us define sets $A_i^k = L_{k-1} \cap R_i$ and $B_i^k = \{v | v \in L_k \wedge parent(v) \in A_i^k \wedge rank(v) < i\}$. I.e., the set A_i^k is a set of nodes with the rank i in the distance $k-1$ from the source. The set B_i^k is a set of all neighbor of nodes in A_i^k that are in the distance k from the source and their ranks are strictly less than i. The graph $G_i^k = (V_i^k, E_i^k)$ is defined as

- $V_i^k = A_i^k \cup B_i^k$, and
- $E_i^k = \{(u,v) | (u,v) \in E \wedge u \in A_i^k \wedge v \in B_i^k\}$.

Each graph G_i^k is a bipartite graphs. It is also a subgraph of the graph G with parts of the bipartition in two neighboring BFS layers. Note that if the graph G belongs to a graph class $\mathcal{G}$ defined by a hereditary property, the graph G_i^k belongs to the graph class $\mathcal{G}$ as well. Note also that some graphs G_i^k can be empty or can have empty B-part. Definition of sets A_i^k implies that each node belongs to exactly one set A_i^k and at most one set B_i^k.

Let $\mathcal{A}_b$ be an algorithm that for a given bipartite graph of a class $\mathcal{G}$ generates a schedule of transmissions realizing the information dissemination in bipartite graphs. Denote as $time_b(G)$ the maximal length of a schedule produced by $\mathcal{A}_b$ for an arbitrary bipartite subgraph of the graph G. Let Π_i^k

be a schedule generated by the algorithm $\mathcal{A}_b$ for the bipartite graph G_i^k with the set A_i^k as the informed part. Obviously, $|\Pi_i^k| \leq time_b(G)$.

Now, we describe a schedule of transmissions realizing the radio broadcasting in the graph G with the source s. Let $v \in L_k$ be a node with the rank i, i.e., $v \in A_i^{k+1}$. Further, let $\{t_1, t_2, \ldots, t_q\}$ be a set of rounds in which the node v transmits in the schedule Π_i^{k+1}. Finally, let us denote $t^F(v) = k + 3 \cdot (time_b(G) + 2) \cdot (rank(s) - i)$. The node v transmits the source message in rounds

- $t^F(v)$, if and only if the node v has a tree child in F_i^{k+1}, and
- $t^F(v) + 3 \cdot t_j$, for all $1 \leq j \leq q$.

Transmission of the node v in the round $t^F(v)$ is referred to as a *fast transmission.* All other transmissions of the node v are referred to as *slow transmissions.* Now, we show some properties of the designed schedule.

Lemma 2.2.3. *Fast and slow transmissions do not interfere each other.*

Proof. Assume that nodes $u \in L_{k_u}$ and $v \in L_{k_v}$ transmits simultaneously. Further, let us assume that the node u realizes a fast transmission and the node v a slow transmission. According to the scheduling principle, it holds that $t^F(u) = t^F(v) + 3 \cdot t_j$ for an appropriate value t_j, $1 \leq t_j \leq time_b(G)$. Due to definitions of $t^F(u)$ and $t^F(v)$, this equality implies that $|k_u - k_v| \geq 3$. Hence, the nodes u and v do not share a common neighbor and their simultaneous transmissions do not interfere. □

Lemma 2.2.4. *Slow transmissions in different bipartite graph $G_{i_1}^{k_1}$ and $G_{i_2}^{k_2}$ do not interfere each other.*

Proof. Assume contrarily that two slow transmissions interfere. It follows that there are two nodes $u \in A_{i_1}^{k_1}$ and $v \in A_{i_2}^{k_2}$ sharing a common neighbor and transmitting simultaneously. I.e., there are values t_{j_1} and t_{j_2} ($1 \leq t_{j_1}, t_{j_2} \leq time_b(G)$) such that $t^F(u) + 3 \cdot t_{j_1} = t^F(v) + 3 \cdot t_{j_2}$. Without loss of generality, let us assume that $i_1 \geq i_2$. The previous equality implies that $k_1 - k_2 = 3 \cdot ((time_b(G) + 2) \cdot (i_1 - i_2) + t_{j_2} - t_{j_1})$. It can be easily shown by a case analysis that either $|k_1 - k_2| \geq 3$ or $k_1 = k_2$, $i_1 = i_2$, $t_{j_1} = t_{j_2}$. It follows that either the nodes u and v do not share a common neighbor or $G_{i_1}^{k_1} = G_{i_2}^{k_2}$. □

Lemma 2.2.5. *Any node in F receives the fast transmission from its parent without any collision.*

Proof. Let us consider a node $u \in F_i^k \subseteq F$. Since $u \in F$, it holds that $rank(parent(u)) = i$ and $parent(u) \in L_{k-1}$. Assume contrarily that a collision occurs during the fast transmission of the node $parent(u)$, i.e., in the round $t^F(parent(u))$. If follows that there is a neighbor $v \in L_j$ ($v \neq parent(u)$) of the node u that transmits in the round $t^F(parent(u))$. Due to Lemma 2.2.3, fast and slow transmissions do not interfere. Therefore, the node v realizes its fast transmission in the round $t^F(parent(u))$, i.e., $t^F(v) = t^F(parent(u))$. The last equality implies that either $|j-k| \geq 6$, or $j = k-1$ and $rank(v) = i$. The former case contradicts directly the assumption that v is a neighbor of u. In the latter case, we get $v \in L_{k-1} \cap R_i$. Since the node v transmits, it follows that there exists a node $w \in L_k$ such that $parent(w) = v$ and $w \in F_i^k$ (see the condition for fast transmissions). Finally, $u, w \in F_i^k$ and the last property of GBST imply that $(v,u) = (parent(w), u) \notin E$ contradicting again that v is a neighbor of u. □

Lemma 2.2.6. *Each node v receives the source message before the round $t^F(v)$.*

Proof. The proof is done by induction on the layer number of a node. For the source, the claim trivially holds. Now, we prove the claim for a node $v \in L_k \cap R_i$ under assumption that induction hypothesis holds. It holds that either $v \in F_i^k$ or $v \in S_i^k$. In the both cases, we get $parent(v) \in L_{k-1}$. The induction hypothesis implies that the node $parent(v)$ is informed before the round $t^F(parent(v))$. Hence, the node $parent(v)$ can realize all its scheduled fast and slow transmissions. Since $rank(v) \leq rank(parent(v))$, it follows that $t^F(parent(v)) < t^F(v)$.

In the former case ($v \in F_i^k$), the fast transmission of the node $parent(v)$ in the round $t^F(parent(v)) < t^F(v)$ is successfully received by the node v due to Lemma 2.2.5. Therefore, the node v is informed before the round $t^F(v)$.

In the latter case ($v \in S_i^k$), it follows that $parent(v) \in L_{k-1} \cap R_l$, where $l > i$. Note that the node v belongs to the uninformed part B_l^k of the graph G_l^k. Note also that the informed part A_l^k is a subset of $L_{k-1} \cap R_l$ and $parent(v) \in A_l^k$. For any node $u \in A_l^k$, the induction hypothesis implies that it is informed before the round $t^F(u)$. Moreover, for any pair of nodes $u, w \in A_l^k$ it holds that $t^F(u) = t^F(v)$. Since $l > i$, for an arbitrary node $u \in A_l^k$ transmitting in rounds $\{t_1, \ldots, t_q\}$ according to schedule Π_l^k, and for any j, $1 \leq j \leq q$, we get $t^F(u) + 3 \cdot t_j \leq t^F(u) + 3 \cdot time_b(G) = k - 1 + 3 \cdot (time_b(G) + 2) \cdot (rank(s) - l) + 3 \cdot time_b(G) < k - 1 + 3 \cdot (time_b(G) + 2) \cdot (rank(s) - i) < t^F(v)$. I.e., all nodes in A_l^k realize their slow transmissions before the round $t^F(v)$. Due to Lemma 2.2.3 and Lemma 2.2.4, slow transmissions according to Π_l^k in G_l^k do not interfere with other transmissions. Since the schedule Π_l^k is a schedule for information dissemination in the bipartite graph G_l^k, the node

v is informed after all slow transmissions according to the schedule Π_l^k are done. □

Theorem 2.2.7. *For a given arbitrary graph $G = (V, E)$, a radio broadcasting schedule with the length $D + O(time_b(G) \cdot \log n)$ can be generated in polynomial time.*

Proof. Lemma 2.2.6 claims that each network node v is informed by the constructed schedule in the round $t^{F}(v)$ at the latest. Due to Lemma 2.2.1, $rank(s) \leq \log n$. Since $k \leq ecc(s)$, it follows that $t^{F}(v) \leq ecc(s) + 3 \cdot (time_b(G) + 2) \cdot \log n = D + O(time_b(G) \cdot \log n)$. The polynomial-time constructibility of the schedule follows from Theorem 2.2.2 showing that a GBST of a given arbitrary graph G can be constructed in polynomial time. □

Observe that the presented schedule has another interesting feature. If a given algorithm $\mathcal{A}_b$ produces k-shot schedules of transmissions, the resulting schedule is $(k + 1)$-shot broadcasting schedule. I.e., this schema can be used for construction of energy efficient broadcasting schedules, provided that schedules generated by the algorithm $\mathcal{A}_b$ are energy efficient.

We conclude this section with corollaries of Theorem 2.2.7. We start with the corollary that shows the main result in [36].

Corollary 2.2.8. *There exists a deterministic polynomial-time algorithm that generates, for any n-node radio network with diameter D, a broadcasting schedule with length $D + O(\log^3 n)$.*

Proof. The algorithms in [9, 48] generate transmission schedules for information dissemination in bipartite graph with length $O(\log^2 n)$. Using one of those algorithms as the algorithm $\mathcal{A}_b$ in the schema from Theorem 2.2.7, we obtain an algorithm generating radio-broadcasting schedules with length $D + O(\log^3 n)$. □

The authors in [18] presented an algorithm producing schedules with length $O(\log \Delta \cdot \log n)$ for information dissemination in n-node bipartite networks with the maximum degree Δ. Generalization of their algorithm for radio networks with long-range interference is presented in Chapter 3 (Theorem 3.2.6). The algorithm in [18] and Theorem 2.2.7 imply the following corollary.

Corollary 2.2.9. *There exists a deterministic polynomial-time algorithm that generates a radio broadcasting schedule with length $D + O(\log \Delta \cdot \log^2 n)$ for any n-node radio network with diameter D and the maximum degree Δ.*

In [37], the following algorithm for information dissemination in bipartite networks was presented. The algorithm is based on *minimal covering sets.*

Definition 2.2.10. *Given a bipartite graph $G = (A \cup B, E)$, a minimal covering set is such a set C, $C \subseteq A$, that every node in B is connected to a node in C and removal of any node from C does not preserve this property.*

Lemma 2.2.11 ([37])**.** *Let $G = (A \cup B, E)$ be a bipartite graph with the informed part A and the uninformed part B. Let Δ be the maximum degree of a node in the graph G. If deg$(v) \geq 1$ for all $v \in B$, all nodes in B can be informed by transmissions of nodes in A during at most Δ rounds.*

Proof. Let $A_0 \subseteq A$ be a minimal covering set of G. Let us denote $B_0 = \{u | u \in B \wedge |N(u) \cap A_0| = 1\}$. Definition of the minimal covering set implies that each node $v \in A_0$ has such a neighbor $u \in B$ that $N(u) \cap A_0 = \{v\}$, i.e., the node v is the only neighbor of the node u in the set A_0. Indeed, if for each neighbor $x \in N(y)$ of a node $y \in A_0$ it holds that $N(x) \cap A_0 \supset \{y\}$, then the set $A_0 \setminus \{y\}$ is covering set as well. This contradicts minimality of the covering set A_0. Obviously, after a simultaneous transmission of all nodes in A_0, all nodes in B_0 receive a message without any collision. Therefore, we can remove all nodes in B_0 from B. Since $|N(v) \cap B_0| \geq 1$ for each node $v \in A_0$, it holds that degree of each node in A_0 is decreased at least by one after removal of B_0 from B. Repeating this process at most Δ times, all nodes in B become informed. Finally, note that a minimal covering set of a given set can be easily found in linear time. □

Utilizing algorithm from Lemma 2.2.11 in the schema from Theorem 2.2.7, we get another corollary.

Corollary 2.2.12. *There exists a deterministic polynomial-time algorithm that generates a radio broadcasting schedule with length $D + O(\Delta \cdot \log n)$ for any n-node radio network with diameter D and the maximum degree Δ.*

In [36], Gąsieniec *et al.* showed also an algorithm generating gossiping schedules with length $2 \cdot D + \Delta \cdot \log n + O(\log^3 n)$. The algorithm consists of two parts: gathering and broadcasting part. At first, all messages are gathered in the central node applying a schedule with the length $D + \Delta \cdot \log n$ rounds. Further, gathered messages (one combined message) is broadcasted in the network during next $D + O(\log^3 n)$ rounds. However, replacing the broadcasting schedule of length $D + O(\log^3 n)$ with a broadcasting schedule of length $D + O(\Delta \cdot \log n)$ generated by the algorithm in Corollary 2.2.12, we get a gossiping schedule of length $2 \cdot D + O(\Delta \cdot \log n)$.

Corollary 2.2.13. *There exists a deterministic polynomial-time algorithm that generates a gossiping schedule of length $2 \cdot D + O(\Delta \cdot \log n)$ for any n-node radio network with diameter D and the maximum degree Δ.*

2.3 Broadcasting in Planar Graphs

Intuitively, if the reachability graph of a radio network belongs to a specific class of graphs, we can utilize this fact and design algorithms producing more efficient (faster) radio broadcasting schedules. If graphs of this class are effectively recognizable by an algorithm, whenever we recognize such a reachability graph, we can replace algorithm for arbitrary graphs by an algorithm for graphs in this class. Moreover, when building a static radio network, we may try to choose locations of devices and directions of antennas in such a way that the resulting reachability graph will have a topology allowing efficient broadcasting.

One of those specific classes is a class of planar graphs. Planarity of a graph can be verified in polynomial time [41]. Recall that a graph is planar if and only if it can be embedded in the plane, i.e., it can be drawn on the plane in such a way that its edges intersect only at their endpoints.

In this section, we show that planarity is a helpful property for centralized communication in radio networks, i.e., we show that planar reachability graphs are radio broadcasting friendly topology in context of centralized communication. Surprisingly, this is not true in the fully distributed setting. Indeed, the graphs showing best known lower bounds [46] for radio broadcasting in unknown symmetric radio networks are planar.

2.3.1 Upper Bounds

The first result concerning the broadcasting problem in radio networks with planar reachability graphs was given by Elkin and Kortsarz in [27]. They presented an algorithm generating radio broadcasting schedules of length $D + O(\log^3 n)$. This result was later improved independently by Gąsieniec *et al.* in [36] and by Galčík in [32]. Both works showed a polynomial-time algorithm that for a given planar reachability graph produces a broadcasting schedule of the length $3 \cdot D$. The key subroutine of those algorithms is a subroutine for information dissemination problem in a bipartite graph $G_b = (A \cup B, E_b)$ that is a subgraph of a planar graph $G = (V, E)$ and

- all nodes of the informed part A are in the same distance i from the source ($A \subseteq L_i$), and

- all nodes of the uninformed part B are in the same distance $i+1$ from the source ($B \subseteq L_{i+1}$).

Hence, we consider only subgraphs of bipartite inter-layer graphs.

Computation of transmission schedule starts with preprocessing of an input graph G as follows.

1. We remove all nodes in the distance at least i from the source, except nodes in the set $A \cup B$.
2. We remove all edges with both endpoints in the set A or in the set B.
3. We contract all edges, except edges with one endpoint in the set A.

Note that node removal, edge removal, and edge contraction preserve planarity. Preprocessing is concluded by use of stereographic projection. We fix a planar embedding of the graph G such that the source node s is located on the outer face. From the computational point of view, the result in [16] implies that we can assume that the planar embedding of the graph G is a straight-line drawing (it can be computed by a polynomial-time algorithm). If a node $b \in B$ has exactly one neighbor in A, any transmission of its unique neighbor in A delivers the source message to the node b. Since we shall guaranty that all nodes in A transmit at least once, we can remove all nodes in B with exactly one neighbor and consider the resulting graph. Moreover, if some nodes in A have no neighbor in B, they can be removed as well. Indeed, construction of a schedule for such nodes is trivial.

Observe, that the resulting graph G consists of two types of edges: a set of edges joining the source s and a node in the set A, and a set of edges joining a node in A and a node in B. Since the source s lies on the outer face, it follows that at least one node from B lies on the border of the outer face. Let $B' \neq \emptyset$ be a set of all nodes from B that lie on the border of the outer face. Due to preprocessing, each node in B has at least two neighbors in A and each node in A has at least one neighbor in B. Therefore, each node, except the source s, appears on the outer-face border at most once. For a node $b \in B$ and two its neighbors $a_L, a_R \in N(b) \subseteq A$, let $R(b, a_L, a_R)$ be a closed region bounded by the edges (s, a_L), (a_L, b), (b, a_R), and (a_R, s). Let $a_L^{(b_i)} \in A$ and $a_R^{(b_i)} \in A$ be a predecessor and a successor of a node b_i on the outer-face border respectively. Notice that a union of regions $\bigcup_{b \in B'} R(b, a_L^{(b)}, a_R^{(b)}, s)$ is a complement of the outer face. Moreover, planarity of the graph embedding implies that for any two different nodes b_i and b_j the intersection of regions $R(b_i, a_L^{(b_i)}, a_R^{(b_i)})$ and $R(b_j, a_L^{(b_j)}, a_R^{(b_j)}, s)$ contains the source s and at most one node from the set $\{a_L^{(b_i)}, a_R^{(b_i)}, a_L^{(b_j)}, a_R^{(b_j)}\}$. Notice also, that the information dissemination process among nodes inside $R(b_i, a_L^{(b_i)}, a_R^{(b_i)})$ can be treated individually with exemption of the nodes $a_L^{(b_i)}$ and $a_R^{(b_i)}$.

Now, we show how to assign a transmission round $round(a) \in \{1, 2, 3\}$ to each node $a \in A$. I.e., each node of A transmits in exactly one of three rounds in the generated transmission schedule for information dissemination in bipartite graphs. The schedule is computed in the following steps.

1. Let $\{b_1, b_2, \ldots, b_r\}$ be a sequence of nodes in B' in order in which they appear on the outer-face border. Assign $round(a_L^{(b_i)}) := 1$ and $round(a_R^{(b_i)}) := 2$ in case that $i \bmod 2 = 0$. Otherwise ($i \bmod 2 = 1$), assign $round(a_L^{(b_i)}) := 2$ and $round(a_R^{(b_i)}) := 1$.

2. Compute assignments for internal nodes of the region $R(b_i, a_L^{(b_i)}, a_R^{(b_i)})$ separately for each b_i utilizing the procedure *Assign-Inside-Region* (see Lemma 2.3.1).

3. Assign $round(v) := 1$ for each node $v \in A$ with unassigned transmission round. Note that in this step, we assign transmission rounds to nodes in the original set A, i.e., to all nodes of the set A which have been removed from the graph G in the process of assignment computation.

Since we assign transmission rounds in one direction of outer-face border traversal, the resulting assignment assigns at most one transmission round to each node in A. Moreover, the resulting assignment satisfies the property that each node in B' receives the source message in two transmission rounds. The third step does not introduce collisions, since we remove from the graph G only such nodes in A that have no neighbors in B or all their neighbors in B have exactly one neighbor in A. In the remaining, we show how to recursively compute transmission rounds for nodes inside a region $R(b_i, a_L^{(b_i)}, a_R^{(b_i)})$ such that all nodes in $R(b_i, a_L^{(b_i)}, a_R^{(b_i)}) \cap B$ receive the source message. Recall again, that the planar embedding of the graph G implies that there are no edges crossing the border of the region $R(b_i, a_L^{(b_i)}, a_R^{(b_i)})$. I.e., computing transmission rounds for nodes in $R(b_i, a_L^{(b_i)}, a_R^{(b_i)}) \cap A$, we have to take into consideration only transmission rounds assigned to nodes $a_L^{(b_i)}$ and $a_R^{(b_i)}$. Observe also, that if transmission rounds assigned to nodes inside $R(b_i, a_L^{(b_i)}, a_R^{(b_i)})$, i.e., to nodes $R(b_i, a_L^{(b_i)}, a_R^{(b_i)}) \cap A$, inform all nodes of the set $R(b_i, a_L^{(b_i)}, a_R^{(b_i)}) \cap B$, then the computed schedule realizes information dissemination in the bipartite graph $G_b = (A \cup B, E_b)$.

Let $R(b, a_L^{(b)}, a_R^{(b)})$ be a region defined by the nodes $b, a_L^{(b)}, a_R^{(b)}$, and s such that

- the nodes $a_L^{(b)}$, $a_R^{(b)}$ have assigned different transmission rounds, i.e., $round(a_L^{(b)}) \neq round(a_R^{(b)})$, and
- no node in the set $R(b, a_L^{(b)}, a_R^{(b)}) \cap A$, except nodes $a_L^{(b)}$ and $a_R^{(b)}$, has assigned transmission round, and
- all neighbors of b are inside of $R(b, a_L^{(b)}, a_R^{(b)})$, i.e., $N(b) \subset R(b, a_L^{(b)}, a_R^{(b)})$.

Note that regions considered in the second step of the assignment computation satisfy this property.

Lemma 2.3.1. *We can assign to each node of $R(b, a_L^{(b)}, a_R^{(b)}) \cap A$ one of 3 transmission rounds in such a way that all nodes in $R(b, a_L^{(b)}, a_R^{(b)}) \cap B$ receive the source message. Computation of the assignment is realized by a polynomial-time algorithm called Assign-Inside-Region.*

Proof. Let $(a_L^{(b)} = a_1, a_2, \ldots, a_p = a_R^{(b)})$ be a clockwise (or counterclockwise) ordering of nodes in the set $N(b)$. Hence, $round(a_1) \in \{1, 2, 3\}$, $round(a_p) \in \{1, 2, 3\}$, $round(a_1) \neq round(a_p)$, and $round(a_i)$ is unassigned for any i, $1 < i < p$. Let us denote $t_1 = round(a_1)$, $t_2 = round(a_p)$, and $t_3 = \{1, 2, 3\} \setminus \{t_1, t_2\}$. We set the transmission round $round(a_i)$ of a node a_i to t_3 in case that $i \bmod 2 = 0$, and to t_2 otherwise. Since $N(b) \subset R(b, a_L^{(b)}, a_R^{(b)})$, defined assignments of transmission rounds guaranty that the node b is informed in the round t_1 by transmission of the node $a_L^{(b)}$. After this assignment, transmission rounds (informing the node b) of all neighbors of the node b are determined and fixed. If there are no other nodes from the set B inside the region $R(b, a_L^{(b)}, a_R^{(b)})$, i.e., $R(b, a_L^{(b)}, a_R^{(b)}) \cap B = \{b\}$, then we are done. The computed assignment realizes information dissemination inside the region $R(b, a_L^{(b)}, a_R^{(b)})$.

In the complementary case when $|R(b, a_L^{(b)}, a_R^{(b)}) \cap B| > 1$, we proceed in the following way. Due to planarity and bipartition, the region $R(b, a_L^{(b)}, a_R^{(b)})$ is a union of disjoint (with exception of border nodes and edges) subregions $R(b, a_i, a_{i+1})$ for i, $1 \leq i < p$. Let us consider each subregion $R(b, a_i, a_{i+1})$ separately. Note that no node from the set A (except a_i and a_{i+1}) located inside this region has assigned transmission round, and $round(a_i) \neq round(a_{i+1})$. Let $G(b, a_i, a_{i+1})$ be a graph induced by the network nodes inside the region $R(b, a_i, a_{i+1})$ with exception of the node b. Removing the node b and all nodes (including incident edges) outside the region $R(b, a_i, a_{i+1})$, we get a planar embedding of the graph $G(b, a_i, a_{i+1})$. Later, removing all nodes in B with exactly one neighbor in A, we remove from the graph all nodes which are informed by any transmission of their unique neighbors in A. Finally, removing all nodes in A without a neighbor in B, we eliminate

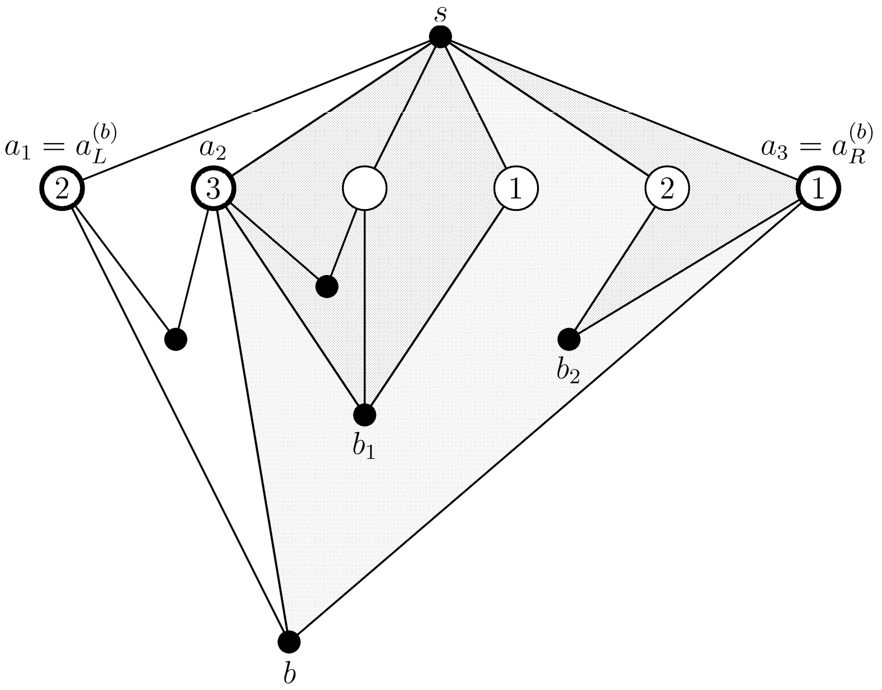

Figure 2.2: A region $R(b, a_L^{(b)}, a_R^{(b)})$ and processing of subregion $R(b, a_2, a_3)$

nodes whose transmissions do not cause interference. Similarly as above, the source s lies on the outer-face border. Let $(b_1, b_2, \ldots, b_q)$ be a sequence of nodes from B in the order in which they appear on the outer-face border of the considered planar embedding of $G(b, a_i, a_{i+1})$. Let $a_L^{(b_i)}$ and $a_R^{(b_i)}$ be the predecessor and the successor of the node b_i on this outer-face border respectively. Note that at most two nodes on this outer-face border have already assigned transmission rounds. Moreover, if there are two such nodes, they have assigned different transmission rounds. Let t_1 be a transmission round assigned to one of those nodes (if no node has assigned transmission round, t_1 can be any transmission round). Further, let t_3 be an unassigned transmission round. Since each node, except the source s, appears on the outer-face border at most once, starting from a node with assigned transmission round t_1 (if no such node exists, we start in any node of the set A lying on the border) and traversing the outer-face border, we can assign transmission rounds t_1 and t_3 to all nodes of set A on this border in such a way that for an arbitrary node $b_i \in B$ on the outer-face border it holds $round(a_L^{(b_i)}) \neq round(a_R^{(b_i)})$. Finally observe that complement of the outer face is a union of disjoint regions $\bigcup_{1 \leq i < q} R(b_i, a_L^{(b_i)}, a_R^{(b_i)})$. Due to planarity, assignment of transmission rounds can be computed in each of those regions independently. For computation of the assignment, we apply the procedure *Assign-Inside-Region* recursively. Indeed, each region $R(b_i, a_L^{(b_i)}, a_R^{(b_i)})$ satisfies all properties to be an input region of the procedure *Assign-Inside-Region* and contains strictly smaller number of network nodes than the input region $R(b, a_L^{(b)}, a_R^{(b)})$. Note that $b \notin R(b_i, a_L^{(b_i)}, a_R^{(b_i)})$.

Correctness of the computed assignment follows from the inductive hypothesis that the assignments computed by the procedure *Assign-Inside-Region* for all subregions $R(b_i, a_L^{(b_i)}, a_R^{(b_i)})$ inform all nodes in B inside them, and the fact that the node b is informed due to transmission rounds assigned to its neighbors. Each call (not counting time taken by recursive calls) of the procedure *Assign-Inside-Region* takes polynomial-time and during whole computation there are at most $|B|$ calls of this procedure. □

In the following theorem, we summarize properties of the presented algorithm for generating transmission schedules which realize information dissemination in planar bipartite graphs.

Theorem 2.3.2. *Let $G = (V, E)$ be a planar graph with a source s. Denote $L_i = \{v | dist(s, v) = i\}$. Let $G_b = (A \cup B, E \cap A \times B)$ be an induced bipartite subgraph of the graph G such that $A \subseteq L_i$ and $B \subseteq L_{i+1}$ for some i. If A is a set of informed nodes and each node in B has at least one neighbor in the set A, then there is a polynomial-time algorithm which generates a 1-shot schedule of transmissions informing all nodes in B during at most 3 rounds.*

The previous theorem provides two corollaries for broadcasting in arbitrary planar reachability graphs.

Corollary 2.3.3. *Let $G = (V, E)$ be a planar reachability graph. There is a polynomial-time algorithm that for a given source node s generates 1-shot radio broadcasting schedules with length not exceeding $3 \cdot ecc(s) - 2$ rounds.*

Proof. Consider a set of layers $L_i = \{v | d(s, v) = i\}$, where $0 \le i \le ecc(s)$. If all nodes in an i-th layer L_i are informed, a schedule generated by the algorithm from Theorem 2.3.2 informs all nodes in the $(i + 1)$-th layer in at most 3 rounds. Therefore, a broadcasting schedule informing all network nodes in at most $3 \cdot (ecc(s) - 1) + 1$ can be constructed. □

The following corollary shows also the main result of [53] by Manne *et al.*

Corollary 2.3.4. *Let $G = (V, E)$ be a planar reachability graph. There is a polynomial-time algorithm that for a given source node s generates 2-shot radio broadcasting schedules with length $ecc(s) + O(\log n)$.*

Proof. The algorithm from Theorem 2.3.2 generates 1-shot schedules for information dissemination in planar bipartite reachability graph. Applying this algorithm as the algorithm $\mathcal{A}_b$ in Theorem 2.2.7, the claim follows. □

2.3.2 Lower Bounds

Utilizing algorithm from Corollary 2.3.3, centralized broadcasting can be always completed in $3 \cdot ecc(s, G) - 2$ rounds in radio networks with planar reachability graphs, i.e., $mintime(s, G) \leq 3 \cdot ecc(s, G) - 2$ for an arbitrary planar graph G. Recall that $mintime(s, G)$ stands for length of a time-optimal broadcasting schedule with respect to a source s in the reachability graph G. Obviously, the broadcasting cannot be completed in less than $ecc(s, G)$ rounds, since $ecc(s, G)$ is the minimal distance between the source s and the most remote node. Another result (Corollary 2.3.4, [53]) implies that there is a constant c such that $mintime(s, G) \leq ecc(s) + c \cdot \log n$ for an arbitrary planar graph G. Considering this, Manne *et al.* stated in [53] a natural question. Is there an algorithm that for any planar reachability graph and a source generates a schedule with the length $ecc(s) + O(1)$? In this section, we present several lower bonds that provide also negative answer to this question.

At first, we show that there are undirected planar reachability graphs such that broadcasting in less than $2 \cdot ecc(s, G)$ rounds is not possible. Particularly, we present a sequence of planar graphs $\{G_k | k \geq 1\}$, such that there is a node $s \in V(G_k)$ satisfying

(1) $ecc(s, G_k) = 2 \cdot k$,

(2) for each radio broadcasting schedule with respect to the source s and reachability graph G_k, there is a node that is informed in the round $4 \cdot k = 2 \cdot ecc(s, G_k)$ for the first time, i.e., $mintime(s, G_k) \geq 2 \cdot ecc(s, G_k)$, and

(3) $ecc(s, G_k) = \Theta(\log |V(G_k)|)$.

Construction of the sequence $\{G_k | k \geq 1\}$ is recursive. The (basic) graph G_1 is defined as follows

- $V(G_1) = \{s\} \cup \{v_i | i = 1, \ldots, 4\} \cup \{u_i | i = 1, \ldots, 8\}$
- $E(G_1) = \{(s, v_i), (v_i, u_4), (v_i, u_{i+4}) | i = 1, 2, 3, 4\} \cup \{(u_i, v_i), (u_i, v_{i+1}) | i = 1, 2, 3\}$.

In order to construct the graph G_k, for $k \geq 2$, we glue a copy of the graph G_{k-1} to each node u_i ($i = 1, \ldots, 8$) of the graph G_1 in such a way that $u_i = s$ where s is the source of G_{k-1}.

Clearly, G_k is a planar graph with the eccentricity of the distinguished source node s equal to $2 \cdot k = ecc(s, G_k)$. Since $|V(G_k)| = 8 \cdot |V(G_{k-1})| + 5$, for $k \geq 2$, it holds $ecc(s, G_k) = \Theta(\log |V(G_k)|)$.

