國立台灣大學資訊管理研究所碩士論文

指導教授：林永松 博士

顏宏旭 博士

無線移動網路下考量穩定叢聚網路拓樸建置與相關服務品質限制路由演算法

Reliable Cluster Construction and QoS-Constrained Routing Assignment in Wireless Mobile Ad Hoc Networks

研究生： 林明源 撰

中華民國 九十四年七月

無線移動網路下考量穩定叢聚網路拓樸建置與相關服務品質限制路由演算法

Reliable Cluster Construction and QoS-Constrained Routing Assignment in Wireless Mobile Ad Hoc Networks

本論文係提交國立台灣大學

資訊管理研究所作為完成碩士

學位所需條件之一部分

研究生： 林明源 撰

中華民國 九十四年七月

謝　誌

本論文的完成，意味著研究所求學過程的結束，也是人生另一個階段的開始。首先要感謝的是我的父母：林信雄先生與陳金蓮女士，有您們兩位全力的支持與鼓勵，提供良好的環境，使我能順利完成學業。

在台大資訊管理系與資訊管理研究所的求學過程中，感謝許多指導過我的老師的傳道與教誨，讓我在專業知識與技能上能有所成長。感謝恩師 林永松博士，大學專題的指導老師，與研究所兩年的指導，帶領著我深入網路的不同研究領域。感謝您總是耐心的聆聽與指導我的問題，提供良好的建議與方向，使本研究論文能更完備、完善。這三年多來在老師身邊，不僅學習到您實事求是與嚴謹的研究精神，在待人處世方面也多有所啟發。感謝恩師 顏宏旭博士這兩年研究所的指導與教誨，有您細心的聆聽我的問題，給予指正與指導，不僅令我獲益匪淺，也使本論文得以如此完整，成果如此豐碩。

本論文的完成，承蒙趙啟超博士、呂俊賢博士與林盈達博士於口試時對學生的指正與建議，使本論文能更盡善盡美。感謝大學導師蔡益坤博士在大學生活對我的關心與照顧，陪伴著我學習與成長，並在演算法方面給予我啟蒙我指導。

在網路最佳化實驗室的兩年，特別感謝演福學長給予我的指正與建議，並時常耐心地與我一起討論，感謝孝澤這兩年來的陪伴與扶持，感謝書平、建宏、琳智，與你們的共研究、切磋，並一起完成碩士學業。感謝學弟妹中蓮、弘翕、文政、義倫和勇誠在論文口試期間的諸多幫忙與協助。

在高中樂旗隊與台大管樂團期間所認識的好朋友，感謝你們的陪伴、支持與鼓

勵，在論文苦悶難熬的時候，為我打氣與加油。感謝智超、馥瑄、俊雄、婉郁、宗源、俊陞等多位好友的鼓勵與支持，願我們友誼長存。

在台大資管待了六年，有著許多美好的回憶與難以言表的情感，相信我永遠不會忘記在這裡的點點滴滴。

林明源 謹 識

於台大資訊管理研究所

民國九十四年七月

論文摘要

論文題目：無線移動網路下考量穩定叢聚網路拓樸建置與相關服務品質限制路由演算法

學生：林明源 九十四年七月

指導教授：林永松 博士

顏宏旭 博士

隨著無線移動網路的發展，且為提供覆蓋式地無線傳輸與應用架構，如何確保網路拓樸的穩定性與效率性儼然成為網路管理的重要議題。然而，無線裝置的移動行為、無線傳輸的限制、脆弱的路由連結與不可預知的網路變動使得網路管理變得複雜與困難。在此分散式環境下，如何建置高度穩定性的網路拓璞與考量相關服務品質的路由指派也因而成為許多專家學者所研究的熱門議題。

本論文針對穩定叢聚網路拓樸與相關服務品質路由問題提出有效率且具彈性的設計方法。在我們提出的架構下，我們假設有一中央決策系統(例如:全球地理定位系統)可以監控整個無線網路並且散佈決策。藉由數學規劃的方式，我們將該問題模式化為一個整數最佳化問題，其目標函數為最大化建構網路與指派路由上的最短連結傳輸時間。如同其他傳統的叢聚問題，我們首先將無線裝置聚集成不同的叢聚，並決定叢聚首與叢聚成員之間的歸屬關係。接著，藉由建置出的叢聚架構，我們決定路由指派並符合相關路由限制，例如:節點容量限制與點對點傳輸延遲限制。相對於其他演算法，不同的地方在於，對於一個過載的網路節點，我們計算它的網路流量總和，重新路由擁塞的傳輸路徑，以達到負載平衡與最佳化網路的效能和穩定性。

由於該問題的複雜性與困難度，我們採用拉格蘭日鬆弛法作為我們的解題方法。藉由該方法優越的特性與我們所提出的演算法，我們可以有效率的解決這複

雜的最佳化問題，並且不斷地最佳化我們的決策品質。

關鍵字：無線移動網路、叢聚拓樸、網路規劃、考量服務品質之路由規劃、穩定性、最佳化、拉格蘭日鬆弛法、數學規劃

THESIS ABSTRACT

GRADUATE INSTITUTE OF INFORMATION MANAGEMENT

NATIONAL TAIWAN UNIVERSITY

NAME: MING-YUAN LIN　　　　MONTH/YEAR: JULY,2005

ADVISORS: DR. FRANK YEONG-SUNG LIN and

DR. HONG-HSU YEN

RELIABLE CLUSTER CONSTRUCTION AND QOS CONSTRAINED ROUTING ASSIGNMENT IN AD HOC WIRELESS MOBILE NETWORKS

With the development of Mobile Ad Hoc Networks (MENETs), providing ubiquitous communications and a convenient framework for applications requires network management to guarantee that the network topology is reliable and efficient. However, the mobility of wireless devices, wireless communication limitations, frequent route breakdowns and unpredictable topology changes make the network management complex and difficult. In a distributed environment, how to construct a network topology and QoS constrained routing assignment with high stability has thus become a popular issue.

In this thesis, we attempt to solve the problem of reliable cluster construction and the QoS constrained routing assignment. We assume that there exists a central decision system, such as a Geographical Positioning System (GPS), to monitor the entire wireless network and disseminate information. By using a mathematical

technique, we model the problem as an integer optimization model, where the objective function is to maximize the minimum link duration of the constructed network topology and routing assignment. Like conventional clustering problems, we first group devices into different clusters and determine the clusterhead/cluster member relationship. Based on the constructed cluster topology, we jointly determine the routing assignment with QoS constraints, such as nodal capacity and end-to-end delay. The difference between our proposed algorithm and other algorithms is that for a heavily-loaded node, we aggregate the traffic demands of all O-D pairs and reroute some congested routing paths to achieve load balance and optimize the utilization and stability of the network.

Because of the difficulty and complexity of the optimization problem, we adopt Lagrangean Relaxation and the subgradient method. By applying the latter method's properties and getting a primal heuristic, we can solve the complicated optimization problem efficiently and improve the solution quality iteration by iteration.

Keywords: Mobile Ad Hoc Network, Reliable Cluster Construction, QoS Constrained Routing Assignment, Reliability, Mathematical Programming, Optimization, Lagrangean Relaxation

Content

List of Figures

List of Tables

Chapter 1 Introduction

1.1 Background

Recently, the growth of mobile devices usage has made the field of mobile ad hoc networks prevalent and important. The advances in hardware design, communication equipment, and increasing user requirement for mobility and geographical dispersion have placed enormous demands on ad hoc wireless mobile networking and computing **[2]**. Mobile ad hoc networks are widely used for many wireless applications, such as tactical missions, emergency rescue operations, rapid response systems, electronic classroom networks. In these applications, mobile ad hoc networks are ideal topologies for instant, ubiquitous communications and disseminating information without deploying a fixed infrastructure first.

