Research Article

An Optimal CDS Construction Algorithm with Activity Scheduling in Ad Hoc Networks

Chakradhar Penumalli and Yogesh Palanichamy

Department of Information Science & Technology, Anna University, Chennai 600025, India

Correspondence should be addressed to Chakradhar Penumalli; chakradhari13@gmail.com

Received 18 December 2014; Revised 5 April 2015; Accepted 15 April 2015

Academic Editor: Jaesoo Yoo

Copyright © 2015 C. Penumalli and Y. Palanichamy. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

A new energy efficient optimal Connected Dominating Set (CDS) algorithm with activity scheduling for mobile ad hoc networks (MANETs) is proposed. This algorithm achieves energy efficiency by minimizing the Broadcast Storm Problem [BSP] and at the same time considering the node's remaining energy. The Connected Dominating Set is widely used as a virtual backbone or spine in mobile ad hoc networks [MANETs] or Wireless Sensor Networks [WSN]. The CDS of a graph representing a network has a significant impact on an efficient design of routing protocol in wireless networks. Here the CDS is a distributed algorithm with activity scheduling based on unit disk graph [UDG]. The node's mobility and residual energy (RE) are considered as parameters in the construction of stable optimal energy efficient CDS. The performance is evaluated at various node densities, various transmission ranges, and mobility rates. The theoretical analysis and simulation results of this algorithm are also presented which yield better results.

1. Introduction

Mobile ad hoc networks appear in a wide variety of applications such as the military, disaster relief, surveillance, sensing, and monitoring. Mobile ad hoc network (MANET) is a special kind of wireless network environment. It is different from the traditional wireless networks. Ad hoc network is a collection of autonomous and arbitrarily located wireless nodes. It is an infrastructure less network with dynamic topology, limited link bandwidth, variation in links, node capabilities, and energy-constrained resources. The nodes need to be more intelligent as they have to act as sender, receiver, and intermediate forwarder. Because of dynamic topology they frequently advertise the control information using a simple flooding. Simple flooding leads to Broadcast Storm Problem [1]. Broadcast Storm refers to the problem of exhausting the limited resources of the network due to excessive transmission of packets in the network. Considering this environment, several routing protocols have been proposed in order to find out and maintain the multihop route with reliability.

Generally ad hoc routing protocols in MANETs are classified into two types, namely, reactive and proactive. In proactive (table driven) routing protocols like DSDV [2] and OLSR [3] routing information in the network to update the tables is periodically broadcast. In case of on-demand (reactive) ad hoc routing protocols such as DSR [4] and AODV [5] simple flooding is used in route discovery procedure by broadcasting route request (RREQ) packets to the network. Generally in MANETs flooding is also used for *alarm signals*, *paging*, and so forth. However reactive routing protocols do not involve flooding of data packets but flooding of control packets takes place. Simple flooding consumes energy of the nodes and bandwidth of the network and also leads to congestion in the network which is called Broadcast Storm Problem.

In simple flooding a host on receiving a broadcast message for the first time will rebroadcast the message into network. A straightforward approach of simple flooding in spite of the fact that size of control packets is small suffers from the Broadcast Storm Problem which leads to the following problems [1].

- (1) *Redundant Rebroadcasts*. When a mobile host decides to rebroadcast a broadcast message to its neighbours, all its neighbours already have the message.
- (2) *Contention for Medium*. After a mobile host broadcasts a message, if many of its neighbours decide to rebroadcast the message, then they contend with each other.
- (3) *Collision*. Collisions are more likely to occur due to lack of efficient back-off mechanism.

The effect of the Broadcast Storm Problem can be reduced efficiently and effectively using Connected Dominating Set (CDS). A Connected Dominating Set is constructed assuming the MANET as a graph, generally unit disk graph [6] for better approximation. CDS is analogous to the backbone network of traditional wired networks. The activity of broadcasting is confined to the nodes of CDS. The smallest possible CDS is called Minimum Connected Dominating Set (MCDS) and the problem of constructing MCDS is an NP hard problem. Many standard methods exist in the literature to construct the Minimum Connected Dominating Set [7–23]. Each has its own merits and demerits. Activity scheduling can be used to prolong the lifetime of the network by rotating the role of various nodes [24]. In this paper we propose an energy efficient optimal CDS algorithm with activity scheduling. Scheduling between the similar neighbourhood backbone nodes (dominators) improves the energy efficiency at the same time. Here the activity scheduling is distributed and synchronous which suits much MANETs. It has also less message complexity because only the dominator nodes involve scheduling locally.

The rest of the paper is organized as follows: The definitions, mathematical calculations, and notations used in this paper are given in Section 2. The existing solutions are reviewed in Section 3. The proposed work is discussed in Section 4. The performance analysis and simulation results are shown in Section 5. The conclusion and future work are provided in Section 6.

2. Definitions, Mathematical Calculations, and Notations

Some important definitions, mathematical equations, and the notations used in this work are discussed here.

2.1. Definitions. $G = (V, E)$ is a unit disk graph [6] which represents ad hoc wireless network. “ V ” is the set of mobile nodes in the network and “ E ” represents all the links in the network. We assume that all the nodes are deployed in a 2-D plane and their maximum transmission range is the same initially.

