ENERGY EFFICIENT BROADCASTING IN WIRELESS AD HOC NETWORKS

by
Serkan ÇİFTLİKLİ

Submitted to the Graduate School of Engineering and Natural Sciences
in partial fulfillment of
the requirements for the degree of
Master of Science

Sabancı University
Summer 2008

ENERGY EFFICIENT BROADCASTING IN WIRELESS AD HOC NETWORKS

APPROVED BY

Assist. Prof. Dr. Kerem BÜLBÜL ....
(Thesis Supervisor)

Assist. Prof. Dr. Özgür ERÇETİN ....
(Thesis Co-supervisor)

Assist. Prof. Dr. Özgür GÜRBÜZ ....

Assist. Prof. Dr. Tonguç ÜNLÜYURT ....

Assist. Prof. Dr. Hüsnü YENİGÜN ....

DATE OF APPROVAL: ....

to my family...

Acknowledgments

I would like to express my gratitude to my thesis advisors, Kerem BÜLBÜL and Özgür ERÇETİN for their enthusiasm, constant inspiration, patience, and sincerity. They have provided me the opportunity to study on my topic as the motivation and stimulation that have prompted me to put all my efforts to this work. Without their trust, completion of this thesis would have been impossible. Their invaluable guidance has not just accompanied me through the research and writing of my thesis but will lead me through my life.

First of all, I would like to thank to my brother Orhan SAYGIN for being the guarantor of my scholarship and his continuous encouragement during my graduate education. In addition, I am grateful to my very close friends Sevgin RÜSTEMOĞLU, Ünal ERKASLAN, Ece ERKOL, Gülistan KARA and Adil SARIKAYA for being with me, and sharing my thoughts, passion and depression. I would also like to present my special thanks to my room-mate U. Mahir YILDIRIM and semi-room-mate Burak AKSU with whom we have supported each others in the name of the fellowship of 311 and semi-fellowship of 311, respectively. I feel very happy that I had the chance of working with such brilliant and anonymous friends; Elvin ÇOBAN, Ayfer BAŞAR, Duygu TAŞ, Nihan ÖZŞAMLI, Birol YÜCEOĞLU, Sevilay GÖKDUMAN and Nurşen AYDIN. In addition, hereby I present my respect to my anonymous Ph.D. friends; M. Sarper GÖKTÜRK, Figen ÖZTOPRAK, L. Taner TUNÇ and İbrahim MUTER.

Lastly, I would like to thank my beloved friend Sinem YILDIZ and my family for the concern, caring, and endless love that they provided.

Abstract

In recent years wireless multi-hop networks have attracted significant attention due to their wide range of potential civil and military applications. Broadcasting is a fundamental data dissemination scheme for these networks. The transmission power control is an important issue in wireless ad hoc networks and still has no satisfactory solution methods. The wireless networking environment presents formidable challenges to the study of broadcasting problems. In particular, the properties of the wireless medium and the presence of battery-powered devices require novel modeling and algorithmic approaches concentrating on judicious use of limited energy resources in wireless networks. In addition, networks are often required to provide certain quality of service (QoS) guarantees in terms of the end-to-end delay along the individual paths from the source to each of the destination nodes. Moreover, the received signal at each receiving node must be strong enough to be successfully decoded. In this study we address the minimum-energy broadcast problem in multi-hop wireless networks with respect to two different constraints: (*i*) each node must receive broadcast message within a given delay bound Δ, and (*ii*) signal-to-interference-plus-noise ratio (SINR) of the received signal must be above a given threshold γ so that the received signal can be successfully decoded at the receiving node. We propose two distinct algorithms Distributed Tree Expansion (DTE) and SINR-BIP which aim to generate minimum power broadcast tree with respect to constraint (*i*) and (*ii*), respectively and exclusively. DTE is based on an implementation of a distributed minimum spanning tree algorithm in which the tree grows at each iteration by adding a node that can cover the maximum number of currently uncovered nodes in the network with minimum incremental transmission power and without violating the delay constraint. In SINR-BIP, we apply the similar idea of well-known Broadcast Incremental Power (BIP) algorithm while considering the SINR values of received powers. In addition, we use an embedded pruning procedure in SINR-BIP, so that the myopic effect of the algorithm is mitigated. Both the algorithms DTE and SINR-BIP are constructive in nature since the broadcast tree grows at each iteration. We observed that the DTE outperforms the existing algorithms and the total energy consumptions of the generated broadcast trees by DTE is within 20% percent of the solutions obtained by Integer Programming.

Özet

Son yıllarda çok sekmeli kablosuz şebekeler sivil ve askeri alanlardaki çok çeşitli potansiyel uygulamaları nedeniyle önemli ölçüde dikkat çekmiştir. Bu tür şebekeler için çoğa gönderim temel bir veri dağıtım yöntemidir. İletim güç kontrolünün kablosuz tasarsız ağlar için çok önemli bir konudur ve hala tatmin edici çözümlerin bulunmamaktadır. Kablosuz ağ oluşturma ortamı çoğa gönderim problemleri için zorlu bir alandır. Özellikle kablosuz ortam özellikleri ve pille çalışan araçların bulunması kablosuz şebekelerde kısıtlı olan enerji kaynaklarının makul kullanılmasını amaçlayan yeni modelleme yaklaşımları ve algoritmik yaklaşımlar gerektirmektedir. Buna ek olarak, şebekelerin kaynaktan hedef cihazlara kadar olan her iletim yolu üzerinde ileti gecikmesi ile ilgili olarak belli bir servis kalitesi garantisi sunması gerekmektedir. Dahası, her alıcı cihazda algılanan sinyalin başarılı bir şekilde çözülebilmesi için sinyalin yeteri kadar güçlü olması gerekmektedir. Bu sebeble, bu çalışmada çok sekmeli kablosuz ağlarda en az enerjili çoğa gönderim problemini iki farklı kısıt doğrultusunda incelemekteyiz: (i) her cihaz çoğa gönderim mesajını belli bir gecikme sınırı olan Δ içerisinde almalıdır, ve (ii) alınan sinyalin işaret-parazit-artı-gürültü-oranı (SINR) eşik değer γ'nın üzerinde olmalıdır ki sinyal başarı ile çözülebilsin. Bu nedenle, sırasıyla kısıt (i) ve (ii) doğrultusunda en az enerjili çoğa gönderim ağacı oluşturacak DTE ve SINR-BIP adlarında iki farklı algoritma önermekteyiz. DTE minimum örten ağaç algoritmasının dağıtık uygulanmasını baz almaktadır. Her yinelemede, gecikme sınırı kısıtını ihlal etmeden en az enerji artışı gerektiren ve mevcut durumda ulaşılamamış düğümlerden en fazlasına ulaşabilen düğüm eğaca eklenerek çoğa gönderim ağacı büyür. SINR-BIP algoritmasında ise alınan sinyallerin SINR değerleri göz önünde bulundurularak iyi bilinen çoğa gönderim güç-artışı (BIP) algoritmasındaki ana fikre benzer bir yaklaşım uygulanmıştır. Buna ek olarak, algoritmanın miyop etkisini azaltmak için SINR-BIP'in içine gömülü olarak bir budama yordamı kullanılmıştır. Çoğa gönderim ağacı her yinelemede büyüdüğü için her iki algoritma da doğası gereği yapıcı algoritmalardır. DTE'nin mevcut algoritmalardan daha iyi performans gösterdiğini ve DTE ile elde edilen çoğa gönderim ağacının toplam enerji tüketiminin Tamsayı Programlama ile elde edilene %20 oranında yakın olduğu gözlenmektedir.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In recent years, wireless multi-hop networks in the form of ad hoc networks have attracted significant attention due to their potential applications in civil and military domains. Wireless ad hoc networks consist of numerous communication devices such as laptops, personal digital assistant or any mobile devices. Each communication device is equipped with processing, memory and wireless communication capabilities. In addition, communication links are established by short-range ad hoc radio connections. Moreover, no wired backbone infrastructure is installed in ad hoc wireless networks like in wired or cellular networks. A communication between the source and destination nodes is achieved either through a single-hop transmission, or through relaying by intermediate nodes. Since it is a wireless environment and each node operates unattended, each node in the network has limited resources. For example, energy is supplied by batteries and for some applications it is non-renewable. Therefore, the efficient use of available resources is an important design consideration for these networks.

In broadcast communication scheme, messages are concurrently sent to all nodes in the network. Such communication scheme is critical in applications where close collaboration of the network components is required to carry out a given task. During the broadcast/multicast sessions, efficient use of network resources is crucial and energy consumption is one of the most important issues related to ad hoc wireless networks because devices are usually equipped with batteries with limited lifetime.

We focus on a specific type of ad hoc network where all nodes are stationary, equipped with omnidirectional antenna, and the transmission range of the transmitter can be adjusted. The main concern of this study is the efficient use of energy while considering some issues related to ad hoc networks such as delay constraint and interference. For some applications, certain quality of service (QoS) guarantees are required in wireless networking environments. QoS is usually defined as the set of service requirements that need to be met by the network while transporting a packet stream from a source to destination nodes [2]. The QoS guarantee considered in this work is the end-to-end delay bound for the broadcast sessions where each node in the network desires to receive message within the delay bound. Since the delay of a broadcast message is highly correlated with the number of hops that a message travels, we require that the number of hops of individual paths from the source to each of the destination nodes is not greater than a pre-specified value. Another important issue that is considered in this study is the interference and environmental noise which affects the success of the signal reception.

Our contribution in this study is twofold; first, we propose a distributed algorithm Distributed Tree Expansion (DTE) for delay constrained minimum power broadcasting problem (DCMPB), and second we propose a centralized heuristic algorithm SINR-BIP for SINR constrained minimum power broadcasting (SCMPB) problem. Both of the proposed algorithms exploit the broadcast incremental power structure proposed in [3] while considering the other problem specific requirements.

First, we focus on constructing a minimum power broadcast tree with a maximum depth Δ, which corresponds to the maximum tolerable end-to-end delay in the network. DCMPB problem is considered to be NP-hard [5] by a reduction from the (unconstrained) minimum power broadcasting problem, shown to be NP-complete in [4], thus efficient heuristic approaches are required. In [5] and [7], a centralized approach for the problem where the source node has the global network topology information and performs the broadcast tree calculations is discussed. On the other hand, a distributed approach is proposed in [6]. In the present work, we investigate the construction of the delay bounded minimum power wireless broadcasting tree in a distributed fashion when each node in the network has only limited topology information and contributes to the construction of the tree.

Structure of the DTE algorithm is constructive in nature, where the broadcast tree is extended by adding one node or a group of nodes to the tree. DTE algorithm can be considered as a modified version of the distributed implementation of Minimum Spanning Tree (MST) algorithm. Basically, the algorithm begins initially with only the source node in the tree and the broadcast tree iteratively grows. At each iteration, a single node or a group of nodes are covered by considering the additional energy requirement of current partial tree and the delay bound. Note that the proposed algorithm only keeps 1-hop neighborhood information and is similar in nature to the one given in [8].

Second, we focus on constructing a minimum power broadcast tree while considering interference and environmental noise around the receiving nodes. In literature, energy efficient broadcasting problems are generally divided into two subproblems which are routing and scheduling problems. Routing algorithms [3, 18, 29, 27] construct the broadcast tree, then scheduling algorithms [36, 37, 39] schedule the transmissions so that the interference effect of the received signals is taken into account. In this work, we propose the algorithm SINR-BIP which takes into account the interference and environmental noise by considering SINR values of received signals while constructing the broadcast tree so that no scheduling algorithm is required. To the best of our knowledge, this is the first study that aims to construct energy efficient broadcasting tree while considering the interference and environmental noise by a single algorithm.

SINR-BIP is also a constructive algorithm in nature, and exploits the broadcast incremental power structure. Basically, a node with the minimum incremental power requirement is added to the broadcast tree at each iteration as in well known BIP algorithm [3]. However, SINR-BIP differs from BIP in two ways: (*i*) SINR-BIP takes into account the interference and environmental noise for signal reception, and (*ii*) In some iterations, SINR-BIP prunes the constructed broadcast tree in order to obtain better solutions for the next iterations. Note that the algorithm SINR-BIP is a centralized algorithm where the entire topology information is required.

The study is organized as follows. In Chapter II, we give brief background information and review the related literature on the minimum power broadcasting problem. In Chapter III and Chapter IV proposed heuristic algorithms DTE and SINR-BIP are described, respectively. The study is concluded in Chapter VI.

Chapter 2

Literature Review

2.1 Wireless Ad Hoc Networks

In recent years various types of wireless networks are preferred for different applications. In traditional wireless communication networks, such as cellular networks and wireless local area networks (WLAN), mobile devices associate with a network infrastructure which provides access to a wired backbone. This type of networks have single-hop communication structure where the nodes directly communicate with the host over a specified single link [9]. On the other hand, communication can be achieved by more than one link between two communicating devices for wireless ad hoc networks. Wireless ad hoc networks are referred to the type of communication architecture where devices, each equipped with a wireless transmission interface, establish a self-organizing network environment [10]. Since these types of networks are used in an ad hoc settings, no wired backbone infrastructure is installed. Thus, the communicating devices act as a router as well as the host with packet forwarding capabilities in order to communicate with each other [1]. In ad hoc wireless networks, resources including energy, band-width, processing capacity and memory are strictly limited when compared to the wired environment [2]. Consequently, the efficient use of available resources is an important design consideration for these networks.

Wireless ad hoc networks have a wide area of applications: i.e., emergency search and rescue operations, data acquisition operations in hospitable environments, decision making in the battlefield, etc [11]. Mostly the communication in ad hoc wireless networks is supported by multi-hop transmissions because of the physical limitations of resources and wireless links. In such a scenario, each communicating device should cooperate with each other by relaying the packets from the source device to destination

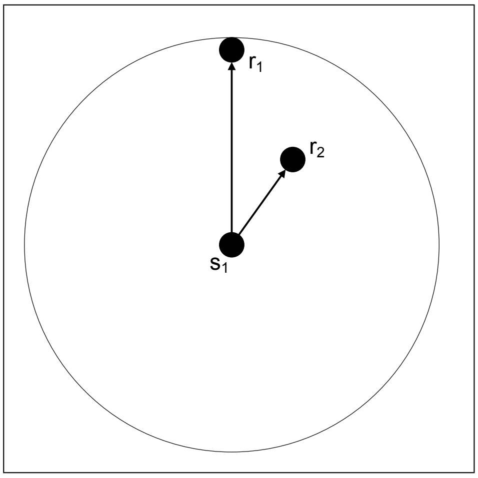


Figure 2.1: Direct transmission

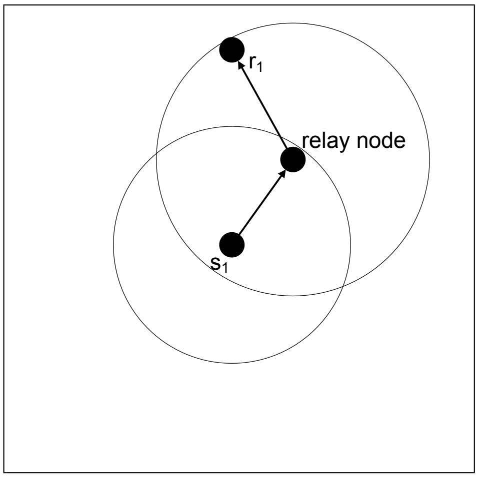


Figure 2.2: Relayed transmission

devices. Figure (2.1) and Figure (2.2) clearly illustrate the direct communication and relayed communication, respectively.

In wireless ad hoc networks, it is generally assumed that the signals are sent and received by using omnidirectional antennas. Note that an omnidirectional antenna radiates or receives equally well in all directions. Since a wireless network is connected by logical links, we say that the link between two nodes is established, if the required transmission power is assigned to the transmitting node. Since the signal power attenuates at the rate of $r^{-\alpha}$, where r is the distance from the signal source and α is the path loss coefficient, the received signal at a node w must be strong enough to be decoded successfully. The strength of a received signal is measured by signal-to-interference-plus-noise-ratio (SINR) which represents the dominance of the received signal power to the environmental noise and interference at the receiving node. Suppose that the node v transmits with power level $P(v)$ and distance between node v and w is $d(v, w)$, then the SINR value at a node w is calculated as follows;

$$SINR(w) = \frac{P_{rec}}{C + I} = \frac{P(v)\, d(v, w)^{-\alpha}}{C + I}$$

where P_{rec} is the received signal power from v at node w, C is the environmental noise around w, and I is the sum of interfering signal power received at node w. If the SINR value of a signal is above the threshold γ, then it is said that the signal is strong enough to be successfully decoded. However, the simplified version of the signal propagation model where the interference and environmental noise are ignored is commonly preferred. In the simplified model, the transmission power of the node v

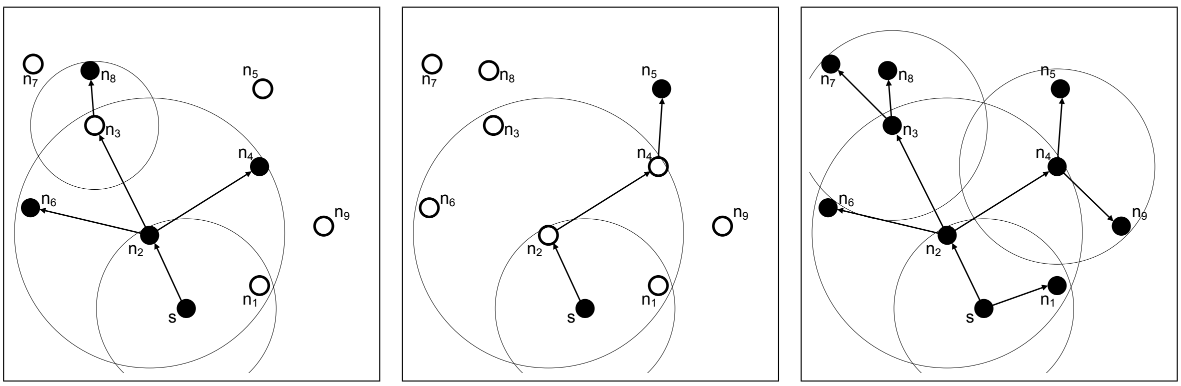
is proportional to r^α, and calculated as $P(v) = \xi\, r^\alpha$, where ξ is the signal detection threshold of node w and usually normalized to 1.

Assuming that each communication device is equipped with omnidirectional antenna, signal propagation occurs in all directions which results in an area of coverage and all nodes located in this area receive messages with a single transmission as shown in Figure (2.1). Reaching several nodes by a single transmission is called as "*wireless multicast advantage* [3]. The broadcasting and multicasting applications exploit the obvious benefits of "*wireless multicast advantage*". However, mathematically the energy related problems of wireless broadcasting and multicasting application become harder to solve [11].

2.2 Data Dissemination

In a wired or wireless network, data dissemination aims to transfer the information to the nodes in the network. Basically, data dissemination schemes can be classified into three groups; (*i*) multicasting, (*ii*) unicasting, and (*iii*) broadcasting.

Multicasting is the delivery of information to a specific group of destinations simultaneously. Note that the non-destination nodes may or may not receive the message and they may or may not be used for relaying, but each destination node must receive the message in multicasting. Suppose that the destination set is $D = \{n_2, n_4, n_6, n_8\}$, then the Figure (2.3)-a illustrates the multicasting structure.