In mobile ad hoc networks, any device with microprocessor and communication equipment, which can be mobile or stationary, is potentially depicted as a node. These devices include mobile telephones, motor vehicles, roadside information stations, and desktops or handheld computing devices **[2]**. A wireless link can be established between a pair of nodes only if they are within wireless transmission range of each other. Compared to traditional wired networks, there is no fixed infrastructure for coordination, scheduling, and resource allocation in wireless ad hoc networks. Hence, traditional multicast and routing protocols which are designed for wired networks can not be suitable in mobile ad hoc networks for the following reasons: (i) the routes in mobile ad hoc networks change frequently; (ii) the central infrastructure is not available; and (iii) the wireless communication limitation, such as the power, coverage and bandwidth are not sufficient **[1]**. In this kind of distributed environment, any device for transmitting information should be programmed to ask neighboring nodes for assistance in forwarding a packet. All nodes have to make decisions collectively, and the routing process is necessary to be progressing through multi-hops. This means that the traffic source node first selects some intermediate node from its neighboring nodes for forwarding the packet. Then, the selected relay node becomes a new source and is eligible to select the next intermediate node for forwarding the received packet until reaching the destination node. In this scenario, more number of relay nodes incurs more transmission delay; and the power consumption of the intermediate node for processing and communication would reduce its battery capacity. Besides of the power limitation, the available transmission rate of the wireless node is usually limited. It is essential and necessary to well control and schedule the traffic load of different source/destination pairs to different wireless nodes for the avoidance the transmission collision in some heavily-loaded node. Due to bandwidth and power limitation, the lifetime of a wireless network is usually evaluated by the residual battery capacity (lifetime) of its nodes. Therefore, efficient power utilization of all relay nodes and the routing assignment should be included in mobile ad hoc networking and computing.

In addition, in mobile ad hoc networks, the wireless link and the routing assignment are still unreliable and fragile because of the mobility of the device or the exhaustion of the device's battery capacity. The mobility of devices enlarges the distance of a node pair and causes the link to be disconnected. Besides, the exhaustion of the device's battery capacity makes the device can not be reached anymore. Such networks are envisioned to have dynamic, sometimes rapidly-changing, random, multi-hops topologies which are likely composed of relatively bandwidth-constrained wireless links as well as the unidirectional links existing because of wireless communication limitation. In these situations, topology change and breakage routing assignment may need extra communication and processing overheads to reconstruct the network topology and decide the routing assignment again. In reality, they will even become a disaster, due to packet loss and delay, such as loss of commands in a military network or loss of contact of rescuing teams in emergency situations. Hence, in mobile ad hoc networks, how to construct a temporary, reliable, and well-controlled communication networks is an important and significant issue. The solution is usually evaluated by the reliability and stability of the constructed network.

As mentioned above, the mobility of wireless devices usually makes the link unreliable and fragile. Frequent route breakage and unpredictable topology changes also make the network inherently unscalable with respect to number of nodes, control overheads, degree of mobility, or network density **[2]**. With the fixed transmission range, the mobility of a node pair changes the distance between each other. Thus, if the distance exceeds the transmission range, the link would be disconnected and not stable enough to finish the packet transmission. This will cause packet loss, retransmission and more communication overheads. In this scenario, it is costly to reconstruct the network topology and reroute the packet because of high communication and control overheads to collect updated network information, connect the decomposed segments, and disseminate the new routing assignment. Hence, how to predict the mobility of mobile nodes and evaluate the link duration of the route path is another critical issue for the network management of mobile ad hoc networks.

There are two suggested approaches for designing the routing and multicasting protocols in mobile ad hoc networks. The first is constructing a reliable multicast routing topology with minimum power consumption and related QoS constraints, such as tree or mesh structures. Second is dividing the network into autonomous zones and electing respective coordinators to manage the transmission and the routing assignment regionally. This approach is called as clustering or grouping. The clustering approach first partitions nodes into sub-sets according to the similarity of nodes. Then, we determine the cluster relationship of nodes and routing assignment based on the constructed topology. Because of location-based factors like regional network management, resource allocation, and scalability, the later is usually adopted by much research. However, due to the mobility issues and wireless transmission limitation, there are still many challenges and a lot of room for improvement. In this paper, we adopt the clustering approach with consideration of related mobility issues to design a reliable, stable network topology and the routing assignment with lower packet loss and delay.

1.2Motivation

As the demand for wireless communication and computing in mobile ad hoc networks has increased, reliable routing and multicasting protocols have become major research topics. However, because such networks have dynamic, sometimes rapidly changing, random, multi-hops topologies, it is difficult to construct a reliable and well-controlled network topology. In addition to limited network resources and the lack of a central infrastructure, the movement of nodes, location management functions, stability and scalability of network management also create complex problems in mobile ad hoc networks.

As mentioned above, for location management and scalability of networks, the clustering approach and hierarchical structure are attractive and usually adopted. Many researchers have proposed heuristic approaches or protocols for cluster construction, but they circumvent the mobility issues by comparing the packet lost and delay with each other. Although mobility information and node capacity may be considered as the criteria of cluster construction, it is still insufficient, as mobility needs to be measured quantitatively. Besides, it is also necessary to quantify the negative effects or penalties caused by the improper routing assignment, processing and communication overheads to reconnect the segmented network topology and disseminate the new routing decision.

Hence, in this paper, we propose a mathematical formulation to construct a reliable cluster topology and the QoS-constrained routing assignment in mobile ad hoc network. In this model, we use the Gauss-Markov mobility model to describe the mobile behavior of nodes and the formula of ODMRP **[7]** predict the link duration at the decision instance. For nodal capacity limitation, we calculate the aggregate traffic working load of the node by summing the transmission rate of different O-D pair and then restrict that the aggregate traffic working load should not exceed the maximum available transmission rate of the node. In our mathematical equations, we denote the stability of the constructed cluster topology as objective function to evaluate the quality of computational results; and jointly determine the clusterheads, cluster members, the intra-clustering routing assignment of each clusterhead/cluster-member pair, and the inter-cluster routing assignment of each source/destination pair with related QoS constraints.

1.3 Literature Survey

In this section, we summarize some related concepts of reliable cluster construction in mobile ad hoc networks and divide them into four categories, namely, cluster construction, mobility research, integer formulation for clustering; and energy efficient- multicasting and wireless advantage.

1.3.1 Cluster Construction

Clustering is a method for organizing unlabeled nodes into groups (clusters) such that nodes within the same group are more similar to each other than to those in a different group **[2]**. The criteria for clustering are usually represented as feature vectors and evaluated by the common characteristics of nodes, such as similar mobility patterns, specified goals of team work, and geographical proximity. In conventional cellular networks, fixed base stations with special processing and communication capability are usually elected as coordinators of the partitioned sub-zones. In this scenario, clustering is used to group mobile stations with some base station and divide them into different corresponding sub-zones (cells). The base station of the cell acts as the coordinator (clusterhead) of the cluster. Then, each base station and adjacent mobiles stations build up a clusterhead/cluster-member relationship and construct a cluster topology. In wireless cellular networks, base stations connected by wired links form a reliable virtual backbone for the cross-cell routing assignment. Communication between a mobile station and a base station is only a single-hop away. For regional network management, such as, allocating channels and network resources to different cells, the clustering approach can achieve frequency reuse and good utilization of network resources.

In wireless mobile ad hoc networks, we use the clustering approach to group wireless devices into clusters to provide a convenient framework of applications such as routing, bandwidth allocation, scheduling, mobility, and regional management **[2]**. In each cluster, some node is elected as the leader of the cluster, called the clusterhead. It acts as a local coordinator and is responsible for the transmissions and network management of the group. In a homogeneous network, mobile nodes without special hardware could also be elected as clusterheads. However, in a heterogeneous network, the node with extra processing power or capacity would more likely be elected as a clusterhead. Within a cluster, each cluster member should determine at least one path to connect to its clusterhead for the maintenance of connectivity. Figure 1-1 illustrates the clusterhead/cluster-member relationship and the routing assignment within the cluster.