Definition 1. Any node in an UDG is adjacent to at most five independent nodes.

Definition 2. A dominator is a node which rebroadcasts the messages in the network. Generally it is called a BLACK node.

Definition 3. A Dominatee is node which listens to the broadcast message from dominator node. It will not rebroadcast the message. Usually it is called a GREY node.

Definition 4. A dominating set (DS) of a graph $G = (V, E)$ is a subset of “ V ” such that each node in “ V ” is adjacent to at least one node in dominating set.

Definition 5. A Connected Dominating Set (CDS) “ C ” of $G = (V, E)$ is a dominating set of G which induces a connected subgraph of G .

Definition 6. Minimum Connected Dominating Set (MCDS) is a CDS with minimum cardinality.

2.2. Mathematical Calculations for Energy. The energy calculations and equations used in this work are described here. All calculations are in joules. In the network, only the energy of dominator nodes is considered because they consume more energy when compared to the Dominatees. Here dominators consume more power than Dominatees because the set of dominators acts as the backbone. Moreover there are many power consuming roles performed by dominators. First a dominator has to rebroadcast the messages in its transmission range. Second it has to keep track of its connectors to the other dominators. Third it has to maintain a table for list of active neighbours. Fourth it has to perform local repairs. Finally it has to perform activity scheduling with its counterparts. All these indulge in high message exchange and in turn energy consumption.

Moreover the Dominatees are considered to be in promiscuous mode in the case of exchanging control packets alone where they consume very low power to receive the control messages. We considered only the transmission of control packets and not data packets. The Dominatees do not have much role in the construction of the CDS and hence throughout the construction phase Dominatees remain in promiscuous mode. The power consumption at various modes is shown in Table 3. Equations (4), (5), and (6) give a clear picture of how energy is consumed for each bit. When compared to the rate at which energy is depleted in the dominators, the rate at which energy is depleted in the Dominatees is negligible.

In order to increase the lifetime of the dominator as well as network we have performed the activity scheduling to balance the energy consumption. At the same time we can make the network fault tolerant to some extent.

The dominator node’s energy “ E_d ” is a function of the residual energy (RE) and dissipated energy (DE) as given in (1) and the energy dissipation rate (D) given in (2):

$$E_d = f(\text{RE}, \text{DE}). \quad (1)$$

The dissipation rate at a random time

$$D_i(t) = \frac{(E_{di(0)} - E_{di(t)})}{t}, \quad (2)$$

where $E_{di(0)}$ = energy of node i at time $T = 0$ and $E_{di(t)}$ = energy of node i at time $T = t$.

Let the initial energy of the dominator be " $E_{\max}$." Once the dominator starts functioning, the energy is consumed and the available energy is reduced. The energy available at a particular instant of time is called residual energy (RE). Let " s " be the initial duration for which the dominator has functioned. After the time duration " s " the residual energy available at the dominator is computed as in

$$RE_d = E_{\max} - DE_s, \quad (3)$$

where " RE_d " is the residual energy of the dominator and " DE_s " is the energy dissipated during duration s .

The dissipated energy DE_s is calculated as follows.

Let " B_t " be the energy required to transmit a bit and let " B_r " be the energy required to receive a bit. Let " B_p " be the energy consumed during the promiscuous mode of operation in one time unit. " B_r " is nearly half of " B_t " and " B_p " is one-tenth of " B_t ." Let " m^1 " be the number of bits transmitted during " s " and let " n^1 " be the number of bits received during " s ." The depleted energy during duration " s " is computed as in

$$DE_s = m^1 * B_t + n^1 * B_r + s * B_p. \quad (4)$$

For the subsequent time intervals of duration " s " the available energy is computed as in

$$\begin{aligned} E_{\text{cur}} &= RE_d, \\ RE_d &= E_{\text{cur}} - DE_s, \end{aligned} \quad (5)$$

where E_{cur} is the residual energy available at a particular instant in time.

The overall remaining energy of the backbone network is computed as in

$$RE_n = \sum_{i=0}^n (E_{\text{cur}} - DE_s), \quad (6)$$

where RE_n = The overall residual energy of the backbone network.

RE_d is calculated at regular intervals, if this ratio for the dominator becomes less than the threshold (TH) then it will choose another equivalent node and performs activity scheduling. Here in this algorithm we have put TH as 10% of $E_{\max}$ because of energy requirement. We have simulated the network with various values of threshold TH (5%, 10%, 15%, and 20%) of $E_{\max}$ and observed that when TH = 10% of $E_{\max}$ maintains the tradeoff between network lifetime and message cost. When TH = 20% of $E_{\max}$ then frequent disconnections occur which result in high message cost and low network lifetime. When TH = 5% of $E_{\max}$ then network lifetime increases but the dominator node will become unavailable because the node's energy will completely drain out. The results are shown in Figure 15. Here the network is simulated with 200 nodes at various transmission ranges.

So threshold is kept as 10% of $E_{\max}$.