(a) Multicasting (b) Unicasting (c) Broadcasting

Figure 2.3: Data Dissemination Schemes

Unicasting and broadcasting are the special forms of multicasting which assume the destination set consisting of only one node and all the nodes in the network, respectively. Assuming that the $D = \{n_5\}$, Figure (2.3)-b illustrates the unicasting scheme where nodes n_2 and n_4 are the relay nodes even if they are not in the destination set, and Figure(2.3)-c illustrates the broadcasting scheme.

Broadcasting/Multicasting is commonly used in many applications and they are very important to achieve close collaboration of network hosts when the given task is being carried out. Moreover, they are critical for many routing protocols in wireless ad hoc networks because the state information of the nodes and the route information of the network is disseminated by broadcasting/multicasting sessions.

2.3 Energy Efficient Broadcasting/Multicasting

Since the communication devices in wireless ad hoc networks operate unattended and have limited resources, efficient utilization of the resources is crucial for these networks. Among the most important issues related to ad-hoc networks, operating in limited energy environments is crucial because devices are usually equipped with batteries of limited lifetime. Considering that the battery capacity has increased only by a factor of 2.7 in the last eleven years [12], it is essential to develop energy efficient broadcasting/multicating protocols for wireless ad hoc networks.

In [1], energy conservation techniques categorized into two groups; (*i*) power mode control and (*ii*) transmission power control. For our studies, we only consider the transmission power control in which energy consumption is managed by adjusting transmission ranges. Since the energy efficiency is to be considered, we need to clearly define our objective for designing energy efficient networks. Typically, two main objective functions are used for energy efficient network design; (*i*) Minimizing the total transmission power consumption of all nodes involved in the multicast session and (*ii*) Maximizing the operation time until the first node run out of battery [1]. Note that we consider the objective function (*i*) for our studies.

Energy efficiency is the main concern for many wireless ad hoc networks, however for some applications other resources must be utilized efficiently and specific requirements must be satisfied. Due to the emergence of real time applications and the widespread use of wireless devices, many networks are required to provide certain quality of service (QoS) guarantees for customer satisfaction. In [2], QoS is defined as the set of service requirements during the packet stream from the source to destinations, and these requirements must be satisfied by the networks. These service requirements can be end-to-end delay, bandwidth utilization, probability of packet loss, etc.

In order to handle the multi-hop communication fast and economically, better network protocols must be developed. In a wireless environment, finding the optimal protocol for overall communication requirements including the efficient use of all resources and providing the service guarantees can be considered as a huge problem. Joint optimization of all these issues is a challenging problem, because most of the subproblems of the main problem is known as NP-hard. Consequently, most of the studies on the wireless communication problem provide heuristics and approximate solutions for several subproblems of wireless communication.

2.4 Distributed Algorithms

In [13], distributed algorithms are defined as the algorithms which are designed to run on hardware consisting of many interconnected processors. In addition, each processor has limited amount of information and runs independently from the other processors running concurrently.

The term *distributed algorithm* covers a large variety of concurrent algorithms for a wide range of applications including telecommunications, information processing, scientific computing and real-time process control [13]. Especially, wireless ad hoc networks have been a focus of interest in the area of distributed computing, due to its topological properties, which result in interesting algorithmic possibilities [14]. Considering the implementation of a distributed algorithm for wireless ad hoc networks, each node in the network uses only local information for its computations. Note that the local decisions of each node constitute the global result for the entire network.

Centralized algorithms are also applied to wireless ad hoc networks. However, the entire topology information is required for the solutions by centralized algorithms and the collection of the topology information can be very difficult and costly. Therefore, compared to centralized algorithms, distributed algorithms are more appropriate for wireless ad hoc networks since each node is required to solve the global problem in its vicinity.

Since each node makes local decisions and nodes must communicate with each other to inform about their local decisions, distributed algorithms impose messaging overhead for the nodes in the network. In addition, the computational overhead of each node must be considered because each node is supplied by batteries and the more computation is done, the more energy is consumed. Consequently, both the message and computational complexity of the algorithm are the most important design considerations for distributed algorithms.

2.5 Related Works

Each communication device in a wireless environment has a limited energy supply, thus the minimum energy broadcast problem in multi-hop wireless networks has received significant attention over the last few years. However, the NP-completeness of the minimum power broadcasting problem is proved in [15, 4, 16, 17]. Therefore, energy efficient heuristic algorithms which solve the problem in a reasonable time is required. Note that an efficient and effective heuristic algorithm usually produces satisfactory results in polynomial time, but does not always guarantee the global optimality [1]. Guo and Yang classified the solution methods of minimum energy broadcasting problem in [1]. Figure (2.4) clearly illustrates this classification.

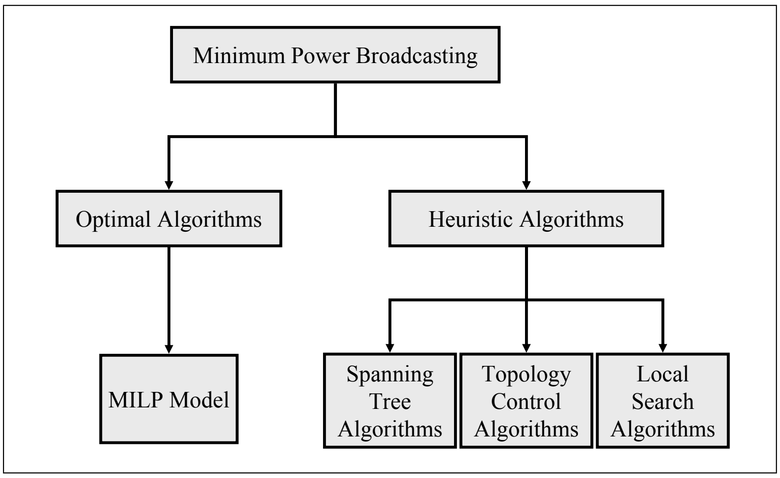


Figure 2.4: Classification for the Solution Methods of Minimum Power Broadcasting Problem

Integer programming formulations can be very useful for obtaining the optimal solutions or best feasible solutions which are important for measuring the strength of developed heuristics. Moreover, these formulations provide insights for designing heuristic approaches. In contrast to the rich literature on solving energy efficient broadcasting problems by heuristic algorithms, approaches based on mathematical programming do not appear extensively in the literature. Das et al. [21] proposed three different mixed-integer linear programming (MILP) formulation for energy efficient broadcasting. Among these formulations, model-C is based on the single-commodity flow formulation, whereas in [22] a model using multi-commodity flow formulation is presented by Yuan. In addition, Altınkemer et al. [23] formulated a model of set covering type, and presented numerical results of a Lagrangian heuristics. Moreover, Guo and Yang [24] introduce a new concept *virtual relay* and formulate another form of MILP model. Recently, Bauer et al. [25] study a number of linear integer models that either use flow model or cuts to characterize feasible solutions of energy efficient broadcasting/multicasting problem. They also proved that the cut-based models and flow-based models are equivalent in strength, and the proposed models are the strongest known models for minimum energy broadcasting/multicasting. Note that the usage of MILP is valuable for theoretical reasons, but the practical usage is usually limited to a small input size.

On the other hand, several heuristic algorithms for minimum power broadcasting problem are available in the literature. As it is illustrated in Figure (2.4), heuristic algorithms are categorized into three groups; (*i*) spanning tree algorithms, (*ii*) topology control algorithms, and (*iii*) local search algorithms [1]. Spanning tree algorithms are usually greedy heuristics and basically construct a spanning tree rooted at source. The algorithms in this category are constructive in nature, thus do not require any initial feasible solution. The second category consists of topology control algorithms which are based on transmission power adjustment. Topology control algorithms assign transmission power to the nodes so that the resulting topology achieves certain connectivity properties while optimizing the energy consumption. Note that the resulting topology of the topology control algorithms is not necessarily in a tree structure. The local search algorithms are the improvement heuristics which start with an initial feasible solution and iteratively improve the objective function while satisfying required constraints.

Over the years, straight greedy approaches minimum spanning tree (MST) and shortest path tree (SPT) are proposed. *Broadcast Least-Unicast-cost* (BLU) and *Broadcast Link-based MST* (BLiMST) algorithms proposed in [29] are based on SPT and MST, respectively. However these two approaches suffer because the standard MST problems reflect the link-based nature of the wired networks and does not capture the node-based nature of the wireless networks, and SPT problems reflect the unicast scheme and does not capture the wireless multicast advantage. The work by Wieselthier et al. [3] is one of the first major contributions in this area. The study in [3] introduces the notion of *wireless multicast advantage* where the node based nature of wireless communication is exploited. Unlike wired networks, wireless networks have logical links which are established if the transmitted signal from the transmitting node is strong enough. The authors propose the Broadcast Incremental Power (BIP) algorithm which is the most notable contribution of [3]. BIP algorithm starts with the source node only and extends the broadcast tree one node at each iteration. Note that the node is selected according to the minimum incremental power rule in which only the additional power consumption of the partial tree is considered. BIP algorithm construct a minimum spanning tree in a similar way that Prim's algorithm [7] does, and BIP algorithm is a centralized algorithm in which the entire topology information of the network is required. Moreover, Wan et al. [18] showed that the BIP algorithm has a constant approximation ratio at least as large as 13/3, however the lower bound is strengthened from 13/3 to 4.6 in [19]. In addition, Klasing et al. [20] showed that the approximation factor is no larger than 12.15. In order to improve the performance of BIP algorithm a simple heuristic named *Sweep* is proposed in [3]. *Sweep* basically detects the redundant transmissions in the resulting broadcast tree and reduce the assigned power of these transmissions.

Using the similar insight with the BIP algorithm, BAIP and GPBE algorithms are proposed in [18] and [26], respectively. However, unlike BIP algorithm, more than one node can be covered at each iteration. In BAIP, the nodes to be covered are chosen according to the *minimum average incremental power* which is defined as the additional transmission power per uncovered nodes to be covered with this transmission power increase. On the other hand, GPBE uses another greedy decision metric *broadcast efficiency* which is defined as the number of newly covered nodes per unit transmission power.

Recently, Ahluwalia and Modiano [4] and Čagalj et al. [27], investigated the distributed calculation of minimum power broadcast trees for wireless networks. Clustering approach for minimum power broadcasting is implemented in [4]. Basically the algorithms consists of two phases: in the first phase the network is divided into clusters where the nodes in each cluster are connected and clusters are formed by considering the minimum energy consumption, then the second phase of the algorithm connects these clusters by calculating the minimum spanning tree of these clusters. In [27], the authors propose the *Embedded Wireless Multicast Advantage* (EWMA) algorithm which is run by exchanging information between 2-hop neighbors. Initially the algorithm runs the distributed MST, then improves the resulting tree. At each iteration, the algorithm checks whether expanding the transmission power of a node to cover the children of one of its children would save some power. If any saving is determined, then existing branches of the tree is replaced by the new one which results in energy savings. Note that the feasibility of the solution is preserved by ensuring that the increase in the transmission power of a node is sufficient to reach all the nodes that were previously covered by the nodes that were excluded. Notice that neither of the solution methods for minimum power broadcasting take into account any of the quality-of-service (QoS) provisions.

Broadcasting is an important communication scheme for implementing real-time applications which usually require the networks to provide QoS guarantees, such as end-to-end delay, delay jitter, bandwidth, etc. [30, 31, 2]. QoS metric considered in this study is end-to-end delay bound for the broadcasting tree. To the best of our knowledge there are only a few studies that consider the end-to-end delay bound for minimum power broadcasting problem. One of the major contributions in this area is presented in [5]. Bulbul et al. [5] rigorously investigate the minimum power broadcasting problem with end-to-end delay constraint. First, authors present an integer programming (IP) formulation of the delay constrained minimum power broadcasting problem. Second, they investigate the optimal solution to a closely related problem in dense networks which they call the Multi-Stage Area Covering Problem (MSACP), then identify several properties of the optimal solution of MSACP. Third, they propose a centralized heuristic which is based on the properties that is obtained by the optimal solution of MSACP. Note that Bulbul et al. consider the end-to-end delay as the number of hops that a message travels from source to the destination nodes.

In [7], the author proposes *Energy-based Link Replacing* (ELR) heuristics for the construction of minimum power delay constrained multicast tree. Since ELR is a centralized algorithm, it requires entire topology information. First, ELR determines a single-hop multicast tree where each node in the network is reached by the source node within a single transmission. In the next iterations of the algorithm, a relay node which decreases the total transmission power most is selected, and a single link is replaced by two links over the selected relay node. Note that the feasibility of the solution is preserved by ensuring that the link replacement does not affect the coverage, and all destinations are still covered after the link replacement. The algorithm proceeds until no such a relay node is obtained without violating the end-to-end delay bound from source to the destination nodes.

The *Distributed Link Substitution* (DLS) algorithm which requires 2-hop neighborhood information is proposed in [6]. DLS can be considered as an improvement algorithm since it starts with an initial feasible solution and improves the objective function while sustaining the feasibility requirements. The main objective for initial phase is to determine a broadcast tree which consists of minimum number of relay nodes, thus the authors adapt the Dominant Pruning Algorithm (DPA) [28]. At each iteration of DLS, new relay nodes with maximum reduction in the total power consumption are selected to be added to the current tree. Note that the distributed implementation of DLS algorithm is similar in fashion to the EWMA algorithm given in [27].

Constructing the broadcast tree is not enough to be implemented directly to the networks because the simultaneous transmissions of nodes cause interference which degrades the received signal power. The performance of an ad hoc network is largely constrained by the interference among these simultaneous transmissions and the metric that is used to determine the link quality between two nodes is signal-to-interference-plus-noise-ratio (SINR). Links with very low SINRs are not typically used due to their extremely poor performance, leading to partial connectivity among all nodes in the network [32]. In general, medium access control (MAC) layer protocols deal with the interference issue, and no routing algorithm considers the interference while constructing the routing tree.

Zhang et al. [32] investigate the routing problem in ad hoc networks considering the SINR values of the links, however they only consider the packet delivery to a destination node and they do not incorporate with multicast or broadcast schemes. The objective of their study is to find the concurrent packets' relay paths associated with the exact relay instants, which can minimize the system-wide energy consumption at all nodes. ElBatt and Ephremides [33], focus on next neighbor transmissions where nodes are required to send information packets to their respective receivers subject to a constraint on the signal-to-interference-and-noise ratio and solve the multiple access problem by two alternating phases, namely scheduling and power control. The objective of their study is twofold: first, to determine the set of users who can attempt transmission simultaneously in a given time slot and second to specify the set of powers needed in order to satisfy SINR constraints at their respective receivers. However, they only consider the interference at a single-hop transmission at each time. In other words, the proposed algorithm is to be executed at the beginning of each time slot in order to cope with excessive interference levels that might be developed in some slots. To the best of our knowledge, there is no study that constructs the broadcast tree and considers the SINR values of the links in multi-hop wireless networks while minimizing the total energy consumption of the transmitting nodes by implementing a single algorithm.

Chapter 3

Delay Constrained Minimum Power Broadcasting

3.1 Problem Statement and System Model

In this chapter, we are interested in the extension of the problem of constructing energy-efficient broadcast trees in ad hoc networks, as given in [3]. The extension considered in this study is the end-to-end delay constraint for each node in the network. End-to-end delay constraint is defined as the maximum delay that a receiving node tolerates, and the delay of a broadcast message is measured on the number of nodes it passes starting from the source to the receiving nodes in the network. Therefore, the problem is called Delay Constrained Minimum Power Broadcasting (DCMPB) problem in [5]. Delay has a strong correlation with the number of hops that a message passes through [39]. Delay in the networks increases as the amount of hopping increases [34]. Therefore, the paths from source to each of the destination node in the network are required to be constructed within a pre-specified number of hops.

A network model similar to the one discussed in [3] is adapted in this study. We consider source-initiated, circuit-switched broadcast sessions. We assume a fixed network topology with $|V|$ nodes, where V represents the set of all nodes in the broadcast application. Each node may receive the broadcast message either directly from the source or over a relay node that is retransmitting the message.

Each node in any particular broadcast tree may transmit with different power levels, and we assume a continuous power level range for these nodes. We use a simplified interference model as in [6]. The simplified interference model assumes that the interference level is independent of network traffic and identical at all nodes. Based on this model, the transmission from node i to node j requires at least P_{ij} amount of transmission power at node i where P_{ij} is proportional to $d(i,j)^{\alpha}$. Note that $d(i,j)$ is the distance between nodes i and j. The path loss coefficient α typically takes on a value between 2 and 4, depending on the characteristics of the communication medium. We further assume that the nodes have omnidirectional antennas which transmit the message equally in all directions. As a result of using omnidirectional antennas and wireless multicast advantage [3], all the nodes within the communication range of a transmitting node receive the transmission. The delay bound is an application specific parameter provided by the user. The delay δ_i experienced by a node i is measured by the number of hops that data travels in order to reach the node i starting from the source. For example, in Figure 3.1, delay of the node s_1, *relay node* and r_1 are $\delta_{s_1} = 0$, $\delta_{\text{relay}} = 1$ and $\delta_{r_1} = 2$, respectively. Note that the depth of the resulting broadcast tree is specified by the overall delay requirement $\delta_i \leq \Delta$, $\forall i \in V$ which is defined by the user.

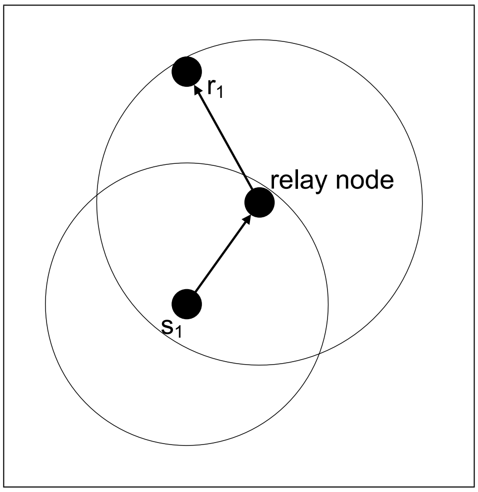


Figure 3.1: Illustrative Example for Delay

In [5], a centralized algorithm is proposed for DCMPB problem. In this algorithm, network topology information is gathered at a node, then this node computes the broadcast tree, and disseminates the resulting broadcast tree to all other nodes in the network. However, such operations at the operating node not only result in considerable time, message complexity, and power consumption, but also require significant resources (power, processor, and memory). These reasons constitute the motivation of this study which considers the localized distributed algorithms that can compute broadcast trees efficiently and effectively. Note that, in a localized algorithm, decision of each node is based on network conditions within some limited distance.

To clearly state the aim of this study, the problem can be defined as determination of a broadcast tree that allows a source node s to send a message to all other nodes within at most Δ hops and consumes the minimum amount of total energy. Thus, we focus on constructing a minimum power broadcast tree with a maximum depth Δ which corresponds to the maximum tolerable end-to-end delay in the network. Moreover, the broadcast tree is desired to be constructed in a distributed manner.

