RLCH: Reliable Link lifetime based Cluster Head election in MANET

by

Dongsheng Chen

A thesis submitted to the Graduate Faculty of
Auburn University
in partial fulfillment of the
requirements for the Degree of
Master of Science

Auburn, Alabama
August 4, 2012

Keywords: Wireless Ad Hoc Networks, Cluster Head Election, Mobility, Link Lifetime, Cluster Update, and Stability

Approved by

Prathima Agrawal, Chair, Samuel Ginn Distinguished Professor of Electrical and Computer Engineering
Shiwen Mao, Associate Professor of Electrical and Computer Engineering
Stanley Reeves, Professor of Electrical and Computer Engineering

Abstract

Clustering has been widely considered in wireless ad hoc networks as a useful mechanism for solving the scalability problems, improving the stability, increasing the network's energy efficiency, etc. The idea behind clustering is to group a flat network into a number of clusters. Cluster heads serve as backbones to establish communication routes between sources and destinations in a mobile ad hoc network (MANET). However, election of a Cluster Head (CH) within a local cluster structure is very important in setting up relatively stable cluster structure. This thesis proposes an efficient method to choose cluster heads in wireless ad hoc networks for improving the network stability and energy efficiency. Considering basic requirements in wireless ad hoc network, we propose a novel reliable link lifetime based cluster head election (RLCH) method to determine which nodes are more qualified to serve as cluster heads.

RLCH takes many factors affecting cluster head election into consideration and, unlike the previous approaches proposed in the literature, employs no weight combination when considering the impact of several factors on choosing a CH. In addition, a link lifetime calculation model and a maximum cluster updating interval model are proposed. An adaptive cluster update interval selection process is developed based on an energy threshold approach for prolonging the network lifetime. Four metrics including average number of clusters, cluster update rate, member exchange rate, and energy consumption distribution are evaluated in our simulations. By comparing with other well-known cluster head election algorithms, our

simulations indicate that RLCH can achieve better stability, energy saving and longer network lifetime in most cases.

Acknowledgments

First and foremost, please allow me to offer my sincerest gratitude to my supervisor, Professor Prathima Agrawal, who supported me throughout my thesis with her patience, encouragement and support. When I came to Auburn University, Dr. Agrawal gave me the chance to purse a Master's degree with her and always encouraged me to keep learning with great patience. It really means a lot in my life. Without her great kindness and patience, this thesis would not be possible.

I am heartily thankful to Professor Shiwen Mao, who triggers me to ponder over the objective during my master's degree when I arrived at Auburn two years ago. It is the first class I took in Auburn University. Also, I wish to express my sincere thanks to Professor Stanley Reeves for serving on my advisory committee. Additionally I want to say thanks to Professor Ming Liao for his help. Finally my thanks also go out to my friend, Dr. Alireza Babaei, Yingsong Huang and Qing Yang, for helping me with my research progress and comments about thesis writing.

Table of Contents

List of Tables

List of Figures

List of Abbreviations

MANETs	Mobile Ad Hoc Networks
CN	Cellular Network
WSNs	Wireless Sensor Networks
QoS	Quality of Services
ODMRP	On Demand Multicast Routing Protocol
PDR	Packet Delivery Ratio
LID	Lowest ID algorithm
HD	Highest ID algorithm
CH	Cluster Head
K-CONID	K-hop connectivity ID clustering algorithm
LCC	Least Cluster Change algorithm
CM	Cluster Member
WCA	Weights-Combination Clustering
WBACA	Weight-based adaptively clustering algorithm
GPS	Global Positioning System
CEMCA	Connectivity Energy & Mobility driven weighted Clustering algorithm
HD	Dominating Set
HEED	Hybrid Energy Efficient Distributed clustering algorithm
TCP	Transport Control Protocol

IP	Internet Protocol
RLCH	Reliable Link lifetime based Cluster Head election

CHAPTER 1

Introduction

1.1 Overview of Wireless Ad Hoc Networks

Mobile ad hoc networks (MANETs) have received considerable attention in research and industry due to their flexible deployment, independence from existing infrastructure and self-configuration, self-healing and self-maintaining properties. In MANETs, mobile nodes can conduct wireless communication without relying on pre-existing network infrastructure like Cellular Networks (CN). Nodes are allowed dynamically to join and leave, frequently, often without warning. Finally, nodes in MANETs can be highly mobile which results in frequently and rapidly changing node constellation, location and link lifetime.

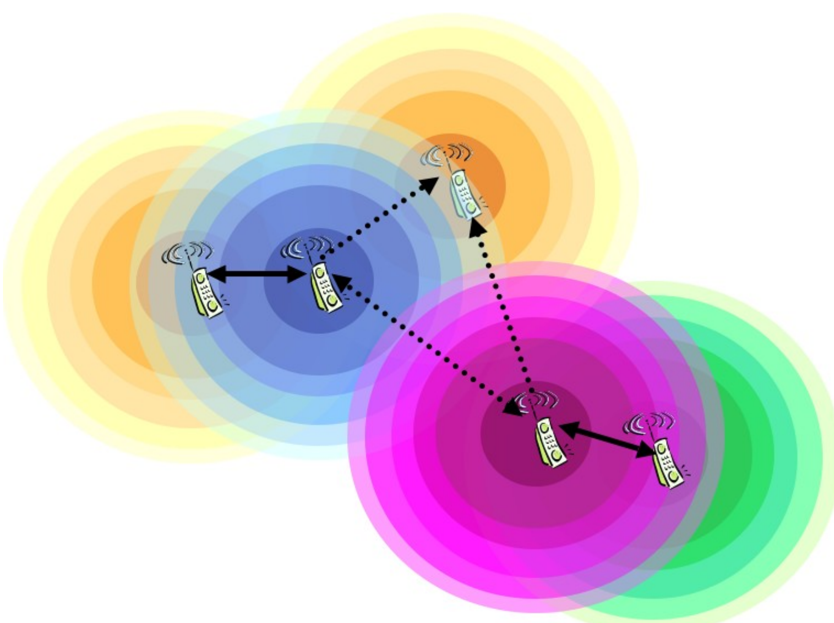

Figure 1.1 Example of Wireless Ad Hoc Network

Figure 1.1 is an example of a MANET, in which mobile users can directly communicate with each other without infrastructure. And some examples of the use of MANETs are list as follows:

- **Security Scenarios**: like communication in urgent scenarios like natural disaster or a global war when fixed infrastructures are unable to work;
- **Tactical Operation**: for fast establishment of military communication in unknown or hostile terrains;
- **Rescue Missions**: provide sufficient wireless coverage when wireless coverage is inadequate;
- **Sensor Networks**: provide communication between intelligent sensors to gather surrounding information.

MANETs are intended to set up a data network, which can adapt to the use of different scenarios and is responsive to change in network topology. Because ad-hoc networks are intended to be deployed anywhere, infrastructure may not be present. Varied mobility patterns and different radio propagation conditions result in intermittent and sporadic connectivity between neighboring nodes. In MANETs, nodes can communicate with each other directly or by multi-hop. In a multi-hop routing protocol, a packet destined to certain nodes is forwarded from intermediate nodes to a specific destination. Of course, appropriate routing determines the relative stability of route, which greatly affects communication quality between sources and destinations.

Because of the possible rapid movement of nodes and different propagation conditions, network information becomes obsolete quickly. Also, network reconfiguration may frequently trigger extra number of control packets, which implicitly reflects the instability of the network.

However, the short lifetime of this gathered information means that a large portion of information may never be used due to the dramatically changing of topology. Thus, the bandwidth used for the distribution of routing update information is wasted. The situation becomes worse with the limited bandwidth and other problems like hidden terminal problem that exist in wireless ad hoc networks.

Based on the above discussion, we can basically generate the features for MANETs as follows:

- **Robust routing and mobility management algorithm** to increase the network's reliability and availability;
- **Adaptive algorithms and protocols** to adjust to frequently changing radio propagation and traffic conditions;
- **Multiple routes** between sources and destinations to provide more reliable communication routes, to reduce congestion, and to increase reliability;
- **Scalable network** to avoid susceptibility to network failures, congestion around high-level nodes and to possess good performance when the number of nodes increases;
- **Quality of Services (QoS)** to preserve the radio communication resource, and to guarantee the communication latency and packet delivery ratio, etc.

Due to mobility, network topology dynamically changes which incurs frequent communication interruptions, larger delay and lower performance. To alleviate this problem and to improve the scalability of large-scale mobile networks, introducing a hierarchical structure has proven to be very effective [1]. Consequently, many clustering algorithms have been proposed, especially in the context of MANETs as well as wireless sensor networks (WSNs).

1.2 Overview of Clustering in Wireless Ad Hoc Networks

Hierarchical techniques can aid in reducing limited energy consumption. Clustering is particularly useful for solving the problem of scalability when the number of mobile nodes increases greatly. For example, it can be used to scale a service location and discovery mechanism by distributing necessary state management to be localized within each cluster. Such a clustering-based technique has been proposed to provide location management of devices for QoS support [2]. Hierarchies based on clustering have been used to solve scalable problem for multi-hop networks [3], [4], [5] and [6]. Clustering can be extremely effective in one to many, one to all (broadcast), or many to many broadcast communications.

In multicast mesh networks, broadcasting is used by a source to find receivers and establish a route to a destination or transmit packets to nodes in a forwarding group. Clustering can be helpful to improve performance metrics including the packet delivery ratio and scalability in this scenario. For example, ODMRP protocol [7] is a well-known protocol in ad hoc wireless mesh networks, which can achieve a better packet delivery ratio and relatively lower latency. However, explosion of control packets may occur when the number of participating nodes increases.

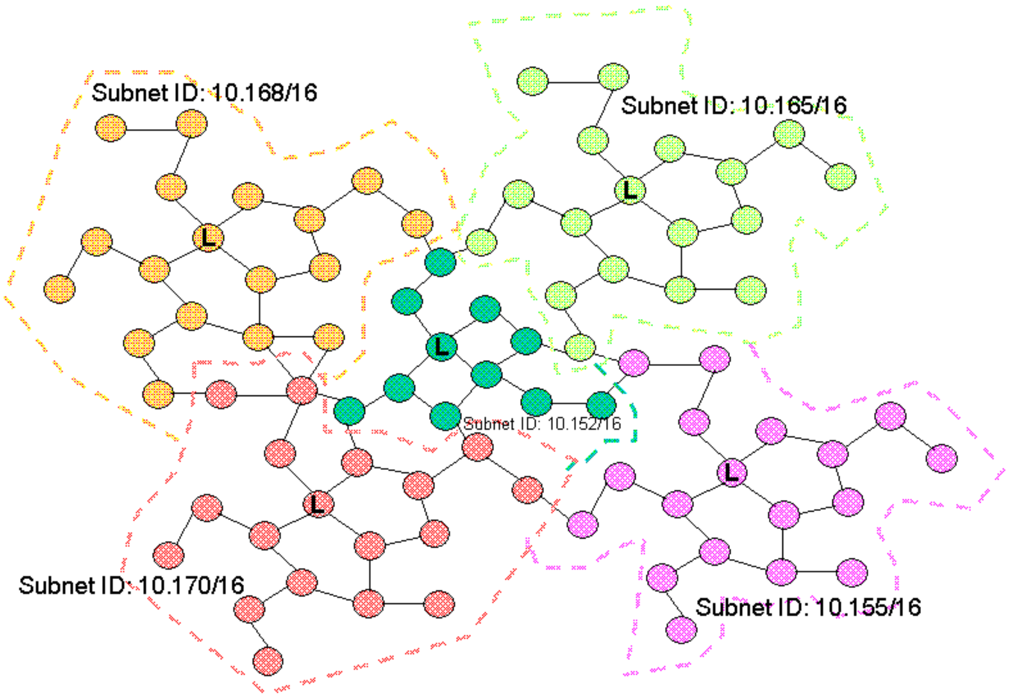

Figure 1.2 Clustering in Wireless Ad Hoc Networks

By using the clustering technique, mobile nodes are divided into virtual groups that are determined based on their proximity. Typically, one of the nodes in a cluster is considered as a cluster head, which serves as a local coordinator for that cluster, other nodes serve as cluster members. A cluster member needs to establish a route or identify destinations through cluster heads if they are not within the same local cluster. As shown in figure 1.2, a virtual but non-stationary infrastructure (black square in figure) is formed in the network, which facilitates data communication, improves the scalability and increases the network stability by local topology updates and limiting the overhead. Also it is likely to construct multiple layer structure shown in figure 1.3 which can be used in the scenarios where multiple frequency bands and transmission power levels are used.

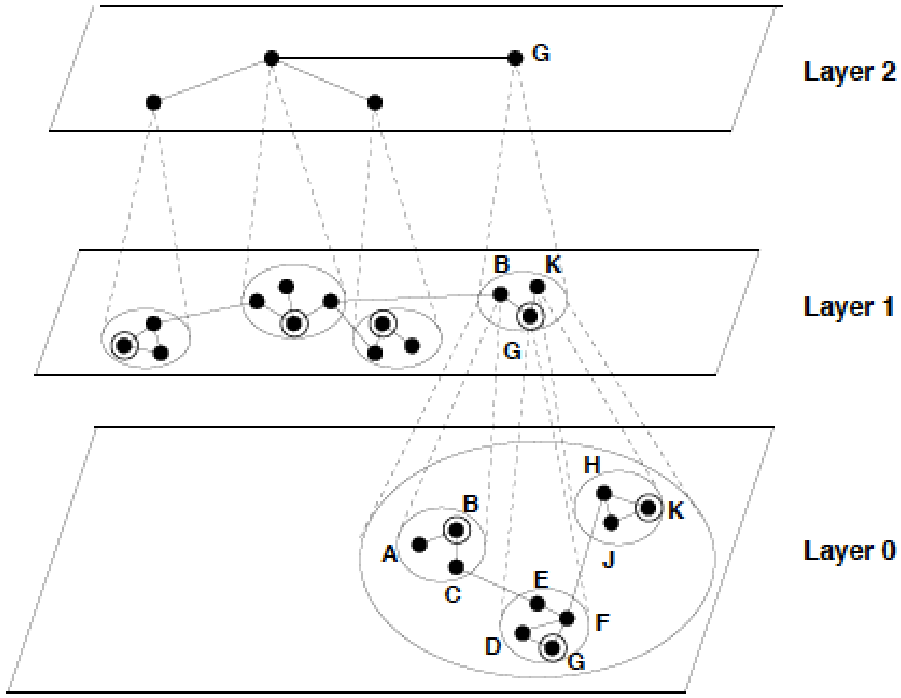

Figure 1.3 Multiple layers in clustering structure

Since Cluster heads serve as relaying nodes in the route between sources and destinations; packet delivery ratio (PDR) can be guaranteed when nodes move between different clusters. If a cluster head dies because of moving away from its cluster members by a large distance or due to exhaustive consumption of energy, it is likely the whole route will be broken. A cluster structure which can adapt to the dynamic changes of network topology and that takes into account the availability of resources is crucial in cluster-based routing protocols.