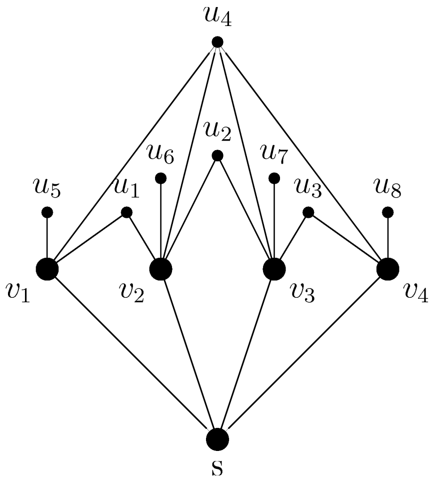

Figure 2.3: Graph G_1

Now, we show that for each graph G_k and for each schedule of radio broadcasting with respect to the source s and reachability graph G_k, there is a node $w \in V(G_k)$ such that w is informed in the round $4 \cdot k = 2 \cdot ecc(s, G_k)$ for the first time.

Theorem 2.3.5 ([E]). *For each $k \geq 1$, any radio broadcasting schedule for graph G_k with respect to the source s has the length at least $2 \cdot ecc(s, G_k)$ rounds, i.e., $mintime(s, G_k) \geq 4 \cdot k = 2 \cdot ecc(s, G_k)$.*

Proof. The proof is done by induction on k. At first, we consider the graph G_1 and show that for any schedule there must exist a node uninformed after 3 rounds. Let us assume an optimal schedule. After the first round, all neighbors of the source are informed. Since the node u_4 (most upper node on Figure 2.3.2) has 4 neighbors, it becomes informed only after a round in which exactly one of its neighbors (v_1, v_2, v_3, or v_4) transmits. Without loss of generality, let the second round be a round after which the node u_4 is informed. After this round, there are 3 uninformed nodes of degree 1 and at least one uninformed node of degree 2. Note that nodes of degree 1 have distinct neighbors. Moreover, all neighboring nodes of uninformed nodes of degree 2 have an uninformed neighbor of degree 1. Thus, it is not possible to inform all nodes after the third round, since in order to inform uninformed nodes of degree 1, all their neighbors have to transmit. However, it causes a collision for uninformed nodes of degree 2.

Consider now the graph G_k, $k \geq 2$, and assume that according to the inductive hypothesis the claim is true for G_{k-1}. In the recursive construction of G_k, we can observe that the broadcasting in a copy of G_{k-1} is started after the corresponding node u_i of the 'basic' G_1 becomes informed. Also note that broadcastings in different copies of G_{k-1} are independent, i.e., broadcasting in one copy has no influence on the broadcasting process in another copy. Claim for $k = 1$ implies existence of a node u_i which becomes informed in the round 4 for the first time. Considering assertion for corresponding copy of G_{k-1}, we obtain $mintime(s, G_k) \geq 4 + mintime(u_i, G_{k-1}) \geq 4 + 4 \cdot (k-1) \geq 4 \cdot k$. □

Typically, the complexity of algorithms for centralized radio broadcasting (the length of a produced radio broadcasting schedule) is described by an expression containing parameter D - diameter of the reachability graph. However, parameter D can be replaced with the eccentricity of the source node $ecc(s, G)$ in known algorithms for the planar case. It is easy to see that the constructed sequence of graphs $\{G_k | k \geq 1\}$ does not provide a better lower bound than the trivial lower bound D. Indeed, $D = 2 \cdot ecc(s, G_k) \leq mintime(s, G_k)$, for each $k \geq 1$. In fact, $2 \cdot ecc(s, G_k) = mintime(s, G_k)$. Using a simple extension of G_k, one can construct a better lower bound according to the parameter D. Let $\overline{G_k}$ be the planar graph constructed from G_k as follows: we add a path of length $ecc(s, G_k)$ starting in the source s. Let $\overline{s}$ be the last node of the added path. Clearly, diameter of the reachability graph is preserved, i.e., $D_{\overline{G_k}} = 2 \cdot ecc(s, G_k) = D_{G_k} = ecc(\overline{s}, \overline{G_k})$. Assume that $\overline{s}$ is the source of the broadcasting. Since each broadcasting have to follow a path from $\overline{s}$ to s of the length $ecc(s, G_k)$, we obtain $mintime(\overline{s}, \overline{G_k}) = ecc(s, G_k) + mintime(s, G_k) \geq 3 \cdot ecc(s, G_k)$. Hence, $mintime(\overline{s}, \overline{G_k}) \geq 3/2 \cdot D$.

Theorem 2.3.6 ([E]). *For each $k \geq 1$, any radio broadcasting schedule for graph $\overline{G_k}$ with respect to the source s has the length at least $3/2 \cdot D$ rounds, i.e., $mintime(\overline{s}, \overline{G_k}) \geq 3/2 \cdot D$, where D is diameter of $\overline{G_k}$.*

In the remaining, we generalize the previous construction. Let G be an undirected planar graph and denote

$$ratio(G) = max\left\{\frac{mintime(v, G)}{ecc(v, G)} | v \in V(G)\right\}.$$

Let $s' \in V(G)$ be a node such that $ratio(G) = \frac{mintime(s', G)}{ecc(s', G)}$. Following introduced notation, since it holds $mintime(s, G_k) \geq 2 \cdot ecc(s, G_k)$, it follows that $ratio(G_k) \geq 2$ for each G_k, $k \geq 1$. We construct a sequence of planar graphs $\{G'_k | k \geq 1\}$, s.t., there is a node $s' \in V(G'_k)$ satisfying

(1) $ecc(s', G'_k) = k \cdot ecc(s', G)$,

(2) for each radio broadcasting schedule with respect to the source s' and reachability graph G'_k, there is a node that is informed in the round $k \cdot mintime(s', G) = k \cdot ratio(G) \cdot ecc(s', G) = ratio(G) \cdot ecc(s', G'_k)$ for the first time, and

(3) $ecc(s', G'_k) = \Theta(\log |V(G'_k)|)$.

Define $G'_1 = G$. For $k \geq 2$, G'_k is constructed recursively from G'_1 and G'_{k-1}. As in the previous section, we glue a copy of the graph G'_{k-1} to each node v of the graph G_1, $v \neq s'$, in such a way that the node v is identified with the source s' of G'_{k-1}. Note that the construction preserves planarity (applying topological isomorphism, there is always a planar embedding of G' with the node s' on the outer face). Using similar argumentation, one can show by induction that all desired properties of G'_k according to s' are valid. Finally, it is easy to see that the properties imply $ratio(G'_k) \geq ratio(G)$, i.e., it is not possible to complete radio broadcasting in less that $ratio(G) \cdot ecc(s', G'_k)$ rounds with respect to the source s' and the reachability graph G'_k.

Theorem 2.3.7 ([E]). *Let G be an undirected planar graph. There exists a sequence of planar graphs $\{G'_k | k \geq 1\}$, s.t., for any $k \geq 1$, G'_k contains a node s' satisfying $mintime(s', G'_k) \geq ratio(G) \cdot ecc(s', G'_k)$.*

Let $\overline{G'_k}$ be a planar graph constructed in such a way that we add a path of the length $ecc(s', G'_k)$, which starts in the node s', to the graph G'_k. Let $\overline{s}$ be the last node of the added path. Considering $\overline{s}$ as a source of a broadcasting, one can prove the following theorem providing generalized lower bound according to diameter of a graph.

Theorem 2.3.8 ([E]). *Let G be an undirected planar graph. There exists a sequence of planar graphs $\{\overline{G'_k} | k \geq 1\}$, s.t., for any $k \geq 1$, $\overline{G'_k}$ contains a node $\overline{s}$ satisfying $mintime(\overline{s}, \overline{G'_k}) \geq (1 + ratio(G))/2 \cdot D$, where D is the diameter of the graph $\overline{G'_k}$.*

Algorithms for centralized radio broadcasting from Section 2.3.1, which complete the broadcasting in $3 \cdot ecc(s, G) - 2$ rounds, imply that $ratio(G) \leq 3 - \frac{2}{ecc(s,G)}$ for an arbitrary planar graph G. The presented construction implies that there is an infinite sequence of planar graphs satisfying $ratio(G) \geq 2$. Also, the generalized construction shows that in order to get a better lower bound of time of the centralized radio broadcasting for the planar case, we have to search for a planar graph with larger $ratio(G)$ (in the best case with $ratio(G) = 3 - \frac{2}{ecc(s,G)}$).

Finally observe, that in the case when we take into account the number of network nodes, the presented lower bounds can be easily reformulated as a lower bound $ecc(s) + \Omega(\log n)$ rounds for broadcasting in planar reachability graphs. It results in the claim that algorithms (Corollary 2.3.4, [53]) generating broadcasting schedules with length $ecc(s) + O(\log n)$ generate broadcasting schedules with the asymptotically optimal length.

2.4 Broadcasting in k-degenerate Graphs

The previous section shows that planar reachability graphs allow time and energy efficient radio broadcasting. Since each planar graph belongs to a class of 5-degenerate graphs, the following natural question emerges. Do k-degenerate reachability graphs allow efficient radio broadcasting for a fixed value k or even for an arbitrary value k similarly as planar graphs ?

Let us start with definition and description of the class of k-degenerate graphs.

Definition 2.4.1. *Let k be a non-negative integer. A graph G is called k-degenerate (we write $G \in \mathcal{D}_k$), if for each subgraph H of G, the minimum degree of H does not exceed k.*

The following value plays the fundamental role in the theory of k-degenerate graphs:

$$s(G) = \max_{H \subseteq G} \min_{v \in V(H)} \deg_H(v).$$

This number is called *Szekeres-Wilf number* and it is easy to see that G is k-degenerate if and only if $s(G) \leq k$. The definition implies that each subgraph of k-degenerate graph is k-degenerate as well (for more details see [6]). Moreover, it implies that for each graph G there is a number k such that G is k-degenerate.

Proposition 2.4.2 ([52]). *A $k + m$-node graph G is k-degenerate if and only if the vertex set $V(G)$ can be labelled $v_1, v_2, \ldots, v_{k+m}$ such that in the subgraph $\langle\{v_i, v_{i+1}, \ldots, v_{k+m}\}\rangle$ of G $\deg(v_i) \leq k$ for each $i = 1, 2, \ldots, m-1$.*

Note that the labelling of k-degenerate graph G satisfying the previous proposition can be computed in such a way that we remove a node with the lowest degree in every step. Obviously, this computation takes polynomial time. Also note that k-degenerate graphs have no general bound on the maximal degree of a node. On the other hand, it was shown in [52] that the number of edges of a k-degenerate graph is at most $k \cdot n - \binom{k+1}{2}$ where n is the number of nodes.

2.4.1 Upper Bounds

In this section, we focus on upper bounds on length of a radio broadcasting schedule. We present algorithms generating radio broadcasting schedules for arbitrary k-degenerate reachability graphs.

Consider a class of 1-degenerate graphs (remark that every connected 1-degenerate graph is a tree). In such a case we can construct a trivial radio broadcasting schedule $\Pi = (R_1, \ldots, R_q)$ with respect to a graph (tree) G and a source $s \in V(G)$ as follows:

(1) $R_1 := \{s\}$ and

(2) $R_{i+1} := I(R_i) \setminus \bigcup_{j=1}^{i-1} I(R_j)$, for $i \geq 1$.

The condition (2) yields that in the round $i+1$ a message is transmitted by all nodes which received a message in the round i for the first time. It is easy to see that there is a round p such that $R_p = \emptyset$. Letting $q := p-1$ one can prove that Π is a radio broadcasting schedule of the optimal length.

In what follows we present algorithms generating a radio broadcasting schedule for graphs which belong to $\mathcal{D}_k$ for a fixed integer k, $k \geq 2$.

Theorem 2.4.3 ([A]). *Let $G = (A \cup B, E) \in \mathcal{D}_k$ be a bipartite k-degenerate graph ($k \geq 2$) such that $deg_G(v) \geq 1$ for all $v \in B$. There is a polynomial-time algorithm which generates transmission schedules with length at most $\lceil k^2/2 \rceil + k + O((1 + \log k) \cdot \log |B|)$ realizing information dissemination with the informed part A and the uninformed part B.*

Proof. The algorithm works in two phases. During each phase, a part of the resulting schedule is produced. The goal of each part is to inform all nodes in the specific subset of B.

Phase 1: Let G be an input graph and denote $n = |V(G)|$. Since G is a k-degenerate graph, according to Proposition 2.4.2, in polynomial time we can compute labelling $v_1, v_2, \ldots, v_n$ of the nodes of G such that for each $i = 1, 2, \ldots, n$ in the induced subgraph $G_i = \langle\{v_i, v_{i+1}, \ldots, v_n\}\rangle_G$ it holds $\deg_{G_i}(v_i) \leq k$. It means that the nodes of G can be ordered in such a way that there are at most k edges from the node v_i to the nodes of set $\{v_{i+1}, \ldots, v_n\}$.

For each i, $1 \leq i \leq n$, we define a set:

$$N_{deg}(v_i) = \{v_j \in V(G) | (v_i, v_j) \in E(G) \wedge j > i\}$$

Note that $N_{deg}(v_i) \subseteq N(v_i)$ and $|N_{deg}(v_i)| \leq k$ for each $v_i \in V(G)$. The goal of this computation phase is to produce a schedule which ensures that each node $v_i \in B$, s.t., $|N(v_i) \setminus N_{deg}(v_i)| \geq 1$, becomes informed. During the computation, a round, denoted as $round(v_i)$, is assigned to every node $v_i \in V(G)$ such that $round(v_i) \in R \cup \{NIL\}$, where $R = \{1, \ldots, \lceil k^2/2 \rceil + k\}$. Symbol NIL is used for still undefined round. For a node $v_i \in A$, the value $round(v_i)$ denotes a round in which the node v_i will transmit a message during the first part of schedule. For a node $v_i \in B$, it denotes a round (from the other admissible) in which the node v_i receives a message. Initially, we set $round(v_i) := NIL$ for all $v_i \in V(G)$. For each node $v_m \in B$, we shall maintain the following set during the computation:

$$\begin{aligned} Receive(v_m) = \{r \quad | & \quad \text{there exists the unique } v_j \in N(v_m) \\ & \quad \text{such that } round(v_j) = r \neq NIL\} \end{aligned}$$

The nodes are processed in the sequential order from v_n to v_1. After a node v_i is processed the following two invariants hold:

(1) for each $v_j \in A$ such that $j \geq i$ we have:

- $round(v_j) \neq NIL$
- $round(v_m) \neq NIL$, for all $v_m \in N_{deg}(v_j)$.

(2) for each $v_j \in B$ such that $j \geq i$ it holds:

- $Receive(v_j) = \emptyset \Rightarrow round(v_j) = NIL$
- $Receive(v_j) \neq \emptyset \Rightarrow round(v_j) \in Receive(v_j)$

Now we show, how a node $v_i \in V(G)$ is processed. We fix $v_i \in V(G)$ and suppose that all nodes in the set $\{v_{i+1}, \ldots, v_n\}$ have been already processed.

In the case when $v_i \in B$, we process the node v_i as follows. If $Receive(v_i) \neq \emptyset$ then we set $round(v_i)$ to an arbitrary element of the set $Receive(v_i)$. Otherwise, $round(v_i)$ is unchanged, i.e., $round(v_i) = NIL$.

In the case when $v_i \in A$, the processing of the node is more complex. For each $v_m \in B$ we compute the set:

$$Used(v_m) = \{round(v) | v \in N_{deg}(v_m)\}$$

Next we compute the following sets:

$$Unassigned = \{v_j \in N_{deg}(v_i) | round(v_j) = NIL\}$$

$$Assigned = N_{deg}(v_i) \setminus Unassigned$$

$$Used^* = \bigcup_{v_j \in Unassigned} Used(v_j)$$

$$Used = \{round(v_j) | v_j \in Assigned\} \cup Used^*$$

Finally, we set the value $round(v_i)$ to an arbitrary element of the set $R \setminus Used$. Afterwards we set the value $round(v_j) := round(v_i)$, for all $v_j \in Unassigned$.

The first part of the schedule, which corresponds to this computation phase, is produced as follows: a node $v_i \in A$ transmits a message exactly in the round $round(v_i)$, i.e., $R_j = \{v_i \in A | round(v_i) = j\}$ for each $j \in R$.

Correctness of phase 1: We use computational induction to show that during the first phase of the algorithm both mentioned invariants hold after processing of a node.

It is not difficult to see that the claim is true after processing of a node $v_i \in B$. In the case when $v_i \in A$, at first we show the correctness of the assignment, i.e., that $R \setminus Used \neq \emptyset$ and we are able to choose a round-number.

Let $v_m \in Unassigned$. From the definition of the set $Unassigned$ it follows that $v_m \in N_{deg}(v_i) \subseteq B$, $m > i$ and $round(v_m) = NIL$. Since $m > i$ the second invariant implies that $Receive(v_m) = \emptyset$. Thus each round-number in the set $Used(v_m)$ is assigned to at least two nodes from the set $N_{deg}(v_m)$. Otherwise, there is a node $u \in N_{deg}(v_m)$ such the value $round(u) \neq NIL$ and moreover there is no node $w \in N_{deg}(v_m)$ such that $round(u) = round(w)$. But it contradicts to $Receive(v_m) = \emptyset$. Since $|N_{deg}(v_m)| \leq k$ and each round-number is used at least twice, it holds that $|Used(v_m)| \leq k/2$. Again using $|N_{deg}(v_i)| \leq k$ we obtain:

$$|Used| \leq |Assigned| + \frac{k}{2} \cdot |Unassigned| \leq \frac{k^2}{2} + k - 1.$$

Hence, there is at least one free round-number and $R \setminus Used \neq \emptyset$, i.e., the defined assignment is correct.

One can verify that according to the assignments which are created at the moment when the round-number $round(v_i)$ is determined, the first invariant holds.

Now we shall analyze the second invariant. Since during the processing of the node $v_i \in A$ we change only one round-number in the set A, it is sufficient to consider validity of conditions of the second invariant only for nodes of the set $N_{deg}(v_i) \subseteq B$. Since for each node $v_m \in Assigned$ it holds that $round(v_m) \notin R \setminus Used$, validity of the conditions remains unchanged for the nodes of the set $Assigned$. Consider a node $v_m \in Unassigned$. Since after processing of v_i it holds that $NIL \neq round(v_i) = round(v_m) \notin Used$ and $round(v_m) \notin Used \Rightarrow round(v_m) \notin Used(v_m) \subseteq Used$. Since $v_m \in$

Unassigned, the first invariant implies that before processing of v_i there is no node $v_j \in N(v_m) \setminus N(v_m) \subseteq A$ such that $round(v_j) \neq NIL$. We can summarize both these facts into the claim: There is no $v_j \in N_{deg}(v_m)$ such that $round(v_j) = round(v_m)$ and $round(v_j) = NIL$, for all $v_j \in (N(v_m) \setminus N_{deg}(v_m)) \setminus \{v_i\}$. This claim implies that $Receive(v_m) \neq \emptyset$ and $round(v_m) \in Receive(v_m)$.

Since after processing of all nodes both invariants hold, we show that each node $v_m \in B$ satisfying $N(v_m) \setminus N_{deg}(v_m) \neq \emptyset$ becomes informed after execution of the first part of schedule. Let $v_m \in B$ be a node such that $N(v_m) \setminus N_{deg}(v_m) \neq \emptyset$ and let $v_i \in A$ be a node such that $v_i \in N(v_m) \setminus N_{deg}(v_m)$. The definition of $N_{deg}(v_m)$ implies that $i < m$ and thus $v_m \in N_{deg}(v_i)$. Finally, the first invariant guaranties that $round(v_m) \neq NIL$. Hence the second invariant implies that $round(v_m) \in Receive(v_m)$, i.e., v_m receives a message in at least the round $round(v_m)$.

It follows that only the nodes $v_i \in B$, for which $N(v_i) = N_{deg}(v_i)$, can be uninformed. For such uniformed nodes it holds $|N(v_i)| = |N_{deg}(v_i)| \leq k$.

Phase 2: The goal of this phase is to inform all remaining uninformed nodes. Since for every uninformed node $v_i \in B$ it holds that $|N(v_i)| \leq k$, we can use the algorithm from Theorem 3.2.6 (Chapter 3) to produce the second part of the schedule. Using the following input collection of set-pairs $\mathcal{F} = \{(\{j | v_j \in N(v_i)\}, \emptyset) | \, v_i \in B \, \text{is uninformed}\}$, the algorithm produces a collection (ad-hoc selective family) $\mathcal{S} = \{S_1, \ldots, S_p\}$ as an output, where $p = O((1 + \log k) \cdot \log |B|)$. The second part of schedule is constructed using the following definition $R_{j+|R|} = \{v_i \in A | i \in S_j\}$ for each j, $1 \leq j \leq p$, where R is the set of the size $\lceil k^2/2 \rceil + k$ which has been defined in Phase 1. Correctness of the produced schedule follows from Theorem 3.2.6.

Complexity: The total length of produced schedule is $\lceil k^2/2 \rceil + k + O((1 + \log k) \cdot \log |B|)$ rounds. Since both phases take polynomial time, the designed algorithm is polynomial as well. □

Note that this result cannot be improved in general due to the lower bound for 2-degenerate graphs that is shown in the next section (Theorem 2.4.8).

Theorem 2.4.4 ([A]). *Let $G = (V, E)$ be an undirected connected k-degenerate graph ($k \geq 2$), i.e., $G \in \mathcal{D}_k$. Then there exists a polynomial-time algorithm producing radio broadcasting schedules of the length $D.(\lceil k^2/2 \rceil + k + O((1 + \log k) \cdot \log \frac{n}{D}))$.*

Proof. Since each subgraph of a k-degenerate graph is k-degenerate too, we use the algorithm from the proof of Theorem 2.4.3 to inform the nodes of every consecutive layer, i.e., broadcasting is scheduled layer by layer. The length of the schedule follows from the fact that $\sum_{i=1}^{D} \log |L_i|$ is maximized for $|L_i| = n/D$. $\square$

Since k is a fixed constant for any given class of k-degenerate graphs, the previous theorem results in the following corollary.

Corollary 2.4.5 ([A]). *Let $k \geq 2$ be an integer and D_k be a class of k-degenerate graphs. There is a polynomial-time deterministic algorithm which produces radio broadcasting schedules of length $O_k(D \cdot \log n/D)$ with respect to a reachability graph $G \in \mathcal{D}_k$ and a source $s \in V(G)$.*

It is not difficult to see that for k-degenerate graphs with diameter $o(\log n)$ proposed algorithm produces radio broadcasting schedules of shorter length than known algorithms for the general case.

Finally, we get another algorithm for generating radio broadcasting schedules in arbitrary k-degenerate graphs applying Theorem 2.2.7.

Corollary 2.4.6. *Let $k \geq 2$ be an integer and D_k be a class of k-degenerate graphs. There is a polynomial-time algorithm that for a given graph $G \in \mathcal{D}_k$ and a given source node s generates radio broadcasting schedules with length $ecc(s) + O_k(\log^2 n)$.*

2.4.2 Lower Bounds

This section is devoted to a lower bound on the broadcasting time in radio networks whose reachability graph is k-degenerate for $k \geq 2$. In particular, we show that there is a subclass of 2-degenerate graphs such that every radio broadcasting schedule has the length $\Omega(\log n)$ for each graph of this subclass.

At first, we define a set of graphs $\mathcal{G} = \{G_m | m \geq 2\}$. For a fixed integer m, $m \geq 2$, the graph G_m is constructed from the graph K_m with vertex set $V(K_m) = \{v_1, \ldots, v_m\}$ (the complete graph on m vertices) as follows: we add a new node s to K_m and we join it to every node of K_m. Next we subdivide every edge $e_{i,j} = (v_i, v_j) \in E(K_m)$ by a new node $u_{i,j}$. Formally, $G_m = (V_m, E_m)$ is an undirected graph with the vertex set $V_m = \{s, v_1, \ldots, v_m\} \cup \{u_{i,j} | 1 \leq i < j \leq m\}$ and the edge set $E_m = \{(s, v_i) | 1 \leq i \leq m\} \cup \{(v_i, u_{i,j}), (v_j, u_{i,j}) | 1 \leq i < j \leq m\}$.

With respect to the source node s, the graph G_m can be partitioned into layers $L_0 = \{s\}$, $L_1 = \{v_i | 1 \leq i \leq m\}$ and $L_2 = \{u_{i,j} | 1 \leq i < j \leq m\}$. Each layer forms an independent set. Obviously, the radius of G_m is 2. Since every node, except the source s, has degree at most 2, the graph G_m is a 2-degenerate graph with $(m^2 + m + 2)/2$ nodes.

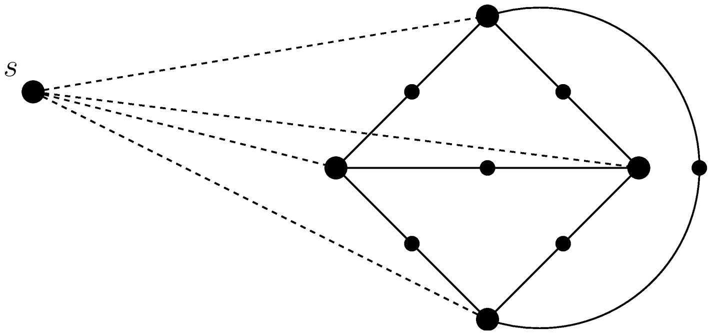

Figure 2.4: Graph G_4 with small nodes constructed by subdivision of edges of K_4

In the following lemma, we show that it is not possible to complete radio broadcasting in the graph G_m with the source s in less than $\lfloor \log n \rfloor + 1$ rounds.

Lemma 2.4.7 ([A]). *Any radio broadcasting schedule for the graph G_m with respect to the source $s \in V(G_m)$ has the length at least $\lfloor \log m \rfloor + 1$, i.e., $mintime(s, G_m) \geq \lfloor \log m \rfloor + 1$.*

Proof. We fix a radio broadcasting schedule $\Pi = (R_1, \ldots, R_q)$. Since $R_1 = \{s\}$, only the source s transmits in the first round. This transmission informs all nodes belonging to the layer L_1. Hence the rest of the schedule informs only the nodes of the layer $L_2 = \{u_{i,j} | 1 \leq i < j \leq m\}$ by the transmissions of nodes of the layer L_1. According to the schedule Π we can associate a binary sequence $s_i = (s_i^1, \ldots, s_i^{q-1})$ of length $q - 1$ with each node $v_i \in L_1$. We set s_i^r to 1 if and only if $v_i \in R_{r+1}$. Otherwise we set s_i^r to 0. It is easy to see that a node $u_{i,j} \in L_2$ receives a message exactly in each round r such that $s_i^{r-1} \neq s_j^{r-1}$. Since Π is a radio broadcasting schedule, every node $u_{i,j} \in L_2$ is informed and it receives a message in at least one round. Thus for each i, j, $i \neq j$, the binary sequences s_i and s_j should differ in at least one position, i.e., $s_i \neq s_j$. It implies that there are exactly $m = |L_1|$ different sequences associated with nodes of the layer L_1.

Clearly, we can construct at most 2^{q-1} different binary sequences of the length $q - 1$. Suppose now that $q - 1 < \lfloor \log m \rfloor$. It implies that $2^{q-1} < 2^{\lfloor \log m \rfloor} \leq 2^{\log m} = m$, i.e., $2^{q-1} < m$. The inequality contradicts the fact that we have m different binary sequences of the length $q - 1$. □

Theorem 2.4.8 ([A]). *There is a subclass $\mathcal{C}$ of the class of 2-degenerate graphs with radius 2 such that:*

(1) for every integer n, $n \geq 9$, there is a graph $G \in \mathcal{C}$ such that $|V(G)| = n$, and

(2) for every graph $G \in \mathcal{C}$ there is a node $s \in V(G)$ such that every radio broadcasting schedule with respect to the graph G and the source s has the length $\Omega(\log n)$, where $n = |V(G)|$ is the number of nodes.

Proof. Fix an arbitrary real number $c \in (0, \frac{1}{4})$. For each $n \geq 9$, we show that there are 2-degenerate graphs on n nodes with the radius 2 for which every radio broadcasting schedule has length at least $\lfloor c \cdot \log n \rfloor$ rounds. It is easy to see that the chosen c, n and $m = \lceil n^c \rceil$ satisfy the inequality $n - (m^2 + m + 2)/2 \geq 0$.

Let $G = (V, E)$ be a graph with n nodes constructed from the graph G_m, where $m = \lceil n^c \rceil$, by adding $n - (m^2 + m + 2)/2$ new nodes and joining them to the node $s \in V(G_m)$. Then G has the vertex set $V(G) = V(G_m) \cup \{w_1, \ldots, w_{n-(m^2+m+2)/2}\}$ and the edge set $E(G) = E(G_m) \cup \{(s, w_i) | 1 \leq i \leq n - (m^2 + m + 2)/2\}$ and obviously graph G is 2-degenerate graph with radius 2. Let $s \in V(G)$ be the source. Since there are no edges between $V(G_m) \setminus \{s\}$ and $(V(G) \setminus V(G_m)) \setminus \{s\}$, the broadcast operation is performed in the subgraph G_m separately. Previous lemma implies that it is not possible to complete broadcasting in G_m (and also in G) in less than $\lfloor \log m \rfloor + 1 \geq \log n^c \geq \lfloor c \cdot \log n \rfloor$ rounds. □

Note that in the previous proof there are more ways how to construct a graph G satisfying desired properties. In more general construction we add $n - (m^2 + m + 2)/2$ new nodes to the graph G_m. Next we add new edges to the graph G between the nodes of the set $W = (V(G) \setminus V(G_m)) \cup \{s\}$ (i.e., between newly created nodes and the source s) in such a way that the induced graph $H = \langle W \rangle_G$ is connected 2-degenerate graph satisfying $ecc_H(s) \leq 2$.

Since $\mathcal{D}_2 \subset \mathcal{D}_k \subset \mathcal{D}_{k+1}$ for each $k > 2$ (see [52]), the following holds:

Corollary 2.4.9 ([A]). *Let k be a positive integer, $k \geq 2$. There is a subclass $\mathcal{C}$ of k-degenerate graphs with radius 2 such that:*

(1) for every integer n, $n \geq 9$, there is a graph $G \in \mathcal{C}$ such that $|V(G)| = n$, and

(2) for every graph $G \in \mathcal{C}$ there is a node $s \in V(G)$ such that every radio broadcasting schedule for G with respect to the source s has the length $\Omega(\log n)$, where $n = |V(G)|$ is the number of nodes.

Chapter 3

Known Radio Networks with Long-range Interference

In the standard graph model, a node v can transmit messages only to the nodes which are located within its transmission range $R_T(v)$. A transmission range of v is a set of network nodes that are located at positions where the signal transmitted by v is intensive enough and quality enough to be successfully decoded. All nodes of a radio network operate on the same radio frequency band. Due to properties of the radio communication medium, simultaneous transmissions of two or more nodes cause interference in the area that is in the range of those transmitted signals. I.e., the nodes can receive and decode at most one message in a given time. The intensity and quality of the signal transmitted by network nodes vary. They depend directly on distance from the transmitting node and properties of the surrounding environment (e.g., presence of large obstacles, signal reflexes, etc). Obviously, the effect of interference depends significantly on the power of signals causing the interference at a given node. Observe that the standard graph model assumes that a transmitted signal can cause interference only inside its transmission range. I.e., it is assumed that the effect (power) of a signal transmitted by a node v is negligible outside of its transmission range $R_T(v)$. However, in some real-world settings, a transmitted signal can reach an area where decoding of the signal is not possible due to its low intensity, but the signal is intensive enough to interfere with other simultaneous transmissions (i.e., to cause an interference). As noted in [60], the interference range of a node is usually at least twice as large as its transmission range. In a more realistic setting, effect of a transmitted signal can even depend on the actual set of simultaneously transmitting nodes in the whole network.

In this chapter, we introduce and formalize new graph model of radio networks. This model generalizes the standard graph model and adds the notion of a long-range interference. We investigate centralized broadcasting in radio networks modelled by the proposed model.

3.1 Introduction

In distributed computing, particular assumptions about a model of communication environment play crucial role when designing an efficient communication protocol. Therefore, finding a proper models of communication environment is one of most challenging issues. Note that more models can be adequate for one communication environment. Their adequacy may depend on properties of real-world communication network and considered application. Generally, with a model closer to reality, it is much more difficult to achieve rigorous theoretical analysis. Hence, our main objective is to focus on models that are similar to abstract, simply, and intensively studied standard graph model of radio networks.

3.1.1 Interference Reachability Graphs - IRGs

Central notion of the standard graph model is the transmission range of a node. In the proposed model, we consider and add a new notion describing properties of a signal transmitted by a node more precisely. An *interference range* $R_I(v)$ of a node v contains all nodes at which a signal transmitted from v causes interference either on its own or together with other simultaneously incoming signals. We say that a node w belongs to the interference range $R_I(v)$ of a node v, i.e., $w \in R_I(v)$, if and only if any transmission of v can interfere with other transmissions reaching the node w. Note that if the transmitted signal is strong enough to be decoded, it is also strong enough to interfere, i.e., $R_T(v) \subseteq R_I(v)$. Therefore, if $w \in R_I(v) \setminus R_T(v)$, any transmission from v causes interference at the node w, but the node w cannot receive any message from v directly.