3.2 Mathematical Model

The optimal solution of DCMPB problem can be obtained by the solution of a mathematical model. In this section, we discuss the Integer Programming (IP) formulation of DCMPB problem proposed in [5]. Bulbul et. al. improved the IP formulation that was proposed in [21] for solving the minimum power broadcasting/multicasting problem in the absence of delay bound constraints by removing the redundant variables and constraints in that formulation, and by adding the delay bound constraints.

Following the terminology in [5], let P_{ij} denote the minimum required transmission power at node i to establish the link in range $d(i, j)$. Since fixed network topology is considered, the minimum required transmission power for each node pair (i, j) can be pre-specified in $|V| \times |V|$ matrix P where each component is;

$$P_{ij} = \xi \, d(i,j)^{\alpha}, \quad \forall i, j \in V \,\text{and}\, i \neq j$$

where $d(i, j)$ is the distance between node i and node j, and ξ is a constant which we assumed as 1.

Note that, more than one node can be reached by a single transmission in a wireless network, and in order to construct a broadcast tree with minimum power consumption, we must keep track of all nodes that are covered by a single transmission from node v to node w. Das et. al. [21] proposed $|V| \times |V|$ reward matrix R where the each entry of R is $|V|$-element binary encoding of all the nodes covered (or not covered) by all possible transmissions in the network. $R_{vw}(p)$ represents the p^{th} element of R_{vw}. The reward matrix is computed as follows;

$$R_{vw}(p) = \begin{cases} 1, & P_{vp} \leq P_{vw} \\ 0, & otherwise \end{cases}$$

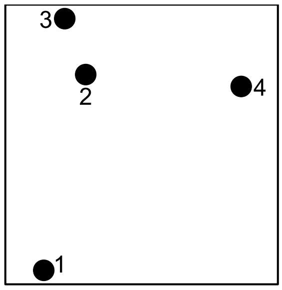


Figure 3.2: Illustrative Example for Reward Matrix

In Figure (3.2) the transmission from 2 to 4 will result in nodes 3 and 4 being covered. This information is encoded in the $(2, 4)$ component of the reward matrix as: $R_{24} = \begin{bmatrix} 0 & 0 & 1 & 1 \end{bmatrix}$. The reward matrix of the wireless network in Figure (3.2) is:

$$R = \begin{bmatrix} \begin{bmatrix} 0 & 0 & 0 & 0 \end{bmatrix} & \begin{bmatrix} 0 & 1 & 0 & 0 \end{bmatrix} & \begin{bmatrix} 0 & 1 & 1 & 0 \end{bmatrix} & \begin{bmatrix} 0 & 1 & 1 & 1 \end{bmatrix} \\ \begin{bmatrix} 1 & 0 & 1 & 1 \end{bmatrix} & \begin{bmatrix} 0 & 0 & 0 & 0 \end{bmatrix} & \begin{bmatrix} 0 & 0 & 1 & 0 \end{bmatrix} & \begin{bmatrix} 0 & 0 & 1 & 1 \end{bmatrix} \\ \begin{bmatrix} 1 & 1 & 0 & 1 \end{bmatrix} & \begin{bmatrix} 0 & 1 & 0 & 0 \end{bmatrix} & \begin{bmatrix} 0 & 0 & 0 & 0 \end{bmatrix} & \begin{bmatrix} 0 & 1 & 0 & 1 \end{bmatrix} \\ \begin{bmatrix} 1 & 1 & 1 & 0 \end{bmatrix} & \begin{bmatrix} 0 & 1 & 0 & 0 \end{bmatrix} & \begin{bmatrix} 0 & 1 & 1 & 0 \end{bmatrix} & \begin{bmatrix} 0 & 0 & 0 & 0 \end{bmatrix} \end{bmatrix}$$

Note that vector R_{vw} is not necessarily equal to R_{wv} because the reward matrix is not necessarily *symmetric*. For the reward matrix (R) of the example in Figure (3.2), it is seen that the transmission $3 \rightarrow 1$ reaches nodes 1, 2 and 4; ($R_{31} = \begin{bmatrix} 1 & 1 & 0 & 1 \end{bmatrix}$, while the transmission $1 \rightarrow 3$ reaches nodes 2 and 3; $R_{13} = \begin{bmatrix} 0 & 1 & 1 & 0 \end{bmatrix}$. Notice that $R_{13} \neq R_{31}$.

Since the precedence relationship is crucial for tree structure, we define the binary variables X_{ijk} which take the value 1, if a transmission in range $d(i, j)$ is assigned to the node i in k^{th} hop; 0, otherwise. As discussed in previous section, Δ_i represents the end-to-end delay bound of node $i \in V \backslash \{source\}$ in terms of number of hops. Note that maximum number of hops (K) for the resulting broadcast tree cannot exceed $|V| - 1$ by the definition of tree structure. Thus, $K = \min\big(\max_{i \in V}(\Delta_i), |V| - 1\big)$. Assuming that the source node is node $1 \in V$, the mathematical formulation of DCMPB problem is given in (3.1)-(3.7).

$$\min \quad Z = \sum_{i=1}^{|V|} \sum_{\substack{j=1 \\ j \neq i}}^{|V|} \sum_{k=1}^{K} P_{ij} X_{ijk} \tag{3.1}$$

$st.$

$$\sum_{j=2}^{|V|} X_{1j1} = 1 \tag{3.2}$$

$$\sum_{i=1}^{|V|} \sum_{\substack{j=1 \\ j \neq i}}^{|V|} X_{ij1} \leq 1 \tag{3.3}$$

$$\sum_{\substack{j=1 \\ j \neq i}}^{|V|} X_{ijk} - \sum_{t=1}^{k-1} \sum_{v=1}^{|V|} \sum_{\substack{w=1 \\ w \neq v}}^{|V|} R_{vw}(i) X_{vwt} \leq 0 \quad , \quad \forall i \in V \backslash \{1\},\ 2 \leq k \leq K \tag{3.4}$$

$$\sum_{t=1}^{k-1} \sum_{v=1}^{|V|} \sum_{\substack{w=1 \\ w \neq v}}^{|V|} R_{vw}(i) X_{vwt} + \sum_{v=1}^{|V|} \sum_{\substack{w=1 \\ w \neq v}}^{|V|} X_{vwk} \geq 1 \quad , \quad \forall i \in V \backslash \{1\},\ 2 \leq k \leq K \tag{3.5}$$

$$\sum_{k=1}^{\Delta_i} \sum_{v=1}^{|V|} \sum_{\substack{w=1 \\ w \neq v}}^{|V|} R_{vw}(i) X_{vwk} \geq 1 \quad , \quad \forall i \in V \backslash \{1\} \tag{3.6}$$

$$X_{ijk} = \begin{cases} 0 \\ 1 \end{cases} \quad , \quad \forall i \in V,\ \forall j \in V \backslash \{1, i\},\ 1 \leq k \leq K \tag{3.7}$$

It is clearly seen that total energy consumption of the broadcast tree is minimized by the objective function (3.1). Since the broadcast session is initiated by the source node only, no other node is allowed to transmit in the first hop by constraints (3.2) and (3.3). Precedence constraint is crucial in tree structure to prevent cycles and connectivity problem. Constraints (3.4) ensure that a node cannot transmit before it receives the broadcast message from any other non-leaf node in the broadcast tree. Suppose that the node $i \in V \backslash \{1\}$ is not covered by any other node in the first $(k-1)$ hops, then the second term in constraint (3.4) becomes zero and enforces the first term to be $\sum_{j=1,\, j \neq i}^{|V|} X_{ijk} = 0$ which implies that node i does not transmit in k^{th} hop. Since we are considering a broadcasting problem, each node $i \in V$ must be covered by at least one node, otherwise new transmission must be assigned to a covered node. Suppose that a node i is not covered in the first $(k-1)$ transmissions, then the first term of the constraint (3.5) becomes zero for such a node i forcing the second term to be positive which represents the total number of transmission in k^{th} hop. Delay bound for each node $i \in V \backslash \{1\}$ is enforced by constraints (3.6) to be $\delta_i \leq \Delta_i$.

Bulbul et.al. investigate the IP formulation of DCMPB problem theoretically in [5], and formulation includes $O(|V|^2)$ constraints and $O(|V|^3)$ variables. Note that Δ_i is $O(|V|)$ since it is bounded by $|V|-1$ by the definition of tree structure. Therefore, IP formulation does not scale well when the number of nodes in the network increases, and heuristics are required for networks of practical size. Thus, for the delay constrained minimum power broadcasting problem, we develop Distributed Tree Expansion heuristic which is constructive in nature. Moreover, the proposed heuristic DTE determines the broadcast tree in distributed manner.

3.3 Proposed Heuristic: DTE

In most of the existing algorithms for minimum power broadcasting, source node plays the role of central decision maker and gathers the entire network topology information. Then, the source node initiates the broadcast sessions. The broadcast tree is constructed by expanding the tree by a single node [3], or a group of nodes [18, 27] in each iteration. Typically, in these algorithms a greedy priority function, such as minimum incremental power or minimum average incremental power, is employed in order to select the next node to be added to the current partial tree. By the definition of broadcasting, each node must receive broadcast message either directly or over relay nodes. Thus the algorithm terminates when all nodes in the network are covered. In this section, we propose a distributed algorithm DTE (Distributed Tree Expansion) that follows this school of thought and keeps expanding the broadcast tree by adding nodes one by one without violating the delay bound constraint. Thus, our contribution is twofold: first, we provide an algorithm that implements the main idea discussed above in a distributed environment for delay constrained minimum power broadcasting in wireless networks while providing provisions for satisfying the delay bound constraint; and second, we propose a priority function which takes into account both the incremental power consumption and the current delay while evaluating a node for extending its transmission range.

The nature of DTE and the distributed algorithm for delay constrained multicasting in wired networks proposed by Kompella et al. [8] are similar to each other. In [8] Prim's algorithm [35] is adapted to the delay constrained case by modifying the link metric. This adaptation opts for the lower delay links to be added to the multicast tree expansion. They also propose an important procedure which is called cycle-make-andbreak procedure in order to reduce the delays along the paths to some specific nodes which violates the delay bound constraint. A distinctive feature of this algorithm is that the source node has the responsibility of selecting the next link to be added to the multicast tree. Thus, the delay information and the cost information is collected by the source node from the nodes that are currently in the tree. Then, the source node conveys its decision on the next link to be added back to the members of the current partial tree.

The proposed distributed algorithm DTE has two main phases for each iteration; (i) Information collection and (ii) Information dissemination. In the first phase, the source node collects the limited information from the current partial tree to be used for decision making on expansion of the tree. In the second phase, the source node decides the next link to be added to the broadcast tree and disseminates this information to the nodes in the current partial tree where the resulting necessary information updates are carried out.

The nature of distributed algorithms require local information storage at each node, and each node must keep track of this information. In DTE, the source node s keeps the following information: the set of transmitting nodes T and the set of nodes S which receive the broadcast message within the delay bound Δ. Initially, all of these sets are considered empty; $T = \emptyset$,$S = \emptyset$, and they are updated over the progress of the algorithm.

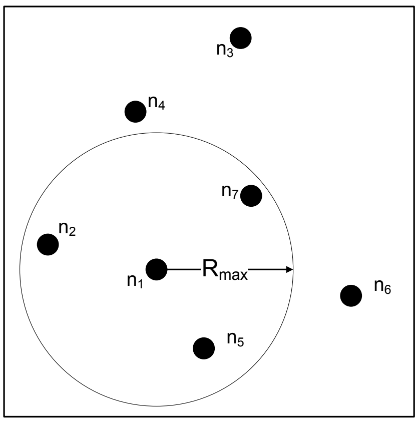


Figure 3.3: 1-hop Neighborhood of Node v; $N_{n_1} = \{n_2, n_5, n_7\}$

We define the 1-hop neighborhood N_i (Figure (3.3)) as the set of nodes within the maximum transmission range of node i, and all nodes, including the source, are only aware of their 1-hop neighborhoods, i.e., they are only informed about the distances of their 1-hop neighbors. 1-hop neighborhood information can be obtained by exchanging *HELLO* messages during the network setup period. Note that any given node i can compute and store the minimum required power to reach each node in its neighborhood N_i during this time period. In addition, each node i stores the information of assigned transmission range R_i, delay δ_i, children C_i, and parent p_i, and keeps track of this information. During the information collection phase, the set of nodes S in the entire network which receive the broadcast message within the delay bound Δ is sent to each node i in the current partial tree from the source. On receiving this information, node i computes the set of nodes S_i and S'_i by using the information of S. Note that S_i and S'_i are local information at node i. While S_i is defined as the set of nodes which are currently uncovered but can be covered within the delay bound by the transmission of node i, S'_i is defined as the set of nodes which are within the maximum transmission range of node i but cannot be reached from node i within the delay bound.

Source node initiates the information collection phase of each iteration by sending a *FIND* message to all nodes i in the current partial tree. The information of the set S and the delay bound Δ is carried by the *FIND* message. Each node receiving the *FIND* message uses the information of S and Δ for its local calculations. On receiving the *FIND* message, node i computes a priority ϕ_{ij} for each uncovered node $j \in N_i$ that falls outside its current assigned transmission range[1], if $\delta_i < \Delta$. Note that each node $j \in N_i$ but currently uncovered corresponds to a specific transmission range extension for node i. Node i identifies the node b_i^* according to its priority as follows;

$$b_i^* = \arg\min_{j \in N_i} \phi_{ij}$$

Then, node i incorporates the nodes that are currently uncovered but may be reached by a transmission from i to b_i^* into S_i. If all nodes in N_i are covered in previous iterations, then b_i^* and S_i are set to null and empty set, respectively. Note that S'_i is empty when $\delta_i < \Delta$. Otherwise, if $\delta_i = \Delta$ and if some nodes in N_i are currently not covered, then these nodes are added to S'_i. In this case, b_i^* is null and $S_i = \emptyset$. After performing these operations, each node i sends *CANDIDATE* message back to the source node. *CANDIDATE* message carries the information of S_i, S'_i, b_i^*, and $\phi_{ib_i^*}$. The source collects the *CANDIDATE* messages until it receives one from each node in the current partial tree. This completes the information collection phase

[1]The exact form of the priority function will be discussed later in this section.

of DTE. The information collection phase is summarized by the flow chart in Figure (3.4).

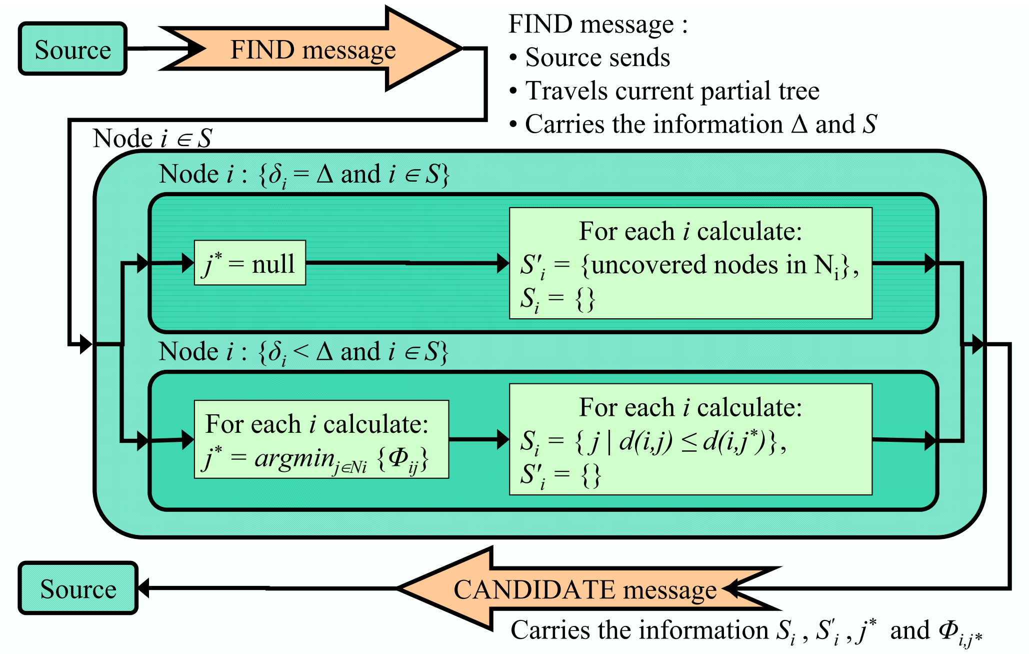


Figure 3.4: Information Collection Phase of DTE

At the end of the information collection phase the algorithm may face three different outcomes; (i) both $\cup_{i\in S}S_i = \emptyset$ and $\cup_{i\in S}S'_i = \emptyset$, (ii) $\cup_{i\in S}S_i \neq \emptyset$ and $\cup_{i\in S}S'_i = \emptyset$, and (iii) $\cup_{i\in S}S_i = \emptyset$ and $\cup_{i\in S}S'_i \neq \emptyset$. If the source faces the case (i), it sends no more *FIND* message because each node in the network is covered within delay bound, and the algorithm terminates. Second case (ii) occurs when there is at least one uncovered node which can be covered within the delay bound. In this case at least one non-empty *CANDIDATE* message is received by the source node, then the source chooses node i^* to extend its transmission range so that $\phi_{i^*b^*_i} = \min_{i\in S}\phi_{ib^*_i}$, adds i^* to T if necessary and updates S as $S = S \cup S_{i^*}$. When there is no uncovered node that can be covered within delay bound but there is at least one uncovered node which cannot be covered within delay bound by the current partial tree, source node faces case (iii). In this case, source node invokes the restore-delay (RD) procedure in order to recover from infeasibility.

If the outcome of the information collection phase is as described in case (ii), then the information dissemination phase is initiated by the source. Note that in this case the source identifies a node i^* to extend its transmission range. Source node sends an *EXTEND_RANGE* message which propagates through the current partial tree. All nodes other than node i^* ignore *EXTEND_RANGE* message. Upon receiving the *EX-*

TEND_RANGE message, node i^* adjusts its transmission range to $R_{i^*} = d(i^*, b_i^*)$. Notice that the transmission range of the node i^* is increased and this increase may cause unnecessary transmissions, i.e. child nodes of some of existing children of node i may fall into the transmission range of node i, thus the transmission of parent of such nodes becomes redundant. In order to detect this event and remove the redundant transmissions in the broadcast tree, node i^* sends an *ELIMINATE* message to its existing child nodes C_{i^*}. *ELIMINATE* message is sent for notifying the child nodes of node i about its new range R_{i^*}. Each node $j \in C_{i^*} \cap T$ checks the new range of node i^* whether the transmission of node j becomes redundant or not. If node $j \in C_{i^*} \cap T$ determines that the new range of its parent i^* covers all of its child nodes C_j, then it sends a *REMOVE_PARENT* message to all nodes $k \in C_j$, and subsequently conveys a *REMOVE_XMIT* message including the set of its child nodes C_j back to its parent i^* before setting $R_j = 0$ and $C_j = \emptyset$. *REMOVE_XMIT* message does not necessarily carry the information of C_j, an empty *REMOVE_XMIT* message is sent from j to i^* if the transmission of node j cannot be removed. At each node $k \in C_j$, on receiving a *REMOVE_PARENT* message, each node $k \in C_j$ sets $\delta_k = \infty$ and p_k as null. Node i^* waits for receiving a *REMOVE_XMIT* message from all of its child nodes $j \in C_j \cap T$. Upon receiving all required *REMOVE_XMIT* messages node i^* expands its set of children C_i by S_i and the set of nodes C_j delivered in *REMOVE_XMIT* messages. Subsequently, i^* sends an *ADD_PARENT* message at its assigned transmission power to all its current child nodes. Each node $j \in C_{i^*}$, on receiving *ADD_PARENT* message, updates the delay and parent information; $\delta_j = \delta_i + 1$ and $p_j = i^*$, respectively. As the final step of the information dissemination phase of DTE, redundant transmissions are sent back to the source by node i^*, and the source node updates the set of transmitting nodes T. The information dissemination phase is summarized by the flow chart in Figure (3.5).