1.3 Desired Goals of Clustering Schemes

To create the hierarchical control structure described above, it is necessary to postulate the following desirable properties that should be present in the clustering mechanisms that exist in different layers of the clustering hierarchy.

Basically, we can put forward several requirements about clustering algorithms in MANETs:

- *Clustering is completely distributed*. Each node can choose to attach to different CHs based on local information. Generally speaking, the election of CH should consider the network coverage and overlapping between different clusters
- *Clustering terminates with a suitable cluster update interval* to establish a more stable structure and prolong the network lifetime; Because of mobility, the whole clustering structure is not stable and should be updated frequently. Even though we can benefit from frequent updating by gathering information of neighbors as soon as possible and make more wise choices to elect CH, more bandwidth and larger communication latency are the price that we need to pay. It doesn't make sense when local clustering structure is relatively stable. Thus clustering update interval should be determined to adjust to different local clustering scenarios.
- *Clustering simplicity:* should be efficient in terms of processing complexity and message exchange; The cluster size or radius is an important parameter to describe complexity of the clustering structure. The larger the size of cluster is, the more complex is the structure. Also clustering size affects the number of CH and battery consumption.
- *Cluster heads* are well distributed and probably elected to set up relatively stable clusters. A good CH can improve the relative stability of structure and should be elected based on the consideration of trade off between many factors that affect the stability, such as mobility, node' density, transmission range and energy.

1.4 Thesis Outline

In this thesis, we propose a novel cluster head election algorithm that utilizes the information in the HELLO messages, which only include one-hop neighbor information. All of the required information is available if nodes are equipped with GPS. RLCH can adapt to different environments very well even in unknown scenarios such as battlefields and mobile instant communication. Factors affecting stability and energy consumption are considered with no weights given to these factors. In addition, less time is required for setting up the cluster reformation. Moreover, we provide a theoretical model to identify local cluster head updating intervals based on the transmission range of nodes and the probability distribution of nodal velocity. We use this model to increase the stability of a cluster and to prolong the network lifetime

Chapter 2

Clustering algorithms in MANET

Numerous clustering algorithms have been proposed in the literature, which are based on different optimization objectives. These algorithms can be classified as identifier-based clustering, connectivity-based clustering, weight-combination clustering, cost-reduced clustering, energy-aware clustering, etc. [8].

2.1 Identifier-based Clustering

In the identifier-based clustering, each node is assigned a unique ID, and this ID is broadcast to its neighbors using HELLO messages periodically. Each node can establish neighbor list by gathering information of ID in HELLO message. In the LID algorithm [9], nodes compare their ID with neighbors' ID and the node with the minimum ID is selected as cluster head. When a node has multiple cluster heads in its neighbor table, it becomes a gateway that is generally used for routing between clusters. LID is simple distributed algorithm but the number of cluster heads may become undesirably large. Furthermore, because of exhaustive energy consumption by the cluster heads, the network lifetime becomes short. In Max-Min d-clustering formation algorithm [10], cluster heads can gather information about neighbors up to d-hops away. Since the cluster radius is large, it is more reasonable to let mobile nodes operate asynchronously. Thus less control overhead is needed to conduct clock synchronization. Limited cluster radius rather than total number of nodes in networks guarantees a good controlled message complexity. A designer can control the number of CH in networks by configuring the value of d, which is an input heuristic. If a node M is largest number of neighbors within d-hop neighborhood of node N, M will be elected a CH, even though M may not be the largest within its own neighborhood. Max-Min d-clustering algorithm allows asynchronous operation of nodes and reduces the clock synchronization overhead, while due to multi-hop it takes longer time for each node to update the information (leave or join) of neighbors up to d hops. Specifically in high mobility environments, neighbor table update becomes a lengthy process and the performance is degraded.

2.2 Degree-based Clustering

Highest Degree (HD) is a typical distributed connectivity-based algorithm [9]. The goal of HD algorithm is to reduce the number of clusters. The nodes' degrees are regarded as the metric for determining cluster heads. By listening to the HELLO message, each node calculates its own degree d (the number of one-hop neighbors) and inserts d into HELLO message. The HELLO messages are periodically broadcasted by each node to establish and maintain the clustering structure. Node with largest degree (largest number of neighbors) is elected as the cluster head (CH). When two nodes have the same degree, the node with smaller ID is selected as CH. Even though HD algorithm can reduce the number of cluster heads in the network, it does not consider the impact of mobility on the stability of the clustering structure. K-hop connectivity ID clustering algorithm (K-CONID) [11] combines the features of HD and LID and considers connectivity (i.e. from the HD algorithm) as the primary criterion. Using only nodal connectivity as a criterion leads to numerous ties between nodes. On the other hand, electing CH based only on a lower ID results in more CHs than necessary. K-CONID generalizes connectivity up to k hops neighbors. Each node in the network is assigned a d-ID represented as (d, ID). d is equal to the degree of the node and ID is the node's identifier. A node is selected as a CH if it has the highest connectivity. In case of a tie, the node with lowest ID is qualified for CH. This algorithm combines the advantages of HD and LID algorithms. However, it takes much longer time to obtain the required information from its k-hops neighborhood and the obtained information may not be reliable. Adaptive multi-hop clustering [12] is a clustering scheme with load-balancing capabilities. There exists upper and lower bounds on the number of members within a cluster. This algorithm can control the number of members in a cluster and the number of clusters in the network. However, the problem about how to choose a stable CH is still unsolved. Degree based algorithms gather neighbors' information by exchange HELLO message in constant interval and select distributed CH, which cannot guarantee the stability of clustering structure.

2.3 Cost-reduced Clustering

In order to reduce the cost of clustering maintenance, least cluster change algorithm (LCC) [13] adopts event-driven re-clustering algorithm only in two cases. One is when two CHs move into mutual communication range, the other is when a mobile node moves out of current existing clustering structure and establishes a new local cluster. In LID and HD algorithms, the clustering scheme is performed periodically to check the local smallest ID or highest degree aspect of a cluster head. When CH finds an alternative with smallest node's ID or higher degree, it is forced to relinquish its cluster head role. This mechanism means a frequent clustering update rate. Hence LCC relinquishes the requirement on-demand and reduces the frequency of CH update, which improves cluster stability significantly.

Passive clustering [14] is another algorithm, which avoids periodic dedicated control packets. Most of clustering algorithms require participating mobile node to send cluster-dependent information repeatedly in the form of beacon message to gather information about the clustering structure. Thus these beacon messages are one of the main sources of control overhead in wireless ad hoc networks. In this scheme, a mobile node can be in one of four states: initial, cluster head, gateway and ordinary node. All mobile nodes start with "initial" state and only a node with "initial" state can be a CH candidate. A node hearing beacon message from a CH becomes an ordinary node. If multiple beacon messages are recovered, it changes to gateway. These states can exchange dynamically due to mobility.

2.4 Weights-Combination Clustering (WCA)

Weights combination clustering algorithm has been widely studied and modified as in [15-18]. WCA [15] chooses a cluster head by striking a balance between factors such as remaining sum of the relative distances of nodes to their neighbors, degree difference, mobility and remaining power. It needs to assign an appropriate weight to each of these factors before selecting or updating a CH. The weight associated to a node u is defined as:

$$S=w_1\Delta\delta+w_2D_d+w_3M_v+w_4P_v$$

The node with minimum weight is selected as CH. The combination of weights satisfies the condition: $w_1+w_2+w_3+w_4=1$. The $\Delta\delta$ is the degree difference, which is used to ensure load balancing and $\Delta\delta=|dv-\delta\,|$, where dv is the actually nodal degree and δ is a pre-defined threshold. The D_d represents the sum of distances from a given node to all its one-hop neighbors. This factor is related to energy consumption since more power is needed for long-distance communication. The measure of mobility is M_v and is computed by average speed of neighbors.

P_v is used to measure the remaining energy and the amount of power consumed. CH consumes more battery power than an ordinary node because it has extra responsibilities. In [16], the authors modify WCA by defining a new factor called relative mobility and a new parameter called the degree of connectivity. A weighted combination of these factors is called the stability factor and minimizing this factor chooses the cluster head. Entropy-based weighted clustering algorithm [17] uses an entropy-based model to evaluate the route stability, which reduces the rate of re-affiliation and the communication overhead. Weight-based adaptive clustering algorithm (WBACA) [18] is based on the availability of Global Positioning System (GPS) and considers parameters such as: transmission power, transmission rate, mobility, battery power and degree. Each node is assigned a weight combination like WCA. The weight of a node is defined as:

$$S = w_1 * M + w_2 * B + w_3 * T_x + w_4 * D + w_5 / T_r$$

, where M, B, T_x, D and T_r represent mobility, battery power, transmission power, degree difference and transmission rate respectively.

CEMCA [19] is based on a combination of several significant metrics like highest node degree, lowest node mobility, highest battery energy and optimized transmission range. Two-hop neighbors are considered when choosing the CH, and each CH stores all of its neighbors' information. Generally speaking, these weight combination algorithms have some challenging problems in common. Frist, it is very likely that one or two factors with larger value dominate the other factors in selecting the CH no matter what weight combination is assigned. Second, it is not easy to assign appropriate weight combination before cluster formation or during updating. Especially in unknown scenarios, it may take a long time to identify proper weight combination for the factors. Moreover, some of these algorithms (like CEMCA) rely on information communicated via multi-hop, which increases the complexity and delay in formation and updating of the clusters.

2.5 Energy-aware Clustering

Some clustering algorithms emphasize on the energy factor while choosing a CH. LBC [20] limits the time that a node can continuously remain a cluster head. It avoids the exhaustive consumption of energy for a few nodes and thereby prolongs the network lifetime. However, the cluster head serving time may not be sufficient to indicate the energy consumption. In [21], HEED algorithm periodically selects cluster heads according to a hybrid of their residual energy and a secondary parameter, such as nodal degree. Clustering is triggered every $T_{CP} + T_{NO}$ ($T_{NO} \gg T_{CP}$) seconds to select new cluster heads where T_{CP} is the clustering process interval and T_{NO} is network operation interval. When a selecting cluster head, the impacts of other factors like relative mobility and relative distance on energy consumption are neglected. S. Banerjee and S. Khuller define clustering problem in a graph theoretic framework [22]. Clusters are formed within root-based tree and each cluster head is chosen when satisfying constraint of sub-tree size. The algorithm doesn't consider the stability of the clustering structure within a tree. In [23][24], cluster heads are elected based on a probabilistic algorithm to minimize energy consumption and achieve load balancing. Power-aware connected dominant set [25] is an energy-efficient clustering algorithm, which reduces the size of the dominating set (DS). It reduces energy consumption by excluding unnecessary mobile nodes from useful dominating set that is a set of CH candidates. Also nodes within DS consume more energy than common nodes outside DS because inside nodes bear extra task. In this algorithm, energy level replacing ID or node degree is used to determine whether a node should serve as a CH. There are some other references like

[26], [27] and [28] that focus on a cross-layer optimization and scheduling framework with the objective to minimize the total power consumption during the steaming period while maintaining the Quality of Experience (QoE) of video users.

2.6 Outline of our proposed RLCH algorithm

Compared to the above clustering algorithms, our proposed RLCH not only considers the impact of many factors such as degree of connectivity, relative velocity, relative distance, and energy distribution on stability of a cluster. In addition, it does not require a weight combination for selecting the cluster heads. Meanwhile all of the information required for RLCH is available if nodes are equipped with GPS, and only 1-hop neighbors within cluster heads are considered. Through HELLO messages, each node can gather up to 2 hop neighbors' information and select a suitable CH based on that. The format of beacon message is shown in figure 2.1. Current value of velocity and moving direction can be calculated based on previously obtained nodal location by GPS.

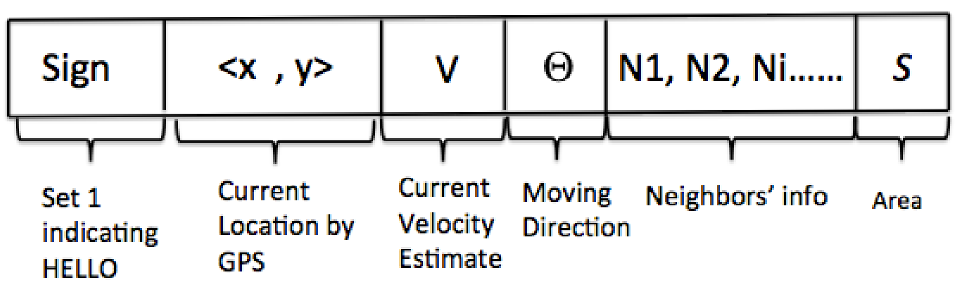

Figure 2.1 Hello message format

As will be shown later in this thesis, in RLCH, the time consumed for cluster updating (reformation) is shorter compared to the other proposed algorithms. This can increase packet delivery ratio and reduce communication delay. Moreover, each local cluster head can dynamically select cluster update interval based on the transmission range, velocity distribution, and remaining power.

Chapter 3

Mathematical Models

In this section, we establish two models for link lifetime and maximum cluster head update intervals. These models will be used for determining cluster update interval and CH election in our RLCH algorithm described in the next chapter.