In our proposed model, a radio network is represented by an undirected (symmetric) graph $G = (V, E = E_T \cup E_I)$ called an *interference reachability graph IRG*. The set of vertices V of the IRG corresponds to the network nodes. The set of edges E in the IRG G is partitioned into two disjoint sets of edges: a set of *transmission edges E_T* and a set of *interference edges E_I*.

An edge $(u, v) \in E_T$ if and only if the node v is located in the transmission range of the node u, i.e., $(u, v) \in E_T \Leftrightarrow v \in R_T(u)$. In this case, the node u is called a *transmission neighbor* of the node v. The induced subgraph $G_T = (V, E_T)$ on the set of transmission edges is called a *transmission subgraph.* We denote the diameter of G_T as D_T, the eccentricity of s in G_T as $ecc_T(s)$, and the degree of v in G_T as $deg_T(v)$. Any path along transmission edges is called a *transmission path.* We denote by $d_T(u,v)$ the smallest integer such that there is a transmission path with $d_T(u,v)$ edges between the nodes u and v.

An edge $(u,v) \in E_I$ if and only if the node v belongs to the interference range of the node u but not to its transmission range, i.e., $(u,v) \in E_I \Leftrightarrow v \in R_I(u) \setminus R_T(u)$. In this case, the node u is called an *interference neighbor* of the node v. We denote the number of incident to v interference edges in IRG as $deg_I(v)$.

We assume that the transmission subgraph is connected, i.e., a message can be delivered from arbitrarily chosen source node to any other network node. Note that a crucial difference between transmission and interference edges is that we cannot transfer a message via an interference edge.

Since we shall often consider a set of nodes in the same distance from a distinguished node s in the transmission subgraph G_T, we introduce a notion of layers for IRGs.

Definition 3.1.1. *Let $G_T = (V, E_T)$ be the transmission subgraph of an undirected IRG $G = (V, E_T \cup E_I)$ with the source node $s \in V$. The i-th BFS layer of G is defined as $L_i = \{v \in V | d_T(s,v) = i\}$.*

Communication process in the standard graph model can be generalized for interference graph model as follows. In each communication round, a node can be in one of two operation modes. The node works either as a transmitter or as a receiver. If a node transmits, the transmitted signal reaches all its transmission and interference neighbors. A node, working as a receiver, receives a message if and only if a set of transmitting nodes contains exactly one of its transmission neighbors and none of its interference neighbors.

It is easy to see that the proposed interference graph model generalizes the standard graph model. Indeed, a radio network modelled by the standard graph model corresponds to the case when for each node v it holds that $R_T(v) = R_I(v)$. It follows that any algorithm designed for IRGs can be applied also in networks modelled by the standard graph model.

Finally, note that a directed variant of interference reachability graphs (when IRG is a directed graph) can be defined and investigated as well.

3.1.2 Difficulty of Fast Broadcasting in IRGs

The broadcasting task can be always completed during at most n rounds where n is the number of network nodes. Indeed, a simple scheduling mechanism realizes it. In each round, we choose an arbitrary informed node that

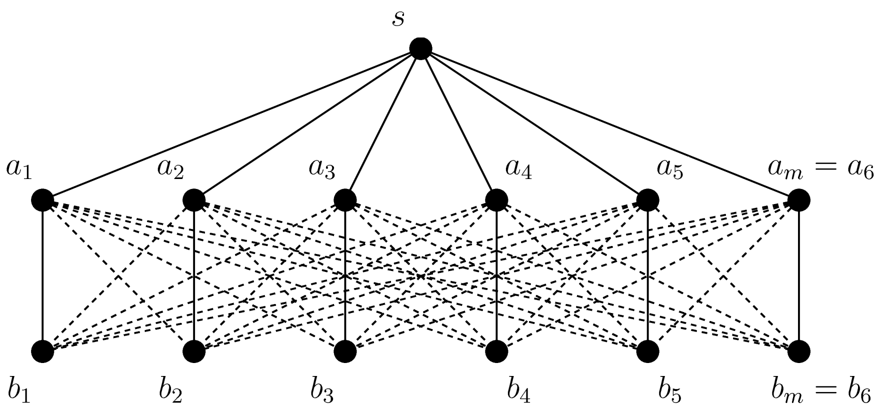

Figure 3.5: Interference reachability graph G_6

is a transmission neighbor of at least one uninformed node. The chosen node is the only transmitting node in this round. Since it is the only transmitting node in this round, all uninformed nodes in its transmission range receive the source message and become informed. Obviously, each node transmits at most once and after at most n rounds all network nodes are informed.

On the other hand, there is an *IRG* with eccentricity of the source equal to 2 such that the broadcasting time is bounded by $\Omega(n)$ rounds. Let us define a $(2 \cdot m + 1)$-node graph $G_m = (V_m, E_T \cup E_I)$ as follows:

- $V(G_m) = \{s, a_1, a_2, \ldots, a_m, b_1, b_2, \ldots, b_m\}$
- $E_T(G_m) = \{(s, a_i), (a_i, b_i) | 1 \leq i \leq m\}$
- $E_I(G_m) = \{(a_i, b_j) | 1 \leq i \neq j \leq m\}$

Now, consider the node s as a source of the broadcasting. Each node a_i becomes informed after first transmission of s. A node b_i can be informed only by a transmission of a_i. However, if a node a_i transmits, no other node b_j, $j \neq i$, can receive the message due to the presence of interference edges. It follows, that at least $m + 1 = \Omega(n)$ rounds are necessary to complete the broadcasting.

The previous example shows that trivial broadcasting schedules are asymptotically optimal in *IRG*s. Moreover, it shows $\Omega(n)$-round lower bound for *IRG*s with a constant eccentricity of the source. In contrast, broadcasting in radio networks with a constant eccentricity of the source can be completed in $O(\log^2 n)$ rounds in the standard graph model.

Intuitively, the slow broadcasting is caused by large density of interference edges. Therefore, in order to study the time complexity of the broadcasting task in the proposed interference model, we should consider other parameters of IRGs and introduce new appropriate parameters reflecting the presence of interference edges in an IRG. Then, we can express the time complexity of a broadcasting schedule much finer with respect to these parameters.

3.1.3 Related Work

Unfortunately, there is no widely adopted model of radio networks with a long-range interference, i.e., in the case when the transmission range and the interference range of a node differ. A radio network is usually modelled on a geometric basis (transmission and interference ranges are defined in terms of Euclidian distance) or designed algorithms are probabilistic.

There are two models that are much closer to reality due to reflecting physical propagation of radio signal. Comparing to other models, they consider physical carrier sensing. The older of them, presented in [44] by Kothapalli *et al.*, is called a *Sensing-Interference-Transmission* (SIT) model. The model is probabilistic and introduces different transmission and interference ranges via a cost function that is based on Euclidian distance between nodes. Particularly, a cost of connection between two nodes u and v, denoted as $c(u, v)$, is an arbitrary number such that $c(u, v) \in [(1 - \delta) \cdot d(u, v), (1 + \delta) \cdot d(u, v)]$, where $d(u, v)$ is the Euclidian distance between nodes u and v, and $\delta \in [0, 1)$ is a constant depending on the properties of the communication environment. Moreover, it is required that the cost function is symmetric, i.e., $c(u, v) = c(v, u)$. With respect to transmission power P, there are defined to monotonic functions $r_t(P)$ and $r_i(P)$ defining a transmission range and an interference range respectively. A node v transmitting a message with the power P can cause interference at a node w with high probability in the case when $c(v, w) \leq r_i(P)$. A node w successfully receives a message transmitted by a node v, if $c(v, w) \leq r_t(P)$ and there is no other simultaneously transmitting node v' such that $c(v', w) \leq r_i(P)$. Observe that if we do not consider the probabilistic aspect of this model, each radio network modelled by this model can be modelled by an interference reachability graph as well. However, the SIT model adds the notion of physical carrier sensing. This feature can be used by communication algorithms very effectively. The model defines two monotonically growing functions $r_{st}(T, P)$, called a *carrier sense transmission range*, and $r_{si}(T, P)$, called a *carrier sense interference range*, where the parameter P is transmission power and T is a sensing threshold. Interpretation of those functions is the following. If a node v transmits a message with the power P, a node w is sensing with a threshold T, and $c(v, w) \leq r_{st}(T, P)$, then w is sensing a message transmission with high probability. Further, if a node w senses a transmission with a threshold T, then there is at least one node v that transmits a message with power P and $c(v, w) \leq r_{si}(T, P)$. Observe that a transmitting node in the carrier sense transmission range of a node always causes carrier sensing with given threshold at this node (with high probability). On the other hand, a transmitting node in the carrier sense interference range may cause carrier sensing. A transmitting node outside of the carrier sense interference range never causes carrier sensing. To conclude description of the SIT model, note that the broadcasting and spanner construction in this model were investigated by Onus *et al.* in [60].

The newer model considering physical carrier sensing is an *extended Signal-to-Interference-plus-Noise-Ratio* $SINR$ model based on interference model from [40] and proposed by Scheideler *et al.* in [64]. In this model, a message sent by a node u is received by a node v if and only if

$$\frac{P_v(u)}{N + \sum_{w \in S} P_v(w)} \geq \beta$$

where $P_x(y)$ is received power of a signal transmitted by a node y at the node x, N is the background noise, S is a set of currently transmitting nodes with exemption of the node u, and β is a constant which depends on technical properties of receivers. Note that the model based on interference reachability graphs can be seen, comparing to the $SINR$ model, as a static model. Indeed, in the $SINR$ model decoding (transmission and interference ranges) of a message depends on actual set of simultaneously transmitting nodes in the whole network. In contrast, successful receiving of a message in a radio network modelled by interference reachability graphs depends only on the set of simultaneously transmitting nodes which contain the receiving node in their statically predefined transmission and interference ranges. The physical carrier sensing in the extended $SINR$ model is defined as follows. A node v senses carrier with threshold T if and only if $N + \sum_{w \in S} P_v(w) \geq T$. Finally, note that most recent explorations of models reflecting physical propagation of signal include a study on characteristics of efficient local broadcasting in $SINR$ model, see work of Goussevskaia *et al.* in [39], and study on connectivity and interference in log-normal shadowing radio propagation model by Muetze *et al.* in [57].

A communication model based on graphs in which the interference range of a node is larger than its transmission range was considered by Bermond *et al.* in [4]. The authors studied time complexity of the gathering task in known topology radio networks. They defined transmission and interference range of a node with respect to distances in an underlying communication graph. Particularly, let us denote the length (the number of edges) of a shortest path between nodes u and v in the graph G as $d_G(u, v)$. Fix the numbers d_T and d_I. In their model, the number d_T, $d_T \geq 1$, is called a *transmission distance* and the number d_I, $d_I \geq d_T$, is called an *interference distance.* The transmission range $R_T(v)$ of a node v is defined as $R_T(v) = \{w|d_G(v,w) \leq d_T\}$ and the interference range $R_I(v)$ of a node v as $R_I(v) = \{w|d_G(v,w) \leq d_I\}$. Note that the standard graph model corresponds to the case when $d_T = d_I = 1$. It is easy to see that there are such settings where the model introduced by Bermond *et al.* is not appropriate, e.g., due to large obstacles or signal reflexes. It means that there are settings for which it is difficult or even impossible to express transmission and interference ranges of the nodes with respect to distances in an underlying communication graph. On the other hand, each radio network modelled by the model introduced by Bermond *et al.* can be described by an IRG. It implies that interference reachability graphs are more general. In particular, this model is more general than the standard graph model, called also a *packet radio network model*, but less general that modelling radio networks by interference reachability graphs.

3.2 Interference Selective Families

Following the work of Clementi *et al.* [18], which is devoted to selective structures related to the standard graph model of radio networks, we define the notion of an *interference selective family.* Later, we show some useful properties of interference selective families. As we will discuss later, they are closely related to the interference graph model of radio networks. In the case when a considered collection of set-pairs satisfies a specific property (defined later), we show the existence of small interference selective families by a probabilistic argument. Finally, we design a deterministic polynomial-time algorithm that computes small interference selective family for a given input collection of set-pairs. Algorithms presented in this section generalize the work [18] of Clementi *et al.*

Selective families are related to intensively studied combinatorial structures called *selectors* (see, e.g., [5], [11], or [42]). One of their applications is in communication algorithms for radio networks modelled by the standard graph model [15] in the case when the nodes are not aware of the network topology. The *k-selectors*, defined and investigated by Chrobak *et al.* in [15], can be seen as a weaker variant of interference selective families (with respect to interference) introduced in this section.

Definition 3.2.1. *Let $\mathcal{F} = \{(T_1, I_1), (T_2, I_2), \ldots, (T_m, I_m)\}$ be a collection of set-pairs such that $T_i \cap I_i = \emptyset$ and $T_i \neq \emptyset$, for all $i = 1, \ldots, m$. Denote $U(\mathcal{F}) = \bigcup_{i=1}^{m} T_i \cup I_i$. A family $\mathcal{S} = \{S_1, S_2, \ldots, S_k\}$ of subsets of $U(\mathcal{F})$ is said to be selective for $\mathcal{F}$ if and only if for any (T_i, I_i) there is a set S_j such* *that $|T_i \cap S_i| = 1$ and $I_i \cap S_i = \emptyset$. We say that the set S_j is selective for (T_i, I_i).*

There is a relationship between interference selective families and the information dissemination in bipartite IRG. Let V_S be the informed part of a bipartite IRG $G = (V_S \cup V_R, E)$ and V_R be the uninformed part. Interference selective families can be utilized to construct a schedule of transmissions such that all nodes in V_R become informed by transmissions of nodes in the set V_S. Indeed, consider a collection $\mathcal{F} = \{(T_v, I_v) | v \in V_R\}$ such that $T_v = \{u \in V_S | v \in T(u)\}$ and $I_v = \{u \in V_S | v \in I(u)\}$. Note that $U(\mathcal{F}) \subseteq V_S$. Let $\mathcal{S} = \{S_1, S_2, \ldots, S_k\}$ be an interference selective family for $\mathcal{F}$. Observe, that if the set of transmitting nodes in the i-th round of a schedule is the set S_i, all nodes in V_R become informed in at most $k = |\mathcal{S}|$ rounds. Hence it seems useful to search for small selective families. Obviously, we can always construct a selective family of the size $min\{|\mathcal{F}|, |U(\mathcal{F})|\}$ by a trivial construction. On the other hand, for any n, there is an instance $\mathcal{F}$, $|\mathcal{F}| = |U(\mathcal{F})| = n$, such that it is not possible to construct interference selective family of the size smaller than n. These instances correspond to the example of "slow" IRG in Section 3.1.2. It follows that a new parameter providing a better characterization of the collection $\mathcal{F}$ should be defined.

Definition 3.2.2. *Let $\mathcal{F} = \{(T_1, I_1), (T_2, I_2), \ldots, (T_m, I_m)\}$ be a collection of set-pairs such that $T_i \cap I_i = \emptyset$ and $T_i \neq \emptyset$, for all $i = 1, \ldots, m$. We say that r is an interference ratio of the pair (T_i, I_i) if and only if $|I_i| \leq r \cdot |T_i|$. Analogously, we say that $r(\mathcal{F})$ is an interference ratio of the collection $\mathcal{F}$, if and only if $|I_i| \leq r(\mathcal{F}) \cdot |T_i|$, for all $i = 1, \ldots, m$.*

Intuitively, an interference ratio is introduced in order to reflect the ratio of the interference edges to the transmission edges of a node, i.e., the local density of interference edges. Now, using a probabilistic argument, we show that there are small interference selective families.

Theorem 3.2.3 ([D])**.** *Let $\mathcal{F} = \{(T_1, I_1), (T_2, I_2), \ldots, (T_m, I_m)\}$ be a collection of set-pairs such that $T_i \cap I_i = \emptyset$, $T_i \neq \emptyset$, and $\Delta_{min} \leq |T_i| + |I_i| \leq \Delta_{max}$, for all $i = 1, \ldots, m$. There is a family $\mathcal{S}$ of the size $O((1 + r(\mathcal{F})) \cdot ((1 + \log(\Delta_{max}/\Delta_{min}))) \cdot \log |\mathcal{F}|)$ that is selective for $\mathcal{F}$.*

Proof. Let us define $\mathcal{F}' = \{(T_i, I_i) \in \mathcal{F}, |T_i| + |I_i| = 1\}$. Since $T_i \neq \emptyset$, we have that $|T_i| = 1$ and $|I_i| = 0$, for all members of $\mathcal{F}'$. It is easy to see that the set $S_0 = \bigcup_{(T_i, I_i) \in \mathcal{F}'} T_i$ is selective for $\mathcal{F}'$. Therefore, in what follows we assume that $\Delta_{min} \geq 2$.

For each $j \in \{\lceil \log \Delta_{min} \rceil, \ldots, \lceil \log \Delta_{max} \rceil\}$, let us consider a family $\mathcal{S}_j$ of l sets, where an unknown parameter l will be determined at the end of the proof. Each set $S \in \mathcal{S}_j$ is constructed by picking each element of $U(\mathcal{F})$ independently with the probability $1/2^j$.

Fix a pair $(T_i, I_i) \in \mathcal{F}$. Let j be an integer such that $2^{j-1} \leq |T_i|+|I_i| < 2^j$. Consider a set $S \in \mathcal{S}_j$. Let us estimate the probability that the set S is selective for (T_i, I_i):

$$Pr[|T_i \cap S| = 1 \wedge I_i \cap S = \emptyset] = |T_i| \cdot \frac{1}{2^j} \cdot \left(1 - \frac{1}{2^j}\right)^{|T_i|+|I_i|-1}$$

$$> |T_i| \cdot \frac{1}{2^j} \cdot \left(1 - \frac{1}{2^j}\right)^{2^j} \overset{(1)}{\geq} \frac{1}{2 \cdot (r(\mathcal{F})+1)} \cdot \left(1 - \frac{1}{2^j}\right)^{2^j} \overset{(2)}{\geq} \frac{1}{8 \cdot (r(\mathcal{F})+1)}$$

The inequality (1) holds because $2^{j-1} \leq |T_i| + |I_i| \leq (r(\mathcal{F})+1) \cdot |T_i|$. The inequality (2) follows from the fact that $\left(1 - \frac{1}{t}\right)^t \geq \frac{1}{4}$, for $t \geq 2$.

The sets in $\mathcal{S}_j$ are constructed independently. Thus the probability that none of l sets of the family $\mathcal{S}_j$ is selective for (T_i, I_i) is upper-bounded by the expression

$$\left(1 - \frac{1}{8 \cdot (r(\mathcal{F})+1)}\right)^l \leq e^{-\frac{l}{8 \cdot (r(\mathcal{F})+1)}}$$

due to the inequality $(1-x)^y \leq e^{-x \cdot y}$, for $0 < x < 1$ and $y > 1$.

Finally, let us define a family $\mathcal{S}$ as the union of the families $\mathcal{S}_j$, for $j \in \{\lceil \log \Delta_{min} \rceil, \ldots, \lceil \log \Delta_{max} \rceil\}$. Now we estimate the probability that $\mathcal{S}$ is not selective for $\mathcal{F}$:

$$Pr[\mathcal{S} \text{ is not selective for } \mathcal{F}] \leq \sum_{(T_i,I_i) \in \mathcal{F}} Pr[\mathcal{S} \text{ is not selective for } ((T_i, I_i))]$$

$$\leq \sum_{(T_i,I_i) \in \mathcal{F}} e^{-\frac{l}{8 \cdot (r(\mathcal{F})+1)}} = |\mathcal{F}| \cdot e^{-\frac{l}{8 \cdot (r(\mathcal{F})+1)}}$$

It follows that the probability of $\mathcal{S}$ not being selective for $\mathcal{F}$ is less than 1 for $l > 8 \cdot (r(\mathcal{F})+1) \cdot \ln |F|$. It implies the existence of an interference selective family $\mathcal{S}$ of the size $O((1 + r(\mathcal{F})) \cdot ((1 + \log(\Delta_{max}/\Delta_{min}))) \cdot \log |\mathcal{F}|)$. □

Using de-randomization method of conditional probabilities, we show that a selective family of the size $O((1 + r(\mathcal{F})) \cdot ((1 + \log(\Delta_{max}/\Delta_{min}))) \cdot \log |\mathcal{F}|)$ can be constructed deterministically in polynomial time.

At first, we fix an ordering of elements of $U(\mathcal{F}) = \{u_1, u_2, \ldots, u_n\}$. For $S \subseteq U(\mathcal{F})$, let us denote $\delta_i(S) = S \cap \{u_i, u_{i+1}, \ldots, u_n\}$. Finally, let us fix a pair $(T, I) \in \mathcal{F}$ and let Δ be a power of 2 (i.e., $\Delta = 2^j$, for some j) such that $\Delta/2 \leq |T| + |I| < \Delta$. For a fixed set $S \subseteq \{u_1, u_2, \ldots, u_{j-1}\}$, we define the conditional probabilities

$$Y_j(S,(T,I)) = Pr[S \cup X \cup \{u_j\} \text{ is selective for } (T,I)]$$
$$N_j(S,(T,I)) = Pr[S \cup X \text{ is selective for } (T,I)]$$

where X is a subset of $\{u_{j+1},\ldots,u_n\}$ constructed by picking each element of $\{u_{j+1},\ldots,u_n\}$ independently at random with the probability $1/\Delta$, i.e., $Pr[u_k \in X] = 1/\Delta$.

Lemma 3.2.4 ([D]). *The conditional probabilities $Y_j(S,(T,I))$ and $N_j(S,(T,I))$ can be computed in $O(n)$ time.*

Proof. Evaluation of the conditional probabilities $Y_j(S,(T,I))$ and $N_j(S,(T,I))$ is based on the following equalities. In these equalities, we use α to denote $|\delta_j(T)| + |\delta_j(I)|$.

- $\delta_j(T) = 0$

$$Y_j(S,(T,I)) = N_j(S,(T,I)) = \begin{cases} \left(1-\frac{1}{\Delta}\right)^{|\delta_j(I)|} & |T \cap S| = 1 \wedge I \cap S = \emptyset \\ 0 & \text{otherwise} \end{cases}$$

- $\delta_j(T) \geq 1 \wedge u_j \in T$

$$Y_j(S,(T,I)) = \begin{cases} \left(1-\frac{1}{\Delta}\right)^{\alpha-1} & (T \cup I) \cap S = \emptyset \\ 0 & \text{otherwise} \end{cases}$$

$$N_j(S,(T,I)) = \begin{cases} 0 & |(T \cup I) \cap S| \geq 2 \\ 0 & |T \cap S| = 0 \wedge \\ & \quad |I \cap S| = 1 \\ \left(1-\frac{1}{\Delta}\right)^{\alpha-1} & |T \cap S| = 1 \wedge \\ & \quad |I \cap S| = 0 \\ 0 & |(T \cup I) \cap S| = 0 \wedge \\ & \quad |\delta_j(T)| = 1 \\ (|\delta_j(T)|-1) \cdot \frac{1}{\Delta} \cdot \left(1-\frac{1}{\Delta}\right)^{\alpha-2} & |(T \cup I) \cap S| = 0 \wedge \\ & \quad |\delta_j(T)| \geq 2 \end{cases}$$

- $\delta_j(T) \geq 1 \wedge u_j \in I$

$$Y_j(S,(T,I)) = 0$$

$$N_j(S,(T,I)) = \begin{cases} 0 & |(T \cup I) \cap S| \geq 2 \\ 0 & |T \cap S| = 0 \wedge |I \cap S| = 1 \\ \left(1-\frac{1}{\Delta}\right)^{\alpha-1} & |T \cap S| = 1 \wedge |I \cap S| = 0 \\ |\delta_j(T)| \cdot \frac{1}{\Delta} \cdot \left(1-\frac{1}{\Delta}\right)^{\alpha-2} & |(T \cup I) \cap S| = 0 \end{cases}$$

- $\delta_j(T) \geq 1 \wedge u_j \notin T \cup I$

$$\begin{array}{l} Y_j(S,(T,I)) \\ N_j(S,(T,I)) \end{array} = \begin{cases} 0 & |(T \cup I) \cap S| \geq 2 \\ 0 & |T \cap S| = 0 \wedge |I \cap S| = 1 \\ \left(1 - \frac{1}{\Delta}\right)^{\alpha} & |T \cap S| = 1 \wedge |I \cap S| = 0 \\ |\delta_j(T)| \cdot \frac{1}{\Delta} \cdot \left(1 - \frac{1}{\Delta}\right)^{\alpha - 1} & |(T \cup I) \cap S| = 0 \end{cases}$$

□

Utilizing those conditional probabilities, we design an algorithm that computes an interference selective family for a given input collection of set-pairs $\mathcal{F}$. Algorithm is based on the procedure *IASF*. It produces an interference selective family for a given collection $\mathcal{F} = \{(T_1, I_1), (T_2, I_2), \ldots, (T_m, I_m)\}$ satisfying the property that there is a power of 2 denoted as Δ (i.e., $\Delta = 2^j$, for some $j \geq 2$) such that the condition $\Delta/2 \leq |T_i| + |I_i| < \Delta$ is valid for all members of $\mathcal{F}$.

```
Input  : $\Delta = 2^j$, $\mathcal{F} = \{(T_1, I_1), (T_2, I_2), \ldots, (T_m, I_m)\}$
Output: $\mathcal{S} = \{S_1, S_2, \ldots, S_k\}$
let $n$ be the number of elements of $U(\mathcal{F}) = \{u_1, u_2, \ldots, u_n\}$;
while $\mathcal{F} \neq \emptyset$ do
    $S \leftarrow \emptyset$;
    for $i \leftarrow 1$ to $n$ do
        $Y_i \leftarrow \sum_{(T,I) \in \mathcal{F}} Y_i(S, (T, I))$ ;
        $N_i \leftarrow \sum_{(T,I) \in \mathcal{F}} N_i(S, (T, I))$ ;
        if $N_i < Y_i$ then $S \leftarrow S \cup \{u_i\}$;
    end
    $\mathcal{F} \leftarrow \mathcal{F} \setminus \{(T, I) \in \mathcal{F} \mid S$ is selective for $(T, I)\}$;
    $\mathcal{S} \leftarrow \mathcal{S} \cup S$;
end
return $\mathcal{S}$
```

Algorithm 1: Procedure *IASF*

Lemma 3.2.5 ([D]). *Let $\mathcal{F} = \{(T_1, I_1), (T_2, I_2), \ldots, (T_m, I_m)\}$ be a collection of set-pairs such that $T_i \cap I_i = \emptyset$, $T_i \neq \emptyset$, and $\Delta/2 \leq |T_i| + |I_i| < \Delta$, for all $i = 1, \ldots, m$, where $\Delta = 2^j \geq 4$ is a power of 2. For the input collection $\mathcal{F}$, the procedure IASF computes an interference selective family $\mathcal{S}$ of the size $O((1 + r(\mathcal{F})) \cdot \log |\mathcal{F}|)$. Time complexity of the procedure IASF is $O(r(\mathcal{F}) \cdot \log |\mathcal{F}| \cdot |\mathcal{F}| \cdot |U(\mathcal{F})|^2)$.*

Proof. We start with analysis of a single execution of the *while* loop in the procedure *IASF*. Let us denote the content of the variable $\mathcal{F}$ before k-th iteration of the *while* loop as $\mathcal{F}_k$, i.e., $\mathcal{F}_1 = \mathcal{F}$.

Let W be a set constructed by picking each element of $U(\mathcal{F}_k)$ independently with the probability $1/\Delta$. Denote as $E(X)$ the expected number of set-pairs $(T, I) \in \mathcal{F}_k$ that are selected by W. Analogously, for a set Y, $Y \subseteq U(\mathcal{F}_k)$, and an integer i, $i \geq 1$, satisfying $Y \cap \delta_i(U(\mathcal{F}_k)) = \emptyset$, we denote as $E(X|(Y, i))$ the expected number of set-pairs $(T, I) \in \mathcal{F}_k$ that are selected by a random set $W_{Y,i}$. The set $W_{Y,i}$ is the union of the set Y and a set of independently (with probability $1/\Delta$) picked elements of the set $\delta_i(U(\mathcal{F}_k))$. Clearly, it follows from the proof of Theorem 3.2.3 that

$$E(X|(\emptyset, 1)) = E(X) \geq \frac{|\mathcal{F}_k|}{8 \cdot (r(\mathcal{F}_k) + 1)}.$$

Now we prove by induction on i that the inequality $E(X|(S, i + 1)) \geq E(X)$ is valid after i ($i \in \{0, \ldots, |U(\mathcal{F}_k)|\}$) iterations of the *for* loop in *IASF*:

- For $i = 0$, it holds $S = \emptyset$. Since $E(X|(\emptyset, 1)) = E(X)$, the claim is true.
- Assume that the claim is true for all j, $j < i$. Denote $S' = S \setminus \{u_i\}$, i.e., S' is the content of the variable S after $i - 1$ iterations of the *for* loop. Due to the definition of the expected value, it holds for $i > 0$ that

  $$E(X|(S', i)) = \frac{1}{\Delta} \cdot E(X|(S' \cup \{u_i\}, i+1)) + \big(1 - \frac{1}{\Delta}\big) \cdot E(X|(S', i+1)).$$

  Obviously, $A = qB + (1-q)C \Rightarrow A \leq max\{B, C\}$, for $A, B, C \geq 0$ and $0 \leq q \leq 1$. Thus it follows

  $$E(X|(S', i)) \leq max\{E(X|(S' \cup \{u_i\}, i + 1)), E(X|(S', i + 1))\}.$$

  Moreover, definition of the expected value implies $Y_i = E(X|(S' \cup \{u_i\}, i + 1))$ and $N_i = E(X|(S', i + 1))$. The choice between adding the element u_i to S or not depends on the values Y_i and N_i. Since the larger value is chosen, it follows that $E(X|(S, i+1)) = max\{Y_i, N_i\} = max\{E(X|(S' \cup \{u_i\}, i+1)), E(X|(S', i+1))\} \geq E(X|(S', i))$. Finally, the inductive hypothesis implies

  $$E(X|(S, i + 1)) \geq E(X|(S', i)) \geq E(X).$$

The previous claim for $i = |U(\mathcal{F}_k)|$ implies that in each iteration of the *while* loop such a set S is computed that at least $\lceil \frac{|\mathcal{F}_k|}{8 \cdot (r(\mathcal{F}_k)+1)} \rceil$ set-pairs of $\mathcal{F}_k$

are selected S. Since $r(\mathcal{F}_k) \leq r(\mathcal{F})$, we get $|\mathcal{F}_{k+1}| \leq \left(1 - \frac{1}{8\cdot(r(\mathcal{F})+1)}\right) \cdot |\mathcal{F}_k|$. It implies that after k iterations of the *while* loop, the number of unselected set-pairs can be upper-bounded by the expression $\left(1-\frac{1}{8\cdot(r(\mathcal{F})+1)}\right)^k \cdot |\mathcal{F}|$. Since, for $z \geq 1$, it holds $\ln(1 - \frac{1}{z}) \leq -\frac{1}{z}$, this expression is lower than 1 for k at least $(8 \cdot r(\mathcal{F}) + 1) \cdot \ln|\mathcal{F}|$. Thus, at most $O((r(\mathcal{F}) + 1) \cdot \log|\mathcal{F}|)$ iterations of the *while* loop are sufficient to select all set-pairs of the collection $\mathcal{F}$. Hence the interference selective family computed by the procedure *IASF* has the size $O((1 + r(\mathcal{F})) \cdot \log|\mathcal{F}|)$.

The time complexity of the computation immediately follows from the proof above and Lemma 3.2.4. □

On the basis of the procedure *IASF*, an algorithm for an arbitrary input collection $\mathcal{F}$ can be constructed.

Theorem 3.2.6 ([D]). *Let $\mathcal{F} = \{(T_1, I_1), (T_2, I_2), \ldots, (T_m, I_m)\}$ be a collection of set-pairs such that $T_i \cap I_i = \emptyset$, $T_i \neq \emptyset$, and $\Delta_{min} \leq |T_i| + |I_i| \leq \Delta_{max}$, for all $i = 1, \ldots, m$. There is a deterministic algorithm that for the given collection $\mathcal{F}$ generates an interference selective family $\mathcal{S}$ of size $O((1 + r(\mathcal{F})) \cdot ((1 + \log(\Delta_{max}/\Delta_{min}))) \cdot \log|\mathcal{F}|)$. Computation takes polynomial time $O((1 + \log(\Delta_{max}/\Delta_{min})) \cdot r(\mathcal{F}) \cdot \log|\mathcal{F}| \cdot |\mathcal{F}| \cdot |U(\mathcal{F})|^2)$.*

Proof. Let us define $\mathcal{F}_j = \{(T_i, I_i) \in \mathcal{F} | 2^j \leq |T_i| + |I_i| < 2^{j+1}\}$. Obviously, $\mathcal{F} = \bigcup\{\mathcal{F}_j | \lfloor\log\Delta_{min}\rfloor \leq j \leq \lfloor\log\Delta_{max}\rfloor\}$, i.e., the collection $\mathcal{F}$ is a union of disjoint subcollections $\mathcal{F}_j$. We compute an interference selective family $\mathcal{S}_j$ for each collection $\mathcal{F}_j$ separately. Then, the resulting interference selective family for the collection $\mathcal{F}$ is a family $\mathcal{S} = \bigcup\{\mathcal{S}_j | \lfloor\log\Delta_{min}\rfloor \leq j \leq \lfloor\log\Delta_{max}\rfloor\}$.