The elimination process is illustrated in Figure 3.6. Suppose that node n_1 is selected to increase its transmission range from $d(n_1, n_2)$ to $d(n_1, n_5)$. It is easily seen that the transmission $n_2 \rightarrow n_3$ becomes redundant with the new transmission range of node n_1. In the beginning of the elimination process, node n_1 notifies its child node n_2 that it is extending its transmission range to node $d(n_1, n_5)$. Node n_2 checks whether its transmission become redundant or not after the range update. Node n_2 identifies that new transmission range of its parent n_1 covers its all child nodes which is only node n_3. Thus, node n_2 sends a *REMOVE_PARENT* message to node n_3 and a *REMOVE_XMIT* message to n_1. Then, nodes n_3 and n_5 receive an *ADD_PARENT*

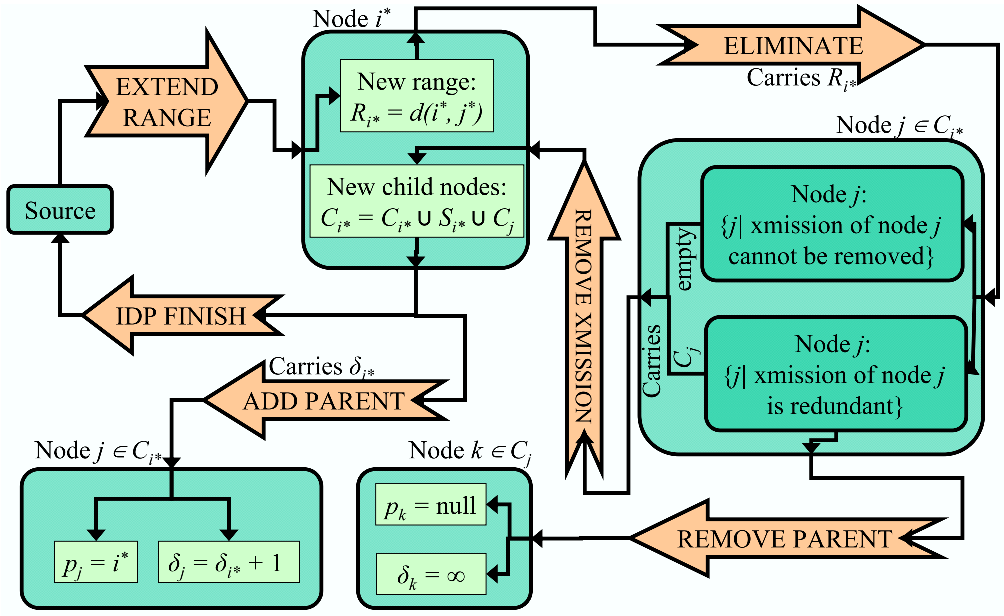
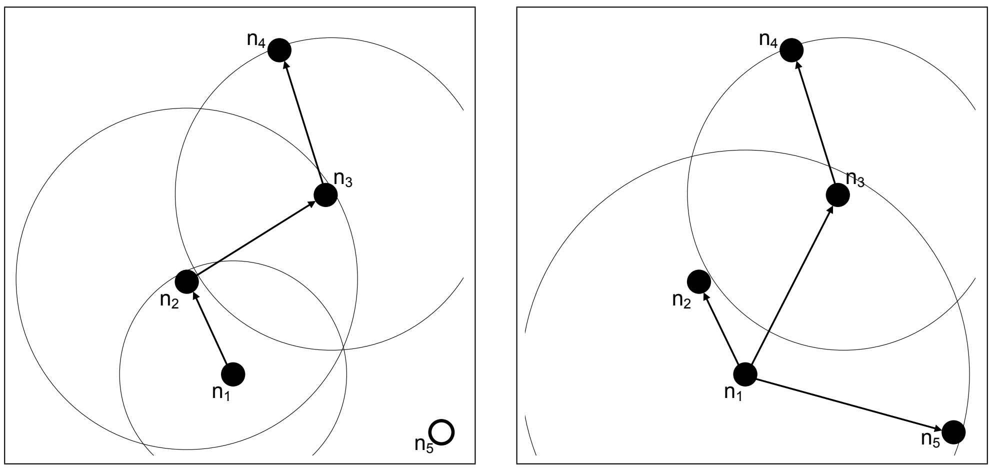


Figure 3.5: Information Dissemination Phase of DTE

message from n_1, and the source is notified by node n_1 that n_2 is no longer a relay node. By using elimination process, redundant transmission of node n_2 is prevented and energy efficiency is provided.



(a) before eliminate process (b) after eliminate process

Figure 3.6: Illustrative Example for Eliminate Process

At the end of the information collection phase, if the source node faces the case (iii), then it decides that the Restore Delay (RD) procedure must be invoked. Notice that in this case there exist at least one uncovered node in the network and it cannot be covered within delay bound by using the current partial tree. As described above, these nodes are detected by nodes $i \in S$ with a delay of $\delta_i = \Delta$ and relayed back to the source in sets S'_i via *CANDIDATE* messages. Defining $H = \{i | i \in S, S'_i \neq \emptyset\}$ and $S' = \cup_{i \in H} S'_i$, the idea of the RD procedure can be explained as follows; if the length of the path from the source to at least one of the nodes $i \in H$ is reduced by one hop, then the broadcast message can be delivered to at least one of the nodes in S' within the delay bound. Thus, RD procedure starts with checking the delay information. The source node sends a *CHECK_DELAY* message carrying the set H to the nodes covered so far. All nodes in $i \in S \setminus H$ respond to this query. If a node $i \in S \setminus H$ has a delay of $\delta_i \leq \Delta - 2$, then it computes $\theta_{ib_i^*} = \min_{j \in H}(P_{ij} - P_i)$ where P_{ij} is the minimum transmission power required to reach node j from node i, P_i is the power assignment corresponding to the current range R_i of node i, and $b_i^* \in H$ is the node that can be covered from node i at minimum incremental transmission power. If node i cannot cover any node $j \in H$ in less than Δ hops, then it sets b_i^* as null and $\theta_{ib_i^*} = \infty$. Subsequently, each node $i \in S \setminus H$ sends the information of b_i^* and $\theta_{ib_i^*}$ back to the source via *REDUCE_DELAY* message. The information collection part of the Restore Delay (RD) procedure is summarized by the flow chart in Figure (3.7).

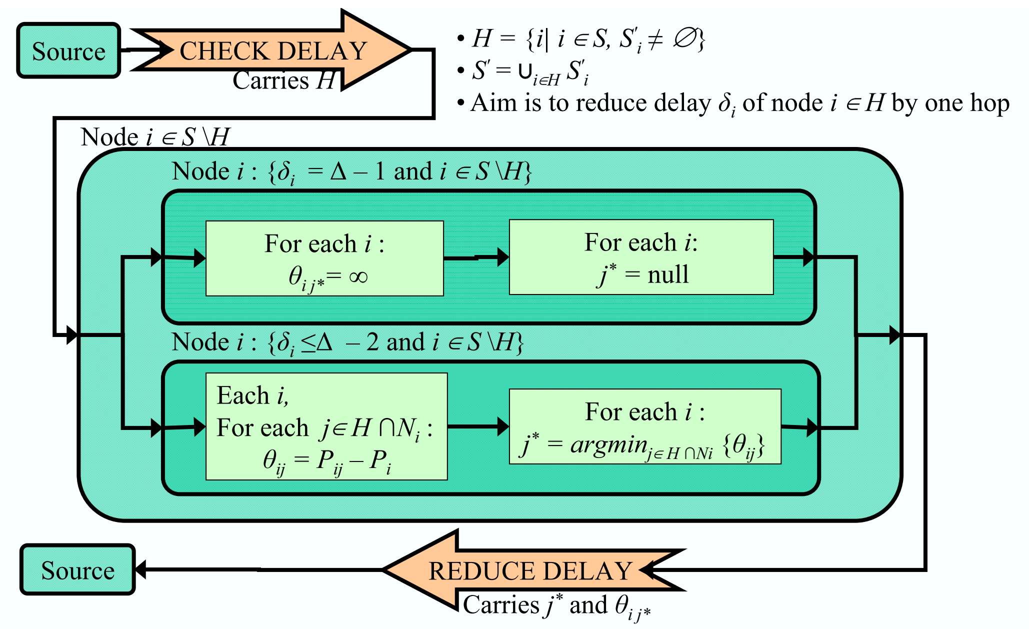
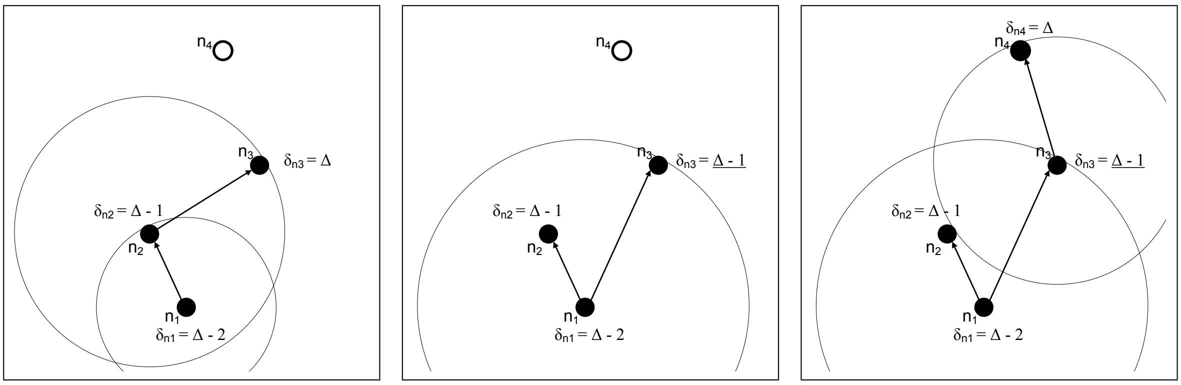


Figure 3.7: Restore-Delay Procedure

On receiving all the *REDUCE_DELAY* messages, the source node determines $\theta_{i^*b_i^*} = \min_{i \in S \setminus H} \theta_{ib_i^*}$. After compiling the *REDUCE_DELAY* messages the source node may face two cases; (i) $\theta_{i^*b_i^*} = \infty$, and (ii) $\theta_{i^*b_i^*} \neq \infty$. If $\theta_{i^*b_i^*}$ is ∞, then the source concludes that feasibility cannot be restored. Then, the source terminates the algorithm DTE. For the cases where $\theta_{i^*b_i^*} \neq \infty$, the source node sends an *EXTEND_RANGE* message destined to node i^*. Then, the algorithm proceeds as described in the information dissemination phase where $S_{i^*} = \{b_i^*\}$, and b_i^* removes itself from the set of children of its current parent.

Although the use of RD procedure is advantageous, for some cases it is insufficient to satisfy feasibility. We note that for a very small portion (1.11%) of our test instances the algorithm could not find a feasible solution if there is at least one feasible solution for these instances. The mechanics of RD is demonstrated in Figure (3.8). Suppose that at the end of the information collection phase node there exists an uncovered node which cannot be covered within the delay bound. During the information collection phase, node n_3 with a delay of $\delta_{n_3} = \Delta$ detects an uncovered node n_4 in its range and sets $S'_{n_3} = \{n_4\}$. This information is relayed back to the source in a *CANDIDATE* message. Assuming that all other nodes receive the broadcast message within Δ hops, the source node initiates the RD procedure at the beginning of the next iteration so that $H = \{n_3\}$ and $S' = \{n_4\}$. Currently, the parent of node n_3 is node n_2, and both nodes n_2 and n_3 are in the maximum transmission range of node n_1 with a delay of $\delta_{n_1} = \Delta - 2$. Source node initiates the RD procedure by sending *CHECK_DELAY* message to the currently covered nodes. Subsequently, node n_1 responds to the *CHECK_DELAY* message with $b^*_{n_1} = n_3$, and the source node prescribes to node n_1 to extend its transmission range to cover node n_3, reducing the delay of node n_3 to $\delta_{n_3} = \Delta - 1$. At the next iteration of DTE, the broadcast message is delivered to node n_4 by node n_3, and feasibility is restored.



(a) before RD (b) after RD (c) next iteration

Figure 3.8: Illustrative Example for Restore Delay Procedure

For marking the start and finish of the various steps of the DTE algorithm, it requires messaging between the nodes because of its distributed nature. It is important to analyze the message complexity of the algorithm to identify the messaging overhead. In order to obtain the message complexity, we start our analysis from one iteration of DTE. At one iteration, both *FIND* and *CANDIDATE* messages require $O(n^2)$ transmissions in the information collection phase. In the information collection phase, *REMOVE_PARENT* messages dominates the message complexity and require

$O(n^2)$ transmissions. Note that *REDUCE_DELAY* messages also requires $O(n^2)$ transmissions for Restore Delay procedure if RD is invoked. Thus, one iteration of DTE results in $O(n^2)$ messages exchanged. Since the algorithm DTE covers at least one node in each iteration, total number of iterations is $O(n)$. As the result of our analysis, DTE algorithm requires $O(n^3)$ messages exchanged.

Finally, we discuss the priority function ϕ employed in the information collection phase. Note that the priority function ϕ is used for determining the node that will extend its transmission range. Since we are considering the DCMPB problem, the delay concept must be embedded to this priority function along with the energy consumption. For this purposes, we modify the minimum incremental power rule of BIP [3]. Note that the partial tree is expanded by one node which requires minimum additional power to be covered at each iteration in BIP algorithm. When the range of a transmitting node i extends its transmission range from $d(i,j)$ to $d(i,k)$, then the incremental power consumption at node i to cover the node k is proportional to $d(i,k)^\alpha - d(i,j)^\alpha$. However, when the delay concept is considered for energy efficient broadcasting, using only incremental power consumption for the decisions of tree expansion is not sufficient to construct good broadcast trees, especially when the delay bound is tight. In [5], it is observed that the nodes with lower delays transmit at substantially higher power levels than those with delays close to Δ. This results shows that the nodes with lower delays are preferable to transmit in order to achieve feasibility for delay bound constraint and obtain good solutions in terms of energy efficiency. These motivations lead us to define a new priority function ϕ_{ij} as follows:

$$\phi_{ij} = \frac{P'_{ij}}{\nu^{\alpha/2}} \times \frac{1}{\log(\Delta - \delta_i + 1)}, \tag{3.8}$$

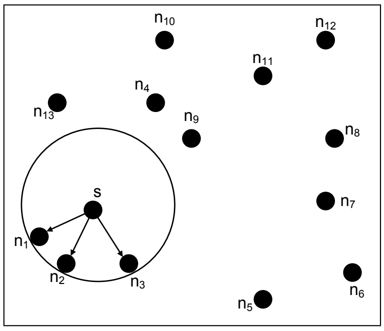
where P'_{ij} is the incremental power spent for transmission from node i to node j, δ_i is delay of node i, Δ is the maximum number of hops allowed, and ν is the number of currently uncovered nodes that are covered by the transmission from i to j. Note that the lower values of ϕ_{ij} are preferred. The incremental power used per newly covered node is represented by the first term in the equation (3.8). For a node with a current range of zero, this term is proportional to the density of nodes currently uncovered and favors transmissions into densely populated regions of such nodes. Moreover, the second term in (3.8) corresponds to the characteristics which is observed in [5] by ensuring that the area covered by the first transmission from the source is sufficiently large. We use $\log(\Delta-\delta_i+1)$ instead of $(\Delta-\delta_i)$ as it demonstrated a better performance in our preliminary experiments.

To sum up, the source node and all other nodes in the network have only limited local information about the network topology. In fact, the source node is aware of the identities of the forwarding nodes in the broadcast tree, but not of their transmission powers or the nodes covered by those transmissions. Nevertheless, in DTE, the source node is the sole node responsible for adding a new forwarding node to the tree, and thus, a significant number of messages are exchanged between the source node and the nodes in the network.

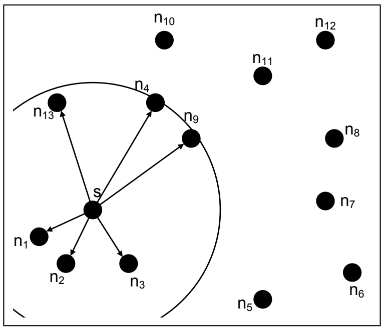
3.3.1 Example

In order to make sure that the DTE algorithm is clearly understood, we implement the DTE on an illustrative example. For the sample network illustrated in Figure (3.9), suppose that we have 13 nodes to be covered within a delay bound of $\Delta = 5$ and source node s initiates the broadcast session. Further suppose that the path loss coefficient α is 2, and each node in the network has limited transmission range. The algorithm DTE obtains the broadcast tree by the iterations represented in Figure (3.9).