3.1 Link Lifetime Model

In this part, we estimate each link lifetime with respect to neighbors based on the information extracted from the HELLO messages periodically broadcast by the 1-hop neighbors of each node. To simplify our link lifetime model, we start with the following assumptions:

1. Each node is equipped with GPS and periodically sends HELLO messages to its neighbors.
2. The velocity and the direction of nodes during the short cluster head election period are fixed.
3. Each node moves with constant velocity and direction with respect to cluster head until moving out of the transmission range of its current cluster head. Since those mobile nodes have relatively higher speed plus short communication range in MANETs, this assumption is without lost of generality.

Here we take the example of link (0,1) connected by node 0 and node 1 to estimate the link lifetime of link (0,1) based on the acquired information. We define the following variables:

1. v_1 and v_2 are current velocities of nodes 0 and 1, respectively.

2. (x_1, y_1) and (x_2, y_2) are current updated locations of nodes 0 and 1, respectively.

3. $\overrightarrow{v_1}$ and $\overrightarrow{v_2}$ are the velocity vectors, and θ_1 and θ_2 are the velocity directions of nodes 0 and 1 respectively; also the relative velocity is $v_{(1,2)} \triangleq |\overrightarrow{v_2} - \overrightarrow{v_1}|$.

4. $d_{link(i,j)}$ is the estimated length of movement of node j while it is in the transmission range of node i (i.e., while there is a link between nodes i and j) and $t_{link(i,j)}$ is the link lifetime.

Let us assume that node 0 fixed and calculate the relative velocity $\vec{v}_{(1,2)}$ of node 1 with respect to node 0, and the relative distance d between node 0 and node 1. Imagine a Cartesian coordinate system with orthogonal unit vectors: $\vec{i}$ along the moving direction of node 0 and $\vec{j}$ on its orthogonal direction, respectively, as shown in Fig. 3.1.

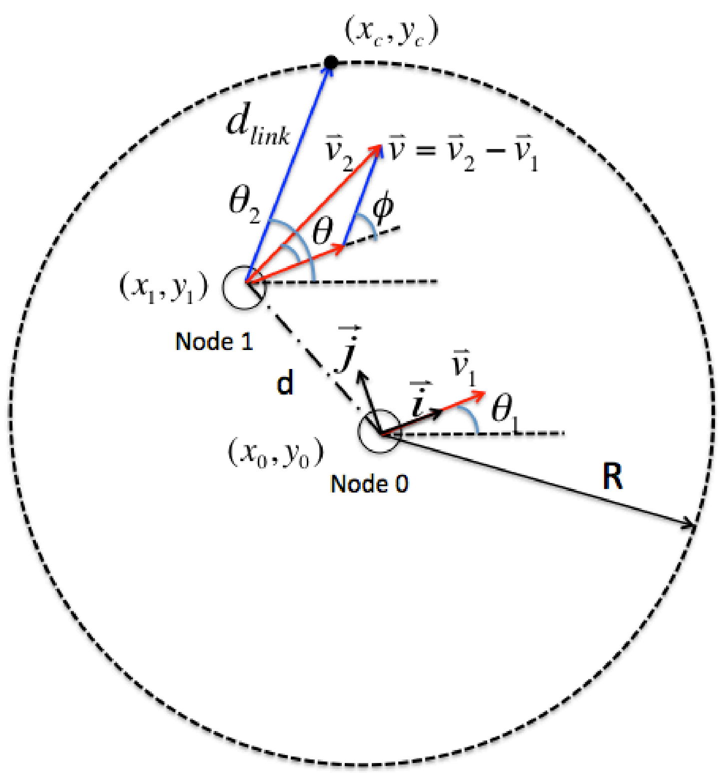

Figure 3.1 Example for calculating link lifetime between node 0 and node 1

In fig 3.1, d denotes the distance between two nodes, and (x_0, y_0) denotes the new location of node 1 with respect to node 0 on which the Cartesian coordinate is fixed; and $\emptyset$ is the angle of relative velocity in new Cartesian coordinate system. We can derive:

- $d=\sqrt{(x_2-x_1)^2+(y_2-y_1)^2}$;
- $x_0=d\cos(\text{atan2}\left(\frac{y_2-y_1}{x_2-x_1}\right)-\theta_1)$;
- $y_0=d\sin(\text{atan2}\left(\frac{y_2-y_1}{x_2-x_1}\right)-\theta_1)$;
- $\emptyset \triangleq \text{atan2}(\frac{\sin\theta}{\cos\theta-v_1/v_2})$;
- $v_{(0,1)} \triangleq |\vec{v}_{(0,1)}| = \sqrt{v_1^2+v_2^2-2v_1v_2\cos\theta}$;

Also let:

$$x_c=\begin{cases}\frac{-(ky_0-k^2x_0)+\sqrt{-(kx_0-y_0)^2+R^2(1+k^2)}}{1+k^2}, & \text{if } |\emptyset|\epsilon[0,\pi/2) \\ \frac{-(ky_0-k^2x_0)-\sqrt{-(kx_0-y_0)^2+R^2(1+k^2)}}{1+k^2}, & \text{if } |\emptyset|\epsilon(\frac{\pi}{2},\pi]\end{cases} \quad (1)$$

and

$$d_{link(0,1)}=\begin{cases}|x_c-x_0|\cdot\sqrt{1+k^2}, & \text{if } |\emptyset|\epsilon[0,\pi/2)\cup(\frac{\pi}{2},\pi] \\ |\sqrt{R^2-x_0^2}-y_0|, & \text{if } |\emptyset|=\pi/2\end{cases} \quad (2)$$

where R is the transmission range and $k=\frac{\sin\theta}{\cos\theta-{v_1}/{v_2}}$.

Based on (2), we can derive: (3)

$$t_{link(0,1)}=\frac{d_{link(0,1)}}{v_{(0,1)}}$$

Thus, each node can update the link lifetime with each neighbor in its neighbor list when receiving HELLO messages from its neighbors.

3.2 Expected Link Lifetime Model

We realize that the cluster update interval, denoted by T, is a very important parameter in designing clusters. If T is too long, it is likely that a cluster becomes unstable because of high mobility of its member nodes, which results in frequent exchange of nodes between different clusters. Consequently, more control packets are needed to reestablish the disconnected links (routes). Also being a cluster head for a long time leads to exhaustive energy consumption and thereby reducing the average lifetime of the network. If T is too short, cluster reformation procedure is conducted too frequently, which results in bandwidth waste, larger communication latency and lower packet delivery ration. As a result, more energy is wasted in choosing a new cluster head and transmission delay may increase. However in [9], [15], [16] and [17]; constant clustering update interval is utilized.

We believe that during the expected link lifetime interval, a common mobile node can be relatively stable inside a local cluster structure. That means that nodes move inside a cluster during this interval and move out when expected link expires. So expected link lifetime reflects relative stability of local clustering structure and should adjust to adapt to different local clustering scenarios.

Cluster update interval T is influenced by many factors, including battery power left, mobility, transmission range, interference and so on. In many algorithms such as those considered in [15] and [16], the cluster update interval is considered to be constant. These papers do not provide analytical results for the cluster update interval and the effects of nodal velocity and transmission range on the cluster update interval are not analyzed. In this part, we try to establish a model for determining the maximum cluster head update interval T.

Let us introduce our assumptions and the notations used in our derivations:

1. Nodes are uniformly distributed in the network.
2. Both nodal speed magnitude and angle satisfy uniform distribution; i.e. v is uniformly distributed in $(0, a]$ (a is maximum speed), and θ is uniformly distributed in $(-\pi, \pi]$. Also speed and angle are uncorrelated random variables.
3. We first focus on a typical node with random angle and distance of γ and d, respectively, where γ and d are independent variables. Both random variables are used to describe certain cluster member (CM)'s initial location with respect to CH during cluster update interval.
4. A link can be established between two nodes if they are in the transmission range of each other,. i.e. interference has been mitigated like [29] and other factors affecting communication link are ignored.

In [30], the authors have established an expected link lifetime model. This model is used to describe the interval between link generation and termination; therefore the expected link lifetime is based on the assumption of long time observation. However, this assumption is not correct for describing the link lifetime with respect to cluster head during the cluster head update interval as we only consider the current existing links while conducting cluster reformation. It is more reasonable to assume cluster members are randomly distributed within the transmission range of the cluster head. Also the result of [30] is based on long-term observation, while these clustering update intervals are not long enough. Reference [31] proposes to divide the link lifetime into many small intervals. We assume that each interval is unrelated and independent from other intervals. However, this assumption is not realistic when the transmission range is not large and the mobility is high. On the other hand, it is not accurate to assume that each epoch is randomly independent especially at high mobility.

First, we define the two variable v and $\emptyset$ to denote the node's respective speed and moving angle with respect to another node. Before establishing the expected link lifetime model to a certain node, we need to derive the probability density function (PDF) of relative speed v and $\emptyset$. Let us define

$$g_{v_1,v_2,\theta}(v_1, v_2, \theta) \triangleq f_{v,\emptyset}(v, \emptyset).$$

where $f_{v,\emptyset}(v, \emptyset)$ is the joint pdf of v and $\emptyset$ and $g_{v_1,v_2,\theta}(v_1, v_2, \theta)$ is the joint pdf of v_1, v_2 and θ. According to the assumptions in link lifetime model in Section 3.1, we know $f_{v_1}(v_1) = f_{v_2}(v_2) = \frac{1}{a}$, and $f_\theta(\theta) = \frac{1}{2\pi}$; Therefore we can conclude:

$$g_{v_1,v_2,\theta}(v_1, v_2, \theta) = f_{v_1}(v_1) \cdot f_{v_2}(v_2) \cdot f_\theta(\theta) = \frac{1}{2\pi a^2}$$

with parameter a denoting the upper limit of nodes' velocity in the assumption.

Next, we use an auxiliary variable v_1 in the Jacobian's transform. We get

$$f_{v,\emptyset,v_1}(v, \emptyset, v_1) = g_{v_1,v_2,\theta}(v_1, v_2, \theta) \cdot \frac{1}{|J(v_1, v_2, \theta)|}$$

Also

$$\mathrm{J}(v_1, v_2, \theta) = \begin{vmatrix} \frac{\partial v}{v_1} & \frac{\partial v}{v_2} & \frac{\partial v}{\theta} \\ \frac{\partial \emptyset}{v_1} & \frac{\partial \emptyset}{v_2} & \frac{\partial \emptyset}{\theta} \\ \frac{\partial v_1}{v_1} & \frac{\partial v_1}{v_2} & \frac{\partial v_1}{\theta} \end{vmatrix} = \frac{v_2}{\sqrt{v_2^2 + v_1^2 - 2v_1 v_2 \cos\theta}} . \Lambda(v, v_1, \emptyset) \qquad (4)$$

where

$\Lambda(v, v_1, \emptyset) =$

$$u\left(\sqrt{v^2 + v_1^2 + 2vv_1 \cos\emptyset}\right) - u(\sqrt{v^2 + v_1^2 + 2vv_1 \cos\emptyset} - a)$$

and $u(t)$ is the unit step function. Then

$$f_{v,\emptyset}(v, \emptyset) = \int_{v_1} g_{v,v_1,\emptyset}(v, v_1, \emptyset) \cdot dv_1 \qquad (5)$$

Now we can analyze the expected link lifetime (maximum cluster head update interval) based on the derived $f_{v,\emptyset}(v,\emptyset)$. In [32], the authors simply assume that the distance of each neighbor moving within transmission range of cluster head is uniformly distributed with PDF function equal to $\frac{1}{(\pi R^2)^2}$, where R refers to transmission range. However, this assumption is not accurate, which ignores the possible condition of the nodal moving outside with same distance. Nodal movement should not be constrained within a cluster when considering their movement.

Based on the assumptions 1 and 3 in this section, we can show the PDF of random variables d, γ as

$$f_d(d) = \begin{cases} \frac{2d}{R^2} & when\ d \in [0, R] \\ 0 & otherwise \end{cases}$$

$$f_\gamma(\gamma) = \begin{cases} \frac{1}{2\pi} & when\ r \in [0, 2\pi) \\ 0 & otherwise \end{cases}$$

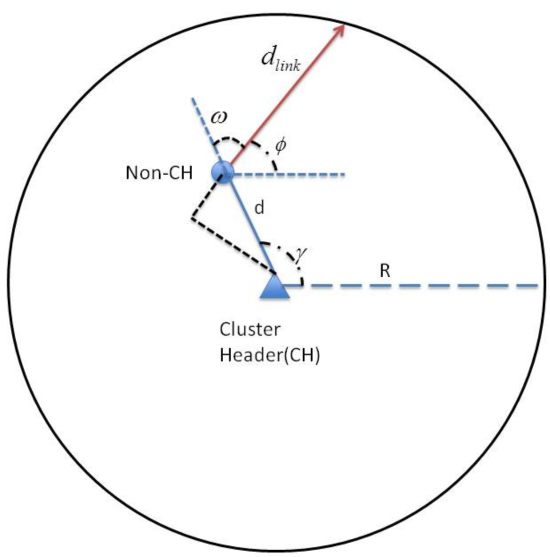

Figure 3.2 Link lifetime model of CH with Cluster Member

In Fig 3.2, we illustrate the location of a typical node within transmission range R of the cluster head. In this figure, d and γ are used to describe the location of this typical node. Variable d indicates relative distance, and γ indicates relative angle. Using this notation, we can get:

$$d_{link} = \sqrt{R^2 - (d \cdot \sin w)^2} \text{ - d} \cdot \cos w = \sqrt{R^2 - (d \cdot \sin(\gamma - \emptyset))^2} \text{ - d} \cdot \cos(\gamma - \emptyset)$$

where $w \triangleq \gamma - \emptyset$ and d_{link} is described in Sub-section 3.1 as the estimated length of movement of typical node while it is in the transmission range of the CH. Also, we have $t_{link} = \frac{d_{link}}{v}$. We can therefore write

$$\bar{\tau} = \mathrm{E}(t_{link}) = \int_0^{2a}\int_0^{2\pi}\int_0^{R}\int_0^{2\pi} t_{link} \cdot f_{v,\emptyset}(v,\emptyset) \cdot f_d(d) \cdot f_\gamma(\gamma)\, dv\, d\emptyset\, dd\, d\gamma$$

$$= \int_v \int_\emptyset f_{v,\emptyset}(v,\emptyset) \cdot \frac{1}{v} \cdot \left\{ \int_\gamma \left[\int_d D_{link} \cdot f_d(d)\, dd \right] \cdot f_\gamma(\gamma)\ d\gamma \right\} dv d\emptyset$$

$$= \int_v \int_\emptyset f_{v,\emptyset}(v,\emptyset) \cdot \frac{1}{v} \cdot \left\{ \int_0^{2R} \frac{2}{\pi R^2} \cdot z \cdot \sqrt{R^2 - (z/2)^2} dz \right\} dv d\emptyset$$

$$= \frac{8R}{3\pi} \cdot \int_v \int_\emptyset f_{v,\emptyset}(v,\emptyset) \cdot \frac{1}{v} dv d\emptyset$$

$$= \frac{8R}{3\pi} \cdot \frac{1}{2\pi a^2} \cdot \int_{v_1} \int_\emptyset \ln\left| \frac{a + \sqrt{a^2 - v_1^2 \sin^2\emptyset}}{v_1 + v_1 \cos\emptyset} \right| d\emptyset dv_1 \quad (6)$$

The above equation cannot be simplified further. Meanwhile the formula (6) is very important in our algorithm, since it can be used to dynamically and adaptively choose cluster update interval T in the following section. We can obtain expected link lifetime under varied transmission ranges or velocity distributions. Figures 3.3 and 3.4 are plotted based on formula (6) and show the numerical results of expected link lifetime under scenarios with different maximum node velocities (i.e. a in (6)) and transmission ranges (i.e. R in (6)).