That is,

$$TH = 0.1 (E_{\max}). \quad (7)$$

RE ratio is given by

$$RE \text{ (ratio)} = \frac{(E_{\max} - RE_{i-1})}{E_{\max}}. \quad (8)$$

Generally, $0 < RE(\text{ratio}) < 1$.

2.3. Notations. The notations used in this work are given in the Notations section:

Δ : the degree of the node that has highest number of neighbours in the graph,

H : harmonic function,

BigO: the Asymptotic upper bound of a function,

Ω : the Asymptotic lower bound of a function,

θ : tight bound, that is, both upper and lower bounds are of the same order,

G : unit disk graph,

V : number of vertices (nodes),

E : number of edges (links),

E_d : energy of a dominator,

B_t : transmission energy of dominator,

B_r : receiving energy of dominator,

B_p : energy required in promiscuous mode,

N_1 : 1-hop neighbors,

N_{i1} : 1-hop neighbor list of node i , where $i = 1, 2, \dots, n$,

ΔT : random time interval,

δT : small time interval used for activity scheduling,

Φ : null set (or) empty set,

n : number of nodes in the network,

s : time duration,

RE_d : the residual energy of the dominator,

E_{cur} : current residual energy,

DE_s : the energy dissipated during duration " s ,"

|OPT|: the optimal value,

|MCDS|: size of MCDS.

3. Literature Survey

3.1. Existing Solutions. There are various solutions to alleviate this Broadcast Storm Problem [1]. Some important solutions are probability based scheme, area based scheme, counter based scheme, neighbor knowledge based scheme, and CDS based scheme [1, 7–23, 25–27]. However except CDS based scheme none of these schemes are able to form the virtual backbone network. Connected Dominating Set (CDS) [7–23] belongs to graph based method. It can be used as a virtual backbone or spine of wireless ad hoc networks. CDS provides efficiency not only in broadcasting but also in multicasting and power management. Most of the above CDS algorithms aimed at small size of CDS but not considered

the lifetime of CDS. Our algorithm considers both the size of CDS and lifetime of CDS. Generally they are classified into global, quasi-global, local, quasi-local ones based on the type of information it gathers. Generally the CDS construction algorithms are classified into two types: centralized and distributed.

3.1.1. Centralized CDS Algorithms. Centralized algorithms require entire network topology (global) of the MANET. They produce small sized CDS and better performance ratio when compared to distributed CDS algorithms. But they are prone to the problems caused by the mobility of nodes. Guha and Khuller [10] propose two polynomial time algorithms to construct a CDS in a general graph G . These algorithms are greedy and centralized. The first one has an approximation ratio of $2(H(\Delta) + 1)$, where “ H ” is a harmonic function and “ Δ ” is the degree of the node that has the highest number of neighbours in G . Here a spanning tree is developed with maximum degree node as root. The nonleaf nodes in the tree forms the CDS. The second algorithm constructs a Steiner tree that connects all dominator (BLACK) nodes to form CDS. The size of CDS is at most $(\ln(\Delta) + 3)|OPT|$, where $|OPT|$ is the size of an optimal MCDS. The message complexity is $O(n^2)$ and time complexity is also $O(n^2)$. Alzoubi et al. [14] propose a one-step greedy approximation algorithm with performance ratio at most $3 + \ln(\Delta)$. Butenko et al. [16] propose a two-phase algorithm; in first phase Maximum Independent Set (MIS) is constructed and second phase forms the CDS based on Steiner tree with minimum number of Steiner nodes. It has a performance ratio of $6.8|MCDS|$. Cheng et al. [11] proposed a three-phase algorithm (centralized) to construct MCDS in UDG. It has time complexity of $O(n)$ and message complexity of $O(n)$. Stojmenovic et al. [17] proposed a pruning based CDS construction. It has time complexity of $O(|V| \cdot |E|)$ and message complexity of $O(n^2 \log^3 n)$. In centralized solutions mobility is a concern. Due to mobility topology changes which reflect in the frequent update cost. In our work we follow a distributed approach for the construction of the CDS.

3.1.2. Distributed CDS Algorithms. CDS is constructed based on localized information. These algorithms are further classified into prune based and MIS based ones. Wan et al. [8] proposed a distributed algorithm based on MIS this algorithm has time complexity of $O(n)$ and message complexity of $O(n \log(n))$. The resulting CDS has a size of at most $8|OPT| + 1$. Wu and Li's algorithm [7] is very simple. The localized property makes the CDS maintenance easier. The algorithm has a linear performance ratio. This algorithm needs at least two-hop neighbourhood information. It is presented based on the general graph model and it has time complexity of $\theta(m)$ and message complexity of $O(\Delta^3)$. They analysed the movement of nodes with an on-model or off-model as leaving from one small management domain and entering another management domain. Das and Bharghavan [12] propose a three-staged algorithm; first stage identifies dominating set, second stage constructs spanning forest, and third stage constructs the spanning tree. It has a time complexity of