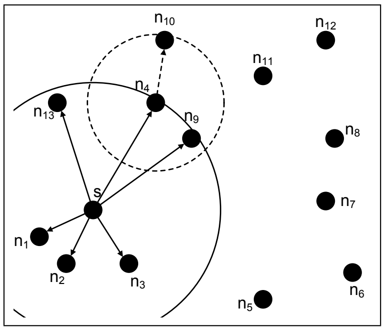
As it is seen in Figure (3.9)-a, the nodes n_1, n_2, and n_3 are covered by the source node s. At the end of the iteration 1, the sets S and T which are initially empty are updated as $S = \{s, n_1, n_2, n_3\}$ and $T = \{s\}$. Each node in S receives a *FIND* message sent by the node s in the beginning of the information collection phase of the second iteration. Note that there is no uncovered node within the neighborhood of nodes n_1, n_2 and n_3, thus the *CANDIDATE* message sent by these nodes to the source carries the empty sets S_i and S_i'. Then, node s decides to increase its transmission power to reach nodes n_4, n_9 and n_{13} which are equally distant to node s. At the end of the second iteration the sets S and T becomes $S = \{s, n_1, n_2, n_3, n_4, n_9, n_{13}\}$ and $T = \{s\}$, respectively. In the third iteration, only nodes n_4, n_9, and n_{13} have uncovered nodes in their neighborhoods. On receiving *FIND* message from the source, nodes n_4, n_9, and n_{13} calculates a priority function for each candidate transmission range. The priority function values for the transmissions of $n_4 \rightarrow n_{10}$ and $n_4 \rightarrow n_{11}$ are $\phi_{n_4 n_{10}} = 2.49$ and $\phi_{n_4 n_{11}} = 3.11$, respectively. The priority function values for the transmissions $n_9 \rightarrow n_8$, $n_9 \rightarrow n_{10}$ and $n_9 \rightarrow n_{11}$ are calculated as $\phi_{n_9 n_8} = 3.31$, $\phi_{n_9 n_{10}} = 3.11$ and $\phi_{n_9 n_{11}} = 4.97$, respectively. For node n_{13}, the priority function evaluates to $\phi_{n_{13} n_{10}} = 13.08$ for a transmission $n_{13} \rightarrow n_{10}$. Since the minimum priority



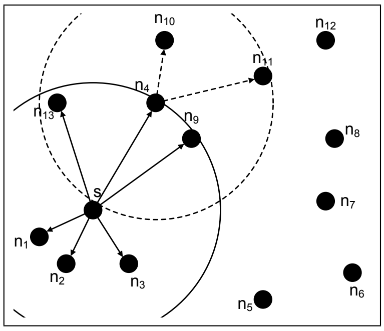
(a) iteration 1



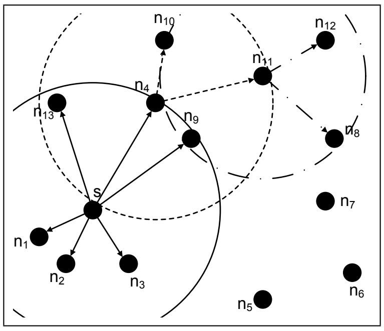
(b) iteration 2



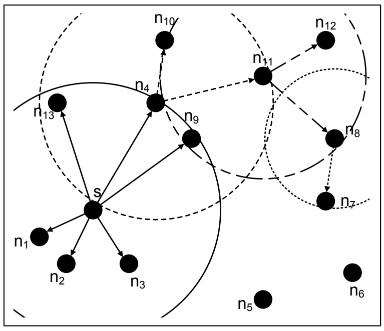
(c) iteration 3



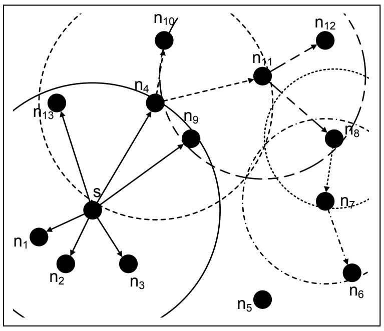
(d) iteration 4



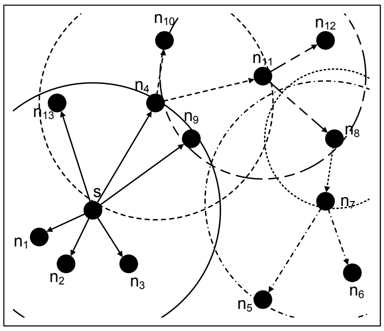
(e) iteration 5



(f) iteration 6



(g) iteration 7



(h) iteration 8

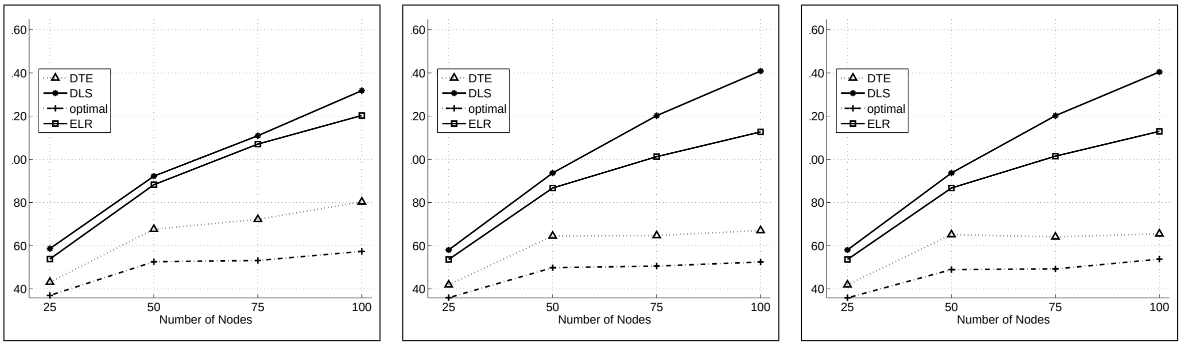
Figure 3.9: Illustrative Example for DTE

function is calculated for transmission $n_4 \rightarrow n_{10}$, node n_4 is selected to transmit to node n_{10} and added to the relay node set T. At the end of the 3^{rd} iteration the sets S and T becomes $S = \{s, n_1, n_2, n_3, n_4, n_9, n_{13}, n_{10}\}$ and $T = \{s, n_4\}$, respectively. In the 4^{th} iteration, candidate transmissions that are evaluated with a priority function are $n_4 \rightarrow n_{11}$, $n_9 \rightarrow n_{11}$, $n_{10} \rightarrow n_{11}$ and $n_9 \rightarrow n_8$. The associated priority functions are $\phi_{n_4 n_{11}} = 3.73$, $\phi_{n_9 n_{11}} = 4.97$, $\phi_{n_{10} n_{11}} = 7.21$ and $\phi_{n_9 n_8} = 4.97$. Since the candidate transmission $n_4 \rightarrow n_{11}$ evaluates the minimum priority function, node n_4 is selected to extend its transmission range from $d(n_4, n_{10})$ to $d(n_4, n_{11})$. Then the sets S and T are updated as $S = \{s, n_1, n_2, n_3, n_4, n_9, n_{13}, n_{10}, n_{11}\}$ and $T = \{s, n_4\}$, respectively. DTE terminates after 8 iterations with a maximum delay of 5 hops. The resulting broadcast tree of the sample example consist of the nodes in T where $T = \{s, n_4, n_7, n_8, n_{11}\}$.

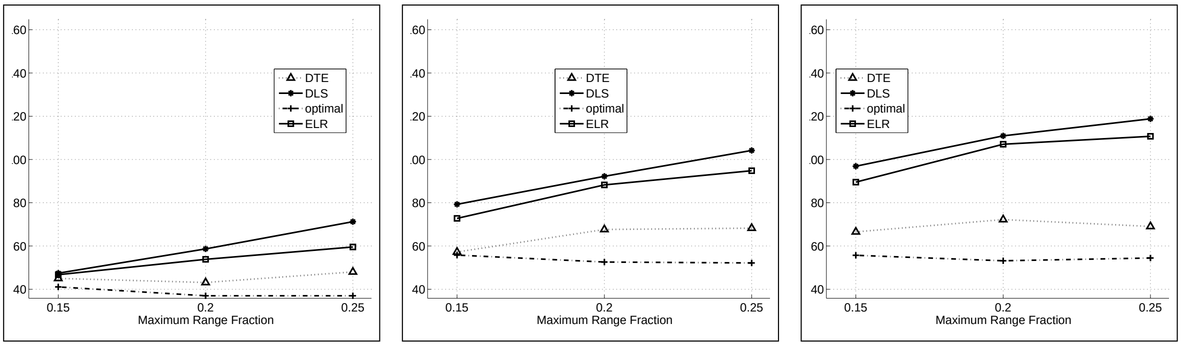
3.4 Numerical Results

In this section, we investigate the performance of DTE on randomly generated instances. The performance of DTE algorithm is analyzed by comparing the results of DTE algorithm with the results of *Energy-based Link Replacing* (ELR) algorithm [7] and Distributed Link Substitution (DLS) algorithm [6]. We also provide the optimal or the best feasible solutions calculated by solving two different integer programming formulations [5, 25] for a maximum CPU time of 2700 seconds each, using ILOG OPL Studio 5.2 running on ILOG CPLEX 10.2. For each network consisting of 25, 50, 75 and 100 nodes, 10 random network topologies are generated in a 10 by 10 unit square area and spatial Poisson distribution is used for determining the node locations. Then the performance of DTE is analyzed for changing number of nodes in the network (N), delay bound (Δ) and maximum range fraction (R_{max}). Note that the maximum transmission range is represented by maximum range fraction (R_{max}) which is the fraction of the length of the diagonal of the square area. The results presented in Figure (3.10) are the averages of the 10 randomly generated network topologies for each parameter settings.

We first fixed the R_{max} to one fifth of the length of the diagonal of the network area, then analyzed the results for changing end-to-end delay bound Δ. The variation of the total transmission power of the resulting broadcast trees with respect to the node density in the network is depicted in Figures (3.10)-a, (3.10)-b, and (3.10)-c. As it is seen in the figures, for all node density settings DTE outperforms DLS and ELR for changing delay bound values, i.e. $\Delta = 6$, $\Delta = 8$ and $\Delta = 10$. In addition, the total transmission power of the broadcast tree generated by DTE is close to the best feasible solution calculated by integer programming formulations in [5], [25]. The main reason of better performance of DTE can be explained by the selection rule of the uncovered nodes to be added to the broadcast tree which makes DTE less greedy. Notice that the priority function that we use for node selection is based on the insights observed in [5]. Since the priority function considers the delay information, it provides better transmission power assignment for the next expansion of the tree. In DTE, nodes with lower delays cover larger areas, thus more nodes in early steps, then the later steps require fewer relay nodes which also reduce the total transmission power. On the other hand, ELR and DLS do not apply such a rule, the relay nodes are added to the broadcast tree without checking whether it is close to the source node or not. Thus, the nodes close to the leaf nodes may have larger transmission power assignment which also increase the total power consumption. Another powerful aspect of DTE against DLS and ELR is that DTE less prone to the increase of the number of nodes in the network than DLS and ELR.



(a) $\Delta = 6,\ R_{max} = 0.2$ (b) $\Delta = 8,\ R_{max} = 0.2$ (c) $\Delta = 10,\ R_{max} = 0.2$



(d) $\Delta = 6,\ N = 25$ (e) $\Delta = 6,\ N = 50$ (f) $\Delta = 6,\ N = 75$

Figure 3.10: Total Transmission Power of Broadcast Trees Generated by Algorithms DTE, DLS, ELR and the Best Feasible Solution of IPs.

Upon observing that the total power consumption of the resulting broadcast trees less rely on the change of end-to-end delay bound, we fixed the Δ to 6, and analyzed the results for changing number of nodes in the network, i.e. $N = 25$, $N = 50$ and $N = 75$. The variation of the total transmission power of the resulting broadcast trees with respect to the maximum range fraction R_{max} is presented in Figures (3.10)-d, (3.10)-e, and (3.10)-f. For these parameter settings, we observed that DTE performs better than DLS and ELR and the results of DTE is close to the best feasible solutions. As it is seen in the figures, the total transmission power consumption of the broadcast trees generated by IP tends to decrease with the increase in R_{max}. The same result can be slightly observed for DTE but cannot be observed for DLS and ELR. This situation can be explained by the relaxation of constraint for the maximum transmission power level at each node. Since larger values of R_{max} extend the feasible region, the performance of the solutions expected to be at least as it is observed for smaller values of R_{max}. Although DTE slightly reflect this observation, DLS and ELR have very poor performance. ELR and DLS algorithms are usually unable to add many new forwarding nodes to the initially constructed tree, where most of the transmissions are close to the maximum transmission range. Therefore, they terminate with high cost feasible solutions.

Chapter 4

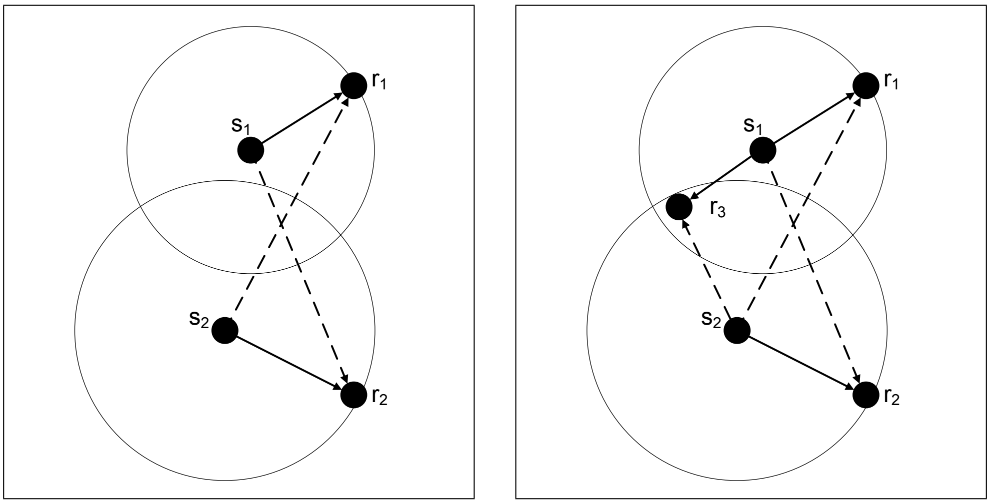
SINR Constrained Minimum Power Broadcasting

4.1 Problem Statement and Model

In the energy-efficient wireless broadcasting literature, problems are generally decomposed into two phases; (*i*) routing phase, and (*ii*) scheduling phase. Although the routing phase of the problem is considered in the network layer, the scheduling phase is considered in the data link layer. The network layer is responsible for end to end (source to destination) packet delivery, whereas the data link layer is responsible for node to node (hop to hop) frame delivery.

In the routing phase of the energy efficient broadcasting problem, the minimum power broadcast tree is constructed with respect to the desired problem specific constraints. However, this tree cannot be directly implemented to the network because broadcast in a multi-hop wireless network has to deal with interference [36]. The message cannot be decoded successfully by the receiver, if the receiver faces strong interference.

Interference occurs at the receiving node when two or more signals are received from different transmitters, in the same time period. Assuming that s_1 and s_2 transmit a message at the same time, Figure 4.1 shows how the transmitting nodes cause interference. In Figure 4.1-a, node s_1 and s_2 send message to r_1 and r_2, respectively. Whether node r_2 (r_1) is in the transmission range of node s_1 (s_2) or not, the transmission by node s_1 (s_2) causes interference at node r_2 (r_1) and the radio signal from node s_1 (s_2) will interfere node r_2 (r_1) from receiving message from node s_2 (s_1). Figure 4.1-b is another example showing that r_3 is in the transmission range of both s_1 and s_2. Notice that straight lines represent the message flow from a sender to receiver and dashed lines represent the interference effect of other transmitting nodes.



(a) (b)

Figure 4.1: Interference

To deal with the interference problems, several broadcast scheduling approaches were developed. Basically, the broadcast session is divided into time slots and a broadcast tree is divided into levels. For each level, transmitting nodes are assigned to a time slot so that the interference is prevented.

There are many studies on routing [3, 18, 29, 27] or scheduling [36, 37, 39] independently, but as far as we know, there is no study that constructs the broadcast tree while considering the interference at the receiving nodes. In other words, there is no single algorithm that produces a broadcast tree that can be directly applied to the network without a scheduling phase.

In this chapter, we are interested in another extension of the energy-efficient broadcasting problem [3] in ad hoc networks when the nodes are covered according to the strength of received signal power. The measure that we use for deciding the sufficiency of received signal strength is called signal-to-interference-plus-noise-ratio, thus we call this problem SINR Constrained Minimum Power Broadcasting (SCMPB) problem.

The system model that we consider for this problem is a source initiated, circuit switched broadcast sessions. It is assumed that we have a wireless ad hoc network which has a fixed topology with $|V|$ nodes where V is the set of nodes. We have a fixed source node which is a priori and it transmits a message to all nodes in the network directly or over another node in the network.

Network connectivity is achieved by the logical links which are established by the required level of transmission power at the transmitting nodes. Each node $v \in V$ can transmit by a power level $P(v)$ which is not restricted with a maximum value. The power level for each node is not necessarily identical and it varies according to (i) the distance between the transmitting node and the receiving node, (ii) environmental noise around the receiving node and (iii) the interference, caused by other nodes, at the receiving node. If a message is clearly decoded at receiving node w, it is said that a link between two nodes (v : sender, w : receiver) separated by distance $d_{v,w}$ is successfully established. The transmitted message is clearly understood at the receiving node w, if the SINR value at the receiving node is above a threshold value γ. SINR value represents the dominance of reception power to the environmental noise and interference at the receiving node.

$$SINR(w) = \frac{P_{rec}(v,w)}{(C_w + I_{vw})} = \frac{P(v)\, d_{vw}^{-\alpha}}{(C_w + I_{vw})}$$

According to the path loss radio propagation model, the radio signal degrades very quickly with respect to the distance that the signal travels [36]. At node w, the received signal power of the transmission of node v, $P_{rec}(v,w)$, is equal to $P(v)\, d_{vw}^{-\alpha}$, where $P(v)$ is the transmission power level of node v, d_{vw} is the distance between nodes v and w, and α is the path loss exponent. The value of α changes according to environmental conditions and it is typically between 2 and 4 [5]. It is assumed that the environmental noise around the receiving node w is constant and represented by C_w. Interference I_{vw} is defined as the total of received signal power at node w from the nodes that transmit in the same time slot with the parent node v. Denoting the nodes which transmits in the same time slot with node v as $L(v)$[1], the interference at node w is formulated as follows;

$$I_{vw} = \sum_{i \in L(v)} P(i)\, d_{iw}^{-\alpha}$$

To clearly state the problem and purpose of the study, we are looking for a heuristic approach that can obtain an optimal/near optimal broadcasting tree which (i) requires minimum power consumption, (ii) considers interference and environmental noise. It is important that the transmitting nodes do not require to be scheduled for transmission,

[1]Notice that $v \cap L(v) = \emptyset$

because the solution does not only produce broadcast tree but also assigns transmission time slots to the non-leaf nodes in the broadcast tree.

4.2 Mathematical Model

One of our contributions in this study is proposing a Mixed Integer Linear Programming (MILP) formulation for the SINR Constrained Minimum Power Broadcasting (SCMPB) problem. The proposed formulation is adapted from the IP formulation in [5] for the Delay Constrained Minimum Power Broadcasting (DCMPB) problem. Note that the formulation presented in this section considers only the broadcasting case; however, this broadcasting case can be generalized to a multicasting problem with minor changes. Moreover, our MILP formulation can be extended to consider the delay bound on the data delivery from source to destination. But these two extensions are out of the scope of this study.

Note that the required transmission power of a transmitting node $v \in V$ to send a message to a receiving node $w \in V$ depends on (i) the distance between transmitting node v and receiving node w, (ii) the interference, caused by other nodes, at the receiving node w, and (iii) the environmental noise around the receiving node w. Thus we cannot pre-define the transmission power levels required to send a message from one node to another. Since we have to consider the interference, two interfering transmitters must be prevented to transmit at the same time or their transmission powers must be rearranged so that the receiving nodes receives the message successfully. We assume that a broadcast session is divided into time slots and each time slot is used for a single transmission or simultaneous transmissions. Thus, the power level of transmitting node $i \in V$ must be defined as P_{ik} where k represents the transmission time slot of node i.

Let Y_{ijk} be a binary variable that assumes the value 1 if node i transmits to node j in the k^{th} time slot in the broadcast tree, and zero otherwise. These binary variables are necessary to ensure the coverage of all the nodes in the network.

Let X_{ik} be a binary variable which represents the existence of a transmission of node i in the time slot k. X_{ik} takes the value 1, if node i transmits in the k^{th} time slot; 0, otherwise.