For $j = 0$, the computation of $\mathcal{S}_0$ is trivial. Otherwise, we utilize the procedure *IASF* with inputs $\mathcal{F}_j$ and $\Delta = 2^{j+1}$ in order to compute the family $\mathcal{S}_j$. Time complexity of the algorithm and the size of the computed family $\mathcal{S}$ follows from Lemma 3.2.5. □

Note that *ad-hoc selective families* defined and investigated by Clementi *et al.* in [18] are the special case of interference selective families with interference ratio equal to 0. In this case, $I = \emptyset$ for each $(T, I) \in \mathcal{F}$.

Interference selective families can be directly transformed to transmission schedules for information dissemination in bipartite *IRG*s. The most simple approach is to use them in the layer-by-layer dissemination of a source message from a source node s. The source message is disseminated in phases. During i-th phase, the source message is received by all nodes of the i-th layer $L_i = \{v | d_T(s, v) = i\}$ owing to transmissions of the nodes in the layer L_{i-1}. Particularly, for each node v of L_i, we construct a set-pair (T_v, I_v) such that

$T_v = \{w \in L_{i-1} | (w, v) \in E_T\}$ and $I_v = \{w \in L_{i-1} | (w, v) \in E_I\}$. Furthermore, for a collection $\mathcal{F}^i = \{(T_v, I_v) | v \in L_i\}$, an interference selective family $\mathcal{S}^i = \{S^i_1, \dots, S^i_m\}$ is obtained by the algorithm presented in Theorem 3.2.6. Finally, transmissions of the phase i are scheduled in such a way that in the j-th round of the phase i exactly the nodes of $S^i_j \subseteq L_{i-1}$ transmit the source message. Therefore, the i-th phase takes totally $|\mathcal{S}^i|$ rounds.

Theorem 3.2.7 ([D]). *Let $G = (V, E_T \cup E_I)$ be an IRG. There is a deterministic polynomial-time algorithm that for a given source node s produces a radio broadcasting schedule with length $O((1 + \log(\Delta_{max}/\Delta_{min})) \cdot R(s))$ rounds, where $R(s) = \sum_{i=0}^{ecc_T(s)-1}((1 + r(\mathcal{F}^i)) \cdot \log |\mathcal{F}^i|)$.*

Note that there are *IRG*s for which layer-by-layer information dissemination approach is not suitable. For instance, consider the following *IRG* $G_m = (V, E_T \cup E_I)$, where

- $V = \{s, v_1, \dots, v_m, w_1, \dots, w_m\}$
- $E_T = \{(s, v_i), (v_i, w_i) | i = 1, \dots, m\} \cup \{(v_i, v_j), (w_i, w_j) | 1 \leq i \neq j \leq m\}$
- $E_I = \{(v_i, w_j) | 1 \leq i \neq j \leq m\}$

Observe, that for each node of the constructed graph G_m the ratio of the incident interference edges to the incident transmission edges is at most 1. It is easy to see that it is not possible to accomplish the broadcasting task from the node s in less then $m + 1$ rounds realizing the layer-by-layer approach. Indeed, all nodes of the layer $L_1 = \{v_1, \dots, v_m\}$ have to transmit in separate rounds. However, broadcasting from the source s can be completed in 3 rounds:

1. the source s transmits and informs all nodes of the layer L_1
2. the node v_1 transmits and informs the node w_1
3. the node w_1 transmits and informs the remaining nodes.

The previous example shows that interference selective families cannot be directly used as a subroutine in algorithms which generate broadcasting schedules for arbitrary *IRG*s. Moreover, it shows that interference ratio is not an appropriate parameter for reflecting global presence and structure of interference edges in arbitrary *IRG*s. Indeed, interference ratio is not a hereditary property. I.e., it is not propagated from graphs to subgraphs. On the other hand, we believe that careful combination of interference selective families with some graph analysis and heuristics can lead to algorithms which generate radio broadcasting schedules of a sufficient length.

3.3 Information Dissemination in Bipartite IRGs

In this section, we focus on two algorithms generating transmission schedules completing information dissemination in bipartite IRGs. In both cases, length of a produced schedule is upper-bounded by an expression containing the maximum degree Δ and the number of network nodes n. Since these parameters are not increasing in subgraphs, designed algorithms are useful as subroutines for broadcasting in arbitrary IRGs. Moreover, generated schedules are 1-shot transmission schedules.

3.3.1 $O(\Delta^2)$-round Algorithm

The first algorithm for information dissemination in bipartite IRGs is based on a simple idea of proper vertex coloring.

Theorem 3.3.1 ([D]). *Let $G = (V_S \cup V_R, E_T \cup E_I)$ be a bipartite IRG and let Δ be the maximum degree in the IRG G, i.e., $\Delta = max\{deg_T(v) + deg_I(v) | v \in V_S \cup V_R\}$. There is a polynomial-time algorithm which generates* 1*-shot transmission schedule realizing information dissemination in the bipartite IRG G with the informed part V_S and the uninformed part V_R. Length of the schedule is at most Δ^2 rounds. Moreover, in the case when $deg_T(v) = 1$ for all $v \in V_S$, length of the generated schedule is at most* $2 \cdot \Delta - 1$ *rounds.*

Proof. Observe that we may assume that for each node $v \in V_R$ it holds $deg_T(v) = 1$. And indeed, if $deg_T(v) \geq 2$, we can change all transmission edges incident to the node v to interference edges. We define an undirected collision graph $G_c = (V_c, E_c)$. The vertex set of G_c is the set of informed nodes V_S, i.e., $V_c = V_S$. An edge $(u, v) \in E_c$ if and only if there is a node $w \in V_R$ such that $(u, w) \in E_T$ and $(v, w) \in E_I$. Hence, two nodes of V_S are connected by an edge in G_c if and only if they have a common neighbor in V_R that is a transmission neighbor of one node and an interference neighbor of the other node. Since each node in V_S has at most Δ neighbors in V_R and each of them has at most $\Delta - 1$ different neighbors in V_S, the maximum degree of the graph G_c is at most $\Delta \cdot (\Delta - 1) < \Delta^2$. Therefore, we can efficiently compute a proper coloring of vertices in G_c that uses at most Δ^2 colors. Denote a color assigned to a node v as $color(v)$. Let us consider a schedule of transmissions in which a node $v \in V_S$ transmits exactly in the round $color(v)$. The length of such a schedule is at most Δ^2 rounds and each node transmits exactly once. Observe that the schedule informs all nodes in V_R. On the contrary, assume that a node $w \in V_R$ is not informed after realizing the transmission schedule. Each of its neighbors in V_S transmits in one of rounds. Let $v \in V_S$ be a node such that $(v, w) \in E_T$, i.e., v is a transmission neighbor of w. The node v transmits in the round $color(v)$. Assume that another neighbor $u \in V_S$ of the node w transmits in the round $color(v)$. It follows that $color(u) = color(v)$. Since the node u is an interference neighbor of the node w, $(u, w) \in E_I$. However, it implies that $(u, v) \in G_c$ and therefore $color(u) \neq color(v)$.

In the case when $deg_T(v) = 1$ for all $v \in V_S$, notice that the maximum degree of the graph G_c is at most $2 \cdot (\Delta - 1)$. Indeed, the only transmission neighbor of a node v can bring to the node v at most $\Delta - 1$ different neighbors in G_c. Each of remaining $\Delta - 1$ interference neighbors of the node v is incident to exactly one transmission edge. Therefore, interference neighbors of the node v bring to the node v at most $\Delta - 1$ additional neighbors in G_c. Summarily, the node v has at most $2 \cdot (\Delta - 1)$ neighbors in G_c. It implies that the vertices of G_c can be colored by at most $2 \cdot \Delta - 1$ and the claim follows. □

3.3.2 $O(\Delta \cdot \log n)$-round Algorithm

The $O(\Delta^2)$-round algorithm is efficient for networks with small maximum degree. Now, we present an algorithm that generates shorter schedules in the case when $\Delta = \omega(\log n)$. Similarly as the $O(\Delta^2)$-round algorithm, it is based on the idea of the collision graph.

Lemma 3.3.2. *For any set of natural numbers $\{d_1, d_2, \ldots, d_n\}$, s.t., $d_i \geq 0$ and $\sum_{i=1}^{n} d_i < h \cdot n$, it holds that $|\{d_i | d_i \leq 2 \cdot h\}| \geq n/2$.*

Proof. Assume contrarily that $|\{d_i | d_i \leq 2 \cdot h\}| \leq n/2$ which is equivalent to $|\{d_i | d_i > 2 \cdot h\}| \geq n/2$. This implies $\sum_{i=1}^{n} d_i > 2 \cdot h \cdot \frac{n}{2}$ which contradicts one of the assumptions of the lemma. □

Theorem 3.3.3 ([F])**.** *Let $G = (V_S \cup V_R, E_T \cup E_I)$ be a bipartite IRG. Assume that all nodes in V_S are informed, i.e., they know the source message, and the nodes in V_R are uninformed. If $deg_T(v) \geq 1$, for all $v \in V_R$, there is a linear time algorithm that generates a 1-shot schedule of transmissions informing all nodes in V_R. The length of the schedule is $O(\Delta \cdot \log |V_S|)$ rounds.*

Proof. Without loss of generality, we may assume that each node $v \in V_R$ is incident to exactly one transmission edge (i.e., $deg_T(v) = 1$) and each node $v \in V_S$ is incident to at least one transmission edge (i.e., $deg_T(v) \geq 1$). And indeed, if $deg_T(v) > 1$ for a node $v \in V_R$, we keep one of its incident transmission edges and consider all other incident transmission edges as interference edges. Furthermore, we remove each node $v \in V_S$ such that $deg_T(v) = 0$. These modifications do not improve chances of nodes in V_R to be informed earlier. Thus, in view of the transmission subgraph, the graph G is a collection of disjoint stars of transmission edges with centers in V_S and terminal nodes in V_R. Each interference edge connects the center of a star with some terminal node of another star. We define the sets of nodes and edges in an undirected graph G_c as follows:

- $V(G_c) = V_S$,
- $E(G_c) = \{(u, v) | \exists w \in V_R, (u, w) \in E_T \wedge (v, w) \in E_I\}$.

The graph G_c is a graph whose nodes correspond to stars, where two nodes are connected by an edge in G_c if and only if there is an interference edge joining the center of one star and a terminal node of the other. Observe that each edge in G_c corresponds to a path of the length 2 in G. The path consists of an interference edge and a transmission edge. Since $deg_T(v) = 1$, for each $v \in V_R$, it follows that an interference edge in G can introduce at most one new edge to G_c. Hence, $|E(G_c)| \leq |E_I| < |E(G)| \leq \Delta \cdot |V_S|$. Denote the degree of a node $v \in V_S$ in the graph G_c as $deg_c(v)$. It follows that $\sum_{v \in V_S} deg_c(v) = 2 \cdot |E(G_c)| < 2 \cdot \Delta \cdot |V_S|$. Applying $h = 2 \cdot \Delta$ to Lemma 3.3.2 we get $|\{v \in V_S | deg_c(v) \leq 4 \cdot \Delta\}| \geq |V_S|/2$. I.e., at least half of the nodes in V_S have their degree in G_c lower than $4 \cdot \Delta$. Now, we remove all nodes with the degree strictly greater than $4 \cdot \Delta$ from G_c obtaining G'_c. Since the maximal degree in G'_c is less than $4 \cdot \Delta$, we can color efficiently [69] the nodes in G'_c using at most $4 \cdot \Delta + 1$ colors. Finally, observe that if the nodes in $V(G'_c)$ transmit in rounds corresponding to the assigned colors, all their transmission neighbors will be successfully informed. And indeed, if there is a node $w \in V_R$ and two nodes $u, v \in V(G'_c)$, s.t., $(u, w) \in E_T$ and $(v, w) \in E_I$, then $(u, v) \in E(G'_c)$. It follows that $color(u) \neq color(v)$, i.e., the nodes u and v transmit in different rounds. Summarizing , at least half of the nodes in V_S inform all their transmission neighbors during at most $4 \cdot \Delta + 1$ rounds. These nodes can be removed together with already informed nodes in V_R from the graph G. And iterating this process at most $\log |V_S|$ times, we obtain a $O(\Delta \cdot \log |V_S|)$-round schedule of transmissions informing all nodes in V_R.

Finally note that each node in V_S is prompted to transmit at most once in the schedule. This is due to the fact that each node in V_S acts as a node of G'_c in at most one iteration of the algorithm and during each iteration, each node transmits at most once. Thus we obtained a 1-shot schedule of transmissions. □

For this algorithm, we provide also more detailed time complexity analysis.

Lemma 3.3.4. *The algorithm from Theorem 3.3.3 computes a transmission schedule in linear time.*

Proof. Assume that the input graph is represented by a list of its edges. We denote an instance of G_c in the i-th iteration of the algorithm by G_c^i. Observe that for any iteration $i > j$, we have $G_c^i = G_c^j[V(G_c^i)]$. I.e., G_c^i is an induced subgraph of G_c^j on a given subset of nodes. In other words, G_c^{i+1} can be obtained from G_c^i by removing appropriate nodes and all edges incident to them.

In what follows we identify all nodes in G with unique numbers from the set $\{1, \ldots, |V(G)|\}$. Initially, we construct the graph G_c using nodes in V_S. During the first iteration over the list of edges of G, we move certain edges from E_T to E_I in order to satisfy condition that each node in V_R is incident to exactly one transmission edge. Simultaneously, we determine the center of a star, denoted as $p(v)$, for each node $v \in V_R$, i.e., a unique node such that $(p(v), v) \in E_T$. Later, we iterate over all edges in E_I. For each edge $(u, v) \in E_I$ where $u \in V_S$ and $v \in V_R$, we add a new pair $(u, p(v))$ (or $(p(v), u)$, for $p(v) < u$) to a list L of pairs. Note that $|L| \leq |E_I|$. We can order all pairs in time $O(|V(G)| + |E(G)|)$ utilizing the *Counting sort* algorithm [21] (stable sorting) twice: first time according to second elements of pairs and second time according to first elements in each pair. Upon the completion of sorting, we remove duplicate pairs using single iteration over the list of pairs. Observe that these pairs correspond to edges of G_c^1. Finally, searching through the list of pairs, we build a double-linked list of incident edges for each node $v \in V_S$ and compute its degree. It follows that the construction of G_c^1 including all related data structures takes time $O(|V(G)| + |E(G)|) = O(|E(G)|)$.

Now, we describe the computation during the i-th iteration of the algorithm ($i \geq 1$), i.e., the processing of G_c^i and the construction of G_c^{i+1}. Let V_i be the list of nodes in the graph G_c^i ($V_i = V(G_c^i)$). Initially, $V_0 = V_S$. Two lists V_i and V_i' are created during single iteration over the list V_{i-1}. The nodes in V_{i-1} with degree smaller than $4 \cdot \Delta$ get inserted to the list V_i' and the remaining nodes to the list V_i. An extra information that $v \in V_i'$ is stored in the data structure associated with v. Later, we go along the list V_i' and calculate for each node $v \in V_i'$ its color, the transmission round. In order to compute a specific color for each node v, we have to pick up a color $\mathit{color}(v) \in \{1, \ldots, 4 \cdot \Delta + 1\}$ which is not used by any adjacent node of v in V_i'. Note that a round number during which the node v transmits in the generated schedule is $(i-1) \cdot (4 \cdot \Delta + 1) + \mathit{color}(v)$. To finalize the coloring of a node in time proportional to the number of incident edges, we use an additional array of integers U of length $4 \cdot \Delta + 1$. The array U is initialized only once with zeros. The node v visits all its incident edges. If

(v, w) is an edge incident to v, $w \in V_i'$, and w has already an assigned color $color(w)$, then we set $U[color(w)]$ to v. This is to make a note that the color $color(w)$ is used in the considered V_i'-neighborhood of the node v. Note that $U[c] \neq v$ when c is a color unused in the neighborhood of v. Moreover, if there is at least one used color in the V_i'-neighborhood of v (which can be easily verified), there must exist an unused color c such that $U[c+1] = v$ or $U[c-1] = v$. I.e., an unused color which is larger or smaller by one in relation to the used color. Hence, during further iteration over all edges incident to v, we are able to determine an unused color in considered neighborhood of v and this color is assigned to the node v. After the colors (transmission rounds) of all nodes in V_i' are computed, we remove the nodes together with all edges incident to them from the maintained graph data structure. Later, we go along the list of nodes in V_i' and for each node along its double-linked list of incident edges. Each edge is removed from the graph and degrees of incident nodes are decremented respectively. Each edge can be removed in $O(1)$ time utilizing the double-linked list of incident edges provided at each node. Since $|V_i| \leq |V(G)|/2^i$ and $\sum |V_i'| = |V(G)|$, the total time devoted to V_i and V_i' lists is $O(|V(G)|)$. Finally, since each edge is processed in $O(1)$ time the total processing time in this case is $O(|E(G)|)$.

The thesis of the lemma follows. □

Observation 3.3.5. *Since an IRG without interference edges corresponds to a reachability graph in the standard graph model, the presented algorithm can be used to generate* 1*-shot schedules for radio networks modelled by the standard graph model. Produced* $O(\Delta \cdot \log n)$*-round schedules is an alternative to* $O(\sqrt{n})$*-time schedules generated by the algorithm in [33].*

3.4 General Schema for Broadcasting in *IRG*s

In the rest of this chapter, we focus on broadcasting in arbitrary *IRG*s. Observe that the transmission subgraph of an *IRG* corresponds to a communication subnetwork that is very close to radio network modelled by the standard graph model. One approach to generate short broadcasting schedules for arbitrary *IRG*s is adaption of known broadcasting algorithms designed for the standard graph model. In particular, a known algorithm can take into account only transmission subgraph during computation of a broadcasting schedule. Naturally, presence of interference edges can cause that the produced schedule does not inform all network nodes. Therefore, we should carefully modify known ideas and algorithms in such a way that they will work properly even in the case when there are interference edges in the network. In this section, we present a schema that is based on ideas of the schema in Theorem 2.2.7 and works correctly for arbitrary IRGs.

Let $\mathcal{A}_F$ and $\mathcal{A}_S$ be algorithms that generate transmission schedules for information dissemination in bipartite IRG which are subgraphs of an input IRG G. Difference between $\mathcal{A}_F$ and $\mathcal{A}_S$ is that the algorithm $\mathcal{A}_F$ accepts as input only such a bipartite IRG $G_b = (V_S \cup V_R, E_T \cup E_I)$ that $deg_T(v) = 1$ for all $v \in V_S$. Therefore, $\mathcal{A}_F$ can generate much shorter schedules for these input IRGs. Let us denote the maximum length of a schedule generated by the algorithm $\mathcal{A}_F$ for an arbitrary vertex induced bipartite subgraph (satisfying the property that each informed node has exactly one uninformed transmission neighbor) of the graph G as $time_F(G)$. Similarly, denote the maximum length of a schedule generated by the algorithm $\mathcal{A}_S$ for an arbitrary vertex induced bipartite subgraph of the graph G as $time_S(G)$. Since the network topology is known, the resulting broadcasting schedule is computed by simulating communication in the network. In certain moments of the communication process, the algorithms $\mathcal{A}_F$ and $\mathcal{A}_S$ are applied to compute schedule of transmissions for some number of subsequent communication rounds. Input graphs for these algorithms are determined at the moment when these algorithms are applied and depend on the actual configuration of the network (sets of informed and uninformed nodes). Recall that algorithms $\mathcal{A}_F$ and $\mathcal{A}_S$ produce schedules for IRGs, i.e., produced schedule work properly even when there is interference caused by interference edges. All transmissions realized as a result of a schedule computed by the algorithm $\mathcal{A}_F$ are called *fast transmissions.* Transmissions realized as a result of a schedule produced by the algorithm $\mathcal{A}_S$ are called *slow transmission.*

The broadcasting schema relies on properties of the gathering-broadcasting spanning tree ($GBST$) computed for transmission subgraph G_T of an IRG G. Construction and properties of gathering-broadcasting spanning trees can be found in Section 2.2.1. In what follows, we assume that $GBST$ of transmission subgraph and ranks of all nodes are already computed. Basic ideas of the broadcasting schema are similar to ideas of the schema from Theorem 2.2.7. However, we should schedule transmission of nodes in a different way in order to properly treat presence of interference edges.

Broadcasting is realized in consecutive composite rounds. Each composite round consists of two rounds, one reserved for fast transmissions (generated by $\mathcal{A}_F$) and the other reserved for slow transmissions (generated by $\mathcal{A}_S$). This time multiplexing strategy eliminates interference between fast and slow transmissions which realize different communication patterns. Schedules for fast and slow transmissions are computed independently. Both subschedules work in stages. The length of one stage of fast transmission is $time_F(G)$ composite rounds and the length of one stage of slow transmissions is $time_S(G)$

composite rounds. Notice that stages can have different length, i.e., stages of fast and slow transmissions are not synchronized, it may happen that $time_F(G) \neq time_S(G)$.

Communication pattern of slow transmissions is computed in the following way. At the beginning of each stage, the network nodes can be partitioned into two sets: a set of informed nodes and a set of uninformed nodes. Let V_S be a set of informed nodes which have at least one uninformed transmission neighbor. Further, let V_R a set of all uninformed transmission neighbors of nodes in the set V_S. These two sets V_S and V_R induce a bipartite IRG $G^S_{V_S,V_R}$ which is subgraph of the input graph G. Note that we do not consider edges among nodes of the same part since their presence does not effect the communication process. Further, we apply the algorithm $\mathcal{A}_S$ to compute transmission schedule for information dissemination in the bipartite graph $G^S_{V_S,V_R}$ with V_S as the informed part and V_R as the uninformed part. The length of generated schedule is at most $time_S(G)$ rounds, i.e., the schedule fits in one stage of slow transmissions. During the whole stage, all network nodes follow this generated schedule of transmissions. Observe that all transmission neighbors of nodes which are informed at the beginning of the stage of slow transmissions become informed at the end of this stage due to properties of the schedule generated by $\mathcal{A}_S$.

Communication pattern realized in the segment of fast transmissions is computed in the similar way as for slow transmissions. A bipartite subgraph is constructed at the beginning of each stage. Let V_R be a set of all uninformed nodes that have informed parent with the same rank. I.e., $V_R = \{v \in F | v \text{ is informed} \wedge parent(v) \text{ is informed}\}$. A set of parents of nodes in the set V_R is denoted as V_S, i.e., $V_S = \{parent(v) | v \in V_R\}$. We construct a bipartite subgraph $G^F_{V_S,V_R} = (V_S \cup V_R, E^F_T \cup E^F_I)$ where $E^F_T = \{(v, parent(v) | v \in V_R\}$ and $E^F_I = \{(v, w) | v \in V_R \wedge w \in V_S \wedge w \neq parent(v) \wedge (v, w) \in E_T \cup E_I\}$. Notice that the edge set of the graph $G^F_{V_S,V_R}$ contains all edges that join a node in the informed part V_S with a node in the uninformed part V_R. However, we move each transmission edge which does not join a node with its parent in the $GBST$ to the set of interference edges. After this change, each node in V_R has a chance to become informed by transmission of its parent. Owing to this change, each node $v \in V_S$ in the bipartite IRG $G^F_{V_S,V_R}$ satisfies the property that $deg_T(v) = 1$. Indeed, the properties of $GBST$ imply that each node, except the source, has exactly one parent and each parent has at most one child with the same rank, i.e., at most one child from the set F. Notice that the graph $G^F_{V_S,V_R}$ is a valid input bipartite IRG for the algorithm $\mathcal{A}_F$. Applying algorithm $\mathcal{A}_F$, we get a schedule of transmissions informing all nodes in V_R. The length of the schedule fits in one stage of fast transmissions. Similarly as in the case of slow transmissions, all network nodes follow the computed schedule during this stage.

The following lemma provides crucial description how fast the source message is disseminated in the network.

Lemma 3.4.1. *Let v be a node with the rank i and in the distance k from the source s in the transmission subgraph G_T, i.e., $rank(v) = i$ and $v \in L_k$. The node v is informed in the composite round $2 \cdot k \cdot time_F(G) + 2 \cdot (rank(s) - i) \cdot time_S(G)$ at the latest.*

Proof. The proof is done by induction on the layer number of a node. For the source, the claim trivially holds. Now, we prove the claim for a node $v \in L_k \cap R_i$ under assumption that induction hypothesis holds. Since the node v is not the source s, it has a parent in the $GBST$. In the transmission subgraph, the parent of the node v is closer to the source than the node v, i.e., $parent(v) \in L_{k-1}$. Denote a composite round in which its parent $parent(v)$ becomes informed as T_P. There are two possible cases. Either the node v has the same rank as its parent in the $GBST$ or not.

Let us analyze the former case, i.e., the case when $v \in F$. The induction hypothesis implies that $T_P \leq 2 \cdot (k-1) \cdot time_F(G) + 2 \cdot (rank(s) - i) \cdot time_S(G)$. Communication pattern for fast transmissions prescribes that if the node v is not informed at the beginning of the nearest stage of fast transmissions which comes after the composite round T_P, both the node v and its parent $parent(v)$ participate in this stage of fast transmissions. The node v participates as a node of the uninformed part of a bipartite IRG which is constructed for this stage, and its parent $parent(v)$ as a node of the informed part. Since the schedule computed for this stage by algorithm $\mathcal{A}_F$ informs all nodes in the uninformed part, the node v becomes informed at the end of this stage at the latest. After the round T_P, we wait at most $time_F(G)$ composite rounds for the beginning of next stage of fast transmissions. Thereafter, the node v becomes informed during at most $time_F(G)$ subsequent composite rounds. It follows that the node v is informed in the composite round $T_P + 2 \cdot time_F(G) \leq 2 \cdot k \cdot time_F(G) + 2 \cdot (rank(s) - i) \cdot time_S(G)$ at the latest.

In the complementary case, we have that $v \in S$. Lemma 2.2.1 implies that $rank(parent(v)) > rank(v)$. According to designed communication pattern for slow transmissions, if the node v is not informed at the beginning of the nearest stage of slow transmission, both the nodes v and $parent(v)$ participate in this nearest full stage of slow transmissions. The node $parent(v)$ is a node of the informed part and the node v of the uninformed part. Schedule generated by the algorithm $\mathcal{A}_S$ informs the node v during this stage. Notice that we wait for the beginning of the next stage of slow transmission at most $time_S(G)$ composite rounds. Thereafter, it takes at most $time_S(G)$ composite rounds to inform the node v. Hence, the node v is informed in the composite round $T_P + 2 \cdot time_S(G)$ at the latest. The induction hypothesis for the node $parent(v)$ implies that $T_P \leq 2 \cdot (k-1) \cdot time_F(G) + 2 \cdot (rank(s) - rank(parent(v)) \cdot time_S(G)$. Since $rank(parent(v)) > rank(v)$, $T_P + 2 \cdot time_S(G) \leq 2 \cdot k \cdot time_F(G) + 2 \cdot (rank(s) - i) \cdot time_S(G)$. □

Theorem 3.4.2. *For an arbitrary given IRG $G = (V, E_T \cup E_I)$ and two algorithm $\mathcal{A}_S$ and $\mathcal{A}_F$, a radio broadcasting schedule with length $O(time_F(G) \cdot ecc_T(s) + time_S(G) \cdot \log n)$ can be generated in polynomial time. Moreover, if the algorithm $\mathcal{A}_F$ generates k_F-shot schedules of transmissions and the algorithm $\mathcal{A}_S$ k_S-shot transmission schedules, generated broadcasting schedules are $(k_F + k_S)$-shot broadcasting schedules.*

Proof. Lemma 2.2.1 implies that $rank(s) \leq \log n$. Therefore, the upper-bound on the length of generated broadcasting schedules follows immediately from Lemma 3.4.1.

Observe that each node participates as a node of the informed part, i.e., as a transmitter, in at most one stage of slow transmission and in at most one stage of fast transmissions (each node has at most one child with the same rank). This implies that generated schedules are $(k_F + k_S)$-shot broadcasting schedules. □

Corollary 3.4.3. *For an arbitrary given IRG $G = (V, E_T \cup E_I)$, a 2-shot radio broadcasting schedule with length $O(\Delta \cdot D_T + \Delta^2 \cdot \log n)$ can be generated in polynomial time.*

Proof. The claim follows from Theorem 3.4.2 applying the second algorithm from Theorem 3.3.1 as $\mathcal{A}_F$ and the first algorithm from the same Theorem 3.3.1 as $\mathcal{A}_S$. □

Corollary 3.4.4. *For an arbitrary given IRG $G = (V, E_T \cup E_I)$, a 2-shot radio broadcasting schedule with length $O(\Delta \cdot (D_T + \log^2 n))$ can be generated in polynomial time.*

Proof. The claim follows from Theorem 3.4.2, if we use the second algorithm from Theorem 3.3.1 as $\mathcal{A}_F$ and the algorithm from Theorem 3.3.3 as $\mathcal{A}_S$. □

3.5 Energy and Time Efficient Broadcasting in *IRG*s

All previously designed algorithms devoted to communication in *IRG*s reflect the presence of interference edges by the combined maximum degree Δ. In the case of interference selective families, we have considered interference ratio as another parameter. Intuitively, if we take into account more parameters, we get finer upper-bounds on lengths of broadcasting schedules. In this section, we introduce and investigate another parameter reflecting structure and presence of interference edges - an *interference distance* of an IRG. We say that a network (represented by an IRG) has the *interference distance* d_I if d_I is the smallest integer, s.t., for any interference edge $e \in E_I$ there exists a transmission path along at most d_I transmission edges connecting the endpoints of e.

In real-world scenarios, if there is an interference edge between two nodes, we expect that these nodes are relatively close to each other. Therefore, we expect that there is a connection (transmission path) between these two nodes which uses small number of transmission edges. Hence, we may assume that interference distance of an underlying IRG is small comparing to size of the network (for instance, $O(1)$ or $O(\log n)$). Note that reachability graphs of the standard graph model correspond to IRGs with interference distance equal to 1.

The algorithm presented in this section is designed with respect to minimization of energy consumption. In particular, broadcasting schedules produced by this algorithm are not only fast, but they are also 1-shot broadcasting schedules.

3.5.1 Overview of the Algorithm

Our next algorithm generating broadcasting schedules adopts an approach in which the set of transmissions is divided into *fast* and *slow* transmissions. As we have seen, this universal paradigm is used with various modifications in almost all algorithms generating broadcasting schedules in known radio networks. In this approach, a source message is disseminated along branches of a BFS tree-like subnetwork using fast transmissions pipelined along selected simple paths and a limited number of slow transmissions based on propagation of information in bipartite graphs. In our approach we utilize, with required modifications, the clustering mechanism presented by Gaber and Mansour in [31], however, the presence of interference edges imposes certain structural changes. A notable difference is that the constructed clusters do not form a connected subgraph, although there is a short transmission path between any two nodes of the same cluster. Another difference refers to the lack of short transmission paths between the clusters of the same color (rather than the lack of direct edges) that enables efficient dissemination mechanism in radio networks with long-range interference. Finally, whereas fast transmissions are realized on the basis of the tree-like communication subnetwork, slow transmission are executed in successive stages in the form of a flooding mechanism where information is disseminated between collections of informed and neighboring uninformed nodes of dynamically evolving bipartite IRG. Note that the slow and fast transmissions despite being treated separately, they must be neatly coordinated to enable energy efficient 1-shot communication protocol.

3.5.2 Construction of Clusters

Layers and super-layers

The broadcast schedule designed for IRGs utilizes a partition of the input graph G into BFS layers, super-layers and a collection of overlapping clusters constructed on the basis of the transmission subgraph G_T in G. Due to Definition 3.1.1, the input IRG G consists of $ecc_T(s)+1$ layers $L_0, L_1, \ldots, L_{ecc_T(s)}$ with respect to the source node s.

In what follows, x denotes a parameter whose value will be determined later in Lemma 3.5.10.

Definition 3.5.1. *For each $i = k \cdot x \leq ecc_T(s)+1$, the layer L_i of G is called a k-th* inter-communication layer. *The union of layers $\mathcal{L}_k = \bigcup\{L_i | k \cdot x \leq i \leq min(ecc_T(s), (k+1) \cdot x)\}$ forms the k-th super-layer of the IRG G. The layers $L_{k \cdot x}$ and $L_{min(ecc_T(s),(k+1)\cdot x)}$ form the highest and the lowest layers in the k-th super-layer respectively.*

The last definition implies that each super-layer consists of $x+1$ layers and exactly two inter-communication layers: the highest and the lowest layer of the super-layer. The exemption is the most distant super-layer that might have smaller number of layers.