$O(n^2)$, message complexity of $O(n^2)$, and cardinality is at most $3H(\Delta)$. Stojmenovic et al. [17] propose a cluster based CDS construction which is based on 2-hop neighbourhood knowledge. Here ranking is based on degree and location of the node. It has time complexity of $\Omega(m)$ where “ m ” is number of edges and message complexity of $O(n^2)$. Khabbazi et al. [22] proposed a local broadcast algorithm based on positional information to construct CDS. It has not focused much on mobility and energy. Sakai et al. [21] proposed a timer based CDS algorithm: two versions, namely, the single initiator CDS algorithm and multi-initiator CDS algorithm. Single initiator algorithm requires less time to construct the small size CDS when compared to multi-initiator. The message complexity for the multi-initiator is very high, but it has not concentrated much on the energy. Li et al. [18] proposed an algorithm with two phases. At the first phase, a Maximal Independent Set (MIS) is formed. At the second phase, a Steiner tree algorithm is used to connect the MIS. Cheng et al. [11] proposed a distributed multileader algorithm. It selects the node with minimum ID in its 1-hop neighbourhood as a leader, builds a tree rooted at each leader, and connect two adjacent trees through one or two nodes. It has a message complexity of $O(n \log n)$ and time complexity of $O(n)$. Tang et al. [28] proposed an energy efficient MCDS algorithm which considered the energy before constructing the MCDS algorithm. It has not much concentrated on mobility and also it is based on reduced neighbour set. Here two stages are used one is CDS construction stage and the second is pruning the CDS to MCDS. It has not concentrated on mobility. Leu and Chang [20] proposed a distributed algorithm for Relaying Mobile Node (RMN) with neighbour table (N) and finally formed a CDS. Most of the above distributed algorithms concentrated either on mobility or on energy but not on both. The complexities of some important basic algorithms are detailed in Table 1.

3.2. Activity Scheduling. The lifetime of a CDS can be significantly increased by allowing some of the neighbouring dominators to sleep intermittently. Activity scheduling must fulfill two requirements: connectivity and coverage and minimization of the energy utilization. There are several sleep scheduling algorithms existing in the literature [24, 29]. These scheduling algorithms are classified into centralized and distributed ones. Distributed scheduling suits mobile ad hoc networks since MANETs lack infrastructure and centralized authority. As in the literature these activity scheduling schemes are further classified into scheduled wakeups and on-demand wakeups [30]. Miller and Vaidya [31] proposed a MAC layer activity scheduling algorithm that uses idle time to sense the channel. Sumi et al. [32] proposed an rotation based activity scheduling for wireless routing protocols where the roles of active and sleep are rotated among the MIS (maximal independent sets) based on power level priority. Kumar et al. [33] proposed a self-scheduling randomized independent scheduling (RIS) which uses synchronized probabilistic method which takes QOS as the metric. Berman et al. [34] proposed scheduling algorithm based on maximization problem to achieve K -coverage. In

- (1) Every node exchanges its unique ID and Residual Energy RE value (in case of reconstruction) with in its transmission range. Obtains 1-hop neighbor list, N_1 .
- (2) Every node again exchanges the list N_1 to obtain the 2-hop neighbor list which also includes the RE energy values
- (3) A node with highest number of neighbors and lowest ID is elected as dominator initially.
- (4) Now the dominator changes its color to BLACK and all its neighbors turn to GREY and broadcast this information
- (5) IF an uncovered WHITE node receives this information from a GREY neighbor
- (6) THEN it converts itself to GREY and its GREY neighbor to BLACK
- (7) This procedure is repeated until no WHITE node exist

ALGORITHM 1

TABLE 1: Complexities of various CDS algorithms.

CDS algorithm	Classification	Time complexity	Message complexity
Guha & Khuller	Centralized	$O(n^2)$	$O(n^2)$
Cheng et al.	Centralized	$O(n)$	$O(n)$
Butenko et al.	Centralized	$O(V \cdot E)$	$O(n^2 \log^3 n)$
Alzoubi et al.	Distributed	$O(n)$	$O(n \log(n))$
Wu & Li	Distributed	$\theta(m)$	$O(\Delta^3)$
Das et al.	Distributed	$O(n^2)$	$O(n^2)$
Stojmenovic et al.	Distributed	$\Omega(m)$	$O(n^2)$
Dai et al.	Distributed	$O(\Delta^2)$	$O(\Delta)$

much of the above scheduling algorithms almost all nodes involve activity scheduling which result in high message complexity.

Here we have designed an activity scheduling that suits MANETs and also the scheduling happens between dominators only. At a time dominator node may be in active mode (WAKEUP) or promiscuous mode (SLEEP). The dominating node in promiscuous mode sniffs the channel for wakeup signal or sleep signal at regular intervals. Moreover, the proposed activity scheduling algorithm in this paper also considers energy of the dominators.