Let $L(i)$ represents the set of nodes which transmit with node i in the same time slot and $i \cap L(i) = \emptyset$. We define an auxiliary variable I_{ijk} for the total interference which is caused by node $v \in L(i)$, at receiving node j, in the k^{th} time slot. Figure

4.2 illustrates the use of the auxiliary variable I_{ijk}. Assume that nodes s_1 and s_2 send messages to nodes r_1 and r_2, respectively. Furthermore, assume that the transmissions are in the same time slot. Then the interference at node r_1 (r_2) is $I_{s_1 r_1 k} = P(s_2)\, d_{s_2 r_1}^{-\alpha}$ ($I_{s_2 r_2 k} = P(s_1)\, d_{s_1 r_2}^{-\alpha}$).

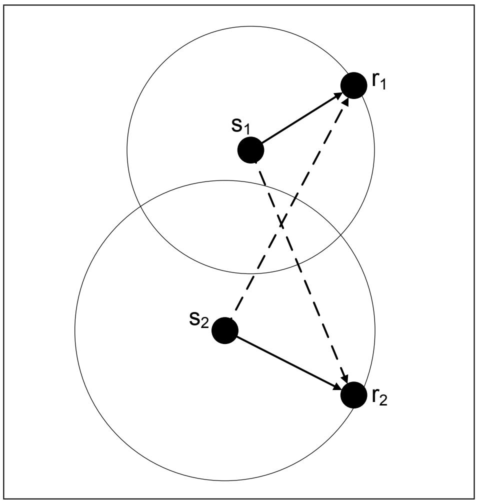


Figure 4.2: Interference at node r_1 and r_2 in time slot k

Other parameters and variables used in the formulation are;

α	: Path loss exponent
γ	: Threshold value of SINR for decoding the message successfully
C_i	: Environmental noise around node i
K	: Maximum number of time slots
M	: Large enough number

$$\min \quad Z = \sum_{i=1}^{|V|} \sum_{k=1}^{K} P_{ik} \tag{4.1}$$

$st.$

$$X_{11} = 1 \tag{4.2}$$

$$\sum_{i=1}^{|V|} X_{i1} \leq 0 \tag{4.3}$$

$$X_{jk} - \sum_{t=1}^{k-1} \sum_{\substack{i=1 \\ i \neq j}}^{|V|} Y_{ijt} \leq 0 \quad , \quad \begin{matrix} \forall j \in V \backslash \{1\} \\ 2 \leq k \leq K \end{matrix} \tag{4.4}$$

$$\sum_{t=1}^{k-1} \sum_{\substack{i=1 \\ i \neq j}}^{|V|} Y_{ijt} + \sum_{m=1}^{|V|} \sum_{\substack{n=1 \\ m \neq n}}^{|V|} Y_{mnk} \geq 1 \quad , \quad \begin{matrix} \forall j \in V \backslash \{1\} \\ 2 \leq k \leq K \end{matrix} \tag{4.5}$$

$$M(1 - Y_{ijk}) + P_{ik} - d_{ij}^{\alpha} \, \gamma \, (C_j + I_{ijk}) \geq 0 \quad , \quad \begin{matrix} \forall i \in V, \quad \forall j \in V \backslash \{1\} \\ 1 \leq k \leq K \end{matrix} \tag{4.6}$$

$$\sum_{k=1}^{K} \sum_{\substack{i=1 \\ i \neq j}}^{|V|} Y_{ijk} \geq 1 \quad , \quad \forall j \in V \backslash \{1\} \tag{4.7}$$

$$I_{ijk} - \sum_{\substack{m=1 \\ m \neq i \\ m \neq j}}^{|V|} P_{mk} \, d_{mj}^{-\alpha} = 0 \quad , \quad \begin{matrix} \forall i \in V, \quad \forall j \in V \backslash \{1\} \\ 1 \leq k \leq K \end{matrix} \tag{4.8}$$

$$P_{ik} - M X_{ik} \leq 0 \quad , \quad \forall i \in V, \quad 1 \leq k \leq K \tag{4.9}$$

$$\sum_{k=1}^{K} X_{ik} \leq 1 \quad , \quad \forall i \in V \tag{4.10}$$

$$P_{ik} \geq 0 \quad , \quad \forall i \in V, \quad 1 \leq k \leq K \tag{4.11}$$

$$X_{ik} = \begin{cases} 0 \\ 1 \end{cases} \quad , \quad \forall i \in V, \quad 1 \leq k \leq K \tag{4.12}$$

$$Y_{ijk} = \begin{cases} 0 \\ 1 \end{cases} \quad , \quad \begin{matrix} \forall i \in V, \quad 1 \leq k \leq K \\ \forall j \in V \backslash \{1, i\} \end{matrix} \tag{4.13}$$

The objective function (4.1) minimizes the total power consumption in the broadcast tree. Constraints (4.2) and (4.3) ensure that only the source node transmits in the first time slot. In constraint (4.4), it is expressed that a node j, other than *source* node, cannot transmit in time slot $k \geq 2$, unless it has been reached by some transmission in the first $(k-1)$ time slots. Note that if node $j \in V \backslash \{source\}$ has not been reach in the first $(k-1)$ time slots, the second term of the constraint (4.4) becomes 0, then the constraint (4.4) becomes $X_{jk} \leq 0$ which means node j cannot transmit in time slot k. The constraint (4.5) is about linking the variables X_{ik} and Y_{ijk}. If there is at least one node $j \in V \backslash \{source\}$ that has not been reached by any transmission in the first $(k-1)$ time slots, constraints (4.5) enforce that there must be at least one transmission in k^{th} time slot where $k \geq 2$. Note that the first term on the left hand side of constraints (4.5) becomes zero for such a node j forcing the second term to be positive. Constraints (4.6)-(4.7)-(4.8) ensure that each node $j \in V \backslash \{source\}$ receives the message successfully from a node in the network. If $Y_{ijk} = 0$, then the corresponding constraint (4.6) becomes redundant. On the other hand, if $Y_{ijk} = 1$, which means node j receives the message from node i in k^{th} time slot, then transmission power level of node i must be greater or equal to $d_{ij}^{\alpha} \gamma (C_j + I_{ijk})$. Constraints (4.7) express that each node, other than *source* node, must receive the message from at least one node in the broadcast tree in some time slot. Interference at a receiving node j is calculated with respect to the candidate transmitting node i and transmission time slot k in constraints (4.8). Constraints (4.9) and (4.10) let each node to transmit at most one time during the broadcast session.

Theoretically, our MILP formulation includes $O(|V|^3)$ constraints, and $O(|V|^3)$ variables since K is $O(|V|)$. Therefore, our formulation does not scale well when the number of nodes in the network increases, and heuristics are required for networks of practical size. Thus, for the SINR constrained minimum power broadcasting problem, we develop a heuristic –referred to SINR-BIP– which exploits exploits the broadcast incremental power (BIP) structure while considering the SINR values of received signals.

4.3 Proposed Heuristic: SINR-BIP

Our objective is the determination of the minimum-power tree, rooted at the source node, that reaches all of the other nodes in the network. The total power associated with the tree is simply the sum of the powers at all transmitting nodes. Clearly, this is a node-based metric because it enables us to exploit the wireless multicast advantage [3]. The proposed SINR-BIP heuristic has a simple and greedy idea. At each iteration, the broadcast tree grows by selecting a covered node to transmit the broadcast message and an uncovered node to receive the broadcast message. Transmitting and receiving node pairs are selected according to the minimum incremental power rule. By using the incremental power rule, we only consider the additional required transmission power of the broadcast tree. This idea is similar to the well known BIP *(Broadcast Incremental Power)* algorithm [3] but the calculation of the transmission power is completely different, because in SCMPB we are also dealing with environmental noise and interference at the receiving nodes.

SINR-BIP heuristic starts with an initialization phase and each iteration of the algorithm consists of the following steps; (i) Finding the next transmission, (ii) Pruning the branches, if necessary , (iii) Updating the network information, (iv) Investigating the SINR values and identifying the parent-child relationships, and (v) Solving a one-level connectivity problem, if necessary.

Network topology is used as an input of the SINR-BIP heuristic. Since the topology is known, a distance matrix d, where d_{vw} represents the distance between nodes v and w, is constructed as an initialization phase. Furthermore, the covered node set S and the set of transmitting nodes T are initially comprised of the source node only. In other words, initially $S = \{source\}$ and $T = \{source\}$. Since we are considering the broadcasting problem, uncovered node list S' is initially consist of all nodes, other than the *source*, in the network. The most important parts of the information structure of each node v are level ($Level(v)$), SINR value ($SINR(v)$), parent ($Parent(v)$), child ($Child(v)$) and power level ($P(v)$) information. Level of each node $w \in V$ represents the transmission time slot of node w, if $w \in T$, and precedence constraints in the broadcast tree are verified by checking the level information. Suppose that $Parent(w) = v$ and $Level(v) = k$, then $Level(w)$ must be strictly greater than $Level(v)$ (generally[2] $Level(w) = Level(v) + 1 = k + 1$).

After the initialization phase, the algorithm starts. In the first step of each iteration, the next transmission, which extends the broadcast tree, is decided. This decision is based on the minimum incremental power rule. A node must receive a message before it transmits, therefore only covered nodes, $v \in S$, are considered as the candidate transmitters. For each covered node $v \in S$, the required transmission power, $P'(v, w)$, to send a broadcast message to an uncovered node $w \in S'$ is calculated as follows;

$$P'(v, w) = d_{vw}^{\alpha} \, \gamma \, (C_w + I_{vw}) \tag{4.14}$$

As discussed in the previous section, $L(v)$ represents the set of nodes which transmit with node v in the same time slot and $v \cap L(v) = \emptyset$. We further define I_{vw} as the total

[2]exceptional cases will be explained later

interference, caused by all node in $L(v)$, at receiving node w. C_w in (Eq. 4.14) represents environmental noise and it is assumed constant, however I_{vw} heavily depends on the other transmissions and it calculated as follows;

$$I_{vw} = \sum_{i \in L(v)} P(i)\, d_{iw}^{-\alpha}$$

where $P(i)$ is the currently assigned transmission power level of node $i \in V$. $P'(v, w)$ represents the required transmission power at node $v \in S$ to establish a link between node v and node $w \in S'$. In terms of energy efficiency, at node v, additional power requirement is more important, because node v might be already in T and $P(v) > 0$. Therefore, we use the minimum incremental power rule. For all node pairs (v, w), where $v \in S$ and $w \in S'$, additional power requirement at node v is obtained from the discrepancy between required transmission power level to reach node w from node v and current transmission power level of node v; $P''(v, w) = P'(v, w) - P(v)$. By using the information of incremental required power, $P''(\cdot, \cdot)$, we obtain the next transmitting node v^* and the receiving node w^* as follows;

$$(v^*, w^*) = \underset{\substack{v \in S \\ w \in S'}}{\arg\min}\, P''(v, w)$$

The minimum increase in power consumption is obtained at node v^* to extend the broadcast tree, thus current power level of node v^* is increased by $P''(v^*, w^*)$ and $P(v^*)$ becomes $P(v^*) \leftarrow P(v^*) + P''(v^*, w^*) = P'(v^*, w^*)$

As the second step, we propose a pruning procedure to mitigate the impact of the myopic strategy of the algorithm. Once we obtain the node pairs (v^*, w^*), the algorithm checks for the necessity of the pruning procedure. If pruning is required, $Cut\_the\_Branches(v^*)$ procedure is invoked; otherwise the algorithm proceeds to the next step. $Cut\_the\_Branches(v^*)$ procedure will be detailed later.

The third step of each iteration updates the information changed by the transmission of node v^*. First of all, node v^* is added to the set of transmitting nodes T; $T \leftarrow T \cup \{v^*\}$. Then, the sets of covered and uncovered nodes are updated; $S \leftarrow S \cup \{w^*\}$ and $S' \leftarrow S' \backslash \{w^*\}$. Increase in $P(v^*)$ affects the received signal power, thus SINR values, at each node $w \in V \backslash \{source\}$, that's why SINR values must be re-calculated. Note that there are $|T|$ number of transmissions which means there are $|T|$ received signals at node w. For each node w, a subset of transmitting nodes in T is considered as candidate parent nodes ($z \in Z(w) \subseteq T$) and SINR values of the received signals from $z \in Z(w)$ are calculated as follows;

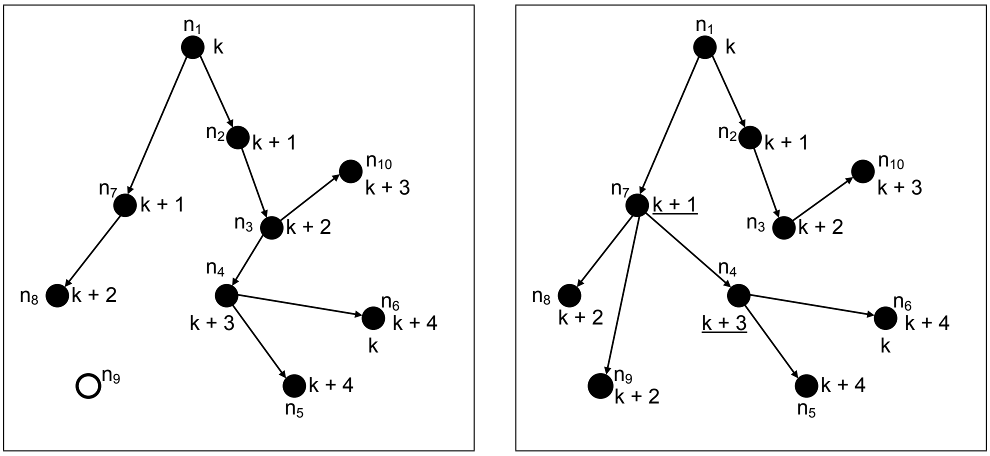
$$SINR(z,w) = \frac{P_{rec}(z,w)}{C_w + I_{zw}} = \frac{P(z)\, d_{zw}^{-\alpha}}{C_w + I_{zw}}$$

where C_w is the environmental noise around node w, $P_{rec}(z,w)$ is the signal power received from node z, and I_{zw} is the interference caused by the other nodes transmitting in the same time slot with z; $I_{zw} = \sum_{i \in L(z) \subseteq Z(w)} P(i)\, d_{zw}^{-\alpha}$. If node w is a leaf node of current broadcast tree, then $Z(w) = T$; otherwise, $Z(w) = \{z \,|\, Level(z) < Level(w) \quad \text{and} \quad z \in T\}$. Once we obtain the $SINR(z,w)$ for all nodes $w \in V \backslash \{source\}$ and $z \in Z(w)$, it is easy to select the most dominant signal among all received signals;

$$\begin{aligned} SINR(w) &= \max_{z \in Z(w) \subseteq T} SINR(z,w) \\ z^*(w) &= \arg\max_{z \in Z(w) \subseteq T} SINR(z,w) \end{aligned}$$

where $z^*(w)$ and $SINR(w)$ are the transmitter of the most dominant signal received at node w and its SINR value, respectively. Then the algorithm proceeds to the next step.

In this step, the algorithm investigates the $SINR(v)$, $\forall v \in V \backslash \{source\}$, and identifies the parent-child relationship of the broadcast tree. Investigation of the SINR values is very important, because an increase in $P(v^*)$ might cause interference at $Level(v^*)$ and decrease in SINR value at some receiving nodes. If SINR value at a receiving node w falls below the threshold γ, then node w cannot receive message successfully. Each node $\hat{w}$ with SINR value less than threshold, is added to the *list*; $list = \{\hat{w} \,|\, SINR(w) < \gamma \quad \text{and} \quad \hat{w} \in S\}$. Note that if $\hat{w} \in T$, then a connectivity problem occurs between $Parent(\hat{w})$ and $\hat{w}$. This connectivity problem will be handled in the following steps by using node set *list*. On the other hand, for the nodes $\bar{w}$ where $SINR(\bar{w}) \geq \gamma$ and $\bar{w} \in S \cup S'$, parent-child relationships and level information are updated. For the level information update two important cases occur; (i) if $Parent(\bar{w}) = \emptyset$, then $Level(\bar{w}) = Level(z^*(w)) + 1$, and ($ii$) if $Parent(\bar{w}) \neq z^*(w)$, then no matter what the $Level(z^*(\bar{w}))$ is, the $Level(\bar{w})$ remains unchanged. These cases are explained with an illustrative example shown in the Figure 4.3. Suppose that Figure 4.3-a is the partial broadcast tree at iteration t, $Level(n_1) = k$, $S = \{n_1, n_2, n_3, n_4, n_5, n_6, n_7, n_8, n_{10}\}$, $S' = \{n_9\}$, and $T = \{n_1, n_2, n_3, n_4, n_7\}$. Further suppose that node pair (v^*, w^*) with minimum incremental power is (n_7, n_9). Increase in the transmission power of node n_7 affects the received signal strength and SINR values of the other receiving nodes. It is trivial that $SINR(n_9) = \gamma$ and $z^*(n_9) = n_7$. Since $Parent(n_9) = \emptyset$, case (i), $Level(n_9) = Level(z^*(w)) + 1 = Level(n_7) + 1 = k + 2$. On the other hand, assuming that $SINR(n_7, n_4) \geq \gamma > SINR(n_3, n_4)$, n_4 is assigned from n_3 to n_7, $z^*(n_4) = n_7$, case (ii), and level of n_4 remains same; $Level(n_4) = k + 3$. The resulting partial broadcast tree is shown in the Figure(4.3)-b.



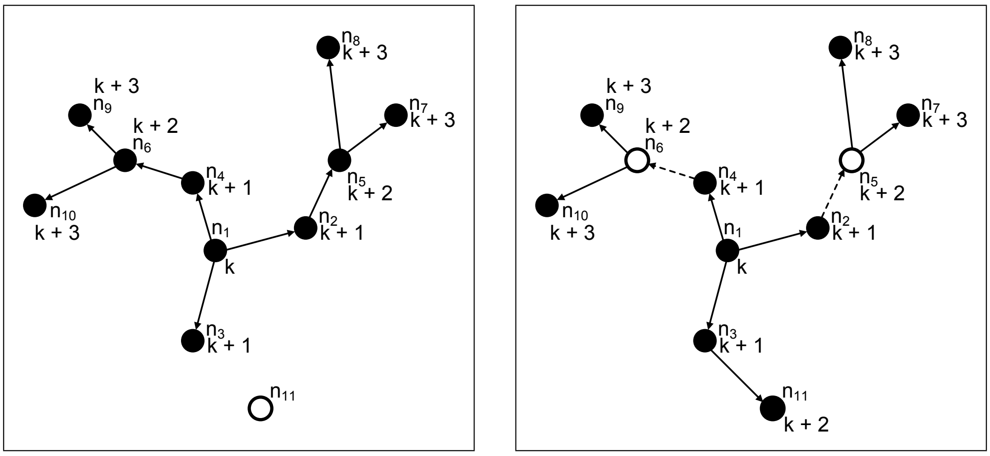
(a) iteration t (b) iteration $t + 1$

Figure 4.3: Illustrative Example for Level Update

The reason of our approach for case (ii) can be explained by using the same illustrative example shown in Figure 4.3. Suppose that case (ii) is ignored and $Level(n_4)$ is assigned $k + 2$, since the parent of n_4 is in the level $k + 1$, then the level of n_6 and n_7 will decrease one and become $k + 3$. As a result of ignoring case (ii), the structure of the broadcast tree will mostly change, because interference effect of all the transmitting nodes located in the branch starting from n_4, will change. This change will cause connectivity problems for most of the branches in the network. In our illustrative example, if we ignore case (ii), transmission of n_4 will cause interference at node n_{10} and connectivity problem will occur between n_3 and n_{10}.