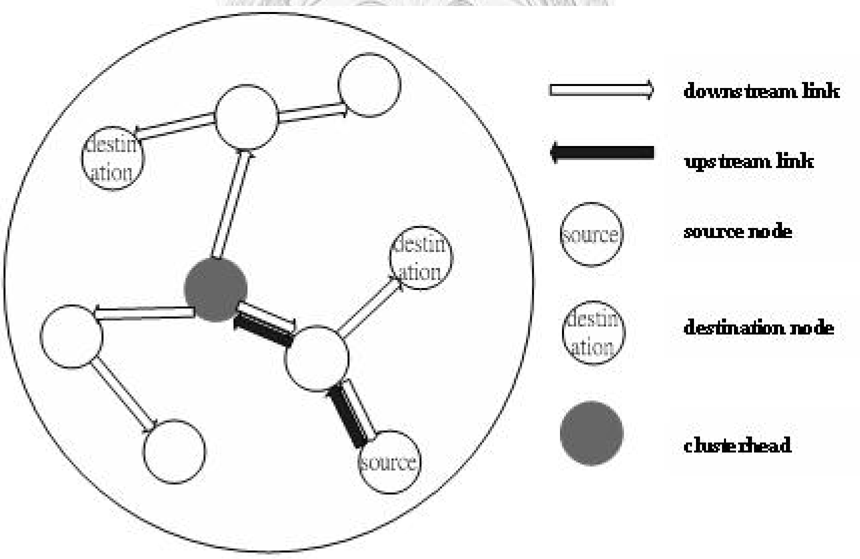

Figure 1-1 Intra-Cluster Transmission

In the figure, the gray node is elected as the clusterhead and white nodes are cluster members. The source and destination nodes are within the same cluster and just need to construct an intra-cluster routing path. The source node first upstream forwards the packet to its clusterhead. Then, the clusterhead simply downstream broadcasts the packet to all cluster members, including the destination nodes. After checking the address on the packet, the destination nodes receive the packet, whereas other nodes discard it.

However, if the source and the destination nodes are in different clusters, it is necessary to determine the inter-cluster routing assignment in different clusters. In a small/medium dense ad hoc network, there may be many available inter-cluster wireless links, which allow the clusterhead be able to forward the packet to adjacent clusterhead directly. In contrast, in a wide area, emitting the transmission frequency directly to another faraway clusterhead would consume a huge mount of power and cause the node exhaust its battery capacity quickly. Hence, connecting to adjacent clusters indirectly by passing through intermediate relay nodes is more effective and reduces the power consumption.

In a cluster, the clusterhead collects the information about links and cluster members and determines how to process the intra-cluster and inter-cluster transmissions. For intra-cluster transmissions, the clusterhead just receives a packet from its cluster member source and simply downstream broadcasts it to its cluster members (see Figure 1-1). However, for inter-cluster transmissions, the clusterhead must determine the routing path to the cluster in which the destination node is located by passing through intermediate relay nodes if no direct links exist. Figure 1-2 shows an illustration of the inter-cluster routing assignment. The source node first forwards the packet to the clusterhead through the intra-cluster routing path. Then, the clusterhead that the source node belongs to selects some cluster members for forwarding the packet to the adjacent cluster. The transmission of the clusterhead and the selected cluster member node is also completed by using the corresponding intra-cluster routing path. This routing procedure continues in different clusters until reaching the cluster which the destination node is located in. Finally, the clusterhead of the destination node simply broadcasts the received packet to its cluster members. Again, by checking the address on the packet, the destination node receives the packet, whereas other nodes discard it. Note that the intermediate nodes on the routing path consume extra processing and communication power, which reduces their battery capacity.

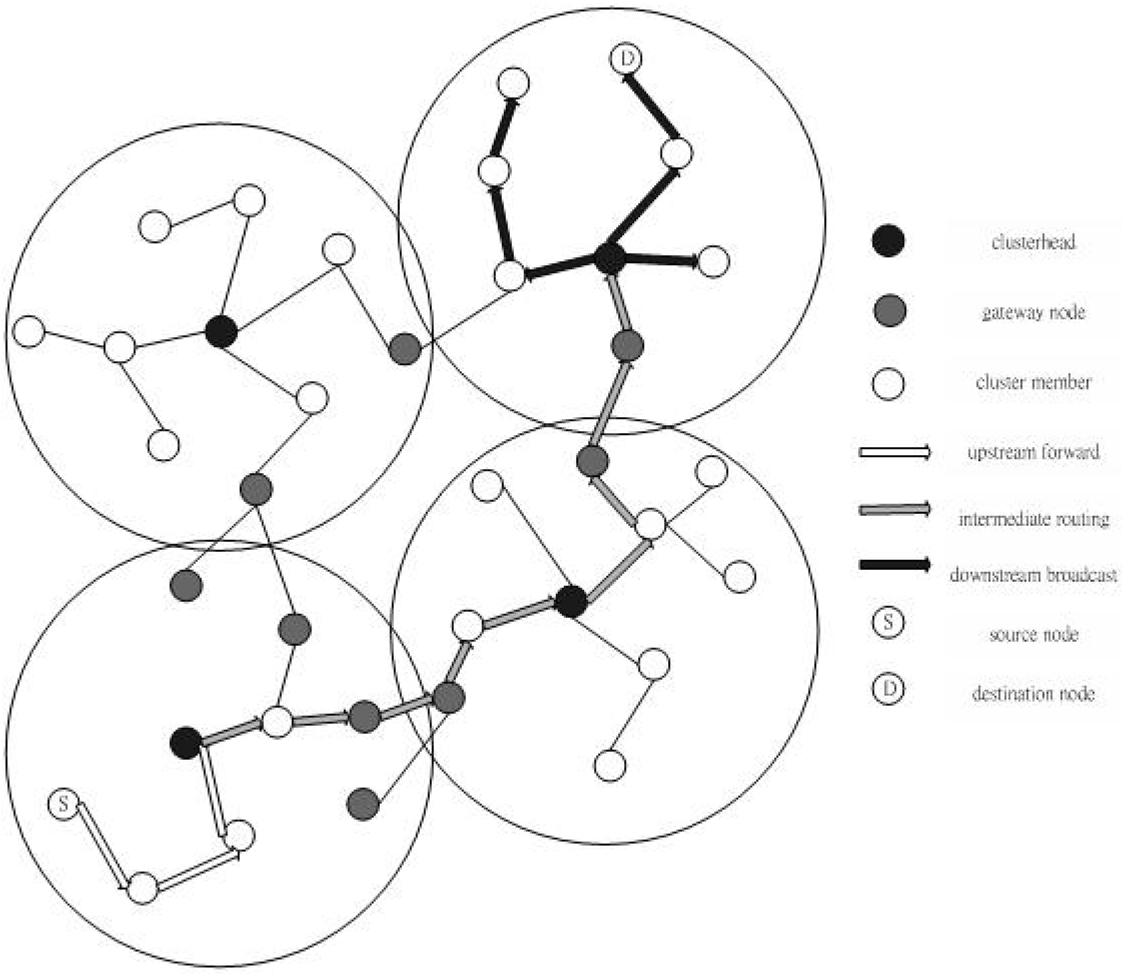

Figure 1-2Inter-Cluster Transmission

A node located on the fringe of the cluster and connected to another cluster's member node could be the gateway node and responsible for the inter-cluster transmissions. If a node pair of a link belongs to two different clusters, the link is a cross link and acts as a bridge. By selecting a gateway node and a cross link, clusterheads can form the inter-cluster routing assignment. Logically, the elected clusterheads and gateway nodes form a virtual mesh backbone and split the intra-cluster and inter-cluster routing into a multiple-layers hierarchical architecture. (See Figure 1-3)