4. Proposed Work

The proposed energy efficient optimal CDS algorithm with activity scheduling is a distributed algorithm. All nodes are assumed to be of equal energy and equal transmission range initially. Our work considers both energy and mobility of the nodes in the construction of MCDS. In addition, we employ activity scheduling algorithm to spend the energy in a more economical way. The actual algorithm is divided into five phases. Phase 1 deals with dominator election. Phase 2 is about obtaining the connectors across the network and forming redundant CDS. Phases 3 and 4 deal with the construction of optimal CDS algorithm along with activity scheduling among dominators to increase the lifetime of dominators and in turn increase the lifetime of the network. Activity scheduling performs localized scheduling among different dominators with same set of N_1 list so that the energy of

the dominators sustain. Phase 5 is about the maintenance of the CDS at frequent intervals. Generally the CDS is disturbed when a dominator moves away or a new node joins the network or the energy and transmission range of the existing dominator goes below the threshold (TH) value. Here the residual energy and the transmission range of a dominator is calculated at regular intervals because initially all nodes have equal energy. In later states the nodes energy reduces according to (1), (2), (3), and (4).

Algorithm Phase 1. This phase is the initial phase, executed distributively over the network. Initially all nodes are in WHITE colour and have equal energy and transmission range. So, residual energy (RE) has no role in selecting the dominating set but in later stages of reconstruction/maintenance of CDS, RE plays an important role in selecting energy efficient CDS algorithm (see Algorithm 1).

This phase is illustrated with sample MANET topology as shown in Figure 1. After the exchange of IDs and RE (initially not counted) as shown in Figure 2, the nodes with the highest number of neighbors and lowest IDs form as dominating set, DS (BLACK nodes) = {5, 13, 18} as shown in Figure 3. After this algorithm, phase 2 is implemented.

Algorithm Phase 2. This is the connection establishment phase. That is, connectors are selected to connect the DS and to form CDS (see Algorithm 2).

This phase is illustrated here in the example MANET; the GREY nodes with two different BLACK nodes are node 1, node 17, and node 30. Now nodes 1, 17, and 30 will turn to BLACK as shown in Figure 3 and connections are established. After this phase CDS is formed as shown in Figure 4. In the next phase equivalent counterparts are selected locally by these dominators for activity scheduling.

Algorithm Phases 3 and 4. This phase plays a crucial role where it prunes the previous CDS to obtain optimal CDS and it performs activity scheduling (Phase 4) among different adjacent dominators (BLACK nodes) with at most the same N_1 list to conserve the dominator's energy. Activity scheduling shifts the role of dominators at intervals, δT and their residual energies RE_s are also calculated as shown in (3). That is, the equivalent adjacent dominators turn on (WAKEUP) and off (SLEEP) alternatively to conserve the energy and increase the lifetime without compromising the delivery ratio. Here a new regular interval based activity scheduling algorithm is

- (1) Every GREY node checks its neighbor list N_1 for more than one BLACK node.
- (2) **IF** (Number of BLACK nodes >1)
- (3) **THEN** convert its color to BLACK (connector) and exchange this information
- (4) **ELSE**
- (5) Two nearest GREY nodes of different BLACK nodes turns themselves as BLACK to establish the connection (these are connectors)

ALGORITHM 2

FIGURE 1: Initial example MANET.

FIGURE 3: MANET after algorithm phase 1 with DS.

FIGURE 2: Initial MANET with ID exchange.

FIGURE 4: Connectors are selected after algorithm phase 2 and (ID, RE, and N_1) are exchanged to get equivalent node for performing activity scheduling.

designed to suit the mobility of nodes. Here we maintained a trade-off between optimal CDS and number of dominators to perform the activity scheduling (see Algorithms 3 and 4).

These phases are illustrated in Figures 4 and 5. According to the algorithm, all the dominators (BLACK) select the counterparts locally and perform activity scheduling. Here in the MANET node 5 selects node 2 as the counterpart and node 1 selects the union of node 8 and node 24 as

counterparts, because their combined N_1 list is equivalent to the neighbour list of node 1. Similarly node 17 and node 30 are counterparts. Node 18 selects union of node 25 and node 23 as the counterparts. Node 13 has no equivalent node or union of nodes with similar N_1 list; in this case node 13 will not perform activity scheduling locally until an equivalent node/nodes join. When compared to other nodes node 13 drains out quickly. These cases are handled by maintenance phase. Activity scheduling is performed as shown

- (1) Every BLACK node exchanges its N_i list in its transmission range
- (2) All its neighbors exchange their N_{i1} list, where $i = 1, 2, \dots, n$
- (3) Every BLACK node performs intersection of N_i list with its own N_{i1} . That is, $(N_i \cap N_{i1})$
- (4) Nodes with at most similar N_i , that is, $(N_i \cap N_{i1} \cong \Phi)$ and $(RE > TH)$ are selected and turned BLACK
- (5) Perform **Activity scheduling** (phase 4) between neighboring BLACK nodes at regular time interval δT
- (6) Black nodes also calculate RE at regular intervals.
- (7) **IF** $(RE \leq TH)$
- (8) **THEN**
- (9) Repeat steps (1), (2), (3), (4) & (5)
- (10) **ELSE**
- (11) Continue