As the last step of each iteration, the algorithm focuses on the connectivity problem at $Level(v^*)$. As mentioned previously, connectivity problem occurs, if $SINR(\hat{w}) < \gamma$ for some node $\hat{w} \in T$, because of the interference caused by the increase in $P(v^*)$. Connectivity problem can be explained with an illustrative example shown in the Figure 4.4. Suppose that Figure 4.4-a is the partial broadcast tree at iteration t, $Level(n_1) = k$, $S = \{n_1, n_2, n_3, n_4, n_5, n_6, n_7, n_8, n_9, n_{10}\}$, $S' = \{n_{11}\}$, and $T = \{n_1, n_2, n_4, n_5, n_6\}$.

Further suppose that node pair (v^*, w^*) with minimum incremental power is (n_3, n_{11}). Notice that $SINR(n_5) = SINR(n_6) = \gamma$ and increase in the transmission power of n_3 causes interference at n_5 and n_6, since $L(n_3) = \{n_2, n_4\}$. Because of this interference, new SINR values at n_5 and n_6 fall below the threshold value; $SINR(n_5) < \gamma$ and $SINR(n_6) < \gamma$. As a result, n_5 and n_6 cannot receive the signal successfully from n_2 and n_4, respectively. Figure 4.4-b depicts the connectivity problem between node pairs (n_2, n_5) and (n_4, n_6). Note that the connectivity problem occurs between nodes $v \in v^* \cup L(v^*)$ and $w \in \bigcup_{i \in v^* \cup L(v^*)} Child(i)$, and transmitting nodes are at $Level(v^*) = (k+1)$.



(a) iteration t (b) iteration $t+1$

Figure 4.4: Illustrative Example for Connectivity Problem

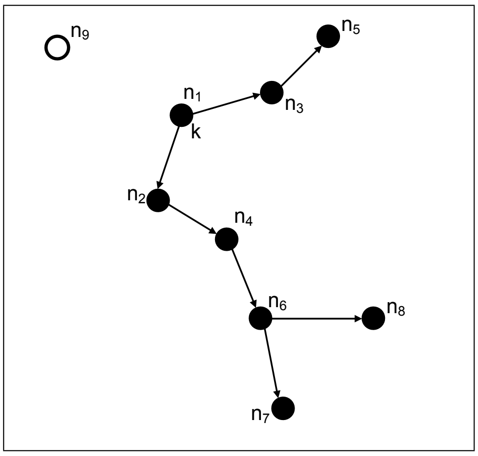
In order to handle this connectivity problem, we propose a subroutine; *Connect*(k). *Connect*(k) procedure is invoked, if the *list* is not empty, and restores connection for level k. Details of *Connect*(k) procedure will be explained later.

In each iteration of the SINR-BIP, calculating the required transmission power for each covered-uncovered node pair and calculating the received signal power dominate the computational complexity of the iteration. These computations both require $O(n^2)$ operations where n represents the number of nodes in the network. In addition, $O(n^2)$ iterations are required in order to cover all the nodes in the network, because at each iteration one node is added to the tree and for some iterations covered nodes can be assigned as uncovered by *Cut_the_Brancehes* procedure. As the result of our analysis, the computational complexity of SINR-BIP is calculated $O(n^4)$.

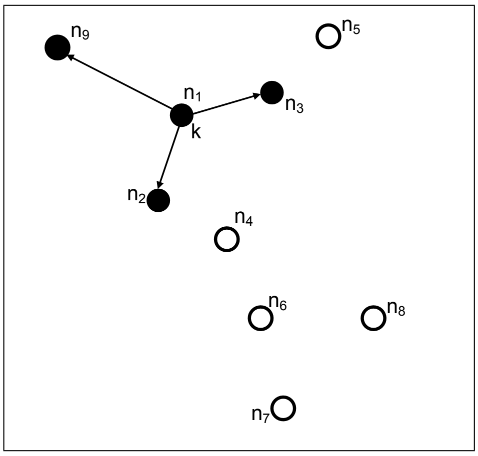
SINR-BIP heuristic iteratively expands the broadcast tree until there is no other uncovered node to be reached; $S' = \emptyset$. Complete pseudo-code of the SINR-BIP heuristic is given in the Appendix.

4.3.1 $Cut\_the\_Branches(v^*)$ Procedure

SINR-BIP algorithm without $Cut\_the\_Branches(v^*)$ procedure considers only the forward expansion of the broadcast tree, however this approach causes redundant transmissions and more power consumption. Thus, we use $Cut\_the\_Branches(v^*)$ procedure to obtain better solutions for the next iterations. The myopic impact of SINR-BIP without $Cut\_the\_Branches(v^*)$ is illustrated in the Figure 4.5.



(a) iteration t, before pruning



(b) iteration t, after pruning

Figure 4.5: Illustrative Example for $Cut\_the\_Branches(v^*)$ Procedure

We define $Branch(v)$ as the set of nodes which are located in the branch of broadcast tree, starting from node v and $v \cap Branch(v) = \emptyset$; e.g., $Branch(n_2) = \{n_4, n_6, n_7, n_8\}$ in Figure 4.5. Notice that, if $Branch(v) \neq \emptyset$, then at least one node in $Child(v)$ is in S or vice versa. On selecting the node v^* to transmit, the algorithm checks whether $Branch(v^*)$ is empty or not; if $Branch(v^*) \neq \emptyset$, then $Cut\_the\_Branches(v^*)$ procedure is invoked. Then, the procedure is basically updates the sets; $S \leftarrow S \backslash \{\dot{w}\}$, $S' = S' \cup \{\dot{w}\}$ and $T = T \backslash \{\dot{w}\}$, $\forall \dot{w} \in Branch(v^*) \backslash Child(v^*)$.

Suppose that Figure 4.5-a is the partial broadcast tree at iteration t, $Level(n_1) = k$, $S = \{n_1, n_2, n_3, n_4, n_5, n_6, n_7, n_8\}$, $S' = \{n_9\}$, and $T = \{n_1, n_2, n_3, n_4, n_6\}$. Further suppose that node pair (v^*, w^*) with minimum incremental power is (n_1, n_9). Since n_1 is already in T, the $Cut\_the\_Branches(n_1)$ procedure is invoked and the broadcast tree is depicted in Figure 4.5 after the $Cut\_the\_Branches(n_1)$ procedure. It is easily seen that if $P''(n_1, n_4) < P''(n_2, n_4)$, then the procedure provides cost reduction in terms of energy consumption. Note that, $Cut\_the\_Branches(v^*)$ procedure results in more iterations to construct the broadcast tree. In our SINR-BIP, we resolve this trade-off in favor of obtaining better solution.

4.3.2 One-Level Connectivity Problem

The connectivity issue very important for ad hoc networks. Suppose that one-level connectivity problem occurrs in level k, and note that $Level(v^*) = k$, then SINR-BIP heuristic invokes $Connect(k)$ procedure in order to provide a feasible and energy efficient solution for level k. Connectivity is established between two sets of nodes; $parent\_set$ and $child\_set$. We define $parent\_set = \{s_i \,|\, s_i \in \{v^*\} \cup L(v^*)\}$ and $child\_set = \{r_i \,|\, r_i \in Child(w),\, \forall w \in parent\_set\}$. Figure 4.6 is the illustration of nodes in $parent\_set$ and $child\_set$.

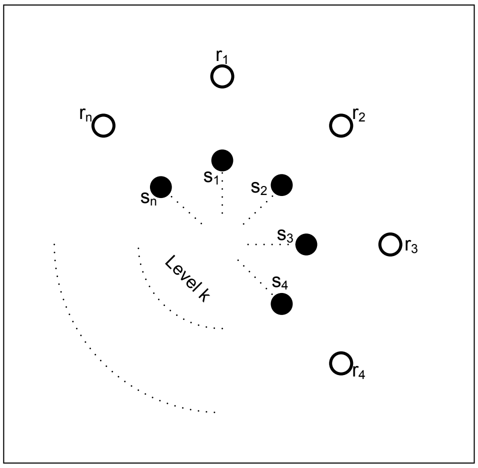


Figure 4.6: One-level Connectivity Problem; $parent\_set$ and $child\_set$

Note that each node $r_i \in child\_set$ must be covered by at least one node $s_i \in T_p \subseteq parent\_set$. It is clearly seen that the one-level connectivity problem is similar to a set covering problem for bipartite graphs.

Mathematical Model

The one-level connectivity problem can be seen as a special case of SCMPB. It only considers the set of nodes in $parent\_set$ and $child\_set$. Thus, the mathematical formulation of the one-level connectivity problem has similar to that of SCMPB.

The variables and parameters that are used in the mathematical model of one-level connectivity problem are mostly similar to the ones in the mathematical model of SCMPB. Binary variable Y_{ij} takes the value 1, if node $j \in child\_set$ receives message from $i \in parent\_set$; 0, otherwise. Total interference which is caused by all nodes $v \in parent\_set \backslash \{i\}$, at receiving node $j \in child\_set$ is represented by I_{ij}. Other variables and parameters are;

P_i	: Transmission power level of node $i \in parent\_set$
d_{ij}	: Distance between node $i \in parent\_set$ and $j \in child\_set$
α	: Path loss exponent
γ	: Threshold value of SINR for decoding the message successfully
C_j	: Environmental noise around node $j \in child\_set$
M	: Large enough number

$$\min \quad Z = \sum_{i \in parent\_set} P_i \tag{4.15}$$

$s.t.$

$$M(1 - Y_{ij}) + P_i - d_{ij}^{\alpha}\,\gamma\,(C_j + I_{ij}) \geq 0 \quad , \quad \forall i \in parent\_set, \; \forall j \in child\_set \tag{4.16}$$

$$\sum_{i \in parent\_set} Y_{ij} \geq 1 \quad , \quad \forall j \in child\_set \tag{4.17}$$

$$I_{ij} - \sum_{\substack{m \in parent\_set \\ m \neq i}} P_m\, d_{mj}^{-\alpha} = 0 \quad , \quad \forall i \in parent\_set, \; \forall j \in child\_set \tag{4.18}$$

$$P_i \geq 0 \quad , \quad \forall i \in parent\_set \tag{4.19}$$

$$Y_{ij} = \begin{cases} 0 \\ 1 \end{cases} \quad , \quad \forall i \in parent\_set, \; \forall j \in child\_set \tag{4.20}$$

Objective function (4.15) minimizes the total power consumption at level k. Constraints (4.16)-(4.17)-(4.18) ensure that each node $j \in child\_set$ receives the message successfully from a node $i \in parent\_set$. If $Yij = 0$, then the constraint (4.16) becomes redundant. On the other hand, if $Yij = 1$, which means node $j \in child\_set$ receives message from node $i \in parent\_set$, then transmission power level of node $i \in parent\_set$ must be greater or equal to $d_{ij}^{\alpha}\,\gamma\,(C_j + I_{ij})$. Constraints (4.17) express that each node $j \in child\_set$ must receive the message from at least one node $i \in parent\_set$ at level k. Interference at a receiving node $j \in child\_set$ is calculated with respect to the candidate transmitting node $i \in parent\_set$ in constraints (4.18).

Theoretically, our MILP formulation for the one-level connectivity problem includes $O(|V|^2)$ constraints, and $O(|V|^2)$ variables. Therefore, our formulation does not scale well when the number of nodes in the network increases, and heuristics are required for networks of practical size. Thus, for the one-level connectivity problem in level k, we propose a heuristic $Connect(k)$ described in the following section.

Heuristic for One-level Connectivity Problem: *Connect*(k) Procedure

We develop *Connect*(k) procedure by using the same idea with SINR-BIP because connectivity problem at level k can be defined as a multi-source and single level version of SCMPB problem. In *Connect*(k) procedure, incremental power structure is exploited considering further energy efficiency. Note that, *Connect*(k) only deals with two distinct sets, *parent_set* and *child_set*, and our objective is the determination of minimum power links between *parent_set* and *child_set* so that each node in *child_set* is covered by at least one node in *parent_set*.

Connect(k) procedure starts with an initialization phase and each iteration of the procedure consists of the following steps; (*i*) Finding the next transmission, (*ii*) Updating the network information, (*iii*) Investigating the SINR values and identifying the parent-child relationships, and (*iv*) Re-updating the network information. Since we are dealing with a single level problem, we omit the pruning procedure mentioned in SINR-BIP. Initialization phase of *Connect*(k) has one more operation than the initialization phase of SINR-BIP;

$$P(x^*) \quad = \quad \max_{i \in parent\_set} \left(\max_{j \in child\_set} \left(d_{ij}^{\alpha} \, \gamma \, C_j \right) \right)$$

Note that $P(x^*)$ and x^* represent the minimum required transmission power of a single transmission which covers all the nodes in *child_set* and the selected node in *parent_set* for the transmission, respectively. $P(x^*)$ and x^* will be used for further energy efficiency.

Finding the next transmission and updating the network information are identical to those in SINR-BIP. Since the problem is in one-level, we implement the step of investigating the SINR values and identifying the parent-child relationships as in SINR-BIP without considering the level information. Note that the level of nodes in *child_set* will automatically be $k+1$ in the solution of *Connect*(k), because all nodes in *parent_set* are in level k.

The set of nodes with SINR values less than the threshold are added to the *sub_list* in the previous step; $sub\_list = \{j \,|\, SINR(j) < \gamma \quad \text{and} \quad j \in child\_set\}$. In the step of re-updating the network information, nodes in the *sub_list* are labeled as uncovered and removed from the set of covered nodes.

Connect(k) heuristic iteratively selects the energy efficient links until there is no other uncovered node in *child_set*. For some special cases, *Connect*(k) is terminated before covering all the nodes in *child_set*. At the end of each iteration, *Connect*(k)

checks for the energy efficiency; if the total energy consumption of the current partial solution exceeds the minimum required transmission power of a single transmission to cover all the nodes in *child_set*, then the heuristic returns $P(x^*)$ and x^* as the solution of $Connect(k)$. Note that $P(x^*)$ and x^* are obtained in the initialization phase of $Connect(k)$.

4.4 Numerical Results

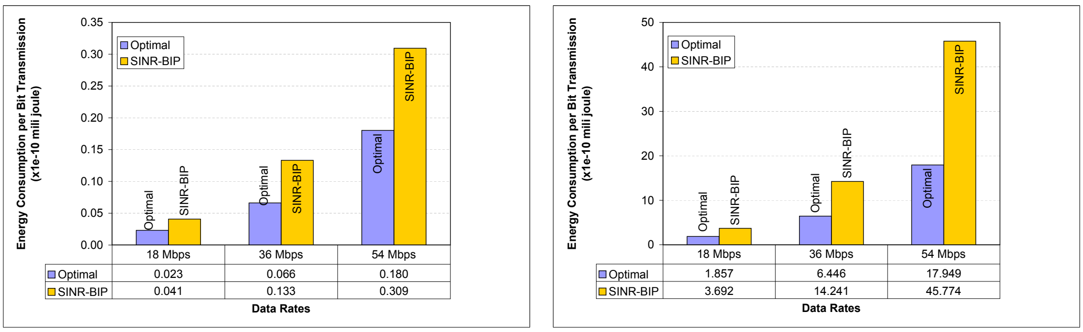
In this section, we investigate the performance of SINR-BIP on randomly generated instances. For each network consisting of 10, 25, and 50 nodes, 20 random network topologies are generated in a 50 by 50 m^2 area and spatial Poisson distribution is used for determining the node locations. Then, the performance of SINR-BIP is analyzed in terms of *energy consumption per bit transmission* for changing number of nodes in the network (N), transmission rate, and path loss coefficient (α). Among different transmission rates in wireless communication 18 Mbps, 36 Mbps, and 54 Mbps are chosen for our analysis. Notice that different SINR threshold values (γ) are required for different transmission rates, thus we consider the threshold values presented in [40] by Lin and Hou. The transmission rates that we consider and the corresponding minimum required SINR values of these data rates are represented in the Table-4.1. In addition, we investigated the performance of SINR-BIP algorithm with respect to two different path loss coefficient values 2.5 and 4.

Transmission Rate (Mbps)	SINR Threshold γ (dB)
18	10.79
36	18.80
54	24.56

Table 4.1: Minimum Required SINR Values for Different Date Rates

For our experimental studies, we considered two major assumptions about the maximum transmission range and the environmental noise that can be faced. First, we simplified the experimental studies by considering that the nodes have no transmission range limits. In other words, one node is able to reach all the nodes by a single transmission. Second, we observed that the environmental noise (or thermal noise) is generally assumed around −90 dBm in the literature [40, 41, 42], thus we consider the environmental noise as −90 dBm.

Since SINR-BIP is the first algorithm that constructs the broadcast tree while considering the SINR values of the received signals, we compared the performance of SINR-BIP with the optimal (for $N = 10$) and best feasible (for $N = 25$ and $N = 50$) solutions. Optimal and best feasible solutions are obtained by solving the mathematical model of the SCMPB problem. Notice that the mathematical model has $O(N^3)$ variables and $O(N^3)$ constraints, thus it is very difficult to obtain the optimal solution for increasing number of nodes. Therefore, we first investigated the strength of the proposed algorithm by comparing the results with the *optimal* solutions of the small size instances where $N = 10$. Then, we further analyzed the performance of SINR-BIP for larger instances by comparing with the best feasible solutions obtained by solving the MILP formulation for a maximum CPU time of 1500 seconds each, using ILOG OPL Studio 5.2 running on ILOG CPLEX 10.2. The numerical results presented in this section are the averages of the 20 randomly generated network topologies for each parameter settings.