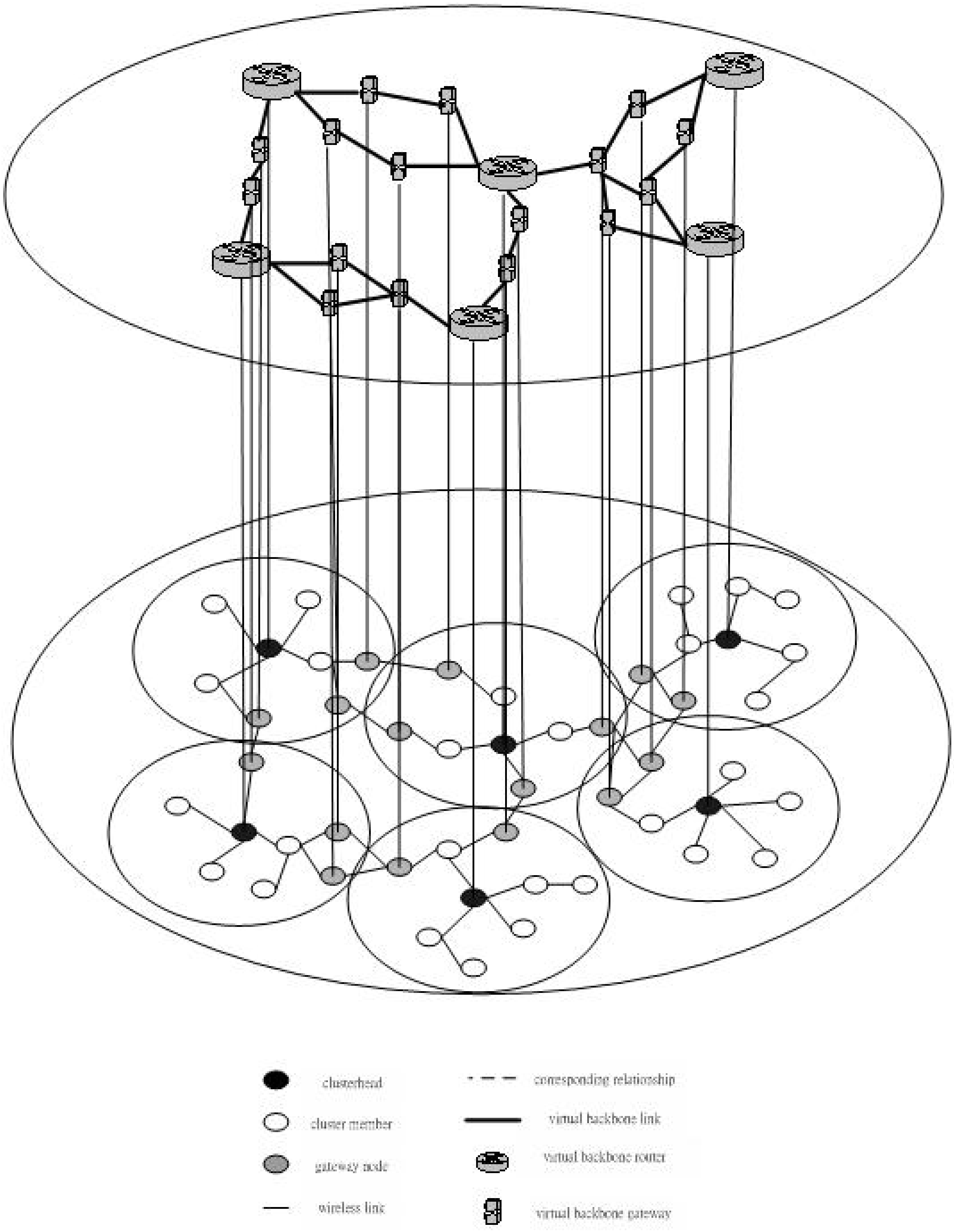

Figure 1-3 Virtual Mesh Backbone

In this architecture, the intra-cluster routing assignment can be served locally by the elected clusterhead and broadcast mechanism, whereas the inter-cluster routing must first be forwarded to the clusterhead that the source node belongs to. We can then determine the routing path between the current clusterhead and the clusterhead that the destination belongs to at the virtual backbone level by using shortest path algorithms, such as downstream flooding, Bellman Ford algorithm, or Dijkstra's algorithm. The architecture in Figure 1-3 is similar to that of real wired networks.

Based on the information collection and communication strategy, we can identify three types of network management architectures: centralized, distributed, and hierarchical architectures. However, because of the special requirements, such as the central infrastructure of a centralized architecture and the synchronization of a distributed architecture, we have excluded these two architectures. Considering network management message costs and node mobility, a three-level hierarchical architecture is proposed as a good tradeoff. Figure 1-4 illustrates this architecture.

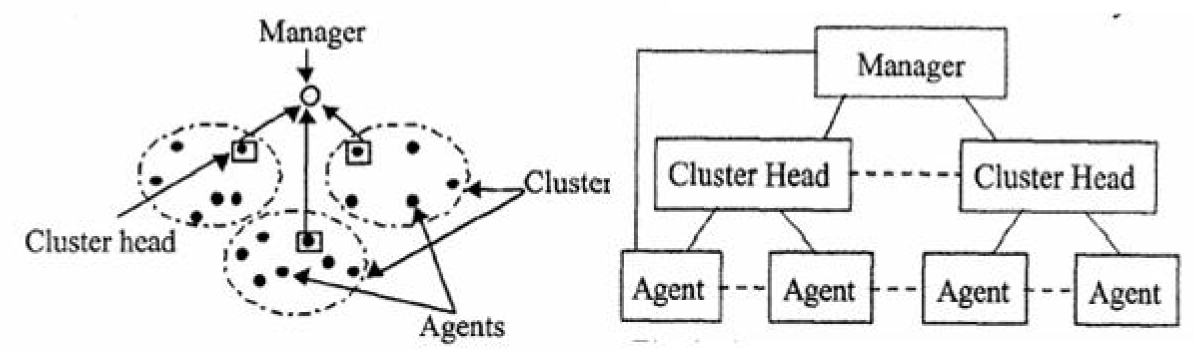

Figure 1-4 Three Level Network Management Framework

The lowest level of the architecture consists of individual managed nodes (cluster member nodes) called agents. Several agents are grouped into clusters and managed by the elected clusterhead. Each clusterhead is managed through the network manager directly. In our proposed mathematical model, we assume that there exists a geographical positioning system (GPS) that can communicate with each wireless device directly and collect the network information. Then, based on the collected information, we can manipulate our algorithm and broadcast the result by GPS to form the cluster topology and determine the routing assignment.

It is noteworthy that there is a little information exchanged between clusterheads and agents, such as the link state, network topology, and relative mobility. The clustering algorithm is more critical in considering the message cost, management information collection, and delay. Although the clusterheads are convenient for location-based management and the scalability of wireless networks, they still increase the number of routing hop counts and cause higher communication overheads and extra power consumption. These issues cause intermediate nodes to consume more battery capacity and become exhausted quickly. Hence, minimizing the power consumption of intermediate nodes without degrading the performance of the network is very important.

1.3.2Mobility Research

In wireless mobile ad hoc networks, links are fragile and change frequently due to the mobility of wireless nodes. The mobility of a node pair, which is described by its velocity and direction, may enlarge the distance and eventually cause the link to be disconnected. Many researchers have focused on the significant impact of different mobility models in the same protocol or scheme and find the performance results of an ad hoc network protocol drastically change as a result of changing the mobility model simulated **[6]**. It is important to select an appropriate mobility model. In **[6]**, the author summarized previous mobility models into two categories: entity mobility models and group mobility models.

In entity models, the mobility of each wireless device is independent of other devices. Each device determines its velocity and direction by its own probabilistic distribution. There are three widely-used entity models: the random walk model, the random waypoint model, and the Gauss-Markov mobility model. In the random walk model, each wireless node moves form its current location to its new location by randomly choosing a new speed and direction in which to travel. At every fixed interval, the node changes its mobility by selecting a new velocity and direction. In this model, the new selected speed and direction are uniformly distributed between [minimum-speed, maximum-speed] and [0, 2π] respectively. In contrast, the random walk waypoint model adds the pause-time mechanism at the interval of change in velocity and direction. In the random waypoint model, each node begins by staying in one location for a certain period of time and then chooses a new speed and direction to move to its destination. Figure 1-5 Random Walk Model Figure 1-6 Random Waypoint Model illustrate the traces of the random walk model and the random waypoint model respectively.