ALGORITHM 3

- (1) Initially all the Adjacent dominators are in active mode
- (2) Dominator 1 turns itself into WAKEUP, sends SLEEP message to the counterparts and starts the timer TI.
- (3) After receiving the SLEEP message the adjacent dominators start their own timer with a higher value and goes in to Promiscuous mode.
- (4) In promiscuous mode dominators listen to the messages
- (5) **WHILE** $(RE \geq TH)$
- (6) **BEGIN**
- (7) **IF** (Message = WAKEUP or Timer = 0)
- (8) **THEN**
- (9) Turns itself to WAKEUP and sends SLEEP message to its counter parts
// this is to balance the energy among the nodes
- (10) **END**

ALGORITHM 4

FIGURE 5: CDS with activity scheduling.

in Figure 6. Nodes $\{(2, 5), (1, (8, 24)), (17, 30), (18, (23, 25))\}$ perform activity scheduling locally. BLACK slashed lines (active connection) and RED slashed lines (inactive connection) indicate the scheduling. When compared to other

algorithms the dominators will not run out of energy and lifetime of the network increases as shown in Figure 5.

Algorithm Phase 5. This phase is called maintenance/repair phase. Here the mobility of the node and the residual energy (RE) are considered for maintaining the optimal CDS. Here four cases are considered when a new node joins the network, existing node moves away, existing dominator moves away, and existing dominator runs out of energy. Every dominator executes this phase at frequent ΔT time intervals. A BLACK (dominator) node turns itself to RED when its energy becomes less than threshold TH; that is, it drains out (see Algorithm 5).

The above phase is illustrated in Figure 6. Here the dominator node 17 moved away so counterpart node 30 has become the BLACK node as shown in Figure 7. Node 5 has drained out and turned itself into RED and exchanged locally. So node 2 has become the dominator as shown in Figure 7. Two new nodes 29 and 31 have joined the network, where node 29 is nearer to the nodes 23 and 25. Node 31 joined network near nodes 11 and 12; now IDs are exchanged locally. Now node 12 will become the additional new dominator; as node 29 is already adjacent to two dominators, it turns itself into GREY as shown in Figure 7. The final optimal repaired CDS $\{(2, (1, (8, 24))), 13, 12, 30, (18, (23, 25))\}$ is shown in Figure 7. Here only the nodes $\{(1, (8, 24)), (18, (23, 25))\}$ perform the activity scheduling.

TABLE 2: Simulation parameters.

Simulator	Ns2 (version 2.34)
Simulation area	1000 × 1000 m
Propagation	Two-ray ground reflection
MAC protocol	IEEE 802.11
Bandwidth	2 Mbps
Traffic	CBR
Transmission range	25~100 meters
Number of nodes	100~450
Maximum speed	5~25 m/sec
Mobility model	Random walk
Broadcast sessions	35
Broadcast rate	0.5~5 pkts/s
Message size	128 bytes
Hello interval	2 sec
Simulation time	60~100 minutes
Number of trials	75

TABLE 3: Power consumption table.

Parameter (node)	Power consumption (Watts)
Transmission	1000 mW
Receiving	500 mW
Promiscuous mode	100 mW

mobility model is used. The terrain area is about 1000 m × 1000 m. Figure 8 shows the size of CDS versus number of nodes in sparse network with node's transmission range of 25 mts, at node speed of 5 mts/s. Figure 9 shows the size of CDS versus number of nodes in sparse network with node's transmission range of 25 mts, at a node mobility of 10 mts/s. In both cases our algorithm performed well irrespective of speed and obtained optimal CDS size in sparse network when compared to the existing CDS algorithms. Figure 10 shows the size of CDS versus number of nodes in dense network with node's transmission range of 50 mts, at a node mobility of 5 mts/s. Figure 11 shows the size of CDS versus number of nodes in dense network with node's transmission range of 50 mts, at node mobility rate of 10 mts/s. In these cases also our algorithm performed well when compared to other algorithms because of its distributed nature. Figure 12 shows the message overhead versus number of nodes with node's transmission range of 50 mts and speed 10 mts/s. Here our algorithm exchanged optimal number of messages when compared to the existing standard CDS algorithms, where the existing CDS algorithms have not performed activity scheduling, so the CDS repair phase message cost increased. Figure 13 shows the delivery ratio versus number of nodes with transmission range of 25 mts and speed of 10 mts/s. Here also our algorithm performed well when compared to other algorithms. The graph in Figure 14 shows the lifetime

FIGURE 8: Number of nodes versus CDS size (transmission range = 25 mts and speed = 5 mts/s).

FIGURE 9: Number of nodes versus CDS size (transmission range = 25 mts and speed = 10 mts/s).

of the optimal CDS versus mobility of nodes with node's transmission range of 50 mts. The lifetime of our algorithm is observed at different speeds of nodes 5 mts/s, 10 mts/s, 15 mts/s, and 20 mts/s. The lifetime of our algorithm in case of mobility is also good when compared to the existing standard CDS algorithms because of activity scheduling among neighbouring dominators in phases 3 and 4. So our algorithm is robust, reliable, and fault tolerant. The graph in Figure 15 shows the lifetime of the optimal CDS at various energy threshold values. Here the network is simulated with

FIGURE 10: Number of nodes versus CDS size (transmission range = 50 mts and speed = 5 mts/s).