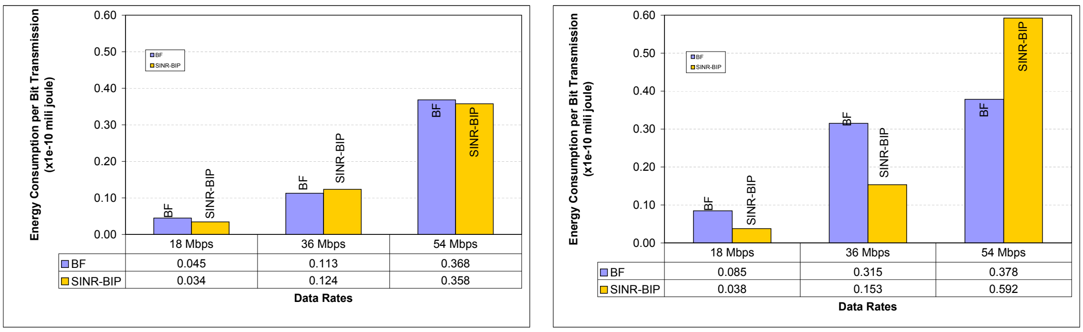


(a) N = 10, $\alpha = 2.5$ (b) N = 10, $\alpha = 4$

Figure 4.7: Energy Consumption per Bit Transmission of the Broadcast Trees Generated by SINR-BIP and the Optimal Solutions ($N = 10$)

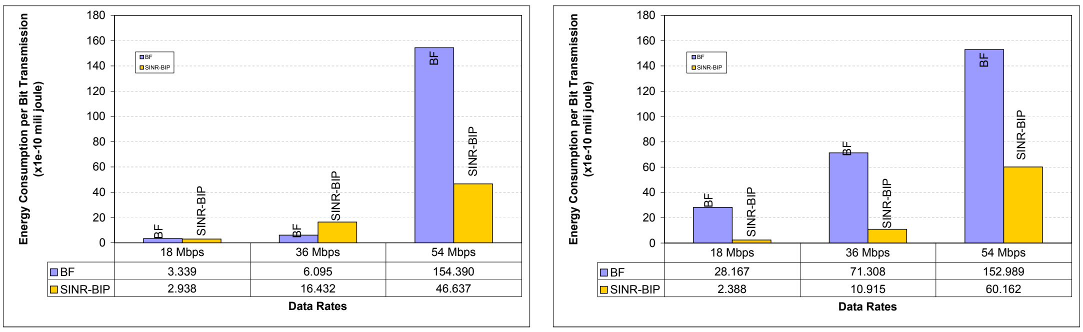
Firstly, Figure (4.7) depicts the results for 10 node instances. It represents energy consumption per bit transmission with respect to changing data transmission rates and path loss coefficient. According to the results illustrated by Figure (4.7)-a and Figure (4.7)-b, on the average, the performance of SINR-BIP algorithm is 80% and 120% more than the optimal solutions, respectively. In addition, it is observed that the deviation of the solutions obtained by SINR-BIP is in between 70% and 160% of the optimal solutions. In other words, the solutions obtained by SINR-BIP is no more than 2.6 times the optimal solution for our 10 node test instances. This is because the SINR-BIP algorithm runs in a greedy manner by adding an uncovered node into the broadcast tree at each iteration according to the greedy minimum incremental power rule. Notice that the SINR-BIP algorithm performs better for smaller transmission rates, because smaller transmission rates require smaller SINR threshold values. Since the SINR threshold values are in dB unit, an increase in the SINR threshold value dramatically increases the energy consumption.

Upon observing the results for small size problems, we enlarged the network size to 25 and 50 nodes. Figure (4.8) and Figure (4.9) depict the energy consumption per bit transmission with respect to changing node density of the network and path loss coefficient. For most of the problem settings the results of the SINR-BIP algorithm is better than the best feasible solutions obtained by solving the mathematical formulation for 1500 seconds. On the average, the performance of the SINR-BIP algorithm is 66% better than the best feasible solutions. In addition, we observed that the deviation of the numerical results of the SINR-BIP is in between -92% and 170% of the best feasible solutions. Moreover, the observation that SINR-BIP performs better for smaller transmission rates remains same in this part of our numerical analysis.



(a) N = 25, $\alpha = 2.5$ (b) N = 50, $\alpha = 2.5$

Figure 4.8: Energy Consumption per Bit Transmission of the Broadcast Trees Generated by SINR-BIP and the Best Feasible Solutions ($\alpha = 2.5$)



(a) N = 25, $\alpha = 4$ (b) N = 50, $\alpha = 4$

Figure 4.9: Energy Consumption per Bit Transmission of the Broadcast Trees Generated by SINR-BIP and the Best Feasible Solutions ($\alpha = 4$)

Notice that the numerical analyses for the instances where $N = 25$ and $N = 50$ are not strong enough. The solutions of these instances are obtained by running the ILOG CPLEX 10.2 for only 1500 seconds. The reason behind this limited running time is the limited memory of the computer. We observed that for longer runs the ILOG OPL Studio 5.2 crashes due to the memory problems. Because our mathematical model includes $O(N^3)$ integer variable and $O(N^3)$ constraints. In addition, numerical instabilities can be faced due to the use of M (large enough number). Therefore, in one point of view the solutions to the 25 node and 50 node instances can be considered as inconclusive. For more realistic solutions, computers with larger memories can be used.

Chapter 5

Conclusion and Future Work

In this study, we consider two extensions of energy efficient broadcasting problem, namely the delay constrained broadcasting problem (DCMPB) and signal-to-interference-plus-noise-ratio constrained broadcasting (SCMPB) problem in wireless ad hoc networks. First, we investigate the distributed construction of minimum power broadcasting trees with a maximum depth of Δ where the delay is represented by the hop count that a message travels from source to the destination nodes. Second, we focus on the communication medium and investigate the construction of the minimum power broadcasting tree where each received signal must be strong enough with respect to the interference and the environmental noise at the receiving node, so that the signal is successfully decoded. Thus, our contribution in this study is twofold: first, we propose a distributed algorithm so-called Distributed Tree Expansion (DTE) for DCMPB problem; second we propose SINR Broadcast Incremental Power (SINR-BIP) for SCMPB problem. Note that both of the proposed algorithms are constructive in nature in which the broadcast tree iteratively grows.

In DTE, the broadcast tree is constructed in a distributed fashion where the source node is the sole responsible node for constructing the tree. At each iteration, broadcast tree is extended by one or more nodes where the expansion decision is given according to a priority function. Each node in the current partial tree uses the local information to decide best transmission range by calculating the priority function values within its neighborhood. Since these candidate transmission ranges are propagated to the source, the source node decides the tree expansion among those candidate transmissions by comparing the priority function values. Then the current partial tree is extended at each iterations by repeating the same procedure until there is no uncovered node in the network.

In SINR-BIP, the broadcast tree is constructed in a centralized fashion where the entire topology information is known. The broadcast tree grows by adding nodes one-by-one to the broadcast tree. The node which requires the minimum additional power to be covered is added to the tree at the end of each iteration. Although SINR-BIP is a constructive algorithm, for some iterations the algorithm prunes the broadcast tree so that the myopic effect of the decisions is mitigated. The algorithms is terminated when the broadcast tree is constructed with all the nodes in the network.

Energy efficient broadcasting with all aspects and requirements is very important for wireless networking. However, energy efficient broadcasting is NP-hard, thus it is very difficult to solve the problem at the optimality. In our future works, first, we will improve the developed heuristics; second, we will investigate the solution methods of other extensions of energy efficient broadcasting.

On improving the DTE, we will explore the use of limited coordination among the nodes so that the messaging overhead is mitigated. In addition, the delay constraint considered in the DTE algorithm refers to the end-to-end hop constraint. However, delay at each hop depends on the queuing, transmission, medium access, and processing delays, thus it is not usually the same. DTE can be extended to take into account the delay as time instead of number of hops. Moreover, we observed that there are some redundant transmissions in the resulting broadcast trees generated by DTE and SINR-BIP, even though we implement a pruning procedure embedded in the algorithms. Therefore, we aim to eliminate these redundant transmissions by developing and implementing a better way of pruning. As an improvement on SINR-BIP, we will try to implement the idea of SINR-BIP in a distributed fashion, so that the algorithm becomes more applicable for the ad hoc settings.

In addition to the current studies, we plan to investigate fast heuristics and/or optimal algorithms for other extensions of the energy efficient broadcasting. We especially plan to focus on the distributed implementation of the algorithms which require limited and localized topology information.

Appendix

INPUT: Network Topology, α, γ, $C_w \, \forall w \in V \backslash \{source\}$

INITIALIZATION

Covered nodes set: $S := \{source\}$

Uncovered nodes set: $S' := V \backslash \{source\}$

Transmitting node set: $T := \{source\}$

Distance between node v and w: d_{vw}

while $|S'| > 0$ **do**

for all $v \in S$ **do**

for all $w \in S'$ **do**

/ Calculate the interference at node w */*

$I_{vw} := \sum_{i \in L(v)} P(i) \, d_{iw}^{-\alpha}$

/ Calculate the required minimum transmission power */*

$P'(v, w) := d_{vw}^{\alpha} \, \gamma \, (C_w + I_{vw})$

end for

end for

/ Calculate the required incremental transmission power */*

for all $v \in S$ **do**

for all $w \in S'$ **do**

$P''(v, w) := P'(v, w) - P(v)$

end for

end for

/ Find the node pairs with minimum incremental power */*

$(v^*, w^*) := \arg\min_{\substack{v \in S \\ w \in S'}} P''(v, w)$

$P(v^*) := P(v^*) + P''(v^*, w^*) = P'(v^*, w^*)$

/ Cut the branches starting from the child nodes of v^* */*

if $v^* \in T$ **then**

for all $\dot{w} \in Branch(v^*) \backslash Child(v^*)$ **do**

$S := S \backslash \{\dot{w}\}$

$S' := S' \cup \{\dot{w}\}$

$T := T \backslash \{\dot{w}\}$

end for

end if

/ Update Covered/Uncovered node and Transmitting node sets */*

$T = T \cup \{v^*\}$

$S = S \cup \{w^*\}$

$S' = S' \backslash \{w^*\}$

/ Calculate the Received Signal Power from each node* $z \in Z(w) \subseteq T$ *and SINR values at node* $w \in V \backslash \{source\}$ **/*

for all $w \in V \backslash \{source\}$ **do**

  for all $z \in Z(w)$ **do**

    $I_{zw} := \sum_{i \in L(z)} P(i)\, d_{iw}^{-\alpha}$

    $SINR(z, w) := \frac{P_{rec}(z,w)}{C_w + I_{zw}} = \frac{P(z)\, d_{zw}^{-\alpha}}{C_w + I_{zw}}$

  end for

end for

/ Find the most dominant signal* $w \in V \backslash \{source\}$ **/*

for all $w \in C$ **do**

  $SINR(w) := \max_{z \in Z(w) \subseteq T} SINR(z, w)$

  $z^*(w) := \arg\max_{z \in Z(w) \subseteq T} SINR(z, w)$

end for

/ Investigate the* $SINR(w)$, $\forall w \in V \backslash \{source\}$ **/*

/ Identify the parent-child relationship and update network information */*

$list := \{\hat{w} \,|\, SINR(w) < \gamma \quad \text{and} \quad \hat{w} \in S\}$

for all $\bar{w} \,|\, SINR(\bar{w}) \geq \gamma \;\&\; \bar{w} \in S \cup S'$ **do**

  Update parent-child relationships and level information

end for

/ Solve one-level connectivity problem by invoking* $Connect(k)$ **/*

if $list \neq \emptyset$ **then**

  Invoke $Connect(k)$

  Update Received Signal information

  Update SINR information

end if

end while

OUTPUT: Broadcast Tree

Bibliography

[1] S. Guo and O. Yang, "Energy aware multicasting in wireless ad hoc networks: A survey and discussion", *Computer Communications*, vol. 30, no. 9, July 2007, pp. 2129-2148.

[2] S. Guo and O. Yang, "QoS-aware minimum energy multicast tree construction in wireless ad hoc networks", *Ad Hoc Networks*, vol. 2, no. 3, July 2004, pp. 217-229.

[3] J.E. Wieselthier, G.D. Nguyen, and Ephremides, "On the construction of energy-efficient broadcast and multicast trees in wireless networks", in *Proceedings of IEEE Infocom 00*, pp. 585-594, 2000.

[4] A. S. Ahluwalia and E. H. Modiano, "On the complexity and distributed construction of energy-efficient broadcast trees in wireless ad hoc networks", *IEEE Trans. Wireless Commun.*, vol. 4, no. 5, pp. 21362147, 2005.

[5] K. Bulbul, O. Ercetin and T. Unluyurt, "Delay Constrained Minimum Power Broadcasting in Wireless Networks", 5^{th} *Intl. Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks(WiOpt)*, April 2007.

[6] S. Ciftlikli, F. Oztoprak, O. Ercetin and K. Bulbul, "Distributed Algorithms for Delay Bounded Minimum Energy Wireless Broadcasting", *The Ninth INFORMS Telecommunications Conference, College Park, MD,* March 27-29, 2008.

[7] W. L. Yang, "Constructing energy-efficient multicast trees with delay constraints in wireless ad hoc networks", in *Proceedings of IEEE AINA 2005*, 2005.

[8] V. P. Kompella, J. C. Pasquale, and G. C. Polyzos, "Two distributed algorithms for the constrained Steiner tree problem", in *Proc. Second Int. Conf. Computer Communications and Networks*, San Diego, CA, June 1993.

[9] S. Raghavan, B. Golden, and E. Wasil, *Telecommunications Modeling, Policy, and Technology*, New York, NY: Springer, 2008.

[10] D. Yuan, "Distributed pruning algorithms based on neighborhood connectivity for broadcast communication in wireless ad hoc networks", *Military Communications Conference* (MILCOM), 17-20 October 2005.

[11] M. G. C. Resende and P. M. Pardalos, *Handbook of Optimization in Telecommunications*, New York, NY: Springer, 2006.

[12] J. L. Wong, G. Veltri and M. Potkonjak, "Energy-efficient data multicast in multi-hop wireless networks", in: *IEEE Workshop on Integrated Management of Power Aware Communications, Computing and Networking*, May, 2002.

[13] N. Lynch, *Distributed Algorithms*, Morgan Kaufmann Publishers, San Mateo, CA, 1996.

[14] P. M. Pardalos and C. A. S. Oliviera, "A distributed optimization algorithm for power control in wireless ad hoc networks", in: *Proceedings of the 18th International Parallel and Distributed Processing Symposium (IPDPS04)*, 26-30 April 2004.

[15] F. Li and I. Nikolaidis, "On minimum-energy broadcasting in all-wireless networks", in: *Proc. IEEE Local Computer Networks*, November, 2001.

[16] W. Liang, "Constructing minimum-energy broadcast trees in wireless ad hoc networks", in: *Proc. ACM MOBIHOC*, June, 2002.

[17] M. Čagalj, J. P. Hubaux and C. Enz, "Minimum-energy broadcast in all-wireless networks: NP-completeness and distribution issues", in: *Proc. ACM MOBICOM*, September, 2002.

[18] P. J. Wan, G. Călinescu, X. Y. Li, and O. Frieder, "Minimum-energy broadcast routing in static ad hoc wireless networks", *ACM Wireless Networks*, vol. 8, no. 6, pp. 607-617, 2002.

[19] J. Bauer, D. Haugland, and D. Yuan, "New results on the broadcast incremental power algorithm (BIP)", 1^{st} *Nordic Optimization Symposium*, April 22, Copenhagen, Denmark, 2006.

[20] R. Klasing, A. Navarra, A. Papadopoulos, and S. Pérennes, "Adaptive broadcast consumption (ABC), a new heuristic and new bounds for the minimum energy broadcast routing problem", *Networking*, 2004. *Lecture notes in Computer Science*,

N. Mitrou, K. Kontovasilis, G. N. Rouskas, I. Iliadis, L. Merakos (Editors), vol. 3042, Springer-Verlag, Berlin, pp. 866-877, 2004,.

[21] A. K. Das, R. J. Marks, M. El-Sharkawi, P. Arabshahi, and A. Gray, “Minimum power broadcast trees for wireless networks: Integer programming formulations”, in: *Proceedings of IEEE Infocom*, 2003.

[22] D. Yuan, “An integer programming approach for the minimum-energy broadcast problem in wireless networks”, in: *Proc.* 2^{nd} *Int. Network Optimization Conf. (INOC)*, Lisbon, Portugal, 2005, pp. 643-650.

[23] K. Altınkemer, F. S. Salman, and P. Bellur, “Solving the minimum energy broadcasting problem in ad hoc wireless networks by integer programming”, in: *Proc. 2nd Int. Network Optimization Conf. (INOC)*, Lisbon, Portugal, 2005, pp. 635-642.

[24] S. Guo, O. Yang, “Minimum energy broadcast routing in wireless multi-hop networks”, in: *IEEE IPCCC*, Phoenix, USA, April 2003, pp. 273280.

[25] J. Bauer, and D. Haugland, “Analysis and computational study of several integer programming formulations for minimum-energy multicasting in wireless ad hoc networks”, *Networks*, *published online*, 2008.

[26] I. Kang, R. Poovendran, “A novel power-efficient broadcast routing algorithm exploiting broadcast efficiency”, in: *IEEE Vehicular Technology Conference (VTC)*, Orlando, October, 2003, pp. 29262930.

[27] M. Čagalj, J. P. Hubaux and C. Enz, “Energy-efficient broadcasting in all-wireless networks”, *Wireless Networks*, vol. 11, no. 1-2, pp. 177-188, 2005.

[28] W. Lou and J. Wu, “On reducing broadcast redundancy in ad hoc wireless networks”, in: *Proceedings of the* 36^{th} *Annual Hawaii International Conference on System Sciences*, 2003.

[29] J.E. Wieselthier, G.D. Nguyen, and Ephremides,“Energy-Efficient Broadcast and Multicast Trees inWireless Networks”, *Mobile Networks and Applications*, vol. 7, pp. 481492, 2002

[30] S. Chakrabarti, and A. Mishra, “QoS Issues in Ad hoc Wireless Networks”, *IEEE Communications Magazine*, Feb. 2001, pp.142-148.

[31] H. K. Wu and P. H. Chuang, "Dynamic QoS Allocation for Multimedia Ad Hoc Wireless Networks", *Mobile Networks and Applications*, vol. 6, 2001, pp. 377-384.

[32] J. Zhang, Q. Zhang, B. Li, X. Luo, and W. Zhu, "Energy-efficient routing in mobile ad hoc networks: Mobility-assisted case", *IEEE Transactions On Vehicular Technology*, vol. 55, no. 1, January 2006.

[33] T. ElBatt, and A. Ephremides, "Joint scheduling and power control for wireless ad hoc networks", *IEEE Transactions On Wireless Communications*, vol. 3, no. 1, January, 2004

[34] A. E. Gamal and J. Mammen, "Optimal hopping in ad hoc wireless networks", in: *Proceedings of IEEE Infocom 06*, Barcelona, Spain, 2006.

[35] T. H. Cormen, C. E. Leiserson, and R. L. Rivest, *Introduction to Algorithms*, Cambridge, MA: MIT Press, 1991.

[36] Z. Chen, C. Qiao, J. Xu, and T. Lee, "A constant approximation algorithm for interference aware broadcast in wireless networks", in: *Proceedings of IEEE Infocom 07*, 2007.

[37] S. C.-H. Huang, P.-J. Wan, X. Jia, H. Du, and W. Shang, Minimum-Latency Broadcast Scheduling in Wireless Ad Hoc Networks", in: *Proceedings of IEEE Infocom 07*, 2007.

[38] A. J. Goldsmith, and S. B. Wicker, "Challenges for energy-constrained ad hoc wireless networks", *IEEE Wireless Communications*, August, 2002

[39] R. Gandhi, S. Parthasarathy, and A. Mishra,"Minimizing broadcast latency and redundancy in ad hoc networks", in: *Proceedings of the* 4^{th} *ACM International Symposium on Mobile Ad Hoc Networking & Computing*, Annapolis, Maryland, 2003.

[40] J. Y. Lin, and J. C. Hou, "Interplay of spatial reuse and SINR-determined data rates in CSMA/CA-based, multi-hop, multi-rate wireless networks", in: *Proceedings of IEEE Infocom 07*, 2007.

[41] T. S. Kim, H. Lim, and J. C. Hou, "Improving Spatial Reuse through Tuning Transmit Power, Carrier Sense Threshold, and Data Rate in Multihop Wireless

Networks”,in: *Proceedings of Mobicom’06, Los Angeles, California,* September 2326, 2006.

[42] A. O. Lim, and S. Yoshida, “A 2-hop path selection protocol (2PSP) in multi-rate ad hoc wireless networks”, *IEICE Transactions on Communications*, vol. E90-B, no. 1, 2007 * Pp. 42-50

left arrow