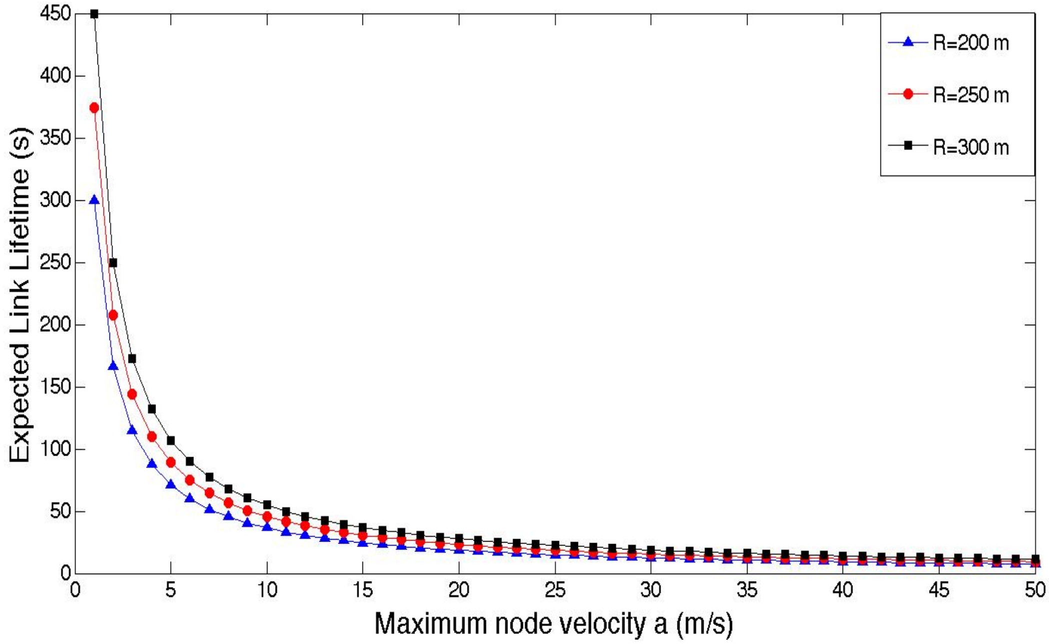

Figure 3.3 Expected link lifetime with varied maximum node speed a (m/s)

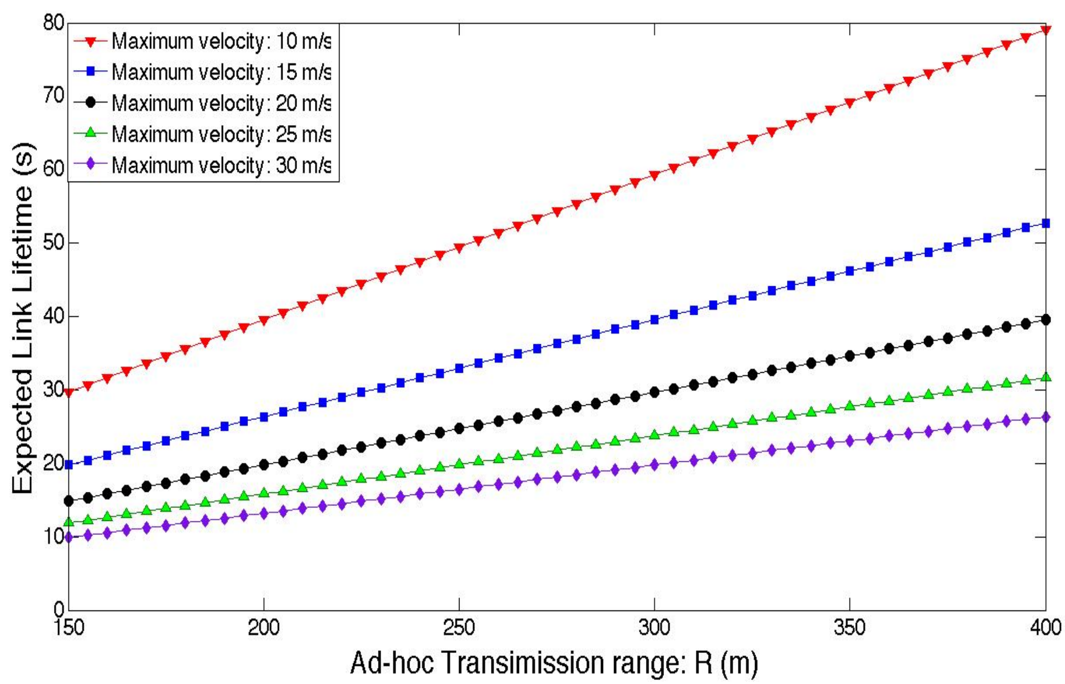

Figure 3.4 Expected link lifetime with varied transmission range R (m)

In the Fig 3.3, we can observe that the expected link lifetime decreases with increasing node velocity; while in Fig 3.4, the expected link lifetime $\bar{\tau}$ is linearly proportional to transmission range R, which can also be seen from equation (6). The intuition behind this result is clear, especially when R is fixed and velocity increases to infinity. In this case, the expect link lifetime approaches to zero. When velocity distribution is fixed and R approaches to infinity, all nodes within network become one-hop neighbors and are always within transmission range which results in infinite expected link lifetime.

CHAPTER 4

Introduce to RLCH

In this chapter, we first describe the desirable characteristics of clustering structure when selecting CH by naming “Double M Principles”. Then an area-based approximation algorithm and cluster head election adaptive algorithm are proposed and analyzed, which includes several advantages of typical algorithms and also is not without a sense of generality.

4.1 Double M Principles

We know that a good cluster should satisfy two basic requirements:

- Include as many neighbors as possible.
- Members do not move in/out frequently between different clusters.

In other words, it is desirable that a cluster head has the maximum number of one-hop neighbors within its transmission range and minimum rate of nodes moving out of its transmission range. Since during CH formation/reformation, it is not easy to predict the number of nodes moving into current local cluster in future and their departing rate after moving in. Therefore, just the number of existing nodes within the current cluster is considered during CH setting up or updating. We denote the number of one-hop neighbors when conducting the cluster formation/reformation as N and the rate of nodes departing as ϑ. Maximizing N can reduce the average number of clusters in the network, which can save energy and reduce the rate of member exchange between different clusters. Minimizing ϑ results in the links between a cluster head and the cluster members to be more stable. In other words, the cluster structure is more resilient to the mobility of nodes, which can reduce the number of control packet overhead, improve the route stability and decrease the transmission delay. Since satisfying the above requirements involve Maximization and a Minimization we use the term *double M principle* to describe these requirements.

4.2 Area-Based Approximation of Double M Principle

A lot of work has been done on implementing the double M principle in the clustering algorithms. HD [9] is the simplest of algorithms, which selects nodes with largest degree as cluster head. Other well-known algorithms are WCA [15] and its modified versions in [16-18] in which the factors including the number of clusters, the node velocities, relative distance, and remaining energy are considered. However, these algorithms either ignore the impact of relative nodal mobility on link stability, or need to identify the weight combination, which is time consuming and complicated to implement particularly in unknown environments.

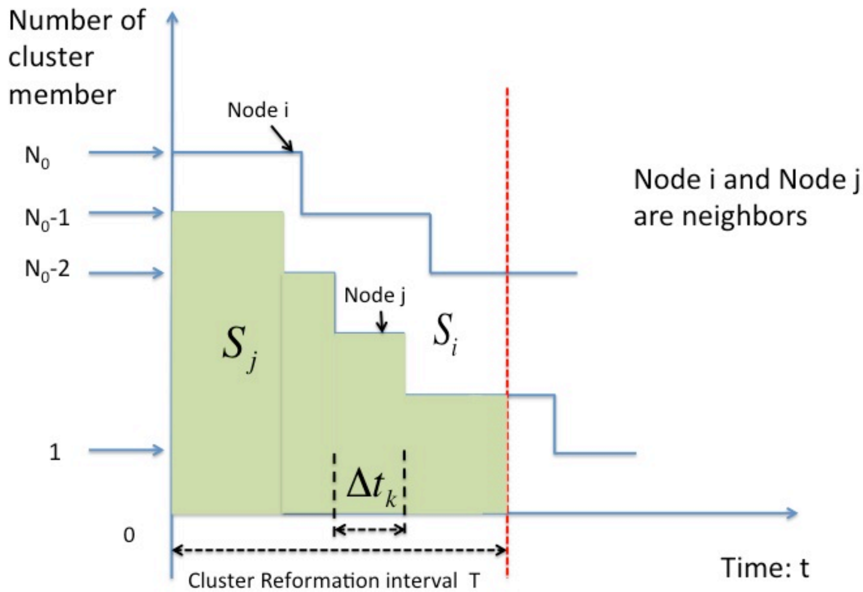

Figure 4.1 Area based cluster head election method

In this thesis, we approximate the requirements of the double M principle. Considering node i, we use the area S_i, shown in Fig. 4.1, to see what is the qualification of node i in being a cluster head. In this figure, the vertical axis shows the number of neighbors within transmission range of a node as time goes on. If the area S_i is large, it implies that first: the number of neighbors of node i when conducting the cluster formation/ reformation is relatively large and second: This number is decreased with a lower rate which means that the number of neighbors remains relatively large as time goes on. As shown in Fig. 4.1, N_0 denotes initial number of neighbors around node i and Δt_k is used to denote the interval during which only one member moves out. The value of the time interval Δt_k and the number of intervals during cluster reformation interval T reflect the impact of factors such as relative mobility and relative distance, transmission range on link stability. If Δt_k is large and the number of Δt_kintervals is small, then the cluster structure is more stable. The requirements of the double M principle are therefore lumped into a single objective function, which is to maximize the area described above. We only consider the nodes that are moving out of the transmission range as the rate of nodes moving in can be reduced if we can decrease the rate of node moving out. After calculating the quality factor S for being cluster head, each node inserts the Q value into HELLO message.

In Fig. 4.1, we are examining the qualifications of nodes i and j in being a cluster head. On the one hand, node i has more number of cluster members inside. In terms of leaving rate of neighbors of node i, it is relatively lower than that of node j. As shown in this figure, $S_i > S_j$ which means that node i satisfies the requirements of the double M principle better than node j and therefore is more qualified for being a cluster head.

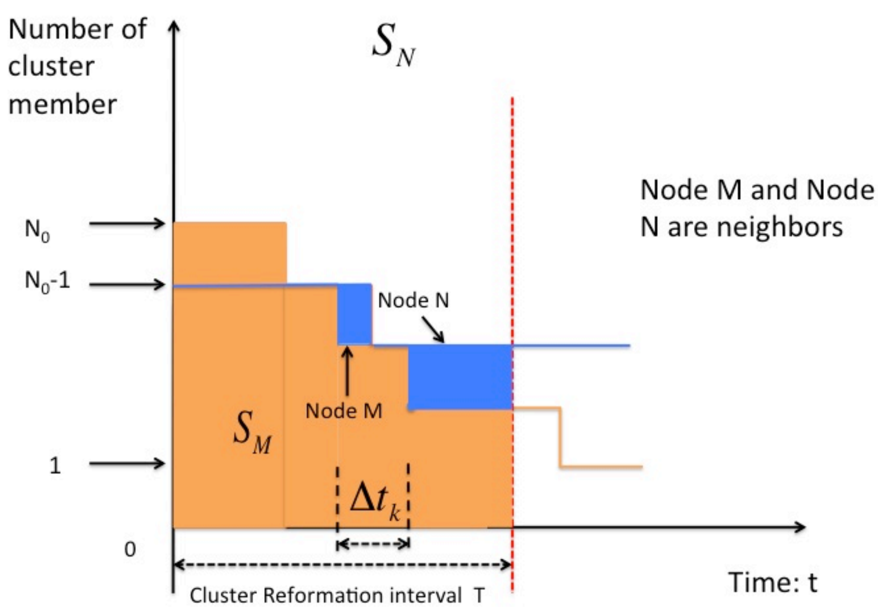

Figure 4.2 Area compassion between Node M and Node N

Another example is given in figure 4.2, even though node M has larger cluster size initially, the leaving rate of node M exceeds that of node N. According to our proposed algorithm, during cluster updating interval T, the area of node M is smaller than node N

($S_M < S_N$). It means that node N is more qualified for being selected as CH and can keep relative stable structure.