FIGURE 11: Number of nodes versus CDS size (transmission range = 50 mts and speed = 10 mts/s).

200 nodes at various transmission ranges of 50 mts, 25 mts, and 10 mts and at a speed of 5 mts/s with threshold values of 0.05, 0.1, 0.15, and 0.2. The value of threshold at 0.1 gives better results and maintains the tradeoff between network lifetime and message cost.

6. Conclusion and Future Work

The major advantage of using activity scheduling algorithm is fast convergence and longer life even if CDS nodes move

FIGURE 12: Number of nodes versus message overhead.

FIGURE 13: Number of nodes versus delivery ratio (%).

out due to mobility or energy drain. Here the energy of the dominators will not drain out due to activity scheduling. Only the dominator nodes are involved in further communication to establish the backbone and in maintenance of optimal CDS. In future work we will extend this to directed graphs and nodes with directional antennas.

Disclaimer

The authors alone are responsible for the content and writing of the paper.

FIGURE 14: Mobility of nodes (mts/s) versus MCDS lifetime (min).

FIGURE 15: Lifetime of CDS (min) versus energy threshold values.

Conflict of Interests

The authors report that there is no conflict of interests regarding the publication of this paper.

Acknowledgment

The authors express their sincere thanks to Anna University for providing the valuable resources to perform the experiments.

References

- [1] S. Ni, Y. Tseng, Y. Chen, and J. Sheu, "The broadcast storm problem in a mobile ad hoc network," in *Proceedings of the 5th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom '99)*, pp. 151–162, August 1999.
- [2] C. E. Perkins and P. Bhagwat, "Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers," in *Proceedings of the Conference on Communications Architectures, Protocols and Applications (SIGCOMM '94)*, pp. 234–244, 1994.
- [3] P. Jacquet, T. Muhlethaler, T. Clausen, A. Laouiti, A. Qayyum, and L. Viennot, "Optimized link state routing protocol for ad hoc networks," Hipercom Project, INRIA Roquencourt, Le Chesnay Cedex, France, 2003.
- [4] D. B. Johnson and D. A. Maltz, "Dynamic source routing in ad hoc wireless networks," in *Mobile Computing*, vol. 353 of *The Kluwer International Series in Engineering and Computer Science*, pp. 153–181, Kluwer Academic Publishers, 1996.
- [5] C. E. Perkins and E. M. Royer, "Ad-hoc on-demand distance vector routing," in *Proceedings of the 2nd IEEE Workshop on Mobile Computing Systems and Applications (WMCSA '99)*, pp. 90–100, February 1999.
- [6] B. N. Clark, C. J. Colbourn, and D. S. Johnson, "Unit disk graphs," *Discrete Mathematics*, vol. 86, no. 1–3, pp. 165–177, 1990.
- [7] J. Wu and H. Li, "On calculating connected dominating set for efficient routing in ad hoc wireless networks," in *Proceedings of the 3rd International Workshop on Discrete Algorithms and Methods for Mobile Computing and Communications*, pp. 7–14, 1999.
- [8] P.-J. Wan, K. M. Alzoubi, and O. Frieder, "Distributed construction of connected dominating set in wireless ad hoc networks," in *Proceedings of IEEE Conference on Computer Communications (INFOCOM '02)*, pp. 1597–1604, June 2002.
- [9] F. Dai and J. Wu, "An extended localized algorithm for connected dominating set formation in ad hoc wireless networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 15, no. 10, pp. 908–920, 2004.
- [10] S. Guha and S. Khuller, "Approximation algorithms for connected dominating sets," *Algorithmica*, vol. 20, no. 4, pp. 374–387, 1998.
- [11] X. Cheng, M. Ding, and D. Chen, "An approximation algorithm for connected dominating set in ad hoc networks," in *Proceedings of the 1st International Workshop on Theoretical Aspects of Wireless Ad Hoc, Sensor, and Peer-to-Peer Networks*, 2004.
- [12] B. Das and V. Bharghavan, "Routing in ad-hoc networks using minimum connected dominating sets," in *Proceedings of the IEEE International Conference on Communications (ICC '97)*, pp. 376–380, Montreal, Canada, June 1997.
- [13] L. Ruan, H. Du, X. Jia, W. Wu, Y. Li, and K.-I. Ko, "A greedy approximation for minimum connected dominating sets," *Theoretical Computer Science*, vol. 329, no. 1–3, pp. 325–330, 2004.
- [14] K. M. Alzoubi, P. J. Wan, and O. Frieder, "Message-optimal connected dominating sets in mobile ad hoc networks," in *Proceedings of the 3rd ACM International Symposium on Mobile Ad Hoc Networking and Computing (MOBIHOC '02)*, pp. 157–164, EPFL, June 2002.
- [15] M. Min, H. Du, X. Jia, C. X. Huang, S. C. Huang, and W. Wu, "Improving construction for connected dominating set