Clusters, pre-clusters and pre-cluster graphs

The clusters are built in each super-layer of G independently. Each cluster is a union of carefully crafted *pre-clusters* defined as follows. For each node $v \in L_{k \cdot x} \subseteq \mathcal{L}_k$ we define a pre-cluster as a set of nodes $S(v) = \{u | u \in L_{k \cdot x + i} \wedge d_T(v, u) = i \wedge 0 \leq i \leq x\}$. Note that the pre-cluster $S(v)$ contains all nodes in the same super-layer that are reachable from v along transmission edges used in the direction away from the source. The node v is called the *top node* of $S(v)$.

We define an undirected *pre-cluster graph* G_k for a given super-layer $\mathcal{L}_k$ as follows:

- $V(G_k) = L_{k \cdot x}$,

- $E(G_k) = \{(u,v) | \exists u', v' : u' \in S(u) \wedge v' \in S(v) \wedge d_T(u',v') \leq d_I(G)+1\}$.

The nodes in G_k correspond to the top nodes of pre-clusters located in the $k-$th super-layer. For any pair of nodes $u, v \in V(G_k)$ there is an edge connecting them in $E(G_k)$ if and only if there is a transmission path of length not exceeding $d_I(G)+1$ that connects some node in $S(u)$ with some node in $S(v)$. The structure of G_k guaranties, e.g., that there are no interference edges between nodes from different pre-clusters whose top nodes are not connected by an edge in G_k. In fact, the structure of G_k implies also several other powerful properties that are summarized in Lemma 3.5.3.

The system of clusters

The following theorem is due to Gaber and Mansour, see [31].

Theorem 3.5.2 ([31]). *Let $G = (V, E)$ be an undirected graph. There exists a clustering $\mathcal{C} = \{C_1, \ldots, C_r\}$ of G with the following properties:*

1. $V(G) = \bigcup\{C_i | 1 \leq i \leq r\}$
2. *$G[C_i]$ is a connected subgraph of G with a diameter at most $2 \cdot \log |V(G)|$*
3. *There is a proper coloring of clusters with at most $\lceil \log |V(G)| \rceil$ colors. The proper coloring of clusters satisfies*

   $(u = v \vee (u,v) \in E(G)) \wedge u \in C_i \wedge v \in C_j \wedge i \neq j \Rightarrow color(C_i) \neq color(C_j)$.
4. $|\mathcal{C}| \leq |V(G)|$

In addition, the clustering can be constructed in $O(|E(G)| \cdot \log |V(G)|)$ time.

Let $\mathcal{C} = \{C_1, \ldots, C_r\}$ be a clustering of the pre-cluster graph G_k obtained by application of the clustering procedure from [31]. A clustering $\mathcal{C}^{(k)} = \{C_1^{(k)}, \ldots, C_r^{(k)}\}$ of pre-clusters in the k-th super-layer $\mathcal{L}_k$ is defined as $C_i^{(k)} = \bigcup\{S(v) | v \in L_{k \cdot x} \wedge v \in C_i\}$. Note that a color assigned to a cluster $C_i^{(k)}$ is the same as the color assigned to the cluster C_i in the clustering $\mathcal{C}$ of G_k. It also follows that coloring of $\mathcal{C}^{(k)}$ uses at most $\lceil \log |V(G_k)| \rceil \leq \lceil \log |\mathcal{L}_k| \rceil$ colors.

Lemma 3.5.3 ([F]). *The clustering $\mathcal{C}^{(k)}$ has the following properties:*

1. $\mathcal{L}_k = \bigcup\{C_i^{(k)} | 1 \leq i \leq |\mathcal{C}^{(k)}|\}$
2. $\forall u, v \in V(G) : u, v \in C_i^{(k)} \Rightarrow d_T(u,v) \leq 6 \cdot (d_I(G) + x) \cdot \log |\mathcal{L}_k|$

3. $\forall u, v, w \in V(G) : u \in C_i^{(k)} \wedge v \in C_j^{(k)} \wedge i \neq j \wedge color(C_i^{(k)}) = color(C_j^{(k)}) \wedge (u, w) \in E_T \Rightarrow (v, w) \notin E_T \cup E_I$

4. $|\mathcal{C}^{(k)}| \leq |L_{k \cdot x}|$

Proof. The property (1) is a consequence of property (1) in Theorem 3.5.2 and the definition of $\mathcal{C}^{(k)}$.

The property (2) says that for any pair of nodes $u, v \in C_i^{(k)}$ there exists a transmission path in G of a length less than $6 \cdot (d_I(G)+x) \cdot \log |\mathcal{L}_k|$. Note that this transmission path can go through nodes located outside of the cluster $C_i^{(k)}$. Hence, any two cluster nodes are connected in G_T, but not necessary in $G[C_i^{(k)}]$, i.e., the cluster $C_i^{(k)}$ can be disconnected. From the definition of $C_i^{(k)}$ one can conclude that there are two nodes $u_h, v_h \in C_i$, s.t., $u \in S(u_h)$ and $v \in S(v_h)$. Recall that C_i is a cluster defined on the pre-cluster graph G_k that determines the content of the cluster $C_i^{(k)}$. The property (2) of Theorem 3.5.2 implies that there exists a path $P = (u_h = w_1, w_2, \ldots, w_t = v_h)$ connecting nodes u_h and v_h in the pre-cluster graph G_k. Moreover, this property implies that the length of P is at most $2 \cdot \log |V(G_k)| = 2 \cdot \log |\mathcal{L}_k|$. From the definition of G_k one can conclude that there is a path of a length at most $d_I(G) + 1$ from a node in $S(w_i)$ to a node in $S(w_{i+1})$ in the transmission subgraph G_T, for any $i = 1, \ldots, t-1$. We denote this path by $P(w_i, w_{i+1})$, its first node by w_i^F and the last node by w_{i+1}^L. Since $w_i^F \in S(w_i)$, we get $d_T(w_i, w_i^F) \leq x$. Similarly, from $w_{i+1}^L \in S(w_{i+1})$ we obtain $d_T(w_{i+1}, w_{i+1}^L) \leq x$. Hence, there is a walk from w_i to w_{i+1} in G_T consisting of three parts: a path from w_i to w_i^F, the path $P(w_i, w_{i+1})$ from w_i^F to w_{i+1}^L, and a path from w_{i+1}^L to w_{i+1}. The total length of the walk is at most $x + (d_I(G) + 1) + x$. Now, we are ready to construct a walk from u to v in G_T. The walk starts with a path from u to $u_h = w_1$. Further, it continues by a sequence of walks from w_i to w_{i+1}, for all $i = 1, \ldots, t-1$. The walk concludes with a path from $w_t = v_h$ to v. The total length of the constructed walk is at most $x + 2 \cdot \log |\mathcal{L}_k| \cdot (x + (d_I(G) + 1) + x) + x \leq 6 \cdot (d_I(G) + x) \cdot \log |\mathcal{L}_k|$. Hence, there exists a transmission path in G from u to v of length not exceeding $6 \cdot (d_I(G) + x) \cdot \log |\mathcal{L}_k|$.

The property (3) is proved by showing that there does not exist a transmission path in G of length smaller than $d_I(G) + 1$ that joins two nodes u, v in two different clusters with the same color. I.e., we must show that $d_T(u, v) > d_I(G) + 1$. If there is no transmission path from u to v, $d_T(u, v) = \infty$. And indeed, for any node $w \in V(G)$ connected to u by a transmission edge, i.e., $(u, w) \in E_T$, the inequality $d_T(u, v) > d_I(G) + 1 \geq 2$ implies that $d_T(w, v) > d_I(G)$. Hence, due to the definition of $d_I(G)$, we get that $(w, v) \notin E_T \cup E_I$. We show now that $d_T(u, v) > d_I(G) + 1$. Assume opposite, i.e., a transmission path from u to v exists and $d_T(u,v) \leq d_I(G)+1$. Since $u \in C_i^{(k)}$, there is a node $u_h \in L_{k\cdot x}$, s.t., $u \in S(u_h)$ and $u_h \in C_i$, where C_i is one of the clusters in G_k. Similarly, $v \in C_j^{(k)}$ implies that there is a node $v_h \in L_{k\cdot x}$, s.t., $v \in S(v_h)$ and $v_h \in C_j$. Recall that we assumed $d_T(u,v) \leq d_I(G)+1$. From the definition of G_k, it follows that $(u_h, v_h) \in E(G_k)$ and from the clustering construction we know that $color(C_i) = color(C_i^{(k)}) = color(C_j^{(k)}) = color(C_j)$. However, this leads to a contradiction since the coloring of clusters in G_k is a proper coloring. I.e., $(u_h, v_h) \in E(G_k)$ implies that $color(C_i) \neq color(C_j)$.

Finally, the property (4) follows directly from the property (4) in Theorem 3.5.2. □

Observe that the clustering $\mathcal{C}_k$ does not contain necessarily only internally connected clusters. In other words, a transmission path connecting two nodes of the same cluster can traverse through nodes outside of pre-clusters contributing to the cluster. On the other hand, the property (2) in Lemma 3.5.3 implies that the nodes of each cluster in $\mathcal{C}_k$ are connected in the graph G by relatively short transmission paths. Note that one can modify the clustering mechanism of Theorem 3.5.2 to obtain a clustering $\mathcal{C}_k$ that consists of internally connected clusters. The key idea of the modification lies in a way, how pre-clusters are gradually introduced to the currently constructed cluster. Instead of adding all unused neighboring pre-clusters of the currently constructed cluster, we insert unused pre-clusters containing at least one node which is connected by a transmission path of a length at most $d_I(G)+1$ to some node of the currently constructed cluster.

3.5.3 Construction of Ranked Trees of Clusters

The tree of clusters defines a parent-child relationship between clusters in neighboring super-layers and vice-versa. It is built in consecutive steps from the lowest (the most distant from the source) super-layer towards the highest super-layer with an index 0. During the bottom-up construction we process each cluster C such that we:

- provide a rank $rank(C)$ to the cluster,
- choose a node, called a *representative* of the cluster, in the highest layer of the cluster,
- select a unique *leading representative* in the lowest layer among representatives of cluster children of C, and

- define the *cluster path* as any shortest transmission path from the representative of C to the leading representative of its cluster children.

The tree of clusters is built on the basis of an arbitrary BFS tree formed of cluster nodes in consecutive super-layers. I.e., for each cluster, we choose a cluster in the neighboring higher super-layer, which will stand as its parent in the constructed tree of clusters. In what follows we show how each cluster in the k-th super-layer is processed in due course. Assume that all clusters in the $(k+1)$-th super-layer have been already processed and that we currently process the cluster $C_i^{(k)}$. Let $child(C_i^{(k)}) \subseteq \mathcal{C}^{(k+1)}$ be a (possibly empty) set of clusters in $\mathcal{C}^{(k+1)}$ whose parent is $C_i^{(k)}$. In the case when $child(C_i^{(k)}) = \emptyset$, the rank $rank(C_i^{(k)})$ of the cluster $C_i^{(k)}$ is set to 0 and an arbitrary node in the highest layer of the cluster is chosen as the representative of the cluster. In this case we do not define the cluster path and the leading representative due to the lack of cluster children. If, however, $child(C_i^{(k)}) \neq \emptyset$, i.e., the cluster $C_i^{(k)}$ is a parent cluster of one or more clusters at the $(k+1)$-th super-layer, we process the cluster as follows. Let $r_{max} = max\{rank(C) | C \in child(C_i^{(k)})\}$ be the maximal rank among all ranks of its cluster children. If at least two cluster children in $child(C_i^{(k)})$ have ranks with value r_{max}, i.e., $|\{C | C \in child(C_i^{(k)}) \wedge rank(C) = r_{max}\}| \geq 2$, the rank of $C_i^{(k)}$ is set to $r_{max}+1$ and the leading representative is chosen arbitrarily among representatives of the cluster children. Otherwise, the rank is set to r_{max} and the representative of the cluster child with the rank r_{max} is chosen as the leading representative of the cluster. Let u be the chosen leading representative. Since each cluster is a union of pre-clusters, there is a node $v \in L_{k \cdot x} \cap C_i^{(k)}$ and a pre-cluster $S(v)$, s.t., $u \in S(v)$. We choose v as the representative of $C_i^{(k)}$ and one of the shortest transmission paths from v to u is set as the cluster path. Since $u \in S(v)$, the length of the cluster path is x. Note that certain clusters with the rank 0 can be formed of less than $x+1$ layers. In such shallow clusters cluster paths are not defined. Finally, the parent of the cluster is an arbitrary cluster of the $(k-1)$-th super-layer $\mathcal{C}^{(k-1)}$ that contains the representative v of the cluster $C_i^{(k)}$.

The following lemma follows directly from the construction of the cluster tree.

Lemma 3.5.4 ([F]). *The cluster tree has the following properties:*

1. *If $C_j^{(k-1)}$ is a parent cluster of $C_i^{(k)}$, it holds $rank(C_j^{(k-1)}) \geq rank(C_i^{(k)})$,*
2. *For any cluster $C_i^{(k)}$, it holds that $|\{C | C \in child(C_i^{(k)}) \wedge rank(C) = rank(C_i^{(k)})\}| \leq 1$,*

3. *Let $C_i^{(k)}$ be a cluster and $C_j^{(k+1)} \in child(C_i^{(k)})$ be a cluster child of $C_i^{(k)}$, s.t., $rank(C_i^{(k)}) = rank(C_j^{(k+1)})$. Then, there is a cluster path of the length x in $C_i^{(k)}$ that connect the representative of $C_i^{(k)}$ with the representative $C_j^{(k+1)}$, which is the leading representative of $C_i^{(k)}$.*

We define a *reverse rank* of a cluster $C_i^{(k)}$ as $rank^*(C_i^{(k)}) = rank(C_0^{(0)}) - rank(C_i^{(k)})$. The following properties of (reverse) ranks in the tree of clusters follow from discussion in [31, 36].

Lemma 3.5.5 ([F]). *The greatest rank in the cluster tree is assigned to the root and its value is at most* $\log n$. *The greatest reverse rank of a cluster is also at most* $\log n$. *Each simple path from the root to any other cluster forms a non-increasing (non-decreasing) sequence of (reverse) ranks.*

3.5.4 Building the Broadcasting Schedule

The broadcasting schedule is implemented as a concurrent (interleaved) execution of two communication patterns of *fast transmissions* and *slow transmissions*. As we will explain later, it is important that fast transmissions do not interfere with slow transmissions and vice versa. Thus transmissions coming from different patterns are executed in disjoint time steps. A produced schedule consists of consecutive *composite rounds*, in which the first round is reserved for fast transmissions and the second one serves slow transmissions.

Pattern of fast transmissions

The main aim of fast transmissions is to disseminate the source message from the cluster representative to the leading representative in the same cluster along the cluster path. In order to avoid interference of simultaneous fast transmissions coming from clusters with different colors, the transmissions are scheduled according to the distance from the source and colors of clusters. In fact, a node on a cluster path has a licence to transmit only in very specific composite rounds. We may picture this permission to transmit as a (time) wave emitted by the source s and descending gradually along the consecutive BFS layers of the input IRG G. Let $t_e = d_I(G) + 2$ be a *period* of this wave, i.e., the frequency of issuing the permission. Further, consider a node $v \in L_z = L_{k \cdot x+y}$, for some $0 \leq y < x$. Let $\mathcal{C}_P(v) \subseteq \mathcal{C}^{(k)}$ be a collection of all clusters in $\mathcal{C}^{(k)}$, s.t., the node v belongs to their cluster paths. Note that we exclude here the nodes of the layer $L_{(k+1)\cdot x} = \mathcal{L}_k \cap \mathcal{L}_{k+1}$, i.e., the lowest layer of the k-th super-layer, from cluster paths of clusters in $\mathcal{C}^{(k)}$. The node v has a license to transmit during t-th composite round, as a part of the pattern of fast transmissions, if and only if $t \equiv (z + c \cdot t_e) \pmod{t_e \cdot \lceil \log n \rceil}$, where $c \in \{\mathit{color}(C) | C \in \mathcal{C}_P(v)\}$. I.e., c is a color of a cluster containing v on its cluster path.

Summarizing, a node v contributes to fast transmissions in a composite round t if:

(1) v is already informed,

(2) v has an uninformed transmission neighbor,

(3) v has enough energy to transmit (e.g., in k-shot protocols v transmitted at most $k-1$ times so far),

(4) $t \equiv (z + c \cdot t_e) \pmod{t_e \cdot \lceil \log n \rceil}$, where $c \in \{\mathit{color}(C) | C \in \mathcal{C}_P(v)\}$ (v is allowed to transmit).

Critical composite rounds

The notion of a *critical composite round* refers the round when a node is prompted by the broadcast schedule to transmit during fast transmission pattern.
For a cluster $C_i^{(k)} \in \mathcal{C}^{(k)}$, we define a *critical composite round* as

$$t_{cc}(C_i^{(k)}) = \mathit{color}(C_i^{(k)}) \cdot t_e + 2 \cdot k \cdot t_e \cdot \lceil \log n \rceil + K(C_i^{(k)}),$$

where $K(C_i^{(k)})$ is the smallest integer, s.t.,

- $K(C_i^{(k)}) \geq 7 \cdot (d_I(G) + x) \cdot \log n \cdot t_b \cdot \mathit{rank}^*(C_i^{(k)})$, and
- $K(C_i^{(k)}) \equiv 0 \pmod{t_e \cdot \lceil \log n \rceil}$.

The parameter t_b corresponds to the length of one stage of slow transmissions and it will be established later. Concerning other parameters the following upper bounds can be established:

- $\mathit{color}(C_i^{(k)}) \cdot t_e$ is the maximum number of composite rounds during which the representative of a cluster has to wait before it starts passing the source message along its cluster path,
- $k \cdot t_e \cdot \lceil \log n \rceil$ is the maximum number of composite rounds wasted by cluster representatives in ancestor clusters while waiting for the first chance to transmit along their cluster paths,

- $rank^*(C_i^{(k)})$ is the maximum number of times the representatives of ancestor clusters were not chosen as the leading representative of their parent clusters, and
- $7 \cdot (d_I(G) + x) \cdot \log n$ is the maximum number of stages with slow transmissions (each stage being of length t_b) in which the cluster representative waits for the source message from its parent cluster in the case when its parent cluster has a different rank.

Finally, a *critical composite round* $t_c(v)$ for the node v ($v \in L_z$) is defined as follows:

$$t_c(v) = min\{z + t_{cc}(C_i^{(k)}) | C_i^{(k)} \in \mathcal{C}_P(v)\}.$$

Pattern of slow transmissions

Recall that the main purpose of slow transmissions is to disseminate the source message from informed cluster representatives and cluster paths to all other nodes in their clusters. Slow transmissions are performed in stages where each stage consists of a fixed number of composite rounds. During each stage, all nodes which are informed pass the source message on all their uninformed transmission neighbors. Hence, one stage of slow transmissions can be seen as information dissemination in a bipartite IRG.

Fix an algorithm $\mathcal{A}_b$ for information dissemination in bipartite IRGs. Let t_b be the maximal length of the schedule generated by $\mathcal{A}_b$ for an arbitrary bipartite IRG which is a subgraph of the IRG G. The k-th stage of slow transmissions, for $k \geq 0$, starts in the composite round $k \cdot t_b$ and finishes in the composite round $(k + 1) \cdot t_b - 1$, i.e., each stage lasts through exactly t_b composite rounds. In the first composite round of each stage we consider a bipartite IRG G_b, for which a schedule of transmissions is computed by the algorithm $\mathcal{A}_b$. The first part of G_b contains all informed nodes with uninformed transmission neighbors with the exception of all nodes whose critical composite rounds occur during this stage of slow transmissions. As we show later, if a node transmits in the pattern of fast transmissions, all its uninformed transmission neighbors become informed. However, this is not true for transmissions contributing to the pattern of slow transmissions. In particular, slow transmissions only guaranty that all uninformed transmission neighbors become informed at the end of the stage. But this could be too slow and inconsistent with the main aim of fast transmissions. The second part of the graph G_b is formed by all uninformed nodes connected by a transmission edge to a node in the first part.

3.5.5 Analysis of the Broadcasting Schedule

Lemma 3.5.6 ([F]). *Any fast transmission results in informing all transmission neighbors of the transmitting node.*

Proof. Fast and slow transmissions do not interfere due to adopted time multiplexing strategy.

Hence, the only case, when a transmission neighbor w of a transmitting node $u \in L_{z_u}$ does not receive a transmitted message from u in a composite round t, is the case when there is a node $v \in L_{z_v}$ transmitting simultaneously in the same round, s.t., an edge $(v, w) \in E_T \cup E_I$. Let c_u be the color of a cluster C_u which determines transmission of u in the composite round t. Note that the cluster C_u must contain u in its cluster path and u is not the last node on this path. Similarly, let c_v be the color of a cluster C_v which determines transmission of v in the composite round t. The scheduling mechanism for fast transmissions implies that

$$t \equiv (z_u + c_u \cdot t_e) \equiv (z_v + c_v \cdot t_e) \pmod{t_e \cdot \lceil \log n \rceil}.$$

If $c_u \neq c_v$, we get $|z_u - z_v| \geq t_e$ using the fact that $c_u, c_v \leq \lceil \log n \rceil$. Thus, $d_T(u, v) \geq t_e = d_I(G) + 2$. Since w is a transmission neighbor of u, the inequality $d_T(u, v) \geq d_I(G) + 2$ implies $d_T(w, v) \geq d_I(G) + 1$. Hence, there is no edge $(w, v) \in E_T \cup E_I$ that contradicts the assumption $(w, v) \in E_T \cup E_I$.

For $c_u = c_v$, the scheduling mechanism for fast transmissions implies $z_u \equiv z_v \pmod{t_e \cdot \lceil \log n \rceil}$. If $z_u \neq z_v$, we get $|z_u - z_v| \geq t_e \cdot \lceil \log n \rceil \geq t_e = d_I(G) + 2$. In this case we also get contradiction with the assumption $(w, v) \in E_T \cup E_I$. Hence, it remains to consider the case when $z_u = z_v$ and $c_u = c_v$. Recall that the fact that a node is a last node on a cluster path does not influence its activity in the pattern of fast transmissions. I.e., the activity of a node depends only on time and cluster colors in exactly one super-layer. Thus, there exists k such that $C_u, C_v \in \mathcal{C}^{(k)}$. Since $c_u = \mathit{color}(C_u) = \mathit{color}(C_v) = c_v$, the property (3) of Lemma 3.5.3 implies that $(v, w) \notin E_T \cup E_I$, which also contradicts the assumption. □

An important implication of Lemma 3.5.6 is the property that each node transmits in the pattern of fast transmissions at most once.

Lemma 3.5.7 ([F]). *Let S be a set of all informed nodes of the graph G at the beginning of a stage of slow transmissions. Each node that is a transmission neighbor of a node in S is informed by the end of this stage.*

Proof. It is enough to only consider the nodes that have an uninformed transmission neighbor at the beginning of the stage of slow transmissions. Initially, all such nodes form an informed part of a bipartite IRG for this stage of slow transmissions with the exception of nodes whose critical composite rounds are scheduled during the stage of slow transmissions. Since every exceptional node has an uninformed transmission neighbor, Lemma 3.5.6 implies that it did not transmit during any of previous composite rounds. The node is informed. Thus, according to scheduling mechanism for fast transmissions, the node will transmit during its critical composite round at the latest. And Lemma 3.5.6 guaranties that its transmission neighbors will become informed. Finally, a schedule of slow transmissions constructed for this stage guaranties that transmission neighbors of all other nodes in the uninformed part become informed. □

Lemma 3.5.8 ([F]). *Assume that the algorithm $\mathcal{A}_b$ that generates schedules for stages of slow transmissions, produces k-shot schedules, i.e., where each node of informed part transmits at most k times. Then, a broadcasting schedule produced by the algorithm for IRG G is a k-shot broadcasting schedule.*

Proof. We already know that every node acts as a transmitter in at most one stage of slow transmissions, where the schedule for that stage is produced by the algorithm $\mathcal{A}_b$. Thus if $\mathcal{A}_b$ produces a k-shot schedule, a node that acts during this stage as a transmitter transmits at most k times. In only problematic case when such a node is also due to contribute to fast transmission interleaved with slow transmissions of this stage, it simply never executes the schedule for slow transmissions. It awaits the round with the fast transmission and then informs all its transmission neighbors instantly. Since every node transmits in the pattern of fast transmissions at most once the limit of transmissions per node is also not exceeded in this case. □

Lemma 3.5.9 ([F]). *All transmission neighbors of a node v that belongs to a cluster path are informed by the end of its critical composite round $t_c(v)$.*

Proof. The proof is done by induction on the layer number of the node.

First, we analyze the base case, i.e., the claim for a source $s \in L_0$. A reverse rank of the source's cluster is 0. If $t_b < t_c(s)$, the informed source participates in the first stage of slow transmissions. From Lemma 3.5.7, all its transmission neighbors become informed by the end of the first stage. In the complementary case, when $t_b \geq t_c(s)$, the source does not participate in the first stage of slow transmissions. Since $t_c(v)$ is a composite round when the source has a permission to transmit, the source transmits and all its transmission neighbors get informed due to Lemma 3.5.6. Hence, in both cases, all transmission neighbors of the source are informed during the composite round $t_c(v)$ at the latest.

Now, we prove the claim for a node $v \in L_z$ under assumption that induction hypothesis holds, i.e., that the claim is true for every node $u \in L_y$, where $y < z$. We consider two cases: (1) when the node v is not a cluster representative, and the complementary case (2).

Lets analyze case (1) first. Let C be a cluster from $\mathcal{C}_P(v)$, s.t., $t_c(v) = z + t_{cc}(C)$ (see definition of the critical composite round). Recall that the node v lies on the cluster path in C. Since the node v is not a cluster representative, it has a predecessor $u \in L_{z-1}$ on the cluster path in C. Since $C \in \mathcal{C}_P(u)$ we also get $t_c(u) = min\{(z-1) + t_{cc}(C') | C' \in \mathcal{C}_P(u)\} \leq (z-1) + t_{cc}(C) < z + t_{cc}(C) = t_c(v)$. Thus, $t_c(u) < t_c(v)$. The induction hypothesis, for $u \in L_{z-1}$, implies that by the end of the composite round $t_c(u)$ all transmission neighbors of u are informed. The node v is a transmission neighbor of u. Therefore, the node v is informed by the end of composite round $t_c(u)$. If between composite rounds $t_c(u)$ and $t_c(v)$ there is room for at least one full stage of slow transmissions, it follows from Lemma 3.5.7 that all transmission neighbors of v are informed by the end of composite round $t_c(v)$. Otherwise, according to the scheduling mechanism for fast transmissions, the node v transmits in the composite round $t_c(v)$ at the latest. Hence, by Lemma 3.5.6, all transmission neighbors of v are informed by the end of composite round $t_c(v)$.

It remains to analyze the case when the node $v \in L_z$ is a cluster representative. Since v is a cluster representative, it holds that $z = k \cdot x$ and $k \geq 1$. Let C be a cluster from $\mathcal{C}_P(v) \subseteq \mathcal{C}^{(k)}$, s.t., $t_c(v) = z + t_{cc}(C)$. Further, let $C' \in \mathcal{C}^{(k-1)}$ be the parent cluster of the cluster C, i.e., $C \in child(C')$. From property (1) of Lemma 3.5.3, it follows that $rank(C) \leq rank(C')$. Next, from the definition of reverse rank, we get $rank^*(C) \geq rank^*(C')$. Two subcases emerge: $rank^*(C) = rank^*(C')$ and $rank^*(C) > rank^*(C')$.

Assume first that $rank^*(C) = rank^*(C')$ and compare critical composite rounds t_{cc} of both clusters. Ranks of both clusters are the same, $C \in \mathcal{C}^{(k)}$, and $C' \in \mathcal{C}^{(k-1)}$. Therefore, we have holds $t_{cc}(C) - t_{cc}(C') = (color(C) - color(C')) \cdot t_e + 2 \cdot t_e \cdot \lceil \log n \rceil \geq t_e \cdot \lceil \log n \rceil$. The last inequality follows from the fact that any cluster color is a number smaller than $\lceil \log n \rceil$. Further, property (3) in Lemma 3.5.4 implies that v is the leading representative of the parent cluster C'. Hence, there is a predecessor u of the node v on the cluster path of the parent cluster C'. Since $C' \in \mathcal{C}_P(u) \subseteq \mathcal{C}^{(k-1)}$, it also holds that $t_c(u) \leq (z-1) + t_{cc}(C')$. Therefore, $t_c(v) - t_c(u) \geq (z + t_{cc}(C)) - ((z-1) + t_{cc}(C')) \geq 1 + (t_{cc}(C) - t_{cc}(C')) \geq 1 + t_e \cdot \lceil \log n \rceil$. Note, that from the induction hypothesis for u it follows that at the end of composite round $t_c(u)$

all transmission neighbors of the node u, including v, are informed. Since also $t_c(v) - t_c(u) \geq 1 + t_e \cdot \lceil \log n \rceil$, the node v has at least $1 + t_e \cdot \lceil \log n \rceil$ composite rounds to inform all its transmission neighbors after a composite round, when it becomes informed, is finished. If this time interval contains at least one full stage of slow transmissions, it follows from Lemma 3.5.7 that all transmission neighbors of v become informed by the end of the composite round $t_c(v)$. In the complementary case, we observe (see the scheduling mechanism for fast transmissions) that this time interval is longer than $t_e \cdot \lceil \log n \rceil$. It also contains a composite round when the node v is allowed to transmit in the pattern of fast transmissions according to the color of C. From Lemma 3.5.6, all transmission neighbors of the node v become informed.

Finally, assume $rank^*(C) > rank^*(C')$. Let $u \in L_{(k-1)\cdot x}$ be the representative of the parent cluster C'. By the induction hypothesis, all transmission neighbors of the node u are informed by the end of the composite round $t_c(u)$. Now, we estimate the length of a time interval between a composite round when all transmission neighbors of u are informed and a composite round when all transmission neighbors of v have to be informed. Since $rank^*(C) - 1 \geq rank^*(C')$, one can show that $t_c(v) - t_c(u) \geq 7 \cdot (d_I(G) + x) \cdot \log n \cdot t_b$. The node v is the representative of the cluster C. Therefore, it belongs to the parent cluster C' of cluster C, i.e., $v \in C'$. Also property (2) of Lemma 3.5.3 implies that $d_T(u, v) \leq 6 \cdot (d_I(G) + x) \cdot \log n$. Certainly, there exists a transmission neighbor u' of the node u such that $d_T(u', v) + 1 = d_T(u, v)$. For any transmission neighbor v' of the node v, it holds that $d_T(u', v') \leq d_T(u', v) + 1 = d_T(u, v) \leq 6 \cdot (d_I(G) + x) \cdot \log n$. Since the node u' is a transmission neighbor of the node u, it gets informed by the end of the round $t_c(u)$. Observe also, that the time interval between $t_c(u)$ and $t_c(v)$ contains at least $6 \cdot (d_I(G) + x) \cdot \log n$ complete stages of slow transmissions. And indeed, $t_c(v) - t_c(u) \geq 7 \cdot (d_I(G) + x) \cdot \log n \cdot t_b \geq (6 \cdot (d_I(G) + x) \cdot \log n) \cdot t_b + t_b$. Since $d_T(u', v') \leq 6 \cdot (d_I(G) + x) \cdot \log n$, Lemma 3.5.7 (iterated $6 \cdot (d_I(G) + x) \cdot \log n$ times) implies that the node v', a transmission neighbor of the node v, becomes informed in the composite round $t_c(v)$ at the latest. □

Lemma 3.5.10 ([F]). *Let G be a given undirected IRG and $s \in V(G)$ be a given source node. Let $\mathcal{A}_b$ be a polynomial-time algorithm which produces schedules of transmissions of length t_b, for information dissemination in bipartite IRGs that are subgraphs of G. There is a polynomial-time algorithm that generates schedules of transmissions with length*

$$4 \cdot ecc_T(s) + O(d_I(G) \cdot \log^3 n \cdot t_b).$$

Moreover, if $\mathcal{A}_b$ generates k-shot schedules in bipartite IRGs, then generated schedules for arbitrary IRGs are also k-shot schedules.

Proof. Lemma 3.5.9 implies that each cluster representative v is informed in the composite round $t_c(v)$ at the latest. Since ranks of clusters are at most $\log n$ and the number of super-layers is $\lceil\frac{ecc_T(s)+1}{x}\rceil$, it follows that $max\{t_c(v)|v \in V(G)\} \leq ecc_T(s) + 3 \cdot \lceil\frac{ecc_T(s)+1}{x}\rceil \cdot t_e \cdot \lceil\log n\rceil + O((d_I(G) + x) \cdot \log^2 n \cdot t_b)$. Thus, all cluster representatives are informed during this composite round at the latest. From property (2) in Lemma 3.5.3 we can conclude that the transmission distance between a node and the nearest cluster representative is at most $6 \cdot (d_I(G) + x) \cdot \log n$. Thus, by Lemma 3.5.7, all nodes in the network become informed after at most $6 \cdot (d_I(G) + x) \cdot \log n$ subsequent stages of slow transmissions. These stages take in total at most $O((d_I(G)+x)\cdot\log n\cdot t_b)$ composite rounds. Therefore, all nodes in the network can be informed in at most $ecc_T(s) + 3 \cdot \lceil\frac{ecc_T(s)+1}{x}\rceil \cdot t_e \cdot \lceil\log n\rceil + O((d_I(G) + x) \cdot \log^2 n \cdot t_b)$ composite rounds. Now choosing $x = 3 \cdot t_e \cdot \lceil\log n\rceil$ we get $3\cdot\lceil\frac{ecc_T(s)+1}{x}\rceil\cdot t_e\cdot\lceil\log n\rceil = ecc_T(s)+O(t_e\cdot\log n)$. Note also that $t_e = d_I(G)+2$. Hence, the total length of the schedule is $ecc_T(s)+(ecc_T(s)+O(d_I(G)\cdot\log n))+ O((d_I(G) + (d_I(G) + 2) \cdot \log n) \cdot \log^2 n \cdot t_b) = 2 \cdot ecc_T(s) + O(d_I(G) \cdot \log^3 n \cdot t_b)$. Finally, since each composite round consists of two rounds, the thesis of the lemma is proved.