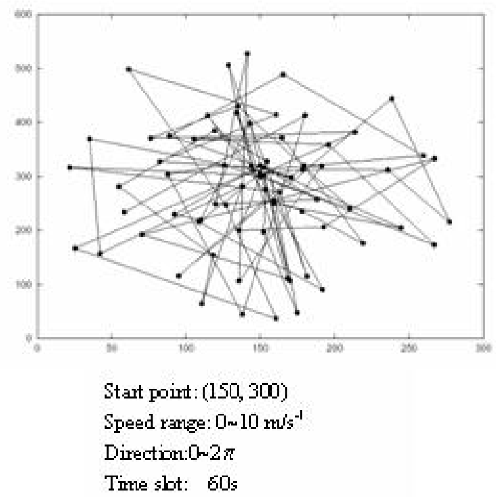

Figure 1-5 Random Walk Model

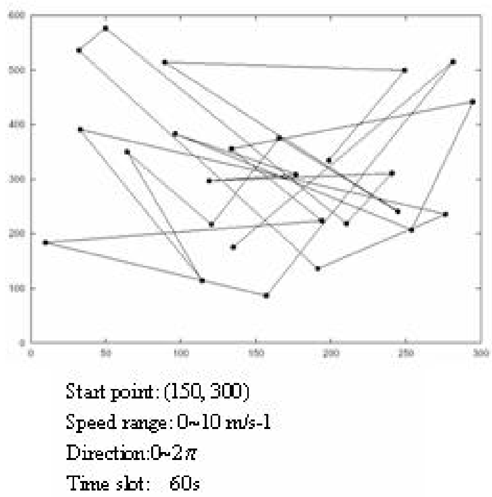

Figure 1-6 Random Waypoint Model

By observation, we see that the random walk model and the random waypoint model are memoryless patterns, because they retain no knowledge of their last location and speed. Although both models are simple to implement, they generate unrealistic movement patterns, such as sudden stops and sharp turns (see Figure 1-5 and Figure 1-6). They may be appropriate for describing the movement of particles or mechanical components, but not the mobile behavior of pedestrians or troops because the mobility pattern of animals or human beings seems smoother and depends on the speed, direction, and position at the last moment. Other models, such as the Gauss-Markov mobility model which we will describe later can resolve these issues. In our proposed mathematical formulation, we use the Gauss-Markov mobility model to depict the mobility of wireless devices, because wireless devices are usually controlled by human beings.

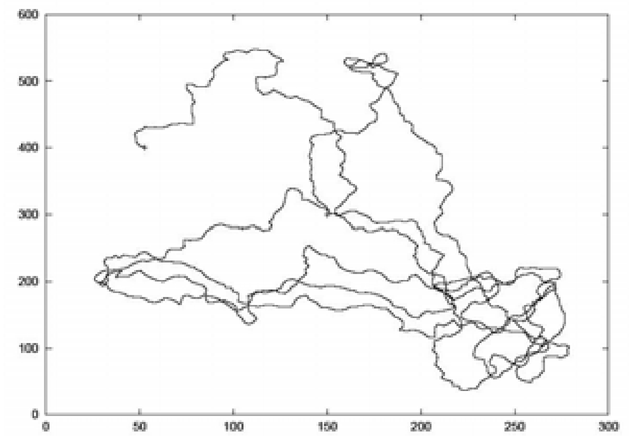

Figure 1-7 Trace of the Gauss-Markov Mobility Model

The Gauss-Markov mobility model was originally proposed for the simulation of personal computing systems. However, it has been used recently in the simulation of mobile ad hoc networks in which each mobile device is assigned an initial speed and direction and updates its mobility at fixed intervals. Specifically, the value of the speed and direction at the n_{th} instance is calculated based upon the value of speed and direction at the $n\text{-}1_{st}$ instance (see Figure 1-8), a new Gauss random variable, and the following equations:

$$s_n = \alpha s_{n-1} + (1-\alpha)\bar{s} + \sqrt{(1-\alpha)^2}\, s_{x_{n-1}} \qquad d_n = \alpha d_{n-1} + (1-\alpha)\bar{d} + \sqrt{(1-\alpha)^2}\, d_{x_{n-1}} \quad \text{(X-axis)}$$

$$s_n = \alpha s_{n-1} + (1-\alpha)\bar{s} + \sqrt{(1-\alpha)^2}\, s_{y_{n-1}} \qquad d_n = \alpha d_{n-1} + (1-\alpha)\bar{d} + \sqrt{(1-\alpha)^2}\, d_{y_{n-1}} \quad \text{(Y-axis)}$$

where s_n and d_n are, respectively, the new speed and direction of the wireless device at time interval n. α, where 0≦α≦1, is the tuning parameter. The Gauss-Markov Mobility Model was designed to adapt to different levels of randomness via this tuning parameter. Totally random values (or Brownian motion) are obtained by setting α=0 and linear motion is obtained by setting α=1. Intermediate levels of randomness are obtained by varying the value of α between 0 and 1. $\bar{s}$ and $\bar{d}$ are constants representing the mean value of speed and direction as $n \to \infty$; σ、ρ are variances, and s_{xn-1} d_{xn-1} are Gaussian random variables **[6]**. (Figure 1-8 illustrates the process of mobility prediction).

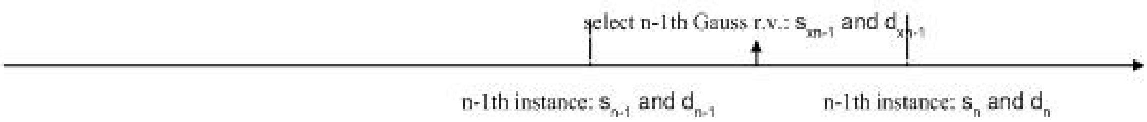

Figure 1-8 Mobility Prediction of the Gauss-Markov Mobility Model

We use the Gauss-Markov mobility model to predict the mobility of wireless nodes and the formula in **[7]** to evaluate the link duration of a pair of nodes. In **[7]**, let (x_i, y_i)、(x_j, y_j) be the coordinates, v_i, v_j be the speeds, and θ_i, θ_j be the directions of node i and node j respectively. Thus, $a = v_i \cos\theta_i - v_j \cos\theta_j$, $c = v_i \sin\theta_i - v_j \sin\theta_j$ are relative speeds and $b = x_i - x_j$, $d = y_i - y_j$ are the distances between node i and node j on the x-axis and y-axis respectively. Then, the amount of time they will stay connected, D_t, is predicted by $D_t = \dfrac{-(ab+cd)+\sqrt{(a^2+c^2)-(ad-bc)^2}}{a^2+c^2}$ from $(at+b)^2+(ct+d)^2=r^2$. at+b and ct+b respectively are the relative distances of node i and node j on the x-axis and y-axis after time interval t. This equation describes the critical point that the distance between node i and node j is just equal to the transmission radius of the node pair. Figure 1-9 gives an illustration.

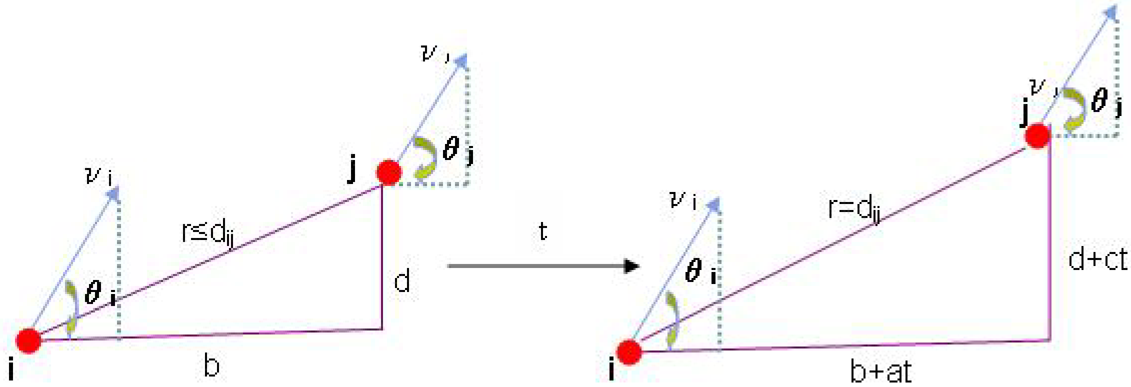

Figure 1-9 Computation of Link Duration

By computing the link duration heuristically at the decision instance, it is convenient for us to quantify the lifetime (stability) of the constructed topology and the routing assignment in order to evaluate the solution quality of the results. Thus, the decision process can be divided into many small cycles and the interval of the cycle can be determined by the minimum link prediction we compute at the decision instance. When the computed lifetime is about to end, a lead time is required to restart the decision procedure, collect the mobility information, compute the new duration of the links, and determine the new cluster topology and routing assignment. The lead time is the sum of the data collection time, t_1, decision processing time, t_2, and decision dissemination time, t_3 (see Figure 1-10). In our proposed model, we assume that there exists a geographical positioning system, such as the central coordinating system in a military network, which can retrieve the location and mobility information of mobile devices at the decision instance. Then, we use the obtained mobility information, the Gauss-Markov mobility model, and the formula in **[7]** to predict the duration of the links. In Figure 1-10, we predict the mobility information of $t_0+t_1+t_2+t_3$

based on the mobility information of the entire mobile ad hoc network at time t_0 and also compute the decision usage period, T, which is the minimum link duration of the constructed cluster topology and the routing assignment.