4.3 Cluster Head Update Interval Adaptation Algorithm

In Section 3.2, an analytical model is introduced to find the maximum cluster head update interval $\bar{\tau}$. Even though RLCH can adapt to different scenarios or even unknown environments, one drawback is that it does not consider the energy factor. However, the energy issue can play an important role in certain scenarios, especially in wireless sensor networks, and needs to be considered. In WCA [15], the battery power left in the nodes is considered as a factor during the cluster formation process. In this thesis, we use a simple energy threshold approach to adaptively change the cluster head update interval, which is determined by the energy available in nodes.

Due to data dissemination and local coordination performed by the cluster heads, they consume more energy compared to other cluster members. To simplify, we adopt a linear energy model to describe the energy consumption of the cluster head and cluster members. We assume that all nodes initially have equal energy: E_o units; also energy consumption rate of the cluster head and the cluster members are α units/s and β units/s respectively where $\alpha > \beta$. The remaining energy $\mathrm{E_{left}}$ can be found as [7]

$$\mathrm{E_{left}} = \begin{cases} \mathrm{E_o} - \alpha \cdot \Delta t & \text{if node i is a CH} \\ \mathrm{E_o} - \beta \cdot \Delta t & \text{if node i isn't a CH} \end{cases} \tag{7}$$

Let us define $E_1 = \alpha \cdot \bar{\tau}$ and consider ε and η as system-defined parameters where $\varepsilon > 1$, $\eta < 1$ and $\bar{\tau}$ is the expected link lifetime. The algorithm works as follows: If the remaining energy of a node is more than $\varepsilon \cdot E_1$, then the cluster update interval is considered to be $\bar{\tau}$.

Otherwise, we should choose the cluster update interval such that the consumed energy is less than $\eta.E_{left}$ and also $T \in (0,\ \bar{\tau})$. You can see the procedure in Figure 4.3. This algorithm avoids a cluster head which has limited energy to die early by decreasing the cluster head update interval.

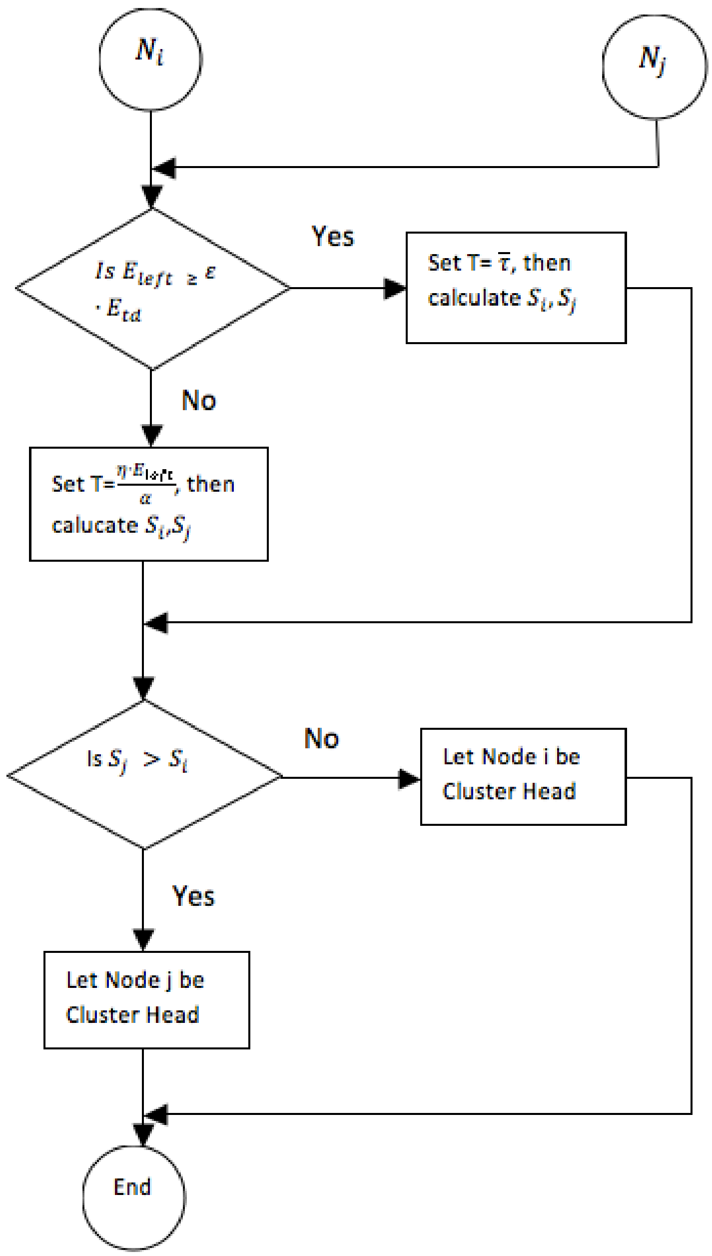

Figure 4.3 Flow chart of cluster updating interval algorithm

In order to choose the values of ε and η, we note that if ε is too large and the remaning energy is less than $\varepsilon \cdot E_1$, then a small value of η can increase cluster head update rate. This avoids a node with limited energy to remain a cluster head for a long period of time and thereby increases the network lifetime. On the other hand, frequent cluster reformation increases the communication delay. A small value of ε and a large value of η result in relatively lower cluster update rate and shorter network lifetime especially when the nodes' remaining energy is below $\varepsilon \cdot E_1$. As for delay, it will be relatively smaller compared to the previous case.

Generally speaking, this algorithm allows the nodes with limited energy to die slowly by limiting the cluster head update interval and the energy consumption of nodes will be uniform throughout the network. The choice of ε and η is based on the requirement of the network lifetime when the amount of remaining nodal energy is small.

4.4 Implementation of Area-Based Approximation

The HELLO messages contain the nodes' current status in cluster, velocity and moving direction and their latest updated S_i and are periodically broadcast by each node in wireless network. Nodes can update the state of their neighbors (connected or disconnected) and calculate or update new link lifetime when receiving a new HELLO message from their neighbors. Also each node can update the new area S_i based on updated link lifetime, relative location, and relative velocity with respect to the neighbors. Several steps are required to calculate S_i which we describe in this section.

Step 1. Assume that the neighbor set of node i is $\{i_1,\ i_2,\cdots,i_s\}$ where s is the number of neighbors of node i. Calculate the link lifetime based on link lifetime model in Section 3.1. Then get the set of link lifetimes as $\{t_{i-i_1}, t_{i-i_2} \cdots t_{i-i_s}\}$.

Step 2. Rearrange the set of nodes' relative link lifetimes in non-decreasing sequence in the form of $\{l_1, l_2, \cdots, l_s\}$.

Step 3. Calculate the area S_i based on three possible cases for the cluster update interval T.

- Case 1: If T$\geq l_s$, then

  $S_i = \sum_{k=0}^{s-1}(s-k)\,(l_{k+1}-l_k)$, where $l_0 = 0$.

- Case 2: If T< l_s and $2 \leq j \leq s$, assume that $l_{j-1} < T \leq l_j$, then

  $S_i = \sum_{k=0}^{j-2}(s-k)\,(l_{k+1}-l_k)$+(s-j+1) ($T$-$l_{j-1}$);

- Case 3: if $j = 1$, then S_i=$s * T$.

In the above cases, T is cluster head update/reformation interval, which is determined by cluster head interval adaption algorithm and maximum cluster head update interval $\bar{\tau}$.

Chapter 5

Simulation Studies of the Link lifetime model and RLCH algorithm

In this Chapter, we introduce the basic simulation configuration; define four metrics to verify the effectiveness of the cluster updating algorithm and compare the performance of our proposed RLCH algorithm with other well-known algorithms and discuss the simulation results

5.1 Simulation Configuration

Simulations are conducted on Ns-2.34 [33], and two groups of simulations are considered. The first group studies the influence of maximum node velocity on the four defined metrics with constant wireless transmission range R. Specific parameter configuration for this group of simulations is available in Table 1. The second group studies the impact of varying transmission range, R, on the four metrics considered. The specific simulation parameter configuration is presented in Table 2. Each simulation in each group was conducted 15 times. To compare with WCA, three groups of weights combinations were simulated [(0.6,0.2,0.1,0.1), (0.7,0.2, 0.05,0.05), (0.75,0.1,0.05,0.1)]. The results shown in each figure are the average of simulation results obtained from each step.

Table 1 Parameter configuration for Group 1

Number of nodes:	50
Transmission range (R)	250 meters
CBR	4 Packets/second
Packet Size	512 bytes
Mobility model:	Random waypoint
Propagation model:	TwoRayGround [34]
Maximum node velocity interval:	5-50 meters/second [35]
Hello message broadcast interval:	2 second
E_0	700 units
α and β	$\alpha = 1, \beta = 0.1$
ε and η	$\varepsilon = 2, \eta = 0.4$
Total Simulation Duration	600 second

Table 2 Parameter configuration for Group 2

Transmission range (R)	150-400 meters
Maximum node velocity interval:	20 meter/second
Other parameters:	same as that in Table 1

5.2 Metric Definition

In this section, we first define four metrics as [15] and their values are obtained through simulation for RLCH as well as other well-known clustering algorithms:

1. *Cluster head update rate*: is defined as the frequency with which cluster head nodes are rotated. It is composed of three cases: Cluster Header (CH) becomes Cluster Member (CM), CM becomes CH and CM moving out of the current cluster structure and becomes a new CH.

Frequent cluster head update results in larger delays and wastes energy. Additionally it can reduce performance by decreasing packet delivery ratio.

2. *Member exchange rate*: is defined as the rate of cluster members moving in/out between different clusters. Higher member exchange rate indicates that cluster structure chosen is not very stable, which causes frequent route disconnection and larger delay.

3. *Standard deviation of remaining energy*: is defined as the standard deviation of all nodes' remaining energy after simulation interval. The larger standard deviation of remaining energy indicates that some of the nodes consume energy exhaustively, which make some nodes die early and reduce whole network lifetime.

4. *Average number of clusters*: The smaller the number of clusters, the more energy can be saved as there will be fewer cluster heads which consume more energy than other nodes. Moreover, the cluster structure will be more stable.

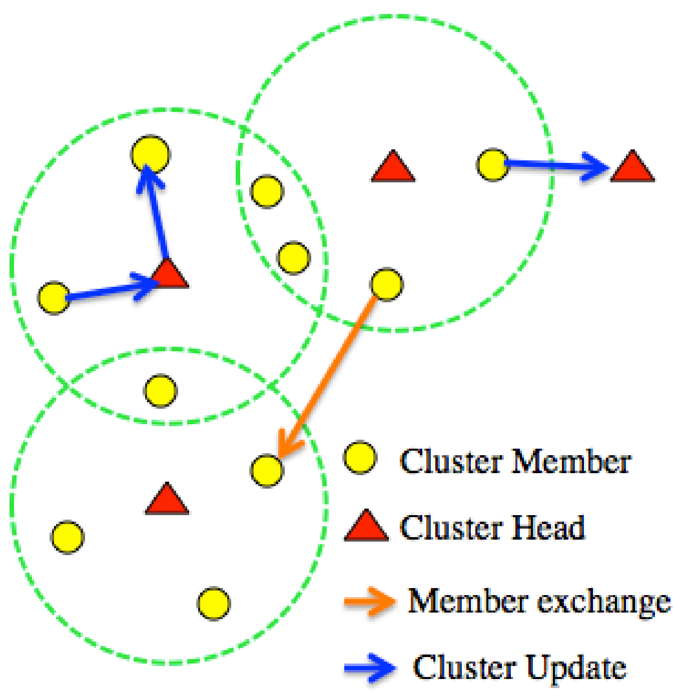

Figure 5.1 Example of defined metrics

In figure 5.1 above, yellow circles denote cluster member and red triangles denote cluster head. CH update rate and CM exchange rate are represented by blue arrow and yellow arrow.

5.3 Analysis of Results

5.3.1 Verification of cluster update model

In section 3.2, we propose expected link lifetime as a heuristic for identifying cluster update interval. However, verification of this model is not provided. In this section, by comparing the performance of three different algorithms: LID, HD and WCA under proposed link lifetime clustering update model and constant cluster update interval, we show the effectiveness of the expected link lifetime model as a vehicle for predicting clustering update interval. Also to select suitable constant cluster update interval as a comparison objective, 15 seconds is given as example as in [15] and [30]. 15 seconds can be regarded as a common suitable value for different scenarios in simulations.

For each of the defined metrics, we conduct our simulations 15 times and take the average value of repeated simulation results in three different clustering algorithms (LID, HD and WCA). In Figs 5.2 and 5.3, we study the first metric, which is the cluster head update rate. We can observe that the cluster head update rate is relatively smaller compared to constant cluster update interval. Also the update rate of different cluster head election algorithms seems very close when the maximum node speed is high (about 30 m/s) or transmission range R is small (about 175 m). That is because the value of the cluster update interval determined by our proposed model approximates that of constant update interval, which can be shown in the Figs 3.3 and 3.4. Moreover, when the mobility is high ($a \geq 35$m/s), the value of the cluster update rate obtained by using adaptive cluster update interval is a bit higher than by using constant update interval. However in this case, the member exchange rate can be reduced. This means that adaptive cluster update interval model can achieve better link stability and route stability within clusters.