Preprocessing phase (construction of pre-cluster graphs, clustering, cluster tree, cluster paths, etc.) is realized in time $O(n^4)$ utilizing Floyd-Warshall algorithm, BFS traversals and the clustering algorithm from [31]. The broadcasting schedule is generated by round-by-round simulation of communication patterns (fast and slow transmission). I.e., when we simulate a round, we compute a set of transmitting informed nodes with respect to defined rules and then we determine new set of informed nodes due to known network topology. The simulation time of one round is bounded from above by $O(n^2) + T(\mathcal{A}_b)$, we simulate at most $O(n \cdot \log^3 n \cdot t_b)$ rounds, and $t_b = O(n)$. It follows that the time complexity of the algorithm is polynomial. □

The main result of this paper follows from Lemma 3.5.10 and Theorem 3.3.3.

Theorem 3.5.11 ([F]). *Let G be a given undirected IRG G where $s \in V(G)$ is the source node. There is a polynomial-time algorithm generating* 1*-shot schedules of transmissions that accomplishes broadcasting task in time*

$$4 \cdot ecc_T(s) + O(\Delta \cdot d_I(G) \cdot \log^4 n) = 4 \cdot D_T + O(\Delta \cdot d_I(G) \cdot \log^4 n).$$

Note that by applying this algorithm to radio networks with no interference edges the time of the broadcasting schedule from [33] is improved in graphs with $\Delta = o(\frac{\sqrt{n}}{\log^4 n})$. The 1-shot broadcasting algorithm proposed in

[33] relies heavily on the concept of *internal ranks* that impose currently an $\Omega(\sqrt{n})$-time bottleneck in the broadcasting schedule.

3.6 Lower Bound on Broadcasting Time

We conclude this chapter devoted to communication in radio networks modelled by *IRG*s with a lower bound on broadcasting time. We show that even for networks with the interference distance $d_I = 2$ any broadcasting schedule requires at least $D_T + \Omega\left(\Delta \cdot \frac{\log n}{\log \Delta}\right)$ rounds. Note that in view of this lower bound, broadcasting schedules generated by algorithm from Theorem 3.5.11 are almost optimal for d_I polylogarithmic in n.

Theorem 3.6.1 ([F]). *There exists an IRG $G = (V, E_T \cup E_I)$ with n nodes, $d_I(G) = 2$, an even maximal degree $\Delta \geq 4$, and the eccentricity of the source in the transmission subgraph $ecc_T(s)$ satisfying $72 \leq 6 \cdot \frac{\log n}{\log \Delta} \leq ecc_T(s) \leq n/2$, s.t., any broadcast schedule requires $ecc_T(s) + \Omega\left(\Delta \cdot \frac{\log n}{\log \Delta}\right)$ rounds.*

Proof. Consider a simple network structure that contributes a $(\Delta/2 - 1)$-round slowdown to the time complexity of information dissemination. Let an IRG $G' = (V', E'_T \cup E'_I)$, where:

- $V' = \{s\} \cup V_a \cup V_b \cup V_c$, where $V_a = \{a_1, \ldots, a_m\}$, $V_b = \{b_1, \ldots, b_m\}$, and $V_c = \{c_1, \ldots, c_m\}$
- $E'_T = \{(s, a_i), (a_i, b_i), (b_i, c_i) | 1 \leq i \leq m\} \cup \{(a_i, a_j) | 1 \leq i \neq j \leq m\}$
- $E'_I = \{(a_i, b_j) | 1 \leq i \neq j \leq m\}$.

Note that the subgraph induced by the nodes in V_a is a complete graph. It follows that $d_T(a_i, b_j) = 2$, for any i and j $(i \neq j)$. Hence, $d_I(G') = 2$. One can observe that any radio broadcasting schedule with the node s as the source requires at least $m + 2$ rounds to be accomplished. And indeed, all nodes in V_a are informed after the first round, when the source s transmits. However, if two or more nodes in V_a transmit simultaneously, no node in V_b receives a message due to presence of interference edges. Hence, it is not possible to inform all nodes in V_b in less than m consequent rounds. Finally, we need one extra round to inform all nodes in V_c. Therefore, the total time required is $1 + m + 1$ rounds. In comparison to a naive 3-round broadcasting for the case without interference edges, we obtain a $(m - 1)$-round slowdown.

Further, we extend this argument to networks with a larger eccentricity of the source. Let $T_{r,h}$ be a perfect r-nary tree of height h. All internal nodes

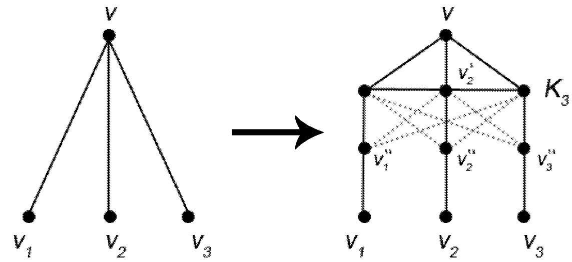

Figure 3.6: Transformation of a node v and its children v_1, v_2, v_3 in $T_{3,h}$ to a set of nodes and edges in $T'_{3,h}$

of $T_{r,h}$ have exactly r children, i.e., their degree is $r+1$. The total number of nodes in the tree $T_{r,h}$ is $\frac{r^{h+1}-1}{r-1}$. We turn a tree $T_{r,h}$ into an IRG $T'_{r,h}$ as follows. First, we insert new nodes v'_i and v''_i, including corresponding edges, between each internal node v and its child v_i, for $i = 1, \ldots, r$. More precisely, the edge (v, v_i) in $T_{r,h}$ is replaced by transmission edges (v, v'_i), (v'_i, v''_i), and (v''_i, v_i) in $T'_{r,h}$. Next, we add an interference edge (v'_i, v''_j) to $T'_{r,h}$, for each $1 \le i \ne j \le r$. These newly added interference edges contribute to the slowdown, because neither pair of nodes v'_i and v'_j is allowed to transmit simultaneously in order to inform nodes v''_i and v''_j. Finally, we add a transmission edge (v'_i, v'_j) to $T'_{r,h}$, for each $1 \le i \ne j \le r$. These edges do not improve the communication process. In fact, they impose the interference distance 2 in the IRG. A similar argument (as we used for the IRG G') implies that in any given radio broadcasting schedule (with s as the source), each internal node $v \in T'_{r,h}$ has a descendant node v_i (a child in $T_{r,h}$) which does not get the message earlier than $r+2$ rounds after v becomes informed. Hence, using inductive argument, any radio broadcasting schedule in $T'_{r,h}$ requires $3 \cdot h + h \cdot (r-1)$ transmission rounds. The resulting IRG $T'_{r,h}$ consists of $3 \cdot \frac{r^{h+1}-1}{r-1} - 2$ nodes and the eccentricity of s is $3 \cdot h$. In order to change the eccentricity of the source to more arbitrary value $ecc_T(s)$, we add a simple path of length $ecc_T(s) - 3 \cdot h$ to $T'_{r,h}$. One end of the path gets connected with the current location of s in $T'_{r,h}$ while the other end becomes the new source s. This new network, denoted by $T'_{r,h,ecc_T(s)}$, consists of $n(h) = 3 \cdot \frac{r^{h+1}-1}{r-1} - 2 + ecc_T(s) - 3 \cdot h$ nodes. In this network, any radio broadcasting schedule requires at least $ecc_T(s) + h \cdot (r-1)$ rounds.

In the following, consider only IRGs $T'_{r,h,ecc_T(s)}$, where $r = \Delta/2$. Let h_m be an integer such that $n(h_m) \le n < n(h_m+1)$. Now assuming that $ecc_T(s) \ge 6 \cdot \frac{\log n}{\log \Delta}$, one can show that for any $h \ge 2$ satisfying the inequality $n(h) \le n$ we get $3 \cdot h \le ecc_T(s)$. Also, the inequalities $n(h_m+1) > n$

and $ecc_T(s) \leq n/2$ imply that $h_m \geq \frac{\log n}{6 \cdot \log \Delta}$. Since $36 \cdot h_m \geq 6 \cdot \frac{\log n}{\log \Delta} \geq 72 = 36 \cdot 2$, we get $h_m \geq 2$. It implies that $3 \cdot h_m \leq ecc_T(s)$. Thus, the construction of $T'_{\Delta/2, h_m, ecc_T(s)}$ is correct. The IRG $T'_{\Delta/2, h_m, ecc_T(s)}$ is an IRG with $n(h_m)$ nodes, the eccentricity of the source $ecc_T(s)$, the maximum degree Δ, $d_I(T'_{\Delta/2, h_m, ecc_T(s)}) = 2$, and the broadcasting time at least $ecc_T(s) + h_m \cdot (\Delta/2 - 1) \geq ecc_T(s) + \frac{\log n}{6 \cdot \log \Delta} \cdot (\Delta/2 - 1) = ecc_T(s) + \Omega\left(\Delta \cdot \frac{\log n}{\log \Delta}\right)$. It remains to transform the IRG $T'_{\Delta/2, h_m, ecc_T(s)}$ to an IRG G with n nodes and satisfying all required properties. Observe that only a relatively small number of nodes is not in $T'_{\Delta/2, h_m, ecc_T(s)}$. Indeed, we can bound from above the number of remaining nodes by $n - n(h_m) \leq n(h_m + 1) - n(h_m) \leq (\Delta/2)^{h_m+3}$.

The number of nodes in a $(\Delta - 1)$-nary tree with the height at most $3 \cdot h_m - 1$ is bounded from above by $\frac{(\Delta-1)^{3 \cdot h_m} - 1}{\Delta - 2}$. For $h_m \geq 2$, this expression gives a value greater than $(\Delta/2)^{h_m+3}$, which is an upper bound on the number of remaining nodes. Since the source of the IRG $T'_{\Delta/2, h_m, ecc_T(s)}$ has degree at most $\Delta - 1$ (it has at most $\Delta/2$ children and no parent), we can arrange the remaining $n - n(h_m)$ nodes into a complete $(\Delta - 1)$-nary tree rooted in a newly created child of the source of $T'_{\Delta/2, h_m, ecc_T(s)}$. The newly constructed IRG constitutes G from the thesis of the theorem. Recall that $ecc_T(s) \geq 3 \cdot h_m$ and the height of the complete tree formed of the remaining nodes is at most $3 \cdot h_m - 1$. Therefore, broadcasting in the tree of remaining nodes can be completed separately and independently from any broadcasting process in $T'_{\Delta/2, h_m, ecc_T(s)}$ during at most $ecc_T(s)$ rounds after first transmission of the source. The thesis of the theorem follows. □

Chapter 4

More Complex Tasks in Unknown Radio Networks

It is known that particular assumptions about initial knowledge of network nodes and their communication capabilities influence how effectively communication tasks can be accomplished. Unlike all other chapters, this chapter is devoted to communication in radio networks in the case when the network topology is not included in initial knowledge of network nodes. This setting is often referred to as a *fully distributed setting* or an *unknown radio network*. In this setting, it is usually assumed that initial knowledge of a node is limited only to its unique identifier (ID). Most of works assume that identifiers are unique integers upper-bounded by $O(n)$, where n is the number of network nodes. Moreover, a node does not possess any information about network in its neighborhood (e.g., identifiers of its neighbors). Note that the lack of knowledge about network topology makes realization of communication tasks very difficult. Designed communication protocols must properly treat not only interference of simultaneous transmissions, but also a lack of knowledge about activity of neighboring nodes.

We focus on communication tasks that are not in the center of investigation related to communication in unknown radio networks modelled by the standard graph model. Our main objective is to study how effectively more complex tasks can be realized by a combination of standard communication primitives (e.g., broadcasting and gossiping) and other non-standard communication protocols working on the network level. We understand tasks like a maximum finding, a computation of some characteristics of an underlying reachability graph, and a computation of grid coordinates under the term "more complex tasks". Also, we investigate the impact of general information about the network topology (for instance, the fact that the network nodes form the grid topology) on efficiency of communication protocols.

4.1 Introduction

This chapter focuses on several communication tasks in slightly different models of radio networks. Although, the models are different, they share some common properties. In our investigation, we consider communication in the fully distributed setting where

- initial knowledge of a node is (at most) its unique identifier,
- spontaneous transmissions are not allowed, i.e., a node is allowed to transmit only after successful receiving of a message or a signal from one of its neighboring nodes,
- internal clocks of nodes are synchronized, but it is not guaranteed that they show the same time, and
- there is one distinguished node (an initiator or a source) that initiates execution of a communication protocol in the whole network.

We adopt the standard graph model of radio network. Here, the network topology is described by an undirected reachability graph. The reachability graph is unknown for the network nodes. Contrary to the centralized setting investigated in previous chapters, a communication protocol is a set of rules (a function) that prescribes activity of a node for each communication round. In each round, activity of a node depends on initial knowledge of the node and whole history of received messages and signals up to this round. I.e., we consider adaptive communication protocols. Though these protocols are adaptive, all our designed protocols require only small memory to store most significant aspects of communication history. Since we focus on communication complexity, we do not count time taken by any internal computation of a node.

4.1.1 Related Work

Communication in unknown radio networks (fully distributed setting) modelled by the standard graph model has been intensively investigated in the context of broadcasting and gossiping problem. Apart from these problems, some attention has been paid to other (closely related) problems like synchronization and wake-up problem.

First distributed algorithms for unknown radio networks were presented by Diks *et al.* in [25]. The authors considered a restricted class of networks having the nodes situated on a line. Moreover, they assumed that each node could reach directly all nodes within a certain distance. Note that for unknown (ad-hoc) radio networks an initial knowledge of a node is limited to its own (unique) identifier. Systematic study of deterministic distributed broadcasting was initiated by Chlebus *et al.* in [10]. In this work, a standard collision-free broadcasting algorithm called *Round-Robin* was presented. It works in stages. In the i-th round of a stage, the node with the identifier i that is already informed, i.e., it received the source message, transmits the source message. Obviously, all nodes of the network become informed after $ecc(s)$ stages. Hence, it completes broadcasting in $O(n \cdot ecc(s))$ rounds for an arbitrary radio network. If no parameters of the radio network are known in advance, one can modify *Round-Robin* to $O(n^2)$-round broadcasting algorithm using a doubling technique. The authors constructed better algorithm completing the broadcasting task in $O(n^{11/6})$ rounds, which is based on the notion k-selective families. In [55], De Marco and Pelc presented $O(n^{5/3} \cdot \log^3 n)$-round broadcasting algorithm. The upper bound was improved to the broadcasting time $O(n \cdot \log^2 n)$ rounds due to Chrobak *et al.* in [15]. Kowalski and Pelc brought a further improvement in work [47]. They constructed an algorithm completing the broadcasting task in $O(n \cdot \log n \cdot \log D)$ rounds. Czumaj and Rytter [22] improved this result and showed an algorithm that accomplish the broadcasting task in time $O(n \cdot \log^2 D)$ rounds. Very recently, De Marco in [54] proposed an algorithm completing the broadcasting in $O(n \cdot \log n \cdot \log \log n)$ rounds. We should note that all known $O(n \cdot \log^c n)$ algorithms are non-constructive. They utilize combinatorial structures (a kind of selective families) with certain desired properties. Existence of these structures is shown using a probabilistic argument, however, no efficient (polynomial-time) constructions are known. Therefore, it is assumed that such a structure is a part of the implicit knowledge of a node (it does not depend on the topology of radio network and it is the same in each node). The best known constructive algorithm (utilizing polynomially constructible combinatorial structures) was presented by Indyk [42]. It realizes the broadcasting task in $O(n^{1+o(1)})$ rounds.

Several lower bounds were proved in the literature. In [10], Chlebus *et al.* showed $\Omega(D \cdot \log n)$-round lower bound. This bound was improved to the currently best lower bound $\Omega(n \cdot \log D)$ rounds due to Clementi *et al.* in [19].

In the case when underlying reachability graph is symmetric, i.e., radio networks is modelled by a undirected graph, more efficient broadcasting algorithm were constructed. If spontaneous transmissions are allowed (the network is globally synchronous and a node can transmit before being informed), Chlebus *et al.* [10] developed an algorithm which broadcasts a source message in $O(n)$ rounds. On the other hand, if spontaneous transmission are not allowed, Bruschi and Del Pinto [7] proved $\Omega(D \cdot \log n)$-round lower bound. In [46], Kowalski and Pelc gave $\Omega(n \cdot \frac{\log n}{\log(n/D)})$-round lower bound for symmetric radio networks and designed $O(n \cdot \log n)$-round broadcasting algorithm.

Broadcasting was studied by Chlebus *et al.* [10] also in the model of radio networks that assumes availability of collision detection mechanism, i.e., the network nodes can distinguish between the interference noise and the background noise. Clearly, all previously mentioned upper bounds remain valid in this setting. Again allowing spontaneous transmissions, the authors proposed algorithms which complete the acknowledged radio broadcasting task in $O(n \cdot ecc(s))$ rounds for strongly connected radio networks and in $O(n)$ rounds for symmetric radio networks. In the case when a source message can be encoded by a binary sequence with length $|M|$, they showed algorithm that completes radio broadcasting in $O(ecc(s) \cdot |M|)$ rounds. Okuwa *et al.* [59] provided an improvement based on pipelining technique that results in $O(ecc(s) + |M|)$-round acknowledged broadcasting algorithm for symmetric radio networks.

Considering random radio networks, Elsässer and Gąsieniec [28] developed a distributed algorithm completing the broadcasting task in $O(\ln n)$ rounds with probability $1 - o(1/n)$

The gossiping problem was mostly studied in the context of ad-hoc radio networks under the assumption that initial knowledge of a node is limited only to its own identifier. Gossiping can be completed by the standard collision-free communication procedure *Round-Robin* (already mentioned as a broadcasting procedure). It runs in stages. In i-th round of a stage the node with the identifier i transmits its whole knowledge. It easy to see that the gossiping task is completed after at most D stages, i.e., *Round-Robin* is a deterministic $O(n \cdot D)$-round gossiping algorithm. In the case when the parameter D is unknown in advance, we obtain $O(n^2)$-round gossiping algorithm.

In [15], Chrobak *et al.* proposed a method to construct a gossiping algorithm from a broadcasting algorithm. Let $B(n)$ be the time complexity of an oblivious broadcasting algorithm. Their method constructs a gossiping algorithm that completes the task in $O(\sqrt{B(n)} \cdot n \cdot \log n)$ rounds. Utilizing the broadcasting algorithm which was proposed in the same paper and whose time complexity is $O(n \cdot \log^2 n)$ rounds, i.e., $B(n) = O(n \cdot \log^2 n)$, one can obtain an algorithm accomplishing the gossiping task in $O(n^{3/2} \cdot \log^2 n)$ rounds. This algorithm was the first deterministic gossiping algorithm with sub-quadratic time complexity. For appropriate (small) values of D or Δ, Clementi *et al.* [20] showed a faster gossiping algorithm working in $O(D \cdot \Delta^2 \cdot \log^3 n)$ rounds. Similarly, Gąsieniec and Lingas [34] presented algorithms completing the gossiping task in $O(\sqrt{D} \cdot n \cdot \log^3 n)$ and $O(D \cdot \Delta^{3/2} \cdot \log^3 n)$ rounds respectively. It should be noted, that $O(n^{3/2} \cdot \log^2 n)$- and $O(\sqrt{D} \cdot n \cdot \log^3 n)$-round gossiping algorithms require identifiers of nodes to be linear in n. Therefore, Gąsieniec *et al.* [35] focused on gossiping in radio networks with polynomially large identifiers. The authors proposed $O(n^{5/3} \cdot \log^3 n)$-round deterministic gossiping algorithm in directed radio networks and $O(n^{4/3} \cdot \log^3 n)$-round deterministic gossiping algorithm for undirected radio networks. These results were improved due to Gąsieniec *et al.* in [38], where an algorithm completing the gossiping task in $O(n^{4/3} \cdot \log^4 n)$ rounds was constructed, even in the case when identifiers are polynomial in n.

In all previous algorithms a node transmits its whole knowledge. Under assumption of bounded size messages, the gossiping problem was studied by Christersson *et al.* in [14]. They showed that 1-gossiping (i.e., gossiping with unit messages) can be in symmetric radio networks completed deterministically in time $O(n^{3/2} \cdot \log^3 n)$ rounds. Furthermore, they proposed a randomized protocol for 1-gossiping in symmetric ad-hoc radio networks whose time complexity is $O(n \cdot \log^3 n)$ rounds.

Chlebus *et al.* [12] studied average time complexity of the gossiping problem. Under the assumption of combined messages, they developed a gossiping protocol with average time $O(n/\log n)$ rounds and showed its optimality. For 1-gossiping, an algorithm having the optimal average running time $O(n \cdot \log n)$ rounds was developed.

4.2 Communication in Anonymous Radio Networks with Collision Detection

In this section, we deal with communication in radio networks with collision detection capability. In such a network, each node operating as a receiver can recognize 3 types of activity in its neighborhood:

- none of its neighbors transmits (the λ-signal is received),
- exactly one of its neighbors transmits (the μ-signal is received and the transmitted message is successfully decoded), and
- two or more its neighbors transmit (the μ-signal is received and no message is decoded).

Our primary focus is on design of a communication protocol for computing the maximal value among the values associated to network nodes. In this setting, we have given a distinguished node called the *initiator* (we shall denote it by s throughout this chapter) and we assume that each node of the network possesses a positive integer value. In some round, the initiator starts the algorithm that computes the maximum. The remaining nodes do not know this starting round.

Our problem is motivated by the following real-world situation: Consider a multihop radio network with a distinguished central node. Every node is able to perform a measurement of a physical quantity. Sometimes, in order to perform a specific operation, the central node must find out the maximal (or minimal) value in the network. One can collect all values into the central node (e.g., by performing a gossiping algorithm), but up to now no efficient suitable algorithm is known. We provide an algorithm that works in pipelined manner and, due to appropriate arrangement of transmissions, reduces the time necessary for completing the task. As a by-product of our main objective, we construct algorithms for computing some parameters of an underlying reachability graph. Namely, we present algorithms for computation of initiator's eccentricity and for computation of nodes' distances from the initiator.

Designed algorithms show another aspect of collision detection capability - how to encode information into collisions. Thanks to this encoding, algorithms presented in this section work even in anonymous radio networks consisting of nodes without initial knowledge. I.e., network nodes have no information about the network topology, the number of network nodes, and even the nodes do not have identifiers. In contrast, deterministic broadcasting cannot be realized in anonymous radio networks without collision detection capability [61].

Finally, note that similar problem of finding the maximum among real values associated to nodes of a multiple access broadcast network was studied by Martel in [56]. Randomized algorithm designed in that paper was used for solving the selection problem. The problem of finding the maximum among integer values associated to nodes of a radio network was treated by Chrobak *et al.* in [15], too.

We start this section with pipelined version of the algorithm *ENCODED-BROADCAST* called *RBEM* and presented by Okuwa *et al.* in [59]. We use this algorithm as important sub-routine. Moreover, it shows basic ideas how to combine pipelining and encoding information into collisions. We continue with two algorithms which focus on computation of some properties of an underlying reachability graph. First of them computes eccentricity of the initiator. Second of them computes a distance of each node to the initiator in a distributed manner. After these two algorithms are executed in the network, all nodes have sufficient knowledge to run algorithm that computes maximum among values stored in network nodes.

4.2.1 Broadcasting

Okuwa *et al.* in [59] presented the pipelined version of an algorithm called *ENCODED-BROADCAST* and originated in [10]. Their algorithm is called *RBEM.* Similarly as other broadcasting algorithms, it adopts the idea of layer-by-layer information dissemination approach. All nodes of a reachability graph G can be partitioned into *layers* according to their distances from the source (initiator) s. Hence, we define the sets

$$L_i = \{v \in V(G) : d(s,v) = i\},\ i = 0, 1, \dots, ecc(s).$$

Let v be a neighbor of w such that $v \in L_{i+1}$ and $w \in L_i$ for some i. We say that v is a $(+1)$-neighbor of w and w is a (-1)-neighbor of v. Algorithm *RBEM* does not broadcast the source message as one unit, but as a sequence of bits. Basic idea of the algorithm *RBEM* is to encode the source message into μ- and λ-signals using transmissions of *contact messages.* The contact message is a message that can be distinguished from all other messages transmitted during work of algorithm.

Theorem 4.2.1 ([59])**.** *The algorithm RBEM broadcasts a source message M with binary length $|M|$ in $O(ecc(s) + |M|)$-rounds, where $ecc(s)$ is the eccentricity of the source s.*

Proof. Let M be a source message and $(m_1, m_2, \dots, m_r)$ be a sequence of bits forming the message M ($r = |M|$). Initially, the message M is transformed into a sequence of bits $M' = (1, 1, 0, m_1, 0, m_2, \dots, 0, m_r, 1, 1)$ with the length $4 + 2 \cdot r$. Observe that the sequence M' can be easily transformed back to the original message M. Moreover, encoding of M' allows properly recognize borders (beginning and end) of the sequence.

Now, we describe rules of the broadcasting protocol. A node v starts its activity in the first round in which it receives the μ-signal, i.e., at least one of its neighbors transmits. The node v sets its round counter $RC(v)$ to 0, i.e., the subsequent round has the local round number 1. The round counter is incremented in each round. In all subsequent rounds, the node v behaves according to the following set of rules:

- $RC(v)$ mod $3 = 1$ (transmission round): If the μ-signal was received in the previous round, the node transmits the contact message. Otherwise, it works as a receiver (no message is transmitted).
- $RC(v)$ mod $3 = 2$ (sleep round): The node works as a receiver. Any signal received in this round is ignored.

- $RC(v) \bmod 3 = 0$ (receive round): The node works as a receiver.

The transmission schedule for the source s is different. The source node s transmits the contact message in the round $1 + 3 \cdot (j - 1)$ if and only if j-th bit of the sequence M' is equal to 1. Since the first bit of M' is 1, the source transmits the contact message in the round 1. Note that we count round numbers from 1.

At first, we show by induction on the layer number that for any $v \in L_i$ it holds $RC(v) = RC(s) - i$. For all nodes in L_1 the claim follows. Let us assume that the induction hypothesis holds for all nodes in the layer L_{i-1}. Let $w \in L_{i-1}$ be an arbitrary (-1)-neighbor of the node v. Due to induction hypothesis, $RC(w) = R(s) - (i - 1)$. It implies that the node w received the μ-signal in the round $i - 1$. According to the first rule of the algorithm, the node w transmits the contact message in the i-th round. Therefore, the node v receives the μ-signal in this round. Moreover, no node in L_{i-1} transmits before the round $i - 1$. It follows that the μ-signal received in the round i is the first μ-signal received by the node v. Hence, $RC(v) = RC(s) - i$.

According to rules of the algorithm, a node $v \in L_i$ transmits contact messages only in rounds when $RC(v) \bmod 3 = 1$. Since $RC(v) = RC(s) - i$, simultaneously transmitting nodes in different layers are in layers with distance at least 3. It follows that a node v can receive the μ-signal in a receive round ($RC(v) \bmod 3 = 0$) only in the case when at least one of its (-1)-neighbors transmits the contact message. Similarly, the contact message (the μ-signal) transmitted by a node v in a transmission round ($RC(v) \bmod 3 = 1$) is received and taken into consideration only by its $(+1)$-neighbors.

To formally prove that the algorithm completes the broadcasting task, we have to show that a node $v \in L_i$ receives the μ-signal in the round $i + 3 \cdot (j - 1)$ if and only if the j-th bit of the sequence M' is equal to 1. For all nodes in L_1, the claim immediately follows from the rules of the broadcasting algorithm. Let us fix j and consider a node $v \in L_i$. Assume that inductive hypothesis holds for all nodes in L_{i-1}. Let $w \in L_{i-1}$ be a (-1)-neighbor of the node v. The induction hypothesis implies that the node w receives the μ-signal in the round $(i - 1) + 3 \cdot (j - 1)$ if and only if the j-th bit of M' is equal to 1. Note that in the round $RC(s) = (i - 1) + 3 \cdot (j - 1)$, we have $RC(w) = R(s) - (i - 1) = 3 \cdot (j - 1)$ and $RC(v) = 3 \cdot (j - 1) - 1$. I.e., in this round $RC(w) \bmod 3 = 0$ and $RC(v) \bmod 3 = 2$. In the subsequent round $RC(s) = i + 3 \cdot (j - 1)$, we have $RC(w) \bmod 3 = 1$ and $RC(v) \bmod 3 = 0$. Due to rules of the algorithm, the node w transmits the μ-signal in this round if and only if the j-th bit of M' is 1. Since only the nodes in L_{i-1} satisfy the condition which allows them to transmit, the claim is proved.

Notice that a sequence of signals received in rounds $RC(v) \bmod 3 = 0$ can be transformed into bits of the sequence M'. Since borders of the sequence M' are easily recognizable, a node can stop its participation in the algorithm after the last bit of the sequence M' is forwarded. Finally, observe that all network nodes receive all bits of the sequence M' in the round $ecc(s)+3\cdot(|M'|-1) = O(ecc(s)+|M|)$ at the latest. □

In the following algorithms, we shall apply the algorithm *RBEM* to broadcast parameters (integer values) computed in the initiator to all network nodes.

4.2.2 Computation of Eccentricity

In this section, we propose an algorithm called *EFC - Eccentricity Fast Counting*. This algorithm computes the eccentricity of the initiator s. However, only the initiator s knows the value of the eccentricity after accomplishing the algorithm. Let us start with a rough description of the algorithm.

We remind that communication in the network is arranged into synchronized rounds. According to our algorithm, each node can be either active or inactive in one round of the algorithm. Initially all nodes, except the initiator, are inactive. During the work of the algorithm, every active node knows its distance from the initiator modulo 3. The work of the algorithm is split into phases. Only the initiator is informed about the number of current phase. But every active node has information about the number of current round within the actual phase (it has its own counter of rounds that is initiated in an appropriate moment). Each phase consists of two parts. First part has 4 rounds and the second one takes 6 rounds. In the first part, the active nodes that have been activated in the previous phase attempt to activate their (+1)-neighbors. An inactive node becomes active whenever it receives the μ-signal in two consecutive rounds. In the second part, active nodes detect whether they have some active (+1)-neighbor. If an active node does not detect any active (+1)-neighbor, it changes its state and becomes inactive. In order to avoid simultaneous transmissions in the consecutive layers, the transmissions in the second part are scheduled in a such a way that only the nodes in layers with mutual distance 3 transmit simultaneously.

Now, we are going to describe the phases of the algorithm *EFC* more precisely. The algorithm is initiated by the initiator s. At that moment only the initiator is active and we consider it to be a node activated in the previous phase. The initiator starts with the tasks prescribed for the first round of first phase of *EFC*.

Part 1 (4 rounds). In this part, only the active nodes which were activated in the previous phase transmit. In the first two rounds they transmit contact messages. If an inactive node receives the μ-signal in two consecutive rounds, it becomes active and sets its counter of the current round within the phase to 2. As it is shown below, an inactive node can become active if and only if it has an active (-1)-neighbor which transmits the contact messages during the first two rounds of a phase. Let $0 \leq i \leq ecc(s)$ and $v \in L_i$ be an active node transmitting in this part of current phase. Since v is active, it knows its distance $d = i \bmod 3$ from the initiator. If $d = 2$ then the node v transmits the contact message in the round 3 of actual phase. If $d = 1$ then v transmits the contact message in the round 4. During the rounds 3 and 4, a node that has been activated in the current phase acts as receiver in order to learn its distance from the initiator. If the μ-signal is received in the round 3 or 4, then it knows that its distance from the initiator is 0 or 2 respectively. Otherwise, it knows that its distance is 1. (The values are considered with respect to modulo 3.)

Part 2 (6 rounds). The second part of the phase is divided into 3 couples of rounds. Since Part 1 consists of 4 rounds we number the round of this part by $5, 6, \ldots, 10$.

If an active node v belongs to a layer L_i for some i, it transmits the contact message in the round with number $5 + (i \bmod 3) \cdot 2$. It means that within Part 2 transmissions occur only in rounds with numbers $5, 7$ and 9 of current phase. If an active node v belonging to L_i receives the μ-signal in the round $5 + ((i + 1) \bmod 3) \cdot 2$ then it remains active in the next phase. If a node was activated during Part 1 of current phase then it will be active at the beginning of the next phase too. All the others nodes will be inactivated.