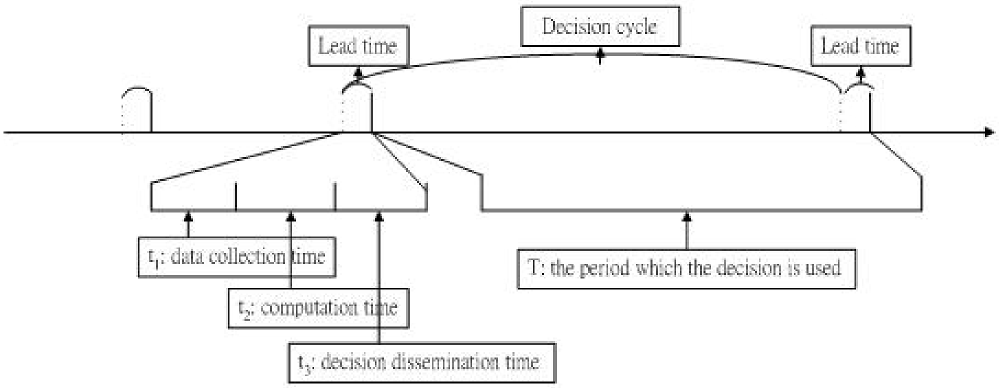

Figure 1-10 Illustration of Decision Cycle

In another kind of mobility model, group mobility models, wireless mobile nodes move together. Their mobility patterns are dependent on some special mobile nodes called reference points. In this kind of model, mobile nodes tune their mobility patterns within a specified range by referencing the speed and direction of the reference points i.e., the leader of a troop or the guide of a travel group. Hence, nodes with the same reference point would have similar mobility modes, move together, and be more likely to be grouped into the same cluster. The reference point is usually elected as the clusterhead for the coordination and network management of the cluster. Although we adopt the Gauss-Markov mobility model to describe the mobility of mobile devices, we believe that our proposed mathematical model could still make a significant contribution in the solution quality of cluster construction in group mobility models.

1.3.3 Integer Formulation for Clustering

The clustering problem has been widely discussed in routing and multicasting protocols in mobile ad hoc networks. Given a graph G (V, E) where V is the node set and E is the link set, we consider the clustering problem that involves in partitioning the graph G into several connected sub-graphs or groups by some specific objectives or criteria. For example, in **[11],** the author indicated that although the clustering approach is convenient for location–based management and scalability of the networks, there are still many defectives, such as that routing through clusterheads and intermediates nodes inevitably increases the hop count, causes extra processing and communication overheads to these nodes and exhausts their battery capacity quickly. Focusing on this point, the author proposed an Integer Linear Programming model, assumed that there are direct link between two adjacent clusters, and put minimizing the number of clusterheads as the objective function to reduce the overheads of intermediate relay nodes on the routing path.

However, the assumption of this model may be reasonable in a small/medium dense network but not in a large sparse network. As mentioned above, it causes huge power consumption for a node to emit the transmission radius directly to another faraway node. In addition, in a more general environment, we are not sure whether there exists a direct link or not. One clusterhead may connect to another clusterhead indirectly by using an inter-clustering routing path which is constructed by intermediate relay cluster members, gateway nodes and clusterheads. Hence, to adjust the assumption and be close to the real-world environment, we reference the concept of **[2]** and propose a mathematical formulation in which we denote decision variables and jointly determine clusterheads, cluster members, intra-cluster and inter-cluster routing paths.

In **[4]**, the clustering problem has been proved as NP-complete and can be reduced from another NP-hard problem, the clique problem. It is difficult to optimally solve the clustering problem in polynomial time and motivates many researchers to propose their heuristic approaches and protocols. To evaluate the solution quality of the experiment results, we should select a good benchmark. Some researchers compared the simulation results with previous works by some performance metric, such as packet delivery rate, packet loss rate or packet delay; and the other researchers solved a designed Integer Linear Programming model optimally to calculate the gap between the optimal solution of the model and the experiment results. The smaller gap indicates the better solution quality.

However, it is insufficient by using only some specific performance metric, and there are many controversial issues in the proposed Integer Linear Programming models, such as the generality of assumptions. Hence, it is necessary and significant for us to quantify the quality of the experiment result mathematically. In mobile ad hoc networks, we observe that due to the nodal mobility, the failed link of the routing path causes the packet loss and packet delay. The longer duration of the routing path produces the better QoS, such as the fewer packet loss or the lower packet delay. Hence, we put maximizing the minimum link duration of the constructed cluster topology and the routing assignment as our objective function to evaluate the solution quality.

By computing the duration of each link, we first determine some nodes as clusterheads and the other nodes as cluster members. Then, based on the constructed cluster topology, we determine the intra-cluster routing assignment. The link is called as a cross link if the two nodes of the link belong to different clusters. The node of a cross link could be the gateway node of and responsible for the inter-cluster communication.

In our proposed mathematical formulation, although we do not denote the gateway nodes as decision variables, they still can be determined in the program by using the information of the link state and the clusterhead/cluster member relationship. After determining clusterheads, cluster members, intra-cluster routing paths and gateway nodes, we execute the Bellman-Ford algorithm to determine the inter-cluster routing path within the hop-count constraint.

As mentioned above, for intra-cluster transmissions, the source node upstream forwards the packet to its clusterhead. Then, the clusterhead downstream broadcasts the packet to other cluster members. In contrast, for inter-cluster transmissions, the routing assignment must satisfy the hierarchical routing specification which defines that a node could be selected as the intermediate relay node only if the clusterhead of the node is also on the same routing path. In this scenario, the source node first upstream forwards the packet to its clusterhead. Then, the clusterhead becomes a new source node and continuing in selecting intermediate relay nodes to determine the inter-cluster routing path until reaching the cluster in which the destination is located. Finally the clusterhead of the destination downstream broadcasts the packet to all its cluster member nodes, including the destination node.

In addition to the hierarchical routing specification, the inter-cluster routing path is also restricted by the hop count and nodal capacity constraints. For the nodal capacity constraint, we aggregate the traffic working load of a node of by summing of the traffic load of different source/destination routing paths which use the node as their common relay node. Then, we ensure that the aggregated traffic working load of a node would not exceed its maximum available transmission rate.

To solve such a complicated problem, we adopt Lagrangean Relaxation approach and decompose the original problem into several subproblems with corresponding decision variables and constraints. By solving the subproblems optimally and adopting our getting primal heuristic procedure, we can improve the solution quality iteration by iteration (see section 3).

1.3.4 Energy Efficient-Multicasting and Wireless Advantage

Unlike the unicast transmission property in wired networks, in mobile ad hoc networks, a wireless device could transmit to multiple neighboring nodes with one transmission simultaneously by using its omni-directional or directional antenna. This phenomenon is called as “the wireless multicast advantage” and illustrated by Figure 1-11.

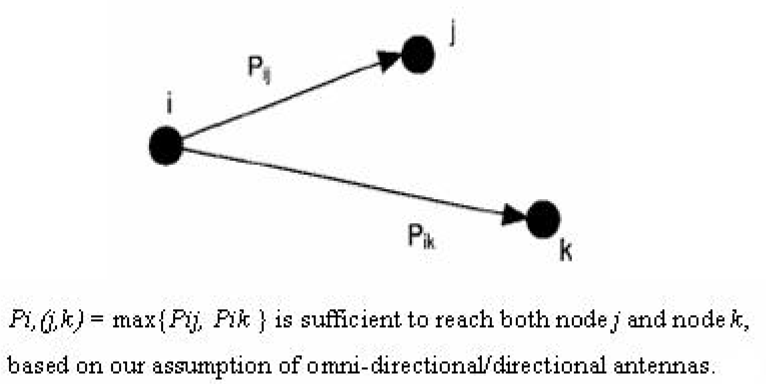

Figure 1-11 Wireless Multicast Advantage

In Figure 1-11, node i can reach node j and node k with one transmission simultaneously and the power consumption is determined by the distance between the farthest relay node k and the source node i. The extra transmission between node i and node j with power P_{ij} would waste the battery capacity of node i. Hence, for computing the transmission cost, we should adopt the node-based approaches instead of the traditional link-based approaches.