- with Steiner tree in wireless sensor networks,” *Journal of Global Optimization*, vol. 35, no. 1, pp. 111–119, 2006.
- [16] S. Butenko, X. Cheng, C. Oliveira, and P. M. Pardalos, “A new heuristic for the minimum connected dominating set problem on ad hoc wireless networks,” in *Recent Developments in Cooperative Control and Optimization*, pp. 61–73, Kluwer Academic Publishers, 2004.
 - [17] I. Stojmenovic, M. Seddigh, and J. Zunic, “Dominating sets and neighbor elimination based on broadcasting algorithms in wireless networks,” in *Proceedings of the IEEE Hawaii International Conference on System Sciences (HICSS ’01)*, IEEE, January 2001.
 - [18] Y. Li, M. T. Thai, F. Wang, C.-W. Yi, P.-J. Wan, and D.-Z. Du, “On greedy construction of connected dominating sets in wireless networks,” *Wireless Communications and Mobile Computing*, vol. 5, no. 8, pp. 927–932, 2005.
 - [19] J. Yu, N. Wang, G. Wang, and D. Yu, “Connected dominating sets in wireless ad hoc and sensor networks—a comprehensive survey,” *Computer Communications*, vol. 36, no. 2, pp. 121–134, 2013.
 - [20] S. Leu and R.-S. Chang, “Simple algorithm for solving broadcast storm in mobile ad hoc network,” *IET Communications*, vol. 5, no. 16, pp. 2356–2363, 2011.
 - [21] K. Sakai, S. C.-H. Huang, W.-S. Ku, M.-T. Sun, and X. Cheng, “Timer-based CDS construction in wireless ad hoc networks,” *IEEE Transactions on Mobile Computing*, vol. 10, no. 10, pp. 1388–1402, 2011.
 - [22] M. Khabbazi, I. F. Blake, and V. K. Bhargava, “Local broadcast algorithms in wireless ad hoc networks: reducing the number of transmissions,” *IEEE Transactions on Mobile Computing*, vol. 11, no. 3, pp. 402–413, 2012.
 - [23] J. Blum, M. Ding, A. Thaler, and X. Cheng, “Connected dominating set in sensor networks and MANETs,” in *Handbook of Combinatorial Optimization*, D.-Z. Du and P. Pardalos, Eds., pp. 329–369, Springer, 2004.
 - [24] T. Yardibi and E. Karasan, “A distributed activity scheduling algorithm for wireless sensor networks with partial coverage,” *Wireless Networks*, vol. 15, no. 6, pp. 213–225, 2009.
 - [25] B. Williams and T. Camp, “Comparison of broadcasting techniques for mobile ad-hoc networks,” in *Proceedings of the 3rd ACM International Symposium on Mobile Ad Hoc Networking & Computing (MobiHoc ’02)*, pp. 194–205, Lausanne, Switzerland, 2002.
 - [26] Q. Zhang and D. P. Agrawal, “Dynamic probabilistic broadcasting in MANETs,” *Journal of Parallel and Distributed Computing*, vol. 65, no. 2, pp. 220–233, 2005.
 - [27] Y. Sasson, D. Cavin, and A. Schiper, “Probabilistic broadcast for flooding in wireless mobile ad hoc networks,” Tech. Rep. IC/2002/54, Swiss Federal Institute of Technology, 2002.
 - [28] Q. Tang, K. Yang, P. Li, J. Zhang, Y. Luo, and B. Xiong, “An energy efficient MCDS construction algorithm for wireless sensor networks,” *EURASIP Journal on Wireless Communications and Networking*, vol. 2012, article 83, 2012.
 - [29] L. Wang and Y. Xiao, “A survey of energy-efficient scheduling mechanisms in sensor networks,” *Mobile Networks and Applications*, vol. 11, no. 5, pp. 723–740, 2006.
 - [30] A. Keshavarzian, H. Lee, L. Venkatraman, K. Chitalapudi, D. Lal, and B. Srinivasan, “Wake up scheduling in wireless sensor networks,” in *Proceedings of the 7th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MOBIHOC ’06)*, pp. 322–333, May 2006.
 - [31] M. J. Miller and N. H. Vaidya, “A MAC protocol to reduce sensor network energy consumption using a wakeup radio,” *IEEE Transactions on Mobile Computing*, vol. 4, no. 3, pp. 228–242, 2005.
 - [32] M. R. A. Sumi, M. H. Imtiaz, S. M. M. Al Mamun, and M. N. Hassan, “A new approach for designing and analysis of distributed routing algorithm and protocols of wireless ad hoc network,” *International Journal of Science and Advanced Technology*, vol. 1, no. 6, 2011.
 - [33] S. Kumar, T. H. Lai, and J. Balogh, “On k-coverage in a mostly sleeping sensor network,” in *Proceedings of the 10th Annual International Conference on Mobile Computing and Networking (MobiCom ’04)*, pp. 144–158, Pennsylvania, Philadelphia, Pa, USA, October 2004.
 - [34] P. Berman, G. Calinescu, C. Shah, and A. Zelikovsky, “An efficient energy management in sensor networks,” in *Ad Hoc and Sensor Networks*, Y. Xiao and Y. Pan, Eds., Nova Science, 2005.