Now we are going to show that if C stands for the number of the first phase in which the initiator is inactive then $C = 2 \cdot ecc(s) + 2$ (the first phase of the work of the algorithm is numbered as 1 and we recall that the initiator knows the numbers of phases). In order to simplify the forthcoming consideration, we introduce two new concepts.

Definition 4.2.2. *A path $(v_0, v_1, \ldots, v_k)$ is called an active path whenever v_0 is the initiator, $v_j \in L_j$ and v_j is active for all j, $0 \leq j \leq k$.*

Definition 4.2.3. *An active path $(v_0, v_1, \ldots, v_k)$ is called an active path in the phase i whenever it is active in the beginning of the first round of phase i.*

The following lemma provides an information about the structure of active paths in a network during the work of EFC.

Lemma 4.2.4 ([B]). *Let d be a length of a longest active path in the phase i. Then for every positive integer i, $1 \leq i \leq ecc(s)+1$, and for every node $v \in L_{i-1}$ there is an active path in phase i of the length d ending in the node v. Moreover, in the first round of phase i, each active node belongs to an active path.*

Proof. We shall proceed by induction with respect to the number of phase i. For $i = 1$ the claim is true, because in the first round of phase 1 only the initiator is active. For $i = 2$ the claim follows immediately from the description of the algorithm. Suppose now, that the assertion is valid for all positive integers less than a fixed number $j \geq 1$. We are going to prove that it is valid also for the phase j. According to the induction hypothesis, for $i = j-1$ every $v \in L_{j-2}$ is the end of a longest active path in phase $j-1$. It implies that all nodes from L_{j-2} are active and all nodes from L_{j-1} are inactive in beginning of the phase $j-1$. Further, for $i = j-2$ we know that all nodes in L_{j-2} became active in the Part 1 of the phase $j-2$, i.e., they transmitted in the Part 1 of phase $j-1$. Therefore all nodes in L_{j-1} are active in the first round of phase j. The rules defined for Part 2 of the phase $j-1$ ensure that if there is an active path in the phase $j-1$ ending in node $v_j \in L_{j-2}$ and v_j has at least one (+1)-neighbor, then all the nodes of this path remain active in the first round of the phase j. By the combination of these two facts we immediately have that the assertion of lemma is true for $i = j$ too and the proof is complete. □

It is not very difficult to see that in the Part 1 of phase i, $1 \leq i \leq ecc(s)$, there are activated exactly the nodes of layer L_i. The following lemma describes the active paths for phases with number at least $ecc(s)+1$.

Lemma 4.2.5 ([B]). *Every longest active path in the phase $ecc(s)+i$, where $1 \leq i \leq ecc(s)+1$, has the length $ecc(s)-i+1$. Moreover, no node from the layer L_j, $j > ecc(s)-i+1$, is active in the first round of the phase $ecc(s)+i$.*

Proof. We proceed by induction on the number i that is related to the number of considered phase of the algorithm. For $i = 1$ the assertion follows from the previous lemma applied for the phase $ecc(s)+1$. Moreover, Lemma 4.2.4 implies that for every node v there is a phase k, $k \leq ecc(s)+1$, such that v is active in the phase k. From the construction of the algorithm it is easy to see that a node transmits in the Part 1 of some phase only once. Further, if an active node has became inactive, then it remains inactive for the rest of the work of EFC. Suppose now, that the assertion of lemma is true for some positive integer i and consider $j = i+1$. From the induction hypothesis it follows that in the first round of the phase $ecc(s)+(j-1)$, there is an active path ending in a node of the layer $L_{ecc(s)-j+2}$ and no node is active in the upper (more remote from the initiator) layers L_k for $k > ecc(s) - j + 2$. Consider an arbitrary longest active path $P = (v_0, v_1, \ldots, v_{ecc(s)-j+1}, v_{ecc(s)-j+2})$ in the phase $ecc(s) + (j - 1)$. Since there is no active $(+1)$-neighbor of the node $v_{ecc(s)-j+2}$, the node $v_{ecc(s)-j+2}$ becomes inactive in the next phase of the algorithm (see the description of the rules for Part 2 of EFC). All the other nodes of the path P receives the μ-signal. Therefore, they remain active and $P - v_{ecc(s)-j+2}$ is an active path in the phase $ecc(s) + j$. Note that this active path is simultaneously a longest active path in the phase $ecc(s) + j$. This completes the proof. □

An application of Lemma 4.2.5 for $i = ecc(s) + 1$ yields that the initiator has no active $(+1)$-neighbor in the phase $2 \cdot ecc(s) + 1$. Since the initiator is active also in all the previous phases, the initiator is inactive for the first time at the beginning of the phase $2 \cdot ecc(s) + 2$. Using these facts we can formulate the following result.

Theorem 4.2.6 ([B]). *The algorithm EFC computes eccentricity $ecc(s)$ of the initiator in $O(ecc(s))$ rounds.*

We remind that after finishing the algorithm EFC only the initiator knows its eccentricity. Now we need to distribute this information to the remaining nodes. In order to broadcast the computed eccentricity of the initiator, we use the algorithm *RBEM* (Theorem 4.2.1). This algorithm broadcasts a message of binary length r in $O(r + ecc(s))$ rounds. In our case, the computed eccentricity of the initiator can be binary encoded to a message with length $\log ecc(s)$. The algorithm *RBEM* completes broadcasting of this message in $O(ecc(s))$ rounds. In general, the algorithm *RBEM* is not acknowledged, but the initiator knows the value of the parameter $ecc(s)$, and therefore it has an implicit information when this task is completed. The algorithm EFC equipped with the previously described broadcasting ability we shall refer as $ExEFC$ - extended EFC.

4.2.3 Computation of Distance from the Initiator

The main objective of algorithm presented in this section is to compute distance from the initiator for each network node. We assume that all network nodes know eccentricity of the initiator s (it can be computed by the previously proposed algorithm). After the completion of this algorithm, every node v knows its exact distance from the initiator that uniquely determines the layer L_i containing v. The basic idea of the suggested algorithm is that the nodes belonging to L_i transmit concurrently binary encoded number $i+1$

(using μ and λ-signals) to their (+1)-neighbors, i.e., to nodes belonging to L_{i+1}. In order to decrease the time complexity of this task, we realize it in the pipelined fashion. We use the fact that knowing k lowest bits of a number i (i.e., the suffix of the binary code of i), we also know the k lowest bits of the number $i + 1$. In order to realize the goal, we modify the algorithm *RBEM*. Particularly, we shall dynamically change the broadcasted message. In the following, we refer this modified algorithm as *DDC-Distributed Distance Counting*. Moreover *DDC* has one useful property. If the eccentricity $ecc(s)$ of the initiator is known for all nodes of the network, it allows us to use this algorithm for a "synchronization of the nodes". It means that the nodes can make an agreement about the round when they would start simultaneously some other task.

Theorem 4.2.7 ([B]). *The algorithm DDC computes the distance from the initiator to each node of the network in $O(ecc(s))$ rounds. Moreover, by an application of the algorithms ExEFC and DDC we can "synchronize" the network in $O(ecc(s))$ rounds.*

Proof. We show how to modify the algorithm *RBEM*. Let M'_i be a binary sequence transmitted by nodes of L_{i-1} to their (+1)-neighbors and received by nodes in L_i. Initially, $M'_1 = (1, 1, 0, 1, 1, 1)$. This sequence informs nodes in L_1 that their distance to the initiator is 1. Let us denote j-th bit (element) of a sequence M'_i as $M'_i[j]$. A sequence M'_{i+1} transmitted by a node $v \in L_i$ is computed as follows. Assume that first k bits of M'_i are already received by the node v. The k-th bit of M'_{i+1} is computed according to first received k bits of M'_i. In particular, $M'_{i+1}[1] = M'_{i+1}[2] = 1$. When the node v receives the third bit of M'_i, it initializes a carry bit α to 0. Further, unless the node recognizes that $M'_i[k] = 1$ for an odd $k \geq 3$, $M'_{i+1}[k]$ is computed by this rule:

- for odd k, $M'_{i+1}[k] = M'_i[k] = 0$, and
- for even k, $M'_{i+1}[k] = (M'_i[k] + \alpha) \pmod 2$ and the carry bit α is set to 1 if $M'_i[k] + \alpha = 2$, and to 0 otherwise.

Finally, the last bits of M'_{i+1} are determined by the current value of the carry bit α. If $\alpha = 0$, we finish the sequence M'_{i+1} with $(1, 1)$. Otherwise, the sequence M'_{i+1} is finished with the sequence $(0, 1, 1, 1)$. In the latter case, the forwarded message is extended, and particularly $|M'_{i+1}| = |M'_i| + 2$.

Notice that the sequence M'_i uniquely encodes the number i. Indeed, the bits $M'_i[|M'_i| - 2], M'_i[|M'_i| - 4] \ldots, M'_i[4]$ correspond to the number i in the binary notation. Therefore, each node in L_i knows after receiving M'_i its distance from the initiator. In the algorithm *RBEM*, it holds for the local round-counter $RC(v)$ of a node $v \in L_i$ that $RC(v) = RC(s) - i$. Since we do not modify first three bits of forwarded sequences and local round-counters are initialized in the first round when the μ-signal is received, this property remains valid in *DDC* as well. Utilizing this property, each node can easily compute the number of rounds from the beginning of the work of algorithm after determining the value i, i.e., to compute actual value of $RC(s)$.

After application of the algorithm $ExEFC$ (as a preprocessing algorithm) the nodes know the eccentricity $ecc(s)$ of the initiator. Using this knowledge and actual value of $RC(s)$, the nodes can compute the last round of *DDC*. It allows us to use this algorithm for mentioned "synchronization of the nodes", i.e., the nodes can make an agreement about the round when they would start simultaneously some task. Similarly, as in case of the algorithm *RBEM*, one can show that the algorithm *DDC* is accomplished in the round $ecc(s) + 3 \cdot (\lceil \log ecc(s) \rceil - 1)$. This value evidently determines also the time complexity of the algorithm *DDC*.

Note that we can use even unary encoding of layer numbers in transmitted sequences without affecting total asymptotical time complexity of the algorithm. □

4.2.4 Computation of Maximal Value

Now, we design an algorithm *CMV* that computes the maximum of the considered values among all nodes in the network. The algorithm consists of three logical parts. In the first step, the initiator estimates the maximal value by determining the minimum number of bits necessary for its binary encoding. In the second step, the initiator broadcasts the estimation to all other network nodes and initiates the computation of the exact value, which forms the third logical part of the algorithm. At the end of the computation, the initiator knows the desired value.

Estimating the maximal value

As we have already mentioned, we suppose that every node of the network possesses a positive integer value. In what follows, we show how to compute the estimation of the maximal value among them. More precisely, for the unknown value Max, the searched maximum, we want to compute the value B_{max} such that $2^{B_{max}-1} \leq Max < 2^{B_{max}}$. Obviously, the value B_{max} specifies how many bits we need to store an arbitrary value associated to a node of the network. We assume that algorithms *ExEFC* and *DDC* have been already performed and every node knows the eccentricity $ecc(s)$ of the initiator, it knows its distance from the initiator and the nodes are synchronized (they know the starting round of the algorithm computing the estimation).

Our algorithm, called *EMV*, works as follows. Every node performs in the loop 3 segments: receive, transmission and sleep. Every segment consists of only one round. (We use concept of segments only in order to use uniform terminology in description of algorithms.) For any node v, belonging to the layer L_i, $0 \leq i \leq ecc(s)$, let us denote by V_v the value associated to v and let B_v be the positive integer satisfying $2^{B_v-1} \leq V_v < 2^{B_v}$.

In the first round of the algorithm, the nodes perform an activity that depends on their layer. The nodes belonging to the layers L_i, where $i = ecc(s) - 3 \cdot k$ for some integer $k \geq 0$, realize transmission segment. The nodes from the layers L_i for $i = ecc(s) - 3 \cdot k - 1$, $k \geq 0$, realize activities prescribed for receive segment and the remaining nodes realize sleep segment.

A node $v \in L_i$ transmits the contact messages according to two rules:

(1) The contact message is transmitted in all rounds r, where $r = ecc(s) - i + 1, ecc(s) - i + 4, \ldots, ecc(s) - i + 3 \cdot (B_v - 1) + 1$. (Note that these rounds are the rounds of the transmission segment.)

(2) The contact message is transmitted in every transmission segment following the receive segment during which the node v received μ-signal.

Let R be the round of the first receive segment with number at least $ecc(s)$ in which the μ-signal is not received by the initiator. It is possible to prove that R is well defined and moreover $(R - ecc(s))/3$ is the maximum among all values B_v with exemption of the value B_s. It results in the following theorem.

Theorem 4.2.8 ([B]). *Let Max be the maximal value among the positive values associated to the nodes. The algorithm EMV computes the value B_{max} such that $2^{B_{max}-1} \leq Max < 2^{B_{max}}$ in $O(ecc(s) + \log Max)$ rounds.*

Proof. Consider an arbitrary node $v \neq s$ of the network. Let i be positive integer such that $v \in L_i$. Firstly, note that v transmits the contact message in at least one round (transmission segment), because $B_v \geq 1$ and, according to the first rule, v transmits in the round $ecc(s) - i + 1$. Secondly, it follows from the second rule of the algorithm that if v transmits the contact message in the transmission segment in a round r, $r \geq ecc(s) - i + 1$, then the μ-signal is received by the initiator s in the round $r + i - 1$. This is not difficult to verify (by induction on the distance from s or the layer number), because the nodes lying on a shortest path from v to s sequentially transmit the received μ-signal toward s.

Let $1 \leq j \leq ecc(s)$ and $w \in L_j$ be a node such that $B_w = \max\{B_v | v \in V, v \neq s\}$. From the first rule we have that the node w transmits the contact message in the round $ecc(s) - j + 3 \cdot (B_w - 1) + 1$. Therefore, the initiator receives the μ-signal in the round $ecc(s) + 3 \cdot (B_w - 1)$.

Suppose now that the initiator receives the μ-signal in the round $ecc(s) + 3 \cdot (B_w - 1) + 3$. It could happen only if there is a $(+1)$-neighbor $v_1 \in L_1$ of the initiator transmitting the contact message in this round. Since $B_{v_1} \leq B_w$ this transmission can be realized only according to the second rule. Further, v_1 could receive μ-signal in the round $ecc(s) + 3 \cdot (B_w - 1) + 2$ only in the case if there is a $(+1)$-neighbor $v_2 \in L_2$ of the node v_1 transmitting the contact message in this round. This transmission can be again done only due to the second rule. By a repeated application of such arguments we can show, that there is a node $v_{ecc(s)} \in L_{ecc(s)}$ transmitting the contact message in the round $ecc(s) + 3 \cdot (B_w - 1) + 4 - ecc(s) = 3 \cdot (B_w - 1) + 4$. Again, since $B_{v_{ecc(s)}} \leq B_w$, this transmission can be approved only by the second rule. But this is not possible, because $v_{ecc(s)} \in L_{ecc(s)}$ has no $(+1)$-neighbors.

It implies that $B_w = (R - ecc(s))/3$, where R is the round of the first receive segment after the round $ecc(s)$, in which the μ-signal is not received by the initiator. After computing B_w the initiator puts $B_{max} = \max\{B_s, B_w\}$ and the proof is complete. □

Computing the maximal value

After finishing *EMV*, the value $B_{\max}$ is known only for the initiator s. Before performing the computation of Max, we have to distribute its estimation B_{max} to the remaining nodes of the network. We can again utilize the algorithm *RBEM* which takes $O(ecc(s) + \log Max)$ rounds. After finishing *RBEM*, all nodes of the network are informed how many bits are needed to store Max. Therefore, the nodes can unify the representation of their values as binary sequences of the length $\lfloor \log Max \rfloor + 1 = B_{max}$. In *CMV* each node v computes the value E_v which is its estimation of Max. According to *CMV*, the nodes that recognize that they cannot improve the calculated value Max eliminate themselves from the process of computation. Similarly as *EMV*, the algorithm *CMV* is working in the loop and repeatedly perform 3 segments: receive, transmission, and sleep. The difference is that every segment consists of two rounds. During work of the algorithm, the nodes can be in one of two states: active or inactive.

Now we are going to describe *CMV* in more details. At the beginning, all nodes are active and the value E_v of each node $v \notin L_{ecc(s)}$ is set to 0. We shall work with binary representation of E_v and therefore E_v can be viewed as a finite sequence of fixed length B_{max} consisting of 0's and 1's. During the work of the algorithm we improve the value of E_v by the modification of the particular bits. It means that initially $E_v = (0, 0, \ldots, 0)$. Moreover, during initialization phase, if $v \in L_{ecc(s)}$ then E_v is set to V_v.

In the first round of the algorithm, the nodes perform an activity that depends on their layer. The nodes belonging to the layers L_i, where $i = ecc(s) - 3 \cdot k$ for some integer $k \geq 0$, realize transmission segment. The nodes from the layers L_i for $i = ecc(s) - 3 \cdot k - 1$, $k \geq 0$, realize activities prescribed for the receive segment and the remaining nodes realize sleep segment.

The inactive nodes cannot transmit. The transmission of an active node $v \in L_i$, $1 \leq i \leq ecc(s)$, is prescribed by the following rules:

(1) The node v transmits the contact message in all rounds r, where $r = 2 \cdot (ecc(s) - i - 2 + 3 \cdot j) - 1$ for some j satisfying $j \in \{1, \ldots, B_{max}\}$, whenever j-th highest bit of binary encoding E_v is 1, i.e., the bit corresponding to $2^{B_{max} - j}$ is 1.

(2) If the μ-signal is received by the node v in the round r, where $r = 2 \cdot (ecc(s) - i - 2 + 3 \cdot j)$ for some j, s.t., $j \in \{1, \ldots, B_{max}\}$, and the node v has not been transmitted in the round $r - 1$, then the node v becomes inactive.

(3) If in the round r, where $r = 2 \cdot (ecc(s) - i - 3 + 3 \cdot j) - 1$ for some j satisfying $j \in \{1, \ldots, B_{max}\}$, the node v receives μ-signal or the j-th highest bit of binary encoding of V_v is 1 and $V_v \geq E_v$, then the node sends the contact message in the following round $r + 1$ and sets the j-th highest bit of the binary encoded value E_v to 1.

Note that the first and the second rule are related to the rounds that belong to the transmission segment of the node v. The third rule concerns rounds of the receive segment of the node v.

The rules can be also interpreted as follows. In the first round of the transmission segment, an active node v transmits according to j-th highest bit of the value E_v, where j is determined by the rules for the given round. Furthermore, all active nodes belonging to the same layer work with the j-th highest bit. Simultaneously a (-1)-neighbor receives the μ-signal during the first round of its receive segment if and only if it has at least one active (+1)-neighbor which has j-th highest bit equal to 1. In the following round these (-1)-neighbors announce to their (+1)-neighbors how they set their j-th highest bits. After this round, every node knows whether its activity in the previous round has influenced some its (-1)-neighbors, i.e., whether its active (-1)-neighbors set their j-th highest bits according to its j-th highest bit. If a node detects that no (-1)-neighbor set its E_v according to its information, it becomes inactive. That is why no value E_v that can be potentially computed by this node cannot be larger than the value which would be computed in one of its active (-1)-neighbors. The next lemma shows that the computed value E_s matches our expectations.

Lemma 4.2.9 ([B|]). *After the round $r = 2\cdot(ecc(s)-3+3\cdot B_{max})$, the value E_s of the initiator s is equal to the maximal value Max among the values associated to the nodes.*

Proof. Note that we can encode the values of the nodes by binary sequences, each of them of the same length, because we have already applied the algorithm *EMV* that computes the value B_{max}.

The following proposition provides relatively straightforward properties of binary sequences utilized for encoding integer numbers. It says that by a comparison of the highest k bits of two numbers we can obtain an important information about the size of these numbers.

Proposition: *Let l, A, B be positive integers and A, B have binary representations $A = (a_1a_2\dots a_l)_2$ and $B = (b_1b_2\dots b_l)_2$ respectively. If $A \leq B$ then for any k, $1 \leq k \leq l$ the following inequality $(a_1a_2\dots a_k)_2 \leq (b_1b_2\dots b_k)_2$ holds.*

Using previous proposition one can easily check that the following two invariants are true during the work of the algorithm CMV.

(1) At the end of any round $r = 2\cdot(ecc(s)-i-3+3\cdot j)$, where j, $1 \leq j \leq B_{max}$, for each node $v \in L_i$ holds the following: $E_v \geq V_v$ and the highest j bits of the binary encoded value E_v remain the same during the rest of the algorithm and they are equal to the highest j bits of the value E_w, where w is an arbitrary active (+1)-neighbour of the node v.

(2) If an inactive node $v \in L_i$ has an active (-1)-neighbor, then there is an active (-1)-neighbor $w \in L_{i-1}$ of v satisfying $E_w > E_v$.

Since the initiator has no (-1)-neighbors, it is active in every round. After the round $r = 2\cdot(ecc(s)-3+3\cdot B_{max})$, any active path consists only of nodes with estimations of Max equal to E_s, because from the first invariant we have that all bits of E_v remain unchanged.

Consider now a node w such that $V_w = Max$. It is easy to see that in every round, every prefix of the binary encoded value E_w is equal to the prefix of the binary encoded value V_w. From the rules of the algorithm it follows, that in every round there is an active path ending in the node w. Indeed, this is true because the prefix of E_w is successively distributed and computed bit by bit in all nodes belonging to an active path ending in w. Therefore all nodes belonging to the considered active path remain active in every next round (there is no greater prefix during the work of the algorithm). Thus, in the considered round $r = 2 \cdot (ecc(s) - 3 + 3 \cdot B_{max})$, the value E_s of the initiator s is equal to the value $E_w = V_w = Max$. □

As a consequence of the previous results, we immediately get the following theorem.

Theorem 4.2.10 ([B]). *Algorithm CMV computes the value Max (in the initiator) in $O(ecc(s) + \log Max)$ rounds.*

Algorithm *CMV* computing the maximal value among values possessed by the network nodes can be applied to realize other communication tasks.

Computation of the minimal value among values possessed by the network nodes can be realized in $O(ecc(s) + \log Max)$ rounds in the following way:

(1) compute the maximal value applying algorithm *CMV* in $O(ecc(s) + \log Max)$ rounds and let Max be the computed maximum,

(2) broadcast the value Max applying algorithm *RBEM* in $O(ecc(s) + \log Max)$ rounds, and

(3) considering $Max - V_v$ as a value possessed by a node v, where V_v is the original value possessed by the node v, compute the maximal value applying algorithm *CMV* in $O(ecc(s) + \log Max)$ rounds. If Max' is the computed maximum, the minimum value among values possessed by network nodes is $Max - Max'$.

Algorithm *CMV* can be utilized to compute logical functions $\wedge$ (1 if the minimum is 1, 0 otherwise) or $\vee$ (1 if the maximum is 1, 0 otherwise).

4.2.5 Lower Bound for Maximum Finding

In this section we show that the algorithm *CMV* (Theorem 4.2.10) is asymptotically optimal. In particular, we reduce the problem of broadcasting in symmetric geometric radio networks (shortly *GRN*) with collision detection to our problem of maximum finding. Model of *GRN* differs from our model of radio networks in two properties: nodes have an additional initial information about their positions and a reachability graph of a *GRN* should satisfy restrictions resulting from geometric locations of nodes. Note that broadcasting algorithms for *GRN* utilize unique identifiers for every node of the network (the existence of such identifiers follows for example from their geometric locations).

Theorem 4.2.11 ([B]). *For any maximum finding algorithm with collision detection there exists a symmetric radio network of diameter* 2 *and such an assignment of values associated to nodes that the algorithm requires* $\Omega(\log Max)$ *rounds.*

Proof. Dessmark and Pelc showed in [24] that for every broadcasting algorithm with collision detection there exists a class of symmetric geometric radio networks with eccentricity of the source node equal to 2 for which this algorithm requires $\Omega(\log n)$ rounds, where n is the number of nodes. More precisely, for a given n, this network has the following form: nodes of the network are labelled $1, \ldots, n$, node 1 is the *source* and node n is the *sink*. The set $\{2, \ldots, n-1\}$ is partitioned into sets X and Y, where $|Y| = 2$. Nodes $1, \ldots, n-1$ form a complete graph. Nodes from Y are connected to the sink n. We shall refer to such a network as *a network of class* $\mathcal{H}$.

In what follows we show how to utilize a maximum finding algorithm in a broadcasting algorithm for networks of class $\mathcal{H}$. Now, let $\mathcal{A}$ be an algorithm for the maximum finding in symmetric radio networks with collision detection and G be a n-node network of the class $\mathcal{H}$. An associated broadcasting algorithm (to the algorithm $\mathcal{A}$) for the network G works as follows: In the first round, the source (node 1) transmits the source message. Nodes in the distance 1 from the source become informed. Next, we perform algorithms *ExEFC* (Section 4.2.2) and *DDC* (Theorem 4.2.7) with the node 1 as the initiator. After $O(1)$ rounds, we can distinguish the sink n (a node in the distance 2 from the initiator 1). In order to distinguish the nodes of the set Y, the sink n transmits the contact message in the following round. All nodes, except the nodes of Y, set their values to 1. Two nodes of the set Y set their associated values to their identifiers. Performing algorithm $\mathcal{A}$ for maximum finding problem, we compute a label of one node from the set Y in the initiator 1 (a label of the node in Y with larger label). After that the initiator transmits a message containing the computed label of one node from Y. In the following round, only the node with this label transmits the source message and the sink n become informed.

Obviously, the asymptotical time complexity of the associated broadcasting algorithm is the same as the complexity of algorithm $\mathcal{A}$. Since $Max \in O(n)$, the previously mentioned result from [24] implies that time complexity of $\mathcal{A}$ is $\Omega(\log Max)$. □

Combining the previous result and the trivial lower bound $\Omega(ecc(s))$ we obtain that the algorithm *CMV* is asymptotically optimal in the view of parameters $ecc(s)$ and Max.

4.3 Communication in Grid Radio Networks

In the rest of this chapter, we focus on communication in a radio grid, i.e., an underlying reachability graph of the radio network is a grid graph. Our aim is to model radio communication in a real-world setting when the nodes (network stations) are spread in a region in a regular way. Namely, we assume that network nodes are located at all grid points of a square mesh and the transmission radius of each node is equal to 1.

Definition 4.3.1. *For $p, q > 1$, an undirected graph $G_{p,q}$ is called a grid graph if and only if:*

$$\begin{aligned} V(G_{p,q}) &= \{(i,j) | 0 \leq i < p, 0 \leq j < q\} \\ E(G_{p,q}) &= \{((i,j),(i',j'))| \\ &\quad (i' = i \wedge j' = j \pm 1) \vee (i' = i \pm 1 \wedge j' = j)\}. \end{aligned}$$

We consider the model of radio networks without collision detection, i.e., the nodes cannot distinguish between background and interference noise. It follows that a receiving node cannot distinguish whether none of its neighbors or more than one of its neighbors transmit in a given communication round.

In this section, we pay attention to both complexity measures: time and energy complexity. Unlike previous algorithms focusing on k-shot schedules, we investigate the total number of transmission in the whole network, i.e., the sum of all transmissions realized by individual network nodes. The complexity of an algorithm is upper-bounded by a function of three parameters of radio network: the number of nodes (denoted as n), an unknown upper bound of identifiers assigned to the nodes (denoted as N), and the largest distance from the source s to any other node of the network (denoted as $ecc(s)$).

Throughout this section, we focus on the deterministic distributed algorithms in unknown radio networks with the grid topology. Initial knowledge of a node is limited only to its unique integer identifier (label). The node is not aware of its position in the grid. Moreover, neither the size of an underlying reachability graph (the number of nodes) nor an upper bound of identifiers are known to the nodes. Most of previous works deal with the assumption that $N = O(n^p)$, for a constant $p > 0$. In this section, we consider that there is no relationship between N (an upper bound of identifiers) and n (the number of nodes), except trivial $N \geq n$. It models the case when each network station possesses a unique factory identifier (e.g., MAC address) from very large set of possible identifiers, but the number of nodes forming a network is relatively small. Note that the considered setting is not a typical real-world case, since the network topology is fixed to a specific graph topology. On the other hand, all considered assumptions reflect the case immediately after appropriate arrangement of nodes (with only factory initial knowledge) in an area. In this sense, presented algorithms can be seen as algorithms computing auxiliary information that serves also for establishment of a fast communication mechanism.

Radio communication in networks with grid topology was investigated by Kranakis *et al.* in [51]. The authors discussed fault tolerant broadcasting in known topology radio grid networks (each node knows its coordinates and dimensions of the grid graph). Bermond *et al.* [4] considered modified model of communication in know topology radio grid networks that follows from a problem proposed by France Telecom.

4.3.1 Broadcasting

This section is devoted to the broadcasting task in unknown radio networks whose underlying reachability graph is a grid graph $G_{p,q}$, for $p, q > 1$. Considering this setting, we design a deterministic distributed broadcasting algorithm that completes the task asymptotically optimal in $O(ecc(s) + \log N)$ rounds. The algorithm consists of three parts. The source selects one of its neighbors during the first part. The goal of the second part is to compute the initialization information, which is later used in the third part of the broadcasting algorithm. We shall compute the initialization information only in neighbors of the source and in nodes that have in their neighborhood two neighbors of the source. Finally, the third part of the algorithm disseminates a source message in the network. Simultaneously, the control information, similar to the initialization information, is computed for newly informed nodes.

Definition 4.3.2. *A node is referred to as a* 2*-neighbor of the source if and only if it is adjacent to two neighbors of the source.*

Common subroutines and techniques

In this subsection, we present some supplementary techniques that are applied in the first and the second part of the broadcasting algorithm.

Definition 4.3.3. *Let v be a node with identifier $ID(v) > 0$ and $(a_1, \ldots, a_k)_2$ be a binary representation of the number $ID(v)$. An infinite binary sequence*

$1, a_k, a_{k-1}, \ldots, a_2, a_1, 0, 0, \ldots$ *is called a transmission sequence corresponding to the identifier* $ID(v)$.

Example. The transmission sequence corresponding to the identifier $11 = (1011)_2$ *is* $1, 1, 1, 0, 1, 0, 0, 0, \ldots$.

Note that the previous definition implies that transmission sequences corresponding to different identifiers differ in at least one position.

Consider the following case. A node u has at least one and at most two active (participating) neighbors, say v and w, that become informed in some unknown rounds (possibly different). We have to deliver the information from one of participating neighbors to the node u as soon as possible. This task can be easily solved applying previously defined transmission sequences. Suppose that a participating neighbor becomes informed in the round 0. In the i-th round, it transmits its information if and only if the i-th element of the transmission sequence corresponding to its identifier is equal to 1. This subroutine is referred to as TAI (transmission according to identifier). We show that at most $O(\log Max(ID(v), ID(w)))$ rounds are enough to inform the node u by one of its participating neighbors.

Lemma 4.3.4 ([C]). *The TAI-subroutine completes the prescribed task in at most* $O(\log Max(ID(v), ID(w)))$ *rounds. The total number of transmissions is* $O(\log Max(ID(v), ID(w)))$.

Proof. Suppose that the node v is informed in a round i and the second participating neighbor w (if exists) is not already informed in this round. Since the first element of the transmission sequence is always 1, v transmits in the round $i+1$ due to the *TAI*-subroutine. The node w does not transmit in the round $i+1$ and thus u becomes informed. Now suppose that v and w are informed simultaneously in a round i. According to TAI, they transmit in the round $i+1$, but the interference causes that u is not informed. Since $ID(v) \neq ID(w)$, the transmission sequences corresponding to $ID(v)$ and $ID(w)$ are different. Hence, there must be an index j such that j-th elements of their binary transmission sequences are different. It implies that exactly one participating neighbor of u transmits in the round corresponding to the index j, i.e., in the round $i+j$. Therefore, the node u is informed in this round. □

In order to avoid interaction during a simultaneous execution of several communication tasks, we use the time division multiplexing strategy. Particularly, the i-th task from the set of k tasks is executed in each round j, s.t., $j \equiv i \ (mod\, k)$. In our setting, we do not allow spontaneous transmission, i.e., a node (except the source or the initiator) cannot transmit before successful receiving of a message transmitted by other node. If we include the number of actual round modulo k in each transmitted message, newly informed nodes can synchronize. Thus all nodes participating in the algorithm can simultaneously execute the same task in a given round.

Selection of a neighbor of the source

Now, we describe an algorithm that selects one of the neighbors of the source. We shall utilize the *TAI*-subroutine. If a message which is transmitted in the *TAI*-subroutine includes the identifier of the sender, successful receiving of the message implies that the receiver can select one of the senders. *TAI*-subroutine works only in the case when there are at most two participating senders. On the other hand, the source has at most 4 neighbors in the grid graph. Hence, a direct application of the *TAI*-subroutine is not possible. With the assistance of 2-neighbors of the source, we split the selection process into several applications of *TAI*-subroutine, where at most two nodes participate in process of transmitting information towards a specific node.