In wireless networks, the intermediate node of a routing path which is responsible for forwarding the packet causes the transmission cost, consumes its battery capacity and reduces its residual lifetime. Hence, the total transmission of a routing path is measured by summing the power consumption of all intermediate relay nodes on the path. Because of the limitation of the bandwidth nodal capacity, it is important to determine an energy efficient transmission strategy. This problem is called as the minimum-energy broadcasting/multicasting problem and becomes more complicated as the number of the source destination nodes increases, because there would be more transmission alternatives. The objective function of this kind of problem is to minimize the total energy consumption.

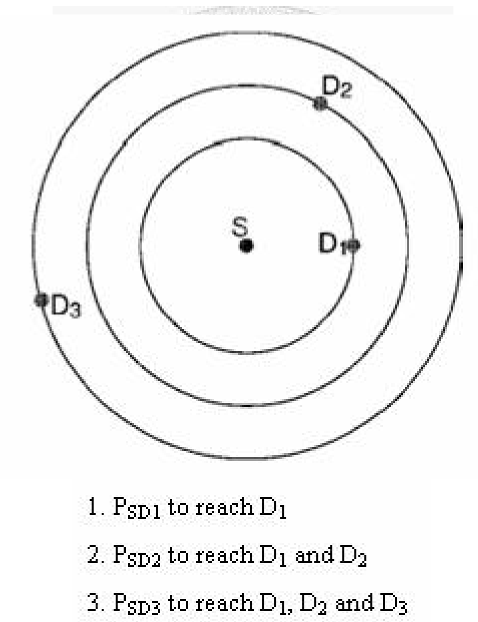

Figure 1-12 Different Transmission Scenarios of the Source Node

Take Figure 1-12 as an example, we enumerate all possible broadcasting alternatives exhaustively: (a) S broadcasts the packet to D_1 , D_2, and D_3 with power P_{SD3}; (b) S multicasts the packet to D_1 and D_2 with power P_{SD2}. Then, one of these two nodes must transmit the packet to D_3. The total power costs $P_{SD2} + P_{D1D3}$ or $P_{SD2} + P_{D2D3}$; and (c) S transmits the packet to D_1 with power P_{SD1}. Then, D_1 must construct a tree to forward the packet to D_2 and D_3. There are three alternatives: (c-1) D_1

transmits the packet to D_2 and D_3 simultaneously with total power P_{SD1} + max $\{P_{D1D2}, P_{D1D3}\}$, (c-2) D_1 transmits the packet to D_2 first and then D_2 forwards the packet to D_3. The total power is $P_{SD1} + P_{D1D2} + P_{D2D3}$, and (c-3) D_1 transmits the packet to D_3 first and then D_3 transmits the packet to D_2. The total power is $P_{SD1} + P_{D1D3} + P_{D3D2}$. Finally, we determine a broadcasting strategy with minimum total power consumption.

Based on the above discussion, the exhaustive search approach may be feasible in a small network with few alternatives, but not in a large and complex network. Many researchers have proposed many heuristic approaches to determine an energy efficient strategy. For example, in [13], the author designed a dominant pruning algorithm by utilizing the 2-hops neighboring information to reduce the redundant transmission when broadcasting in a wireless network. In each round of the transmission, a source node determines some neighboring nodes as its new intermediate relay nodes and prunes the nodes which have received packets. Then, the selected nodes become the new source nodes and are responsible for forwarding the packet until reaching all nodes on the entire wireless network (the termination of the broadcasting).

We adopt the same concept for the multicasting of the constructed cluster topology. By controlling and scheduling the routing assignment efficiently, the source node could forward a packet to many neighboring nodes simultaneously within one transmission and split the routing assignment into multiple individual routing paths with corresponding source node. We give an illustration of the intra-cluster routing assignment with “the wireless multicast advantage” in Figure 1-13.

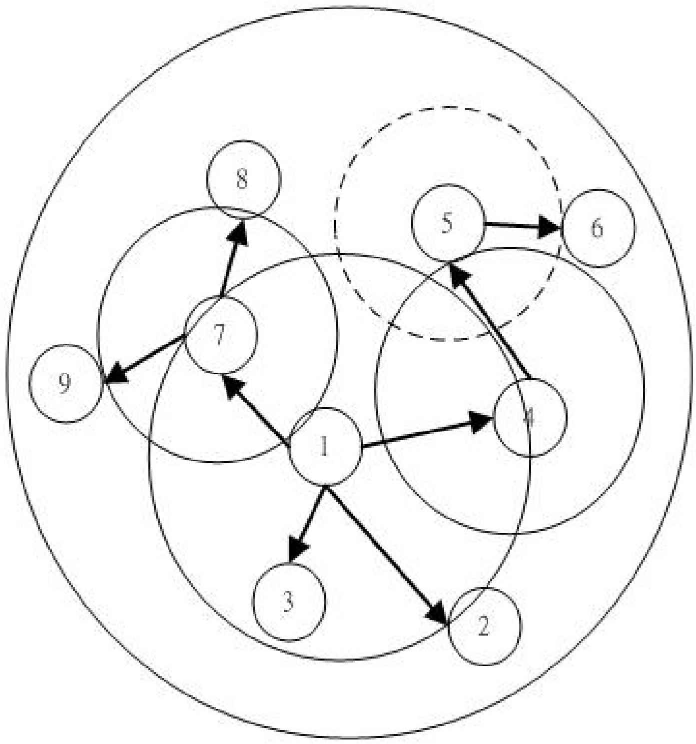

Figure 1-13 Wireless Multicast Advantage in the Intra-Cluster Transmission

In Figure 1-13, the clusterhead (node 1) initializes the intra-cluster broadcasting. Take the first transmission between node 1 and node 2 as an example, node 1 multicasts the packet to the nodes 2, 3, 4 with power P_{12} and splits the routing assignment into four individual paths. Second, the relay nodes 4 and 7 become new sources and broadcast the received packet to node 5, 8 and 9. Finally, the third relay node 5 transmits the packet to node 6 and terminates the broadcasting. By using wireless multicast advantage, it only costs P_{12}, P_{45}, P_{56} and P_{79} instead of the sum of P_{12}, P_{13}, P_{14}, P_{17}, P_{45}, P_{56}, P_{78}, and P_{79}, by using the single transmission iteratively. The former strategy improves the power consumption significantly.

For inter-cluster routing transmissions, to connect multiple clusters, we do not attempt to minimize the number of clusterheads and gateway nodes. Instead of only a gateway node in the fringe of the cluster, many gateway nodes provide more inter-cluster routing alternatives and make the routing assignment more reliable. In our proposed three-level architecture, gateway nodes and clusterheads form a virtual mesh backbone (see Figure 1-3). For inter-cluster transmissions, the source node first upstream forwards the packet to its clusterhead. Then, in the virtual mesh backbone layer, the clusterhead of the source node multicasts the packet to many gateway nodes and clusterheads simultaneously by using the “wireless multicasting advantage". Finally, the clusterhead of the destination node simply downstream broadcasts the received packet to the destination nodes. Figure 1-14 shows an illustration of the inter-cluster transmission with “the wireless multicasting advantage”.