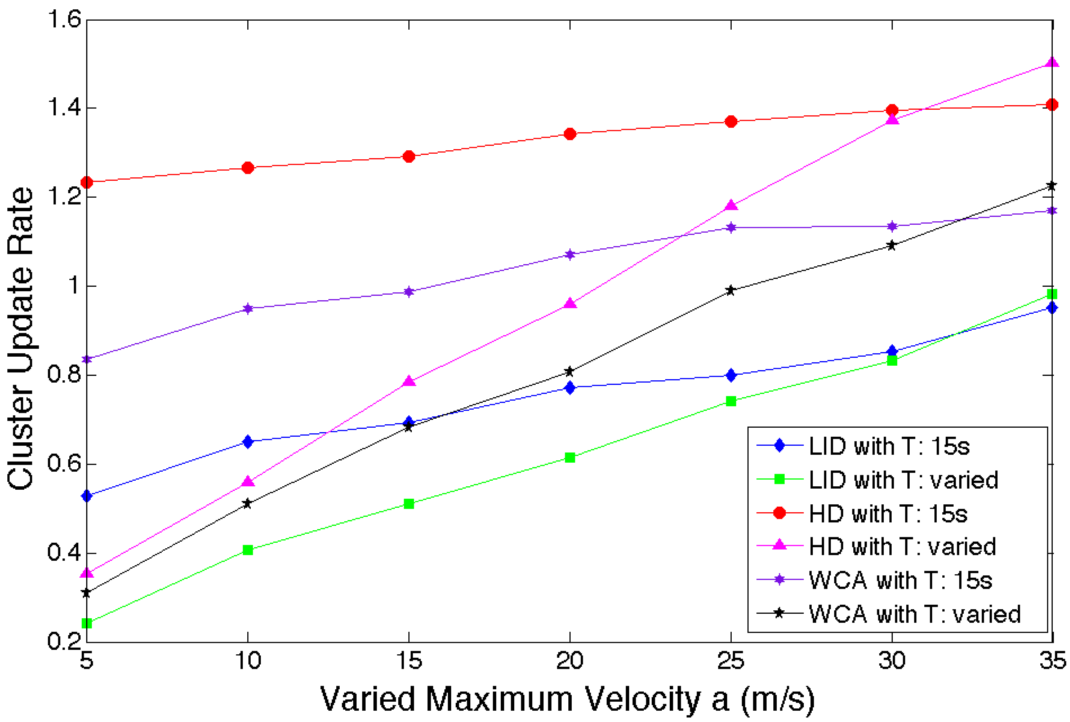

Figure 5.2 Variation of Cluster Head update rate with maximum node

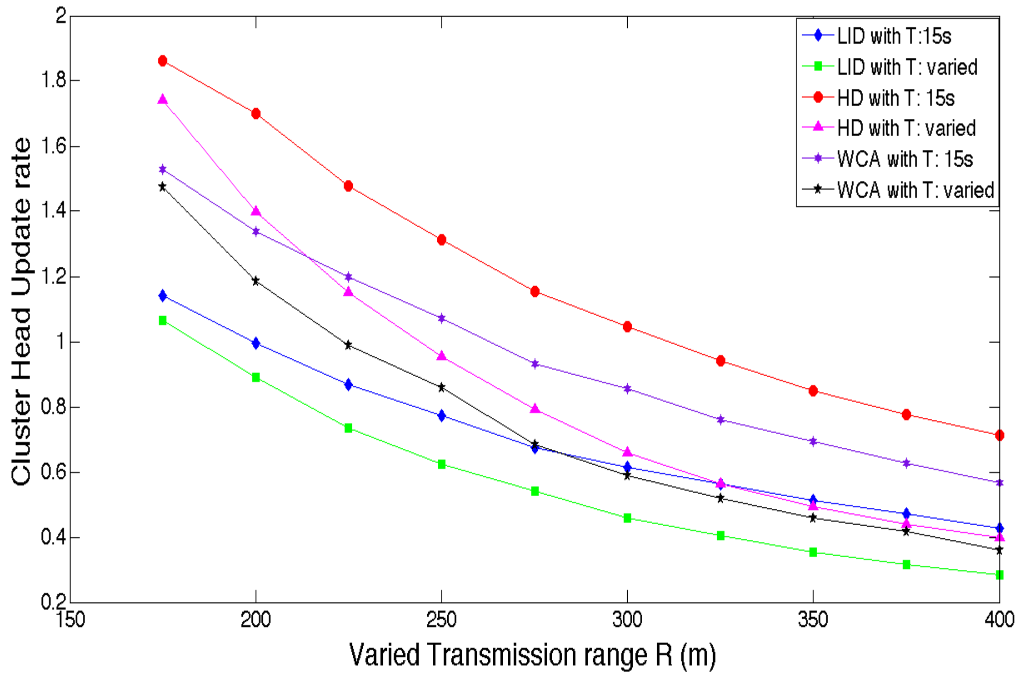

Figure 5.3 Variation of cluster head update rate with transmission range

The second part of simulation study is the analysis of member exchange rate. Through simulation results in Fig. 5.4 and 5.5, we can observer that the member exchange rate under constant cluster update interval and adaptive update rate is very close. In most cases, the cluster member exchange rate under the adaptive cluster update rate decided by our proposed model is a little smaller than that of constant cluster update rate. This is a good proof that adaptive cluster update period decided by expected link lifetime model can enhance the relative stability of cluster structure while improving efficiency of cluster update rate. Therefore, cluster update rate can be reduced and member exchange rate is not affected. It means that the adaptive cluster update approach can reduce unnecessary cluster updates which results in larger communication delay and smaller packet delivery ratio and decreases communication overhead by keeping relatively stable route or link.

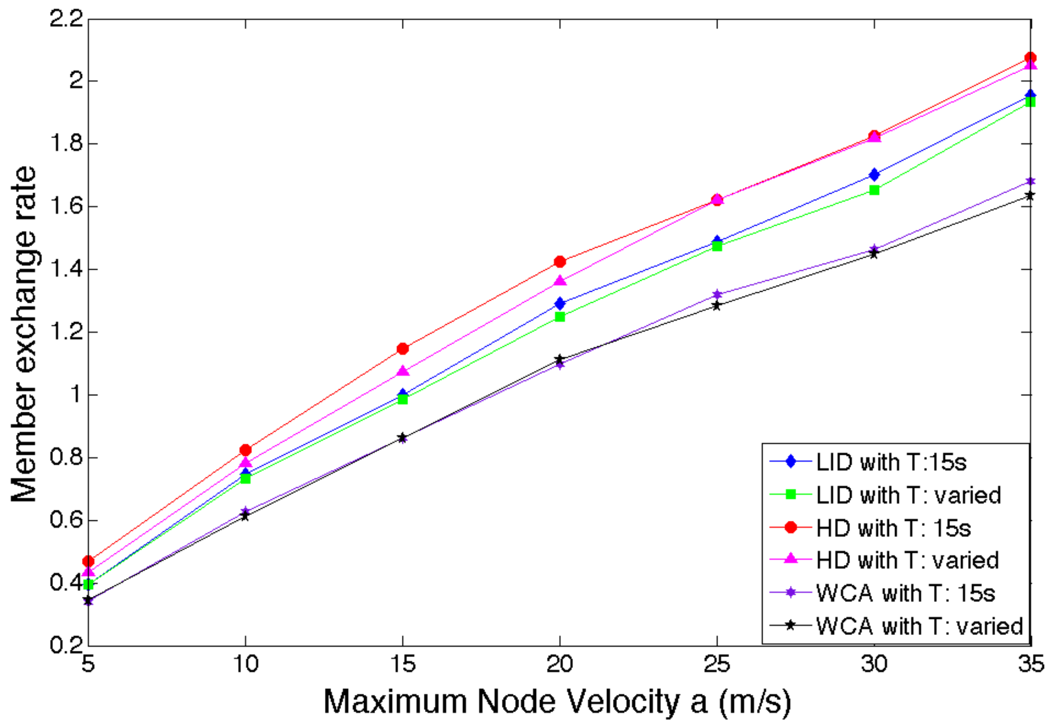

Figure 5.4 Variation of Member exchange rate with maximum node speed

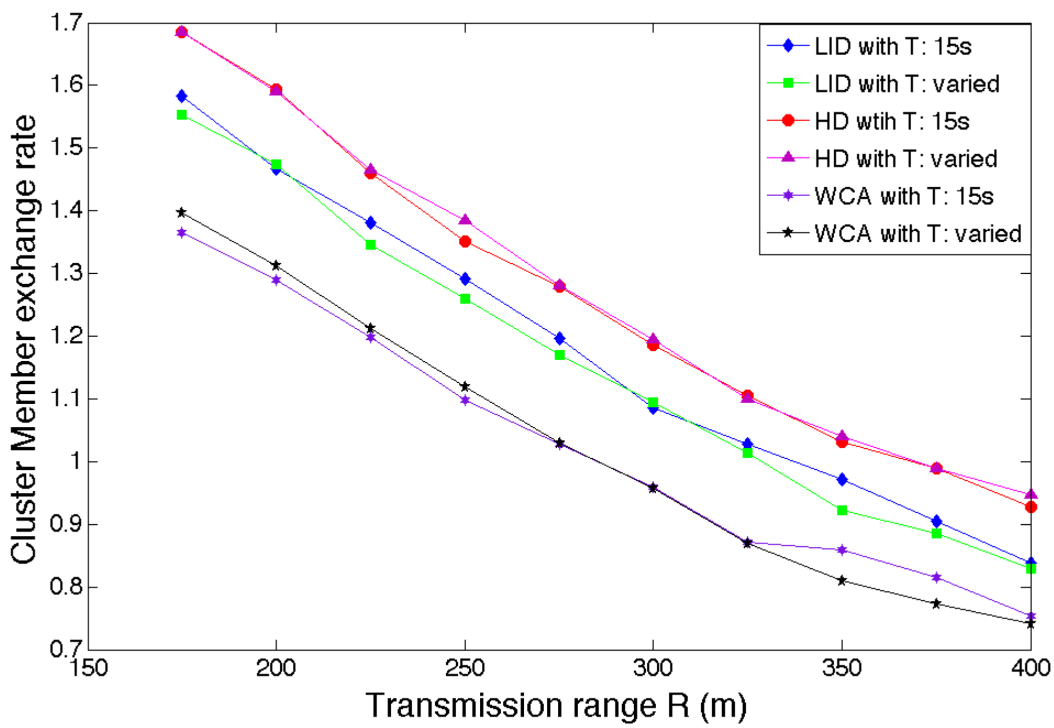

Figure 5.5 Variation of Member exchange rate with transmission range

5.3.2 Performance comparison with others algorithms

In section 5.3.1, we demonstrate the effectiveness of the expected link lifetime model and we will focus on analyzing performance of RLCH with respect to other algorithms. To make a comparison, three different clustering algorithms (LID, HD and WCA) are compared and discussed. For each of the defined metrics, we conduct our simulations to examine the effects of nodal velocity and transmission range for different protocols.

In Figs 5.6 and 5.7, we study the first metric, which is the cluster head update rate. We can observe that our proposed RLCH can achieve better performance compared to the other three algorithms. Even thought HD and LID algorithms are easier to operate by just considering the nodal degree or fixed ID, they cannot guarantee the relative stability of clustering structure. Without consideration of the impact of other factors such as velocity and transmission range makes HD and LID algorithms cannot adapt to high mobility or small transmission range scenarios. RLCH makes a trade-off between different factors like mobility, nodal degree and transmission range. The cluster head update rate seems close to that obtained by WCA when the maximum node velocity a is high, i.e. (40-50m/s) or when R is small (about 150m). This is because large mobility or small transmission range results in small link lifetime and cluster update interval T. The smaller T results in frequent cluster updates that may reduce the performance of RLCH.

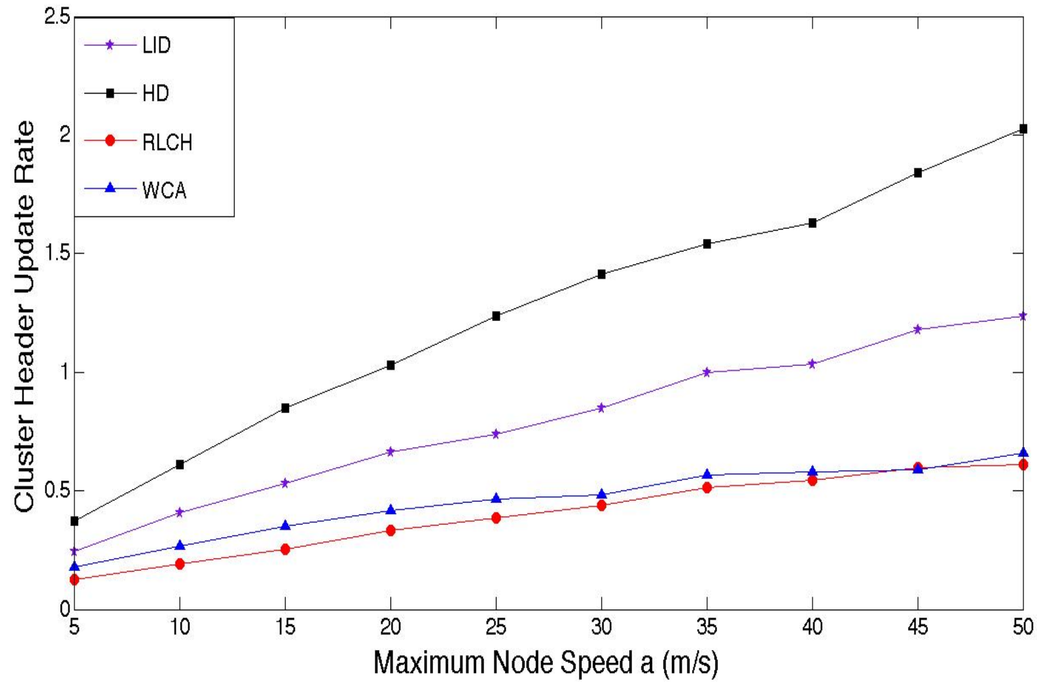

Figure 5.6 Variation of Cluster head update rate with node speed

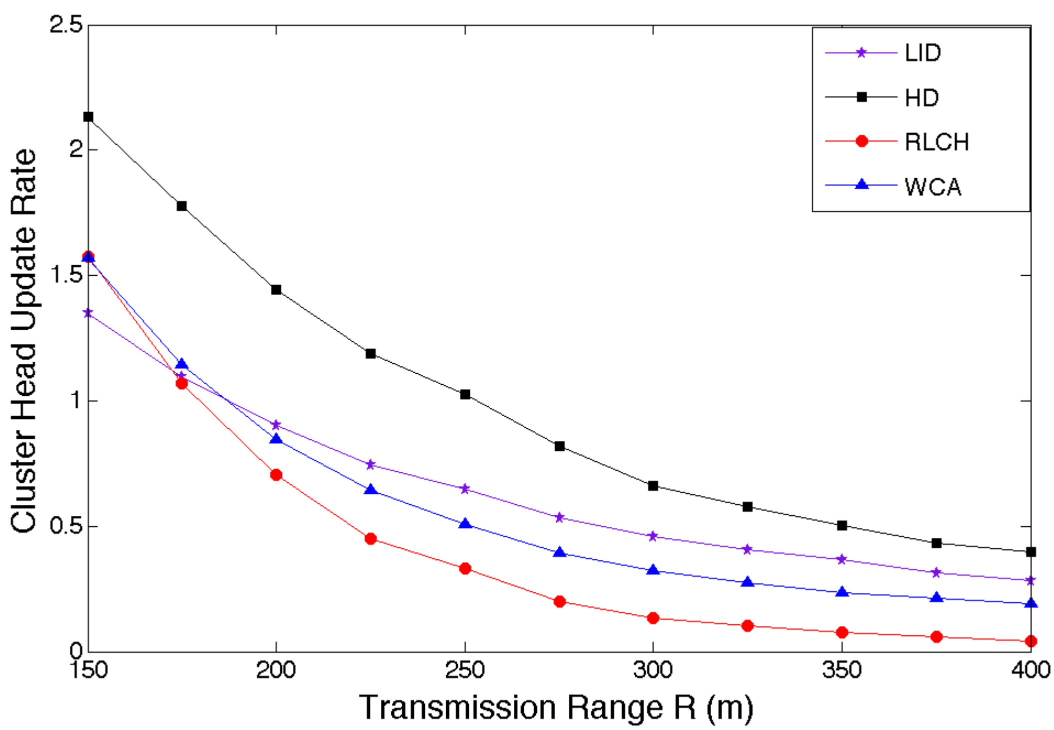

Figure 5.7 Variation of Cluster head update rate with transmission range

The second part is the analysis of member exchange rate. The simulation results in figures 5.8 and 5.9 show that RLCH has a smaller member exchange rate compared with other three algorithms in most cases.