In the first round, the source transmits an initialization message. It is received exactly by all neighbors of the source. Subsequently, we start 4 simultaneous tasks (each of them in a separate time slot modulo 4). In the first task, each neighbor of the source tries to inform adjacent 2-neighbors by its identity. Since exactly two neighbors of the source have to transmit towards a 2-neighbor, we utilize the *TAI*-subroutine. After at most $O(\log N)$ rounds, at least one 2-neighbor becomes informed. Note that the nodes are not aware of the fact whether they are 2-neighbors of the source or not. We solve this problem by a modification of the *TAI*-subroutine in such a way that the first transmission of a node contains a special message. If a node receives this special message, it knows that it is not 2-neighbor. Indeed, all neighbors of the source transmit the special message in the second round of the algorithm (the first round of the *TAI*-subroutine). Due to interference, this message cannot be received by 2-neighbors. An informed 2-neighbor starts its activity during the second task and ignores all messages received in the rounds assigned for the first task. Particularly, it attempts to send a message that includes the received identifier of one its neighbor. Again, we utilize the *TAI*-subroutine. All nodes, except neighbors of the source, ignore transmitted messages during execution of this task. Further, a neighbor of the source that receives its identifier in a round of the second task, starts the execution of the third task and attempts to send its identifier in a message to the source utilizing the *TAI*-subroutine. If a neighbor of the source received identifier of another node, it finishes its participation in all tasks, expect the fourth task. Moreover, each node that is active in the third task transmits a message in all rounds of the first task. Indeed, it blocks receipt of a message by its adjacent 2-neighbors. The source acknowledges the receipt of an identifier of one its neighbor by transmission of a message in the round reserved for the fourth task. This transmission stops execution of the first and the third task by all neighbors of the source because the selection task is accomplished. Besides, the neighbors ignore all received messages transmitted in rounds of the second task. The goals of tasks can be summarized as follows:

(1) the first task - neighbors of the source are sending their identifiers towards 2-neighbors

(2) the second task - 2-neighbors of the source are sending the received identifier (that was received in the execution of the first task) towards neighbors of the source,

(3) the third task - neighbors of the source that are informed in a round of the second task, are sending their identifiers towards the source

(4) the fourth task - the source acknowledges selection of a neighbor

Lemma 4.3.5 ([C]). *The source selects one of its neighbors in $O(\log N)$ rounds and using total $O(\log N)$ transmissions, where N is the unknown upper bound of identifiers assigned to the nodes of the network.*

Proof. Time and energy complexity of this part of the algorithm are obvious. Correctness of the algorithm can be shown by the case analysis of all possible states (and future transmissions due to the algorithm) in the round when a 2-neighbor of the source has received a message at the first time. □

Computation of initialization information

During this part, an initialization process is performed. The goal is to mark neighbors of the source with distinct labels from the set $\{A, B, C, D\}$ in such a way that a node marked by A (B) do not have a common neighbor with the node marked by C (D, respectively). Furthermore, we require that each 2-neighbor of the source knows the labels of both adjacent neighbors of the source. Thus these nodes (2-neighbors of the source) can be marked by distinct labels from the set $\{AB, BC, CD, AD\}$. Desired initial labelling of the nodes is described on Figure 4.7.

Note that only neighbors and 2-neighbors of the source participate in this part of the algorithm. All other nodes ignore transmitted messages.

1. The source transmits an initialization message containing the identifier of a selected neighbor.

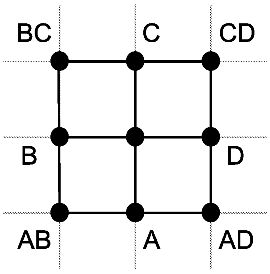

Figure 4.7: The initial marking of nodes

2. Selected neighbor transmits a message and marks itself with label A.
3. 2-neighbors of the source that received the message in the previous round retransmit the received message. A neighbor of the source that does not receive a message in this round marks itself with the label C.
4. Each unmarked neighbor of the source executes the *TAI*-subroutine. It transmits messages containing its identifier as an information content of the node (in the sense of *TAI*). Execution of the *TAI*-subroutine is interleaved with rounds that are reserved for the source. In one of these rounds, the source informs the nodes participating on *TAI* that an unmarked neighbor is selected. This notification is realized by a transmission of the identifier of a selected (unmarked) neighbor.
5. Selected unmarked neighbor sets its label to B and the unselected neighbor (if exists) to D.
6. In one of 4 rounds, each neighbor transmits its label in a round that is designated for its labels.
7. 2-neighbors of the source set the labels according to labels received in previous rounds.

It is easy to see that the initialization schema works properly even in the case when the source has less than 4 neighbors. All steps, except step 4, require constant number of rounds. The step 4 is accomplished in $O(\log N)$ rounds due to lemma 4.3.4. We summarize the time complexity of this part that computes initialization information in the following lemma.

Lemma 4.3.6 ([C]). *The initialization information can be computed in $O(\log N)$ rounds with the energy complexity $O(\log N)$ transmissions.*

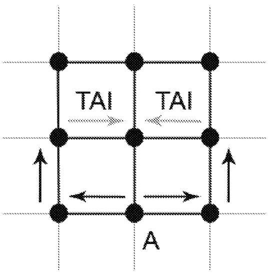

Figure 4.8: Scheme of the initial marking computation

Dissemination of the source message

This part of broadcasting algorithm works in phases. The goal of each phase is to disseminate a source message to the nodes that are in neighborhood of nodes informed during the previous phase. Furthermore, newly informed nodes compute auxiliary information that is used to arrange their transmissions in the next phase. The auxiliary information has similar structure as the initialization information. Particularly, the auxiliary information is a label of a node. Each node can be marked with a label from the set $\{A, B, C, D, A^+, B^+, C^+, D^+, AB, BC, CD, AD\}$. The source message is disseminated from the source in a wave that has the shape of square. All nodes located on one side of square have the same label, however, each side is marked with different label. The corner nodes are marked with a label from the set $\{AB, BC, CD, AD\}$. Intuitively, a label of a corner node expresses that the node belongs to two sides (directions of the broadcasting wave). We mark the nodes adjacent to the corner nodes with a label containing the $+$ sign.

Initially, we change labels of nodes in the neighborhood of the source in such a way that the label A is changed to A^+, B to B^+, C to C^+, and D to D^+. This transformation guarantees that the labels of nodes are compatible with the semantic description of labels stated above. In the first round of this part, neighbors and 2-neighbors of the source are considered as the nodes that were informed in the previous phase.

We assume that each transmitted message contains the source message and the number of actual round of executed phase. Each phase takes 5 communication rounds:

1. A node informed in the previous phase and marked with the label A, A^+, AB, AD, C, C^+, BC, or CD transmits a message containing its label.

2. A node informed in the previous phase and marked with the label B, B^+, C, or C^+ transmits a message containing its label.

3. The nodes that receive a label in the round 1 or 2 of the phase and are not yet informed set the label to the received label. These nodes are referred to as newly informed nodes of given phase. Each newly informed node that received the label A^+, B^+, C^+, or D^+ in the round 1 or 2 of the given phase transmits its label.

4. Newly informed nodes marked with label AB, AD, BC, or CD that received in the previous round the label A^+ or C^+ transmits their labels (received in round 1 or 2).

5. Newly informed nodes marked with label AB, AD, BC, or CD that received in the previous round the label B^+ or D^+ transmits their labels (received in round 1 or 2).

At the end of the phase, we change labels of some newly informed nodes. No communication is required in this step. At first, the nodes with label A^+, B^+, C^+, or D^+ change the labels to A, B, C, or D respectively. Next, the nodes with label AB, AD, BC, or CD change the label to a label received in the round 3. Finally, if a node received messages transmitted in the rounds 4 and 5, it sets its label to the received value and is considered as a newly informed node of the given phase. Note that the labels received in the round 4 and in the round 5 are the same.

It is easy to see that the number of phases is limited by the eccentricity $ecc(s)$ of the source. Each phase takes constant number of round and thus the third part of algorithm is completed in $O(ecc(s))$ rounds. Each node transmits constantly many times. It implies that energy complexity of this part is $O(n)$ transmissions.

Complexity of the broadcasting algorithm

The time and energy complexity of designed broadcasting algorithm is summarized in the following theorem.

Theorem 4.3.7 ([C]). *Consider a radio network such that its underlying reachability graph is a grid graph. There is a distributed deterministic algorithm that completes the broadcasting task in $O(ecc(s) + \log N)$ rounds using total $O(n + \log N)$ transmissions, where n is the number of nodes and N is the unknown upper bound of identifiers in the network. Moreover, designed algorithm is asymptotically optimal.*

Proof. The first part of algorithm (selection of a neighbor of the source) takes $O(\log N)$ rounds and uses $O(\log N)$ transmissions. Time complexity of the second part (computation of the initial information) is $O(\log N)$ rounds. Energy complexity of this part is $O(\log N)$ transmissions. Finally, the third part of algorithm that disseminates the source message, takes $O(ecc(s))$ rounds and uses $O(n)$ transmission. Therefore, the time complexity of the broadcasting algorithm is $O(ecc(s) + \log N)$ rounds. Summing total number of transmissions in each part of algorithm, we obtain that at most $O(n+\log N)$ messages are transmitted by nodes during the execution of the algorithm.

Note that it is usually assumed that $N = O(n^p)$, for a constant $p > 0$. In grid graphs, it holds that $ecc(s) \geq \sqrt{n} \geq \log n$ for sufficiently large n. Hence $O(\log N) = O(\log n) = O(ecc(s))$. In our setting, parameter N cannot be bounded in this way. Let us fix a deterministic broadcasting algorithm. One can show that there is such an assignment of identifiers to the nodes of a grid radio network $G_{3,3}$ that the broadcasting task cannot be completed in less that $\Omega(\log N)$ rounds. The proof could be obtained by adaptation of the argument that was used in the proof of the $\Omega(n\frac{\log n}{\log{(n/D)}})$-round lower bound on broadcasting time in [46]. □

Algorithm for acknowledged broadcasting

Note that the broadcasting algorithm presented in the previous section is not acknowledged, i.e., the source is not aware of the round when the broadcasting task is completed. Furthermore, the source is not able to compute the duration of the algorithm, because parameters of the radio network are unknown for the nodes. Presented principles of the constructed broadcasting algorithm allow to modify the algorithm to an algorithm completing the acknowledged broadcasting task. We present modified algorithm only briefly. The modification is based on the following. First, we add new labels A^*, B^*, C^*, and D^* to the set of labels that is used to mark the nodes. In each phase, new labels are used to mark one node, called a *progress node*, in each direction of the broadcasting wave. In particular, the progress nodes form a cross with the center in the source. The nodes forming a limb of the cross are marked with the same label and each limb of a cross is marked with a different label. We append a new round to each phase of algorithm. In this round, each active progress node informs its neighboring progress node which is closer to the source about the fact that the broadcasting task in the given direction is not yet accomplished. If an active progress node does not receive a message in this round during an appropriate phase, it becomes inactive. Since the nodes located on the border of the grid do not inform new nodes, a progress node on this border does not receive any message in this round. It causes a chain of continuous deactivations of progress nodes. Finally, there must be a phase in which the source is notified (by silent) about completing broadcasting in the given direction. If the source is informed that disseminations of the source message are completed in all direction, it knows that all nodes are informed. In order to avoid interference during the last round, we have to schedule transmissions in appropriate manner. Let v be a progress node that has been informed in the i-th phase (the number of the current phase must be included in each transmitted message). We define a number $P(v)$ as follows:

- $P(v) = (i \, mod \, 3) \, mod \, 4$, if v is marked with A^*
- $P(v) = (i \, mod \, 3 + 1) \, mod \, 4$, if v is marked with B^*
- $P(v) = (i \, mod \, 3 + 2) \, mod \, 4$, if v is marked with C^*
- $P(v) = (i \, mod \, 3 + 3) \, mod \, 4$, if v is marked with D^*

An active progress node v transmits a message in the last round of the j-th phase if and only if $j \equiv P(v) \; (mod \, 4)$.

It is easy to see that asymptotical time and energy complexity are preserved by this modification.

4.3.2 Computation of Coordinates

Since we consider radio networks with a regular topology, it can be assumed that the radio network is static. It means that the topology of the network remains unchanged for a long time period. This assumption heads towards the issue of computation of an communication structure for the collision-free communication. As we show later, the grid coordinates of nodes can serve as the basic information for a collision-free communication schema. In this section we present a distributed algorithm which computes grid coordinates of each node. The algorithm is a modification of the previously presented broadcasting algorithm. Particularly, it takes advantage of the auxiliary information computed during the third part of the broadcasting algorithm. We assume that the task of computation of grid coordinates is initiated by a distinguished node, called a *initiator*.

The algorithm consists of 3 parts. First two parts of the algorithm are identical to first two parts of the broadcasting algorithm. After this two part, an initialization information is computed. Now, we assign to the source coordinates $[0, 0]$ and to its neighbor marked with A^+ (B^+, C^+, D^+) coordinates $[0, 1]$ ($[-1, 0]$, $[0, -1]$, $[1, 0]$ respectively). It is easy to see that the assignment of coordinates is correct. Indeed, it follows from the way how the nodes are marked with labels during the second part of the algorithm. Similarly, we assign to a node marked with AB (BC, CD, and AD) coordinates $[-1, 1]$ ($[-1, -1]$, $[1, -1]$, and $[1, 1]$, respectively). Notice that, in the broadcasting algorithm, the labels of nodes store the information about direction in which the source message is disseminated. We shall use the sense of direction in order to compute grid coordinates of nodes according to information received from some their neighbors.

We modify content of messages sent in the third part of the broadcasting algorithm in the following way. Each message that is transmitted in the round 1 or 2 of the phase, contains coordinates of the sender. During execution of the third part of the algorithm, we preserve the invariant that each informed node (in sense of broadcasting algorithm) already computed its grid coordinates. Thus, the nodes transmitting in first two rounds of a phase have already computed their coordinates. Moreover, we modify messages sent in the round 4 and 5 of the phase. We attach grid coordinates received in one of first two rounds of the phase (in fact it happens in exactly one of those rounds) to the transmitted message. Finally, the newly informed nodes compute their coordinates at the end of the phase. Let $[x, y]$ be coordinates which were included in a message received in the round 1, 2, 4, or 5 of a phase. We apply the following rules to set the coordinates of a newly informed node:

- label of node A or A^+: $[x, y + 1]$
- label of node B or B^+: $[x - 1, y]$
- label of node C or C^+: $[x, y - 1]$
- label of node D or D^+: $[x + 1, y]$
- label of node AB: $[x - 1, y + 1]$
- label of node BC: $[x - 1, y + 1]$
- label of node CD: $[x + 1, y - 1]$
- label of node AD: $[x + 1, y + 1]$.

Theorem 4.3.8 ([C]). *Consider an unknown radio network such that its underlying reachability graph is a grid graph. There is a distributed deterministic algorithm that computes grid coordinates of each node in $O(ecc(s) + \log N)$ rounds with total $O(n + \log N)$ transmissions, where n is the number of nodes and N is an unknown upper bound of identifiers in the network. Designed algorithm is asymptotically optimal.*

Proof. Correctness of the algorithm follows from the properties of the broadcasting algorithm and the rules for computation of coordinates of newly informed nodes. Since we do not allow spontaneous transmission, i.e., to participate in algorithm before receiving a message from another node, the task cannot be accomplished in better time (and energy complexity) than the broadcasting task. It implies asymptotical optimality of designed algorithm. □

Note that we can design an algorithm in which the initiator of the computation is notified that the task is completed. It could be achieved by a similar modification of acknowledged broadcasting algorithm for radio networks with grid topology that is presented in Section 4.3.1.

Finally, note that the this algorithm for computation of grid coordinates of nodes can be adapted to compute dimensions of the underlying grid graph. Another adaptation could be an algorithm that solves the task of assignment of compact identifiers to the nodes (i.e., the nodes have to be labelled by unique numbers from the set $1, \dots, n$).

4.3.3 Collision-free Communication Mechanism

In this section, we discuss a collision-free communication mechanism for radio networks with grid topology. It is based on results concerning 2-distance coloring of grids that have been proposed by Fertin *et al.* in [30]. A 2-distance coloring of a graph is a proper coloring of vertices satisfying that no vertices in distance at most 2 have assigned the same color. Hence no vertex has in its neighborhood two vertices with the same color. Applying algorithm for computing grid coordinates, we may assume that each node is aware of its grid coordinates.

Definition 4.3.9. *Let $[x, y]$ be the coordinates of a node v. The number $TR(v) = (2x + y) \, mod \, 5$ is called a collision free number of the node v.*

Collision free mechanism is defined as follows. A node v is allowed to transmit a message in the i-th round if and only if $i \equiv TR(v)$ $(mod \, 5)$. Correctness of this mechanism follows from that fact that $TR(v)$ corresponds to a 2-distance coloring of a grid with 5 colors, which is moreover shown to be optimal in [30]. Thus we can use algorithms that are not primary designed for the communication in radio networks. The slowdown caused by the presented mechanism is only by a constant factor.

Chapter 5

Conclusion and Further Work

In this dissertation thesis, we studied time efficient communication algorithms for multihop radio networks. We considered several models of radio networks that reflect different communication environments and real-world settings. We took into account also energy complexity of proposed algorithms.

In Chapter 2, we investigated centralized broadcasting in known topology radio networks in the case when an underlying reachability graph satisfies certain topology restrictions, i.e., it belongs to a specific class of graphs. It follows from designed general schema for generating broadcasting schedules that, in order to get much efficient broadcasting schedules, one can look for graph classes in which information dissemination in bipartite graphs can be realized in $o(\log n)$ rounds. Another interesting research problem is whether other approaches (e.g., based on clustering or a sort of generalization of the algorithm from [49]) allow construct better schemas/algorithms for some graph classes.

In Chapter 3, we introduced and studied communication in new graph-based model of radio networks that is much closer to modelling real-world communication environment than the standard graph model. Moreover, it generalizes the standard graph model. As we showed, efficiency of algorithms designed for this model (and for models with long-range interference in general) is very sensitive to what parameters reflecting the presence of interference are taken into account. Therefore, it is important to define and investigate other parameters that can better reflect the presence of interference. Efficiency of schedules generated by all designed algorithms is bounded with respect to the maximum degree Δ. It could be interesting to study whether this parameter can be replaced with another parameter with the same success and supporting lower bounds. Obvious open problem is communication in unknown radio networks with long-range interference. Probably, a sort of (interference) selectors should be defined in order to treat this problem effectively. Graph-based models adopted by algorithmic community are often criticized because they do not provide adequate modelling for real-world communication environments, devices, and their properties. A natural issue arises. How can graph-based models be modified in order to provide better model for real-world networks without affecting their simplicity ?

In Chapter 4, we focused on communication in radio networks with nodes unaware of an underlying reachability graph. Especially, we explored the concept of encoding information into collisions that is possible if the network nodes are capable of collision detection. Presented results showed that this concept is very powerful, since many communication problems can be solved in this setting very effectively even in the case when the nodes are anonymous, i.e., they do not have identifiers. We considered communication tasks that are not in the center of investigation in this area. Generally, all of those tasks can be solved by a combination of standard communication primitives. However, a problem of effective combination of those well-studied communication primitives emerges. More complex communication tasks could define special (restricted) settings for communication primitives that are worth to investigate. The concluding part of Chapter 4 considers a setting when the network nodes do not know an underlying reachability graph, but they know that this graph belongs to a certain graph class. We explored this setting in the case of grid graphs. Interesting open problem is to answer the question what graph classes allow more efficient communication if the nodes have an information that an underlying reachability graph belongs to a certain graph class. Note that, unlike the case of planar graphs, such an information is useful in the case of grid graphs.

Bibliography

[1] N. Alon, A. Bar-Noy, N. Linial, and D. Peleg. A lower bound for radio broadcast. *J. Comput. Syst. Sci.*, 43(2):290–298, 1991.

[2] R. Bar-Yehuda, O. Goldreich, and A. Itai. On the time-complexity of broadcast in multi-hop radio networks: An exponential gap between determinism and randomization. *J. Comput. Syst. Sci.*, 45(1):104–126, 1992.

[3] P. Berenbrink, C. Cooper, and Z. Hu. Energy efficient randomised communication in unknown adhoc networks. In P. B. Gibbons and C. Scheideler, editors, *SPAA*, pages 250–259. ACM, 2007.

[4] J.-C. Bermond, J. Galtier, R. Klasing, N. Morales, and S. Pérennes. Hardness and approximation of gathering in static radio networks. *Parallel Processing Letters*, 16(2):165–183, 2006.

[5] A. D. Bonis, L. Gasieniec, and U. Vaccaro. Generalized framework for selectors with applications in optimal group testing. In J. C. M. Baeten, J. K. Lenstra, J. Parrow, and G. J. Woeginger, editors, *ICALP*, volume 2719 of *Lecture Notes in Computer Science*, pages 81–96. Springer, 2003.

[6] M. Borowiecki, I. Broere, M. Frick, P. Mihók, and G. Semanišin. Survey of hereditary properties of graphs. *Discuss. Math. Graph Theory*, 17:5–50, 1997.

[7] D. Bruschi and M. D. Pinto. Lower bounds for the broadcast problem in mobile radio networks. *Distributed Computing*, 10(3):129–135, 1997.

[8] I. Chlamtac and S. Kutten. On broadcasting in radio networks - problem analysis and protocol design. COM-33(12):1240–1246, 1985.

[9] I. Chlamtac and O. Weinstein. The wave expansion approach to broadcasting in multihop radio networks. COM-39(3):426–433, 1991.

[10] B. S. Chlebus, L. Gąsieniec, A. Gibbons, A. Pelc, and W. Rytter. Deterministic broadcasting in unknown radio networks. In *SODA*, pages 861–870, 2000.

[11] B. S. Chlebus and D. R. Kowalski. Almost optimal explicit selectors. In M. Liskiewicz and R. Reischuk, editors, *FCT*, volume 3623 of *Lecture Notes in Computer Science*, pages 270–280. Springer, 2005.

[12] B. S. Chlebus, D. R. Kowalski, and M. A. Rokicki. Average-time complexity of gossiping in radio networks. In P. Flocchini and L. Gąsieniec, editors, *SIROCCO*, volume 4056 of *Lecture Notes in Computer Science*, pages 253–267. Springer, 2006.

[13] B. S. Chlebus and M. A. Rokicki. Asynchronous broadcast in radio networks. In Královič and Sýkora [50], pages 57–68.

[14] M. Christersson, L. Gąsieniec, and A. Lingas. Gossiping with bounded size messages in ad hoc radio networks. In P. Widmayer, F. T. Ruiz, R. M. Bueno, M. Hennessy, S. Eidenbenz, and R. Conejo, editors, *ICALP*, volume 2380 of *Lecture Notes in Computer Science*, pages 377–389. Springer, 2002.

[15] M. Chrobak, L. Gąsieniec, and W. Rytter. Fast broadcasting and gossiping in radio networks. *J. Algorithms*, 43(2):177–189, 2002.

[16] M. Chrobak and T. H. Payne. A linear-time algorithm for drawing a planar graph on a grid. *Inf. Process. Lett.*, 54(4):241–246, 1995.

[17] F. Cicalese, F. Manne, and Q. Xin. Faster centralized communication in radio networks. In S. K. Madria, K. T. Claypool, R. Kannan, P. Uppuluri, and M. M. Gore, editors, *ISAAC*, volume 4317 of *Lecture Notes in Computer Science*, pages 339–348. Springer, 2006.

[18] A. E. F. Clementi, P. Crescenzi, A. Monti, P. Penna, and R. Silvestri. On computing ad-hoc selective families. In M. X. Goemans, K. Jansen, J. D. P. Rolim, and L. Trevisan, editors, *RANDOM-APPROX*, volume 2129 of *Lecture Notes in Computer Science*, pages 211–222. Springer, 2001.

[19] A. E. F. Clementi, A. Monti, and R. Silvestri. Distributed multi-broadcast in unknown radio networks. In *PODC*, pages 255–264, 2001.

[20] A. E. F. Clementi, A. Monti, and R. Silvestri. Selective families, superimposed codes, and broadcasting on unknown radio networks. In *SODA*, pages 709–718, 2001.

[21] T. Cormen, C. Leiserson, and R. Rivest. *Introduction to Algorithms*, chapter Section 8.2: Counting sort, page 168–170. The MIT Press, Cambridge, MA, 1990.

[22] A. Czumaj and W. Rytter. Broadcasting algorithms in radio networks with unknown topology. In *FOCS*, pages 492–501. IEEE Computer Society, 2003.

[23] A. Dessmark and A. Pelc. Tradeoffs between knowledge and time of communication in geometric radio networks. In *SPAA*, pages 59–66, 2001.

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

[25] K. Diks, E. Kranakis, D. Krizanc, and A. Pelc. The impact of knowledge on broadcasting time in radio networks. In J. Nesetril, editor, *ESA*, volume 1643 of *Lecture Notes in Computer Science*, pages 41–52. Springer, 1999.

[26] M. Elkin and G. Kortsarz. Logarithmic inapproximability of the radio broadcast problem. *J. Algorithms*, 52(1):8–25, 2004.

[27] M. Elkin and G. Kortsarz. Improved schedule for radio broadcast. In *SODA*, pages 222–231. SIAM, 2005.

[28] R. Elsässer and L. Gąsieniec. Radio communication in random graphs. *J. Comput. Syst. Sci.*, 72(3):490–506, 2006.

[29] Y. Emek, L. Gąsieniec, E. Kantor, A. Pelc, and D. Peleg. Upper bounds on broadcasting time in udg radio networks with unknown topology. Technical report, Computer Science and Applied Mathematics, Weizmann Institute of Science, Israel, 2006.

[30] G. Fertin, E. Godard, and A. Raspaud. Acyclic and k-distance coloring of the grid. *Inf. Process. Lett.*, 87(1):51–58, 2003.

[31] I. Gaber and Y. Mansour. Centralized broadcast in multihop radio networks. *J. Algorithms*, 46(1):1–20, 2003.

[32] F. Galčík. Complexity aspects of radio networks. Master's thesis, Faculty of Mathematics, Physics and Informatics, Comenius University in Bratislava, Slovakia, 2005.

[33] L. Gąsieniec, E. Kantor, D. R. Kowalski, D. Peleg, and C. Su. Time efficient k-shot broadcasting in known topology radio networks. *Distributed Computing*, 21(2):117–127, 2008.

[34] L. Gąsieniec and A. Lingas. On adaptive deterministic gossiping in ad hoc radio networks. *Inf. Process. Lett.*, 83(2):89–93, 2002.

[35] L. Gąsieniec, A. Pagourtzis, and I. Potapov. Deterministic communication in radio networks with large labels. In R. H. Möhring and R. Raman, editors, *ESA*, volume 2461 of *Lecture Notes in Computer Science*, pages 512–524. Springer, 2002.

[36] L. Gąsieniec, D. Peleg, and Q. Xin. Faster communication in known topology radio networks. In M. K. Aguilera and J. Aspnes, editors, *PODC*, pages 129–137. ACM, 2005.

[37] L. Gąsieniec, I. Potapov, and Q. Xin. Time efficient gossiping in known radio networks. In Královič and Sýkora [50], pages 173–184.

[38] L. Gąsieniec, T. Radzik, and Q. Xin. Faster deterministic gossiping in directed ad hoc radio networks. In T. Hagerup and J. Katajainen, editors, *SWAT*, volume 3111 of *Lecture Notes in Computer Science*, pages 397–407. Springer, 2004.

[39] O. Goussevskaia, T. Moscibroda, and R. Wattenhofer. Local broadcasting in the physical interference model. In M. Segal and A. Kesselman, editors, *DIALM-POMC*, pages 35–44. ACM, 2008.

[40] P. Gupta and P. R. Kumar. The capacity of wireless networks. *IEEE Transactions on Information Theory*, 46(2):388–404, 2000.

[41] J. Hopcroft and R. Tarjan. Efficient planarity testing. *J. ACM*, 21(4):549–568, 1974.

[42] P. Indyk. Explicit constructions of selectors and related combinatorial structures, with applications. In *SODA*, pages 697–704, 2002.

[43] C.-Y. Koo, V. Bhandari, J. Katz, and N. H. Vaidya. Reliable broadcast in radio networks: the bounded collision case. In *PODC '06: Proceedings of the twenty-fifth annual ACM symposium on Principles of distributed computing*, pages 258–264, New York, NY, USA, 2006. ACM Press.

[44] K. Kothapalli, C. Scheideler, M. Onus, and A. W. Richa. Constant density spanners for wireless ad-hoc networks. In P. B. Gibbons and P. G. Spirakis, editors, *SPAA*, pages 116–125. ACM, 2005.

[45] D. R. Kowalski and A. Pelc. Deterministic broadcasting time in radio networks of unknown topology. In *FOCS*, pages 63–72. IEEE Computer Society, 2002.

[46] D. R. Kowalski and A. Pelc. Broadcasting in undirected ad hoc radio networks. In *PODC*, pages 73–82, 2003.

[47] D. R. Kowalski and A. Pelc. Faster deterministic broadcasting in ad hoc radio networks. In H. Alt and M. Habib, editors, *STACS*, volume 2607 of *Lecture Notes in Computer Science*, pages 109–120. Springer, 2003.

[48] D. R. Kowalski and A. Pelc. Centralized deterministic broadcasting in undirected multi-hop radio networks. In K. Jansen, S. Khanna, J. D. P. Rolim, and D. Ron, editors, *APPROX-RANDOM*, volume 3122 of *Lecture Notes in Computer Science*, pages 171–182. Springer, 2004.

[49] D. R. Kowalski and A. Pelc. Optimal deterministic broadcasting in known topology radio networks. *Distributed Computing*, 19(3):185–195, 2007.

[50] R. Královič and O. Sýkora, editors. *Structural Information and Communication Complexity, 11th International Colloquium , SIROCCO 2004, Smolenice Castle, Slowakia, June 21-23, 2004, Proceedings*, volume 3104 of *Lecture Notes in Computer Science*. Springer, 2004.

[51] E. Kranakis, D. Krizanc, and A. Pelc. Fault-tolerant broadcasting in radio networks. *J. Algorithms*, 39(1):47–67, 2001.

[52] D. Lick and A. White. k-degenerate graphs. *Canad. J. Math*, 22:1082–1096, 1970.

[53] F. Manne, S. Wang, and Q. Xin. Faster radio broadcasting in planar graphs. In *4th Annual Conference on Wireless On demand Network Systems and Services*, 2007.

[54] G. D. Marco. Distributed broadcast in unknown radio networks. In S.-H. Teng, editor, *SODA*, pages 208–217. SIAM, 2008.

[55] G. D. Marco and A. Pelc. Faster broadcasting in unknown radio networks. *Inf. Process. Lett.*, 79(2):53–56, 2001.

[56] C. U. Martel. Maximum finding on a multiple access broadcast network. *Inf. Process. Lett.*, 52(1):7–15, 1994.

[57] T. Muetze, P. Stuedi, F. Kuhn, and G. Alonso. Understanding radio irregularity in wireless networks. In *SECON*, pages 82–90. IEEE, 2008.

[58] K. Nakano and S. Olariu. *Handbook of wireless networks and mobile computing*, chapter Leader election protocols for radio networks, pages 219–242. In Stojmenovic [66], 2002.

[59] T. Okuwa, W. Chen, and K. Wada. An optimal algorithm of acknowledged broadcasting in ad hoc radio networks. In *ISPDC*, pages 178–184. IEEE Computer Society, 2003.

[60] M. Onus, A. W. Richa, K. Kothapalli, and C. Scheideler. Efficient broadcasting and gathering in wireless ad-hoc networks. In *ISPAN*, pages 346–351. IEEE Computer Society, 2005.

[61] A. Pelc. *Handbook of wireless networks and mobile computing*, chapter Broadcasting in radio networks, pages 509–528. In Stojmenovic [66], 2002.

[62] D. Peleg. Time-efficient broadcasting in radio networks: A review. In T. Janowski and H. Mohanty, editors, *ICDCIT*, volume 4882 of *Lecture Notes in Computer Science*, pages 1–18. Springer, 2007.

[63] L. Petrů and J. Wiedermann. A model of an amorphous computer and its communication protocol. In J. van Leeuwen, G. F. Italiano, W. van der Hoek, C. Meinel, H. Sack, and F. Plasil, editors, *SOFSEM*, volume 4362 of *Lecture Notes in Computer Science*, pages 446–455. Springer, 2007.

[64] C. Scheideler, A. W. Richa, and P. Santi. An o(log n) dominating set protocol for wireless ad-hoc networks under the physical interference model. In X. Jia, N. B. Shroff, and P.-J. Wan, editors, *MobiHoc*, pages 91–100. ACM, 2008.

[65] A. Sen and M. L. Huson. A new model for scheduling packet radio networks. *Wireless Networks*, 3(1):71–82, 1997.

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

[67] A. N. Strahler. Hypsometric (area-altitude) analysis of erosional topology. *Bull. Geol. Soc. Am.*, 63:1117–1142, 1952.

[68] X. G. Viennot. A strahler bijection between dyck paths and planar trees. *Discrete Mathematics*, 246(1-3):317–329, 2002.

[69] D. Welsh and M. Powell. An upper bound for the chromatic number of a graph and its application to timetabling problems. *The Computer Journal 10*, 1(10):85–86, 1967.

[70] Q. Xin. *Time Efficient Communication in Multi-hop Radio Networks.* PhD thesis, Department of Computer Science, The University of Liverpool, United Kingdom, 2004.