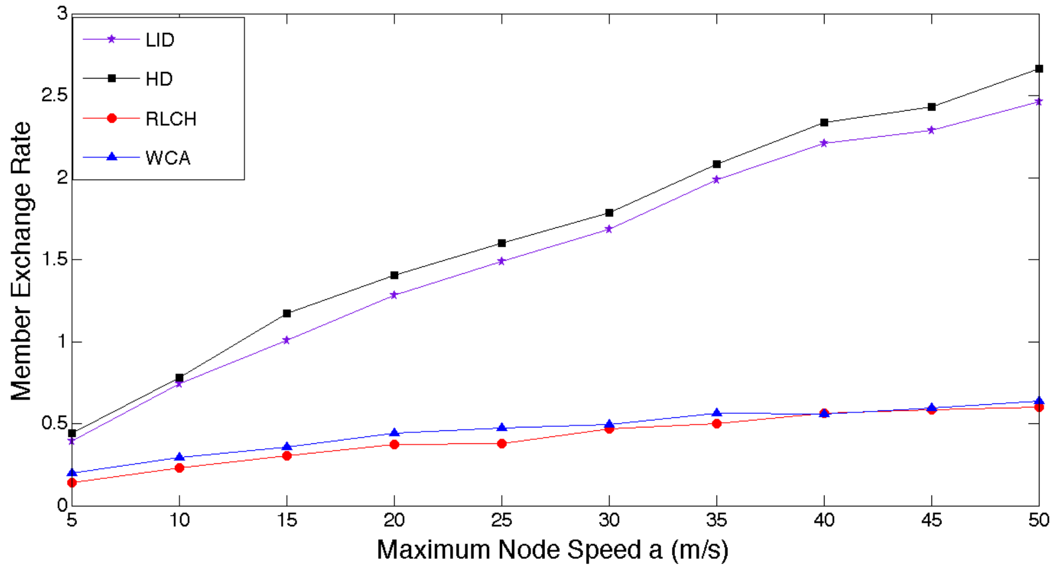

Figure 5.8 Variation of Member exchange rate with maximum node speed

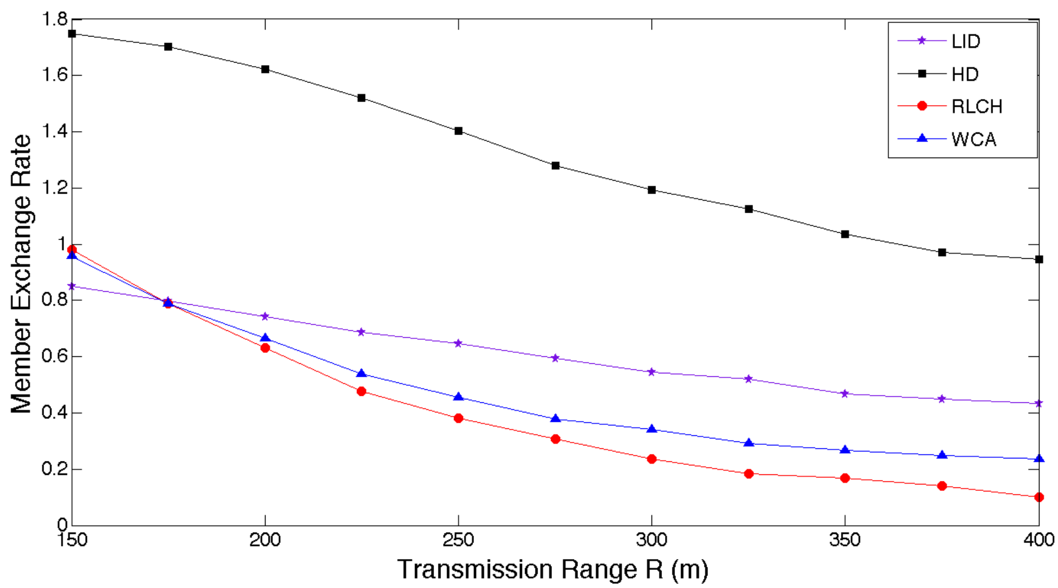

Figure 5.9 Variation of Member exchange rates with transmission range

Since HD, LID and WCA algorithms have a relatively higher rate of cluster update rate compared to the RLCH algorithm as represented in Figs 5.6 and 5.7, RLCH algorithms can achieve small rate of member exchange rate between clusters by prolonging average lifetime of the cluster structure.

Also it is easy to explain the reduction of cluster update rate and member exchange rate with the higher transmission range and smaller maximum nodal velocity. When the nodal velocity is small; the local cluster structure is relatively stable and approaches to a stable condition when nodal speed goes to zero. When the transmission range increases, the large transmission range reduces the impact of mobility on the cluster structure. Stable cluster structure results in smaller value of cluster update rate and member exchange rate.

The third part is related to the remaining energy distribution as shown in figures 5.10 and 5.11. Through the simulation results, we conclude that the proposed RLCH can save energy. This is because RLCH has relatively small number of clusters (shown in the fourth set of results) and lower rate of cluster update. Moreover, the energy distribution is even throughout the network. This can be attributed to the cluster head update interval adaptation scheme described in Section 4.4. In LID algorithms, nodes with small ID have higher probability to be selected as CH, which results in over consumption of nodal energy and decrease network lifetime. Also HD algorithm is able to control the amount of the cluster head, which fails to consider the influence of remaining energy on cluster head selection and update interval. Compared with WCA algorithm, instead of assigning weight factor of remaining energy, RLCH can control the time of being cluster head by updating cluster structure periodically. Meanwhile this cluster update interval also reflects the impact of remaining energy on selecting CH.

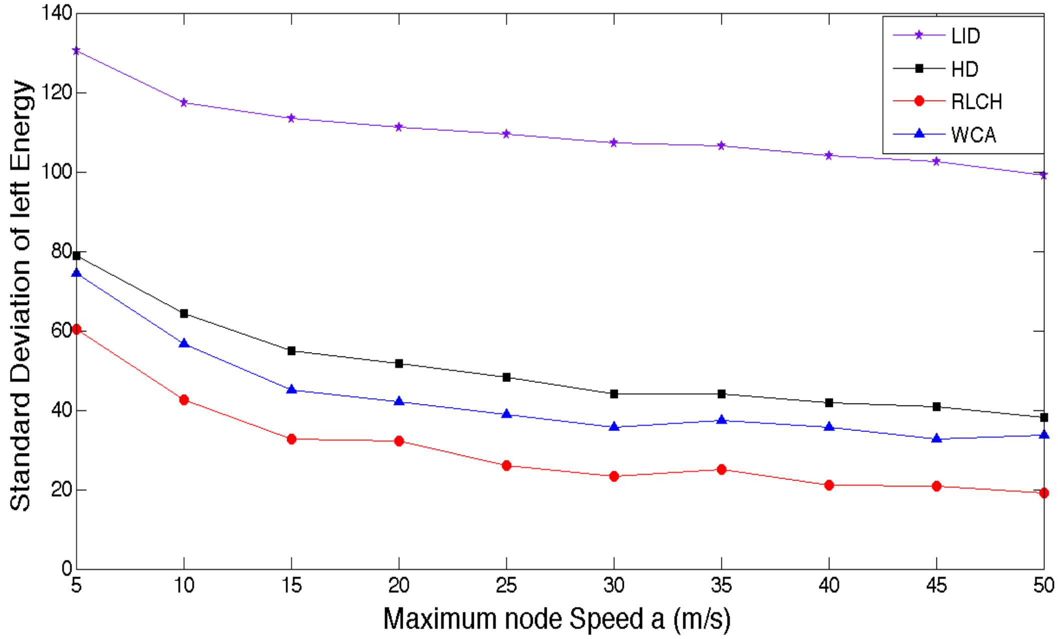

Figure 5.10 Standard deviation of remaining energy with varied maximum node speed

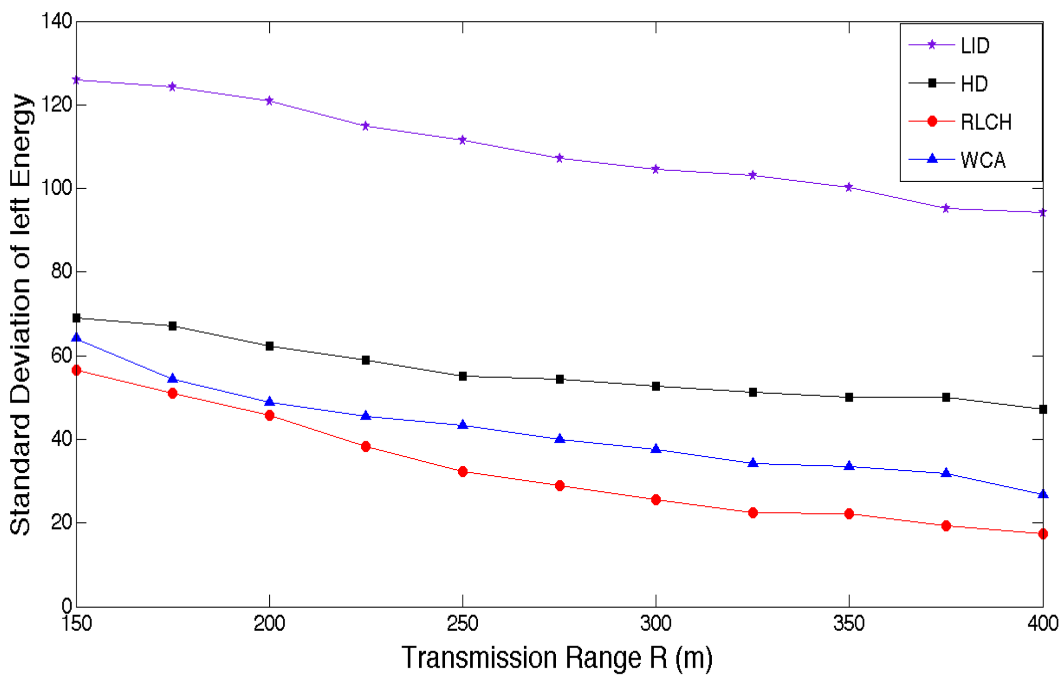

Figure 5.11 Standard deviation of remaining energy with varied transmission range

In Figs 5.12 and 5.13, we observe that there are fewer clusters in RLCH compared to LID, HD and WCA. That is because, RLCH can construct relatively stable clustering structure, which reduces the probability of generation of new cluster heads. When the transmission range is fixed and velocity is small, RLCH algorithm can achieve a better performance than WCA; because it is not easy to identify optimal weight combination for WCA and RLCH can make a good trade-off between these factors. However, the improved performance degrades due to high mobility.

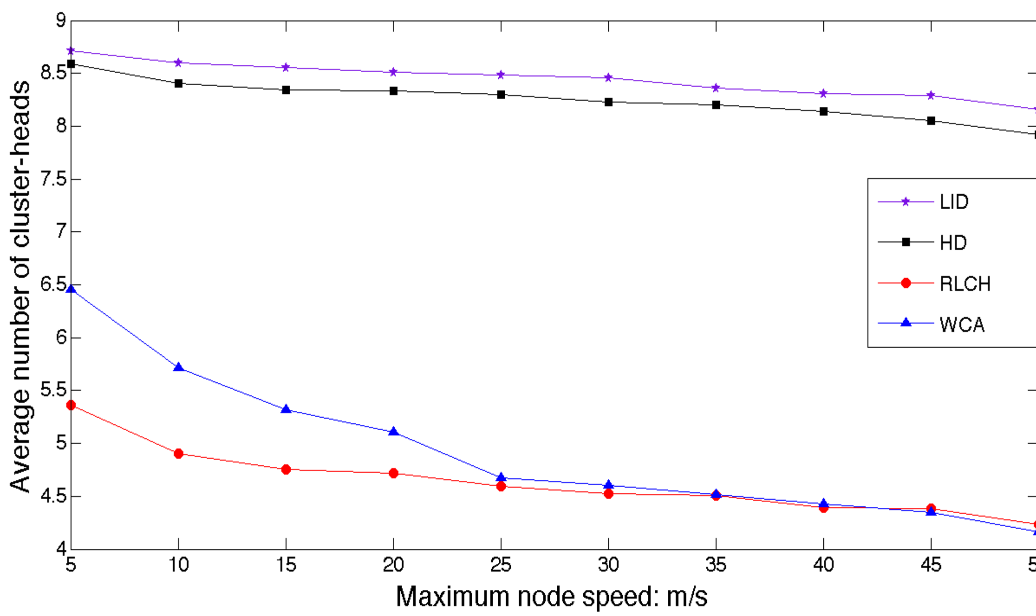

Figure 5.12 Variation of average number of cluster head with maximum nodal speed

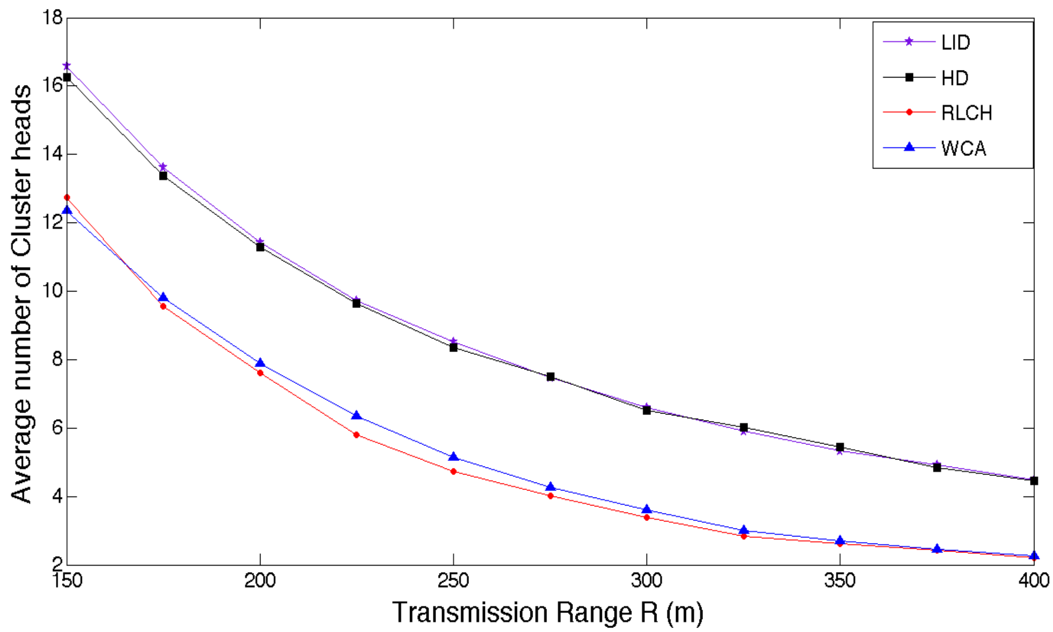

Figure 5.13 Variation of average number of cluster heads with transmission range

High mobility within finite transmission range triggers frequent cluster head contention and increase cluster update rate (i.e., two cluster heads are not allowed to exist within one hop or two clusters cannot be directly connected [36],[37]). Therefore, total number of cluster heads decreases as shown in these figures, and cluster update rate increases are shown in Fig. 5.6. When transmission range increases, average number of nodes within one cluster increases due to the assumption that nodes are uniformly distributed within the network. With large degree of CH and fixed network size, the number of CHs decreases. Theoretically the amount approaches one when the transmission range approaches infinity.

Chapter 6

Conclusion and Future work

In this thesis, we analyze the impact of nodal mobility on the stability of link and clustering structure. Different factors such as nodal velocity, relative distance, remaining energy and nodal degree are considered to determine suitable clustering update interval and select cluster head. To make a trade-off between these factors with no weights combination needed, we propose a novel cluster head election algorithm, which uses an area-based approach to approximate the requirements of the double M principle. These requirements lead to better cluster stability, reduction in the number of clusters in the network, and improving the energy efficiency. We establish a link lifetime model to estimate the lifetime of each link and propose a maximum cluster head update interval (link lifetime) model to identify the clustering update frequency. Also, a simple and effective energy thresholding algorithm is combined with expected link lifetime model to consider the impact on nodal energy. Finally, four metrics are adopted to evaluate the performance of RLCH compared to other algorithms.

With different nodal mobility distributions and transmission ranges, RLCH can achieve relatively smaller cluster update rate, member exchange rate, energy consumption and number of cluster heads in simulation. Simulation results confirm that RLCH can establish much stable clustering structure, save energy and can quickly adapt to unknown environments, when comparing to LID, HD and WCA algorithms.

Also we conclude that our proposed model can serve as a good heuristic to determine a suitable update interval for clustering structure locally. We can avoid the un-necessary bandwidth waste, improve communication quality and guarantee the relative stability within proposed cluster update interval

For future work, the effectiveness of RLCH algorithm will be discussed when used in mesh-based multicast protocol like ODMRP. By comparing to other algorithms [38-41], we can confirm in further that RLCH algorithm outperforms other algorithms by evaluating some metrics such as transmission delay, packet delivery ratio and throughput.

Since the actual activity of mobile node is unknown and hard to predict, it is different to model the activity of mobile user in wireless ad hoc network and many assumptions in theoretical model are doubted. Different mobility model such as random walk will be studied to verify the effectiveness of expected link lifetime as a heuristic of clustering update interval and link lifetime cluster head election algorithm. Even we can try to study the performance of RLCH under the real collected traffic data of vehicle and modify this algorithm.

Bibliography

[1] Zhao, S., and Raychaudhuri, D. 2009. "*Scalability and Performance Evaluation of Hierarchical Hybrid Wireless Networks,"* IEEE/ACM Transactions on Networking (Oct, 2009), vol.17, No.5.

[2] R. Ramamnathan and S. Steenstrup, *"Hierarchically organized, multi-hop mobile wireless networks for quality-of-services support,"* Mobile Networks and Application, vol.3 no. 1, June 1988.

[3] Carlsson, G and Memoli, F. *"Characterization, Stability and Convergence of Hierarchical Clustering Methods,"* The Journal of Machine Learning Research, Mar 2010.

[4] Wu, Y and Wang, W.Y. *"MEACA: Mobility and Energy Aware Clustering Algorithm for Constructing Stable MANETs,"* Military Communications Conference (MILCOM), 2006.

[5] Wang, C.G, Yu, Y.P, Xu, Y.X, Ma, M.J and Diao, S.B. *"A Multi-Hop Clustering Protocol for MANETs"*. Wireless Communications, Networking and Mobile Computing, 2009.

[6] Iwanicki, K and Steen, M. *"Multi-hop Cluster Hierarchy Maintenance In Wireless Sensor Networks: A Case for Gossip-Based Protocols,"* EWSN'09 Proceeding of the 6^{th} European Conference on Wireless Sensor Networks. 2009.

[7] Lee, S.J., Gerla, M., and Chiang, C.C. 1999. *"On-demand multicast routing protocol,"* Wireless Communications and Networking Conference, vol.3, 1298-1302.

[8] Agarwal, R. and Motwani, M. *"Survey of clustering algorithm for MANET,"* International Journal on Compute Science and Engineering, vol.1(2) (2009), pp. 98-104.

[9] Gerla, M. and Tsai, J.T. 1995. "*Multiuser, Mobile, Multimedia Radio Networks*," Wireless Networks, vo1. 1(Oct. 1995).

[10] Amis, A.D., Prakash, R., Vuong T.H.P., and Huynh, D.T. 2000. "*Max-Min D-Cluster Formation in Wireless Ad Hoc Networks,"* In proceedings of IEEE Conference on Computer Communications (INFOCOM), vol.1. pp. 32-41.

[11] Chen, G., Nocettj, F., Gonzalez, J., and Stojmenovic, I. 2002. "*Connectivity based k-hop clustering in wireless networks,"* In Proceeding of the 35th Annual Hawaii International Conference on System Sciences, vol.7, pp.188.3

[12] Ohta, T., Inoue, S., and Kakuda, Y. 2002. "*An Adaptively Multihop Clustering Scheme for Highly Mobile Ad Hoc Network*s," In proceeding of 6th ISADS'03, (Apr. 2003).

[13] C.-C. Chiang et al., *"Routing in Clustered Multihop, Mobile Wireless Networks with Fading Channel,"* Proceeding of IEEE SICON' 97, 1997.

[14] T. J. Kwon et al., *"Efficient Flooding with Passive Clustering an Overhead-Free Selective Forward Mechanism for Ad Hoc/Sensor Networks,"* in proceeding of IEEE, vol. 91, no.8, pp. 1210-20, Aug 2003.

[15] Chatterjee, M., Das, S.K., and Turgut, D. 2000. *"An On-Demand Weighted Clustering Algorithm (WCA) for Ad hoc Networks,"* In proceeding of IEEE Globecom'00, pp. 1697-701.

[16] Wu, H., Zhong, Z.D., and Hanzo, L. 2010. *"A Cluster head Selection and Update Algorithm for Ad Hoc Networks,"* In the proceeding of IEEE Gobecom.

[17] Wang, Y.X., Bao, F.S. 2007. *"An Entropy-Based Weighted Clustering Algorithm and Its Optimization for Ad Hoc Networks,"* Third IEEE International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob 2007).

[18] S.K. Dhurandher and G.V. Singh. *"Weight-based clustering in wireless ad hoc networks,"* IEEE 2005.

[19] Tolba, F.D., Magoni, D., and Lorenz, P. 2007. *"Connectivity, energy & mobility driven Weighted clustering algorithm,"* In proceeding of IEEE Globecom.

[20] Amis, A.D., and Prakash, R. 2000, *"Load-Balancing Clusters in Wireless Ad Hoc Networks,"* In proceeding of 3rd IEEE ASSET'00 (Mar. 2000), 25-32

[21] Younis, O., and Fahmy, S. 2004. *"Distributed Clustering in Ad-hoc Sensor Networks: A Hybrid, Energy-Efficient Approach,"* IEEE Transactions on Mobile Computing, vol.3 (Dec, 2004), 366-379.

[22] Banerjee, S and Khuller, S. 2001. *"A Clustering Scheme for Hierarchical Control in Multi-hop Wireless Networks,"* INFOCOM, 1028-1037 vol.2.

[23] Heinzelman, W.R., Chandrakasan, A., and Balakrishnan, H. 2000. *"Energy-Efficient Communication Protocol for Wireless Sensor Networks,"* Proceeding of the 33th Hawaii International Conference on System Science.

[24] Bandyopadhyay, S and Coyle, E.J. 2003. *"An Energy efficient hierarchical clustering algorithm for wireless sensor networks,"* INFOCOM (April 2003), pp. 1713-1723.

[25] J Wu, F. Dai, M. Gao and I. Stojmenovic. *"On Calculating Power-Aware Connected Dominating Sets for Efficient Routing in Ad Hoc Wireless Networks,"* Journal of communication and networks, vol 4, no.1, mar 2002.

[26] Huang, Y.S, Mao, S.W and Li, Y.H. 2011. *"Downlink power control for VBR video streaming in cellular networks: a majorization approach,"* In Proc. IEEE GLOBECOM (Dec, 2011), pp. 1-6, Huston, TX.

[27] Huang, Y.S and Mao, S.W. 2011. *"Downlink power control for variable bit rat video over multicell wireless networks,"* In proc. IEEE INFOCOM (April 2011), pp.2561-2569, Shanghai, China.

[28] Huang, Y.S, Mao, S.W and Li, Y.H. *"Downlink power allocation for stored variable-bit-rate videos,"* Invited paper, the 7th international Conference on Heterogeneous Networking for Quality, Reliability, Security and Robustness (Qshine'10), Houston, TX, Nov 2010, in Spring Lecture Notes of the Institute for Computer Science, Social Informatics and Telecommunications Engineering (LNICST), vol. 74, pp.423-438, 2011.

[29] Hu, D. and Mao, S.W. *"Multicast in femtocell networks: a successive interference cancellation approach,"* In Proc. IEEE GLOBECOM 2011, Houston, TX, Dec. 2011.

[30] Samar, P., and Wicker, S.B. 2006. *"Link Dynamics and Protocol Design in a Multihop Mobile Environment,"* IEEE Transactions on Mobile Computing, vol.5 (Sep, 2006).

[31] Wu, X.R,. Sadjadpour, H.R., and Garcia-Luna-Aceves, J.J. 2007. *"Link Lifetime as a Function of Node Mobility in MANETs with Restricted Mobility: Modeling and Applications,"* Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks and Workshops. WiOpt 2007.

[32] Xu, Y., and Wang, W.Y. 2007. *"Stability of Hierarchical Mobile Ad Hoc Networks,"* Global Telecommunications Conference, 2007. GLOBECOM (Nov, 2007), 1232-1236.

[33] NS-2: The Network Simulator, http://www.isi.edu/nsnam/ns, 2006

[34] Random Waypoint Model: http://www.netlab.tkk.fi/~esa/java/rwp/rwp-model.html

[35] Goldsmith, A. (2005). *Wireless Communications*. Cambridge University Press.

[36] Rezaee, M., and Yaghmaee, M. 2009. *"Cluster based Routing Protocol for Mobile Ad Hoc Networks,"* Feb, 2009.

[37] Sun, X.M., Liu, W.J., Zhang, Z.Q., and Zhao, Y. 2006. *"CODMRP: Cluster-Based On Demand Multicast Routing protocol,"* Wireless Communications, Networking and Mobile Computing (WiCOM). Sep, 2006.

[38] Jun, L and Liao, B. *"Hybrid Zone based Multicast Routing Scheme for Mobile Ad Hoc Network,"* Wireless Communications, Networking and Mobile Computing 2008. WICOM'08. 4th International Conference on Digital Object Identifier, 2008.

[39] Pathirana, D and Kwon, *"M. RODMRP: Resilient On-Demand Multicast Routing Protocol,"* 21st International Conference on Advanced Information Networking and Applications Workshops (AINAW'07).

[40] Begillo, S.J, Mohamamdzadeh, H., Jamali, S and Norouzi, A. *"Stable Route Selection in ODMRP with Energy Based Strategy,"* In Personal Indoor and Mobile Radio Communications (PIMRC), pp. 26-30, Sept 2010.

[41] Xiang, X.J, Zhou, Z and Wang X., *"Robust and Scalable Geographic Multicast Protocol for Mobile Ad Hoc Networks,"* In Proc of IEEE INFOCOM (May 2007), pp. 2301-2305.