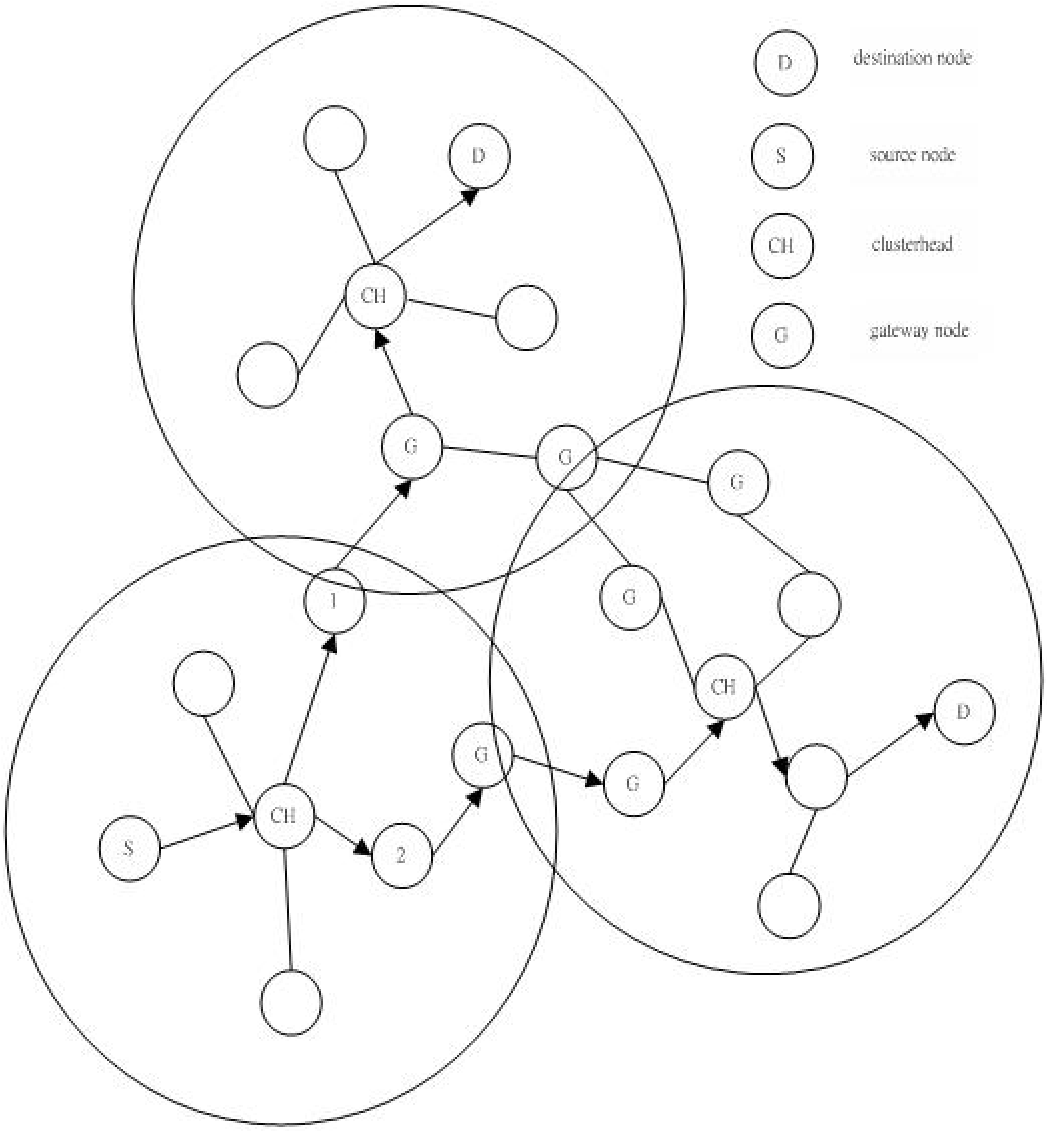

Figure 1-14 Wireless Multicast Advantage in the Inter-Cluster Transmission

In Figure 1-14, the clusterhead of the source node splits the inter-cluster transmission into two individual routing paths with two different cluster members, node 1 and node 2. Then, these two individual routing paths can progress simultaneously until reaching the destination nodes in other clusters. In this scenario, although the three-level hierarchical architecture increases the hop count and reduces the battery capacity of the intermediate nodes on the routing path, we could still schedule the power consumption efficiently by using “the wireless multicast advantage”.

Chapter 2 Problem Formulation

2.1 Problem Description

The network is modeled as a graph where each mobile device can be depicted as a node and each wireless connection as a link. In a mobile ad hoc network, the wireless link and routing path are unreliable and fragile because of the mobility of a device or the exhaustion of a device's battery capacity. This problem causes extra overheads for reconstructing the network topology and routing assignment. Hence, it is significant to improve the stability of a constructed topology and routing assignment.

By adopting the clustering approach, we consider the problem that involves in grouping mobile devices into different clusters. Given a wireless network topology, we jointly predict the mobility pattern of each node, compute the duration of each link, and determine the following five decision variables: (1) the clusterhead of each cluster, (2) the cluster members of each clusterhead, (3) the intra-cluster paths between each pair of the clusterhead and the cluster member, (4) the inter-cluster routing path of each O-D pair, and (5) the minimum link duration of the constructed cluster topology and routing assignment. Our objective function is to maximize (5).

For the mobility prediction, we assume that there is a central network management mechanism, such as a geographical positioning system in a military network, which can monitor the entire network and collect the location and mobility information of each wireless device at the decision instance. Then, we use the collected information and the Gauss-Markov mobility model to predict the mobility pattern of each device at the decision implementation instance. Figure 2-1 shows an illustration of the prediction process.

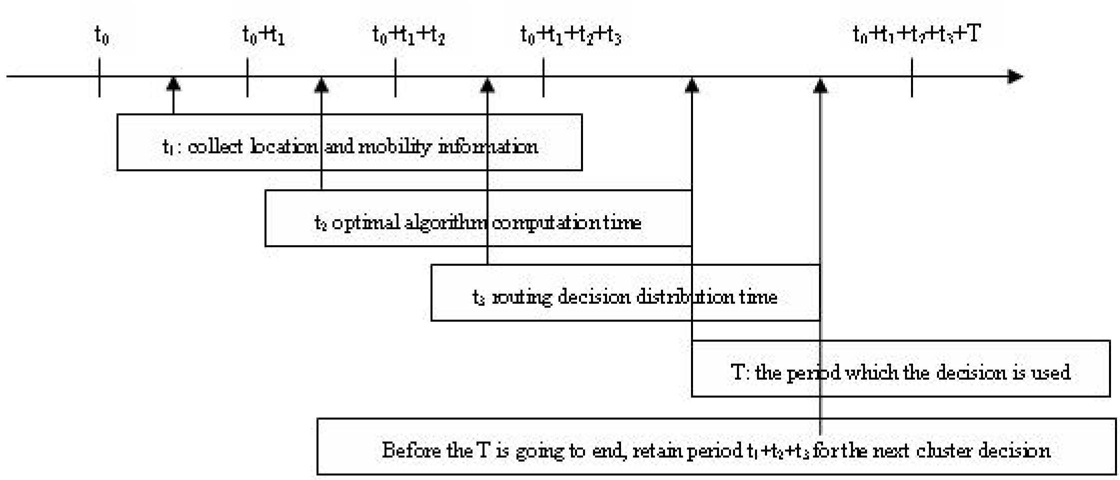

Figure 2-1 Illustration of Decision Cycle

To evaluate the stability of a constructed topology, we use the predicted mobility pattern and the formula of **[7]** to compute the link duration. In **[7]**, let (x_i, y_i)、(x_j, y_j) be the coordinates, v_i, v_j be the speeds, and θ_i, θ_j be the directions of node i and node j respectively. Thus, $a = v_i \cos\theta_i - v_j \cos\theta_j$, $c = v_i \sin\theta_i - v_j \sin\theta_j$ are relative speeds and $b = x_i - x_j$, $d = y_i - y_j$ are the distances between node i and node j on the x-axis and y-axis respectively. Then, the amount of time they will stay connected, D_t, is predicted by $D_t = \frac{-(ab+cd)+\sqrt{(a^2+c^2)-(ad-bc)^2}}{a^2+c^2}$ from $(at+b)^2+(ct+d)^2=r^2$.

Figure 2-2 shows an illustration of the cluster construction, where the black, gray and white nodes are clusterheads, gateway nodes and cluster member nodes respectively. The thin lines and bold lines represent the intra-cluster links and inter-cluster cross links respectively. For intra-cluster transmissions, the source node upstream forwards the packets to its clusterhead. Then, the clusterhead downstream broadcasts the received packet to all cluster members. For inter-cluster transmissions, the source node upstream forwards the packet to its clusterhead. Then, the selected clusterhead becomes the new source node and selects intermediate clusterheads, cluster members and gateway nodes to construct the inter-cluster routing path until reaching the cluster in which the destination node is located. Finally, the clusterhead of the destination node downstream broadcasts the packet to the destination nodes. (See Figure 1-1 and Figure 1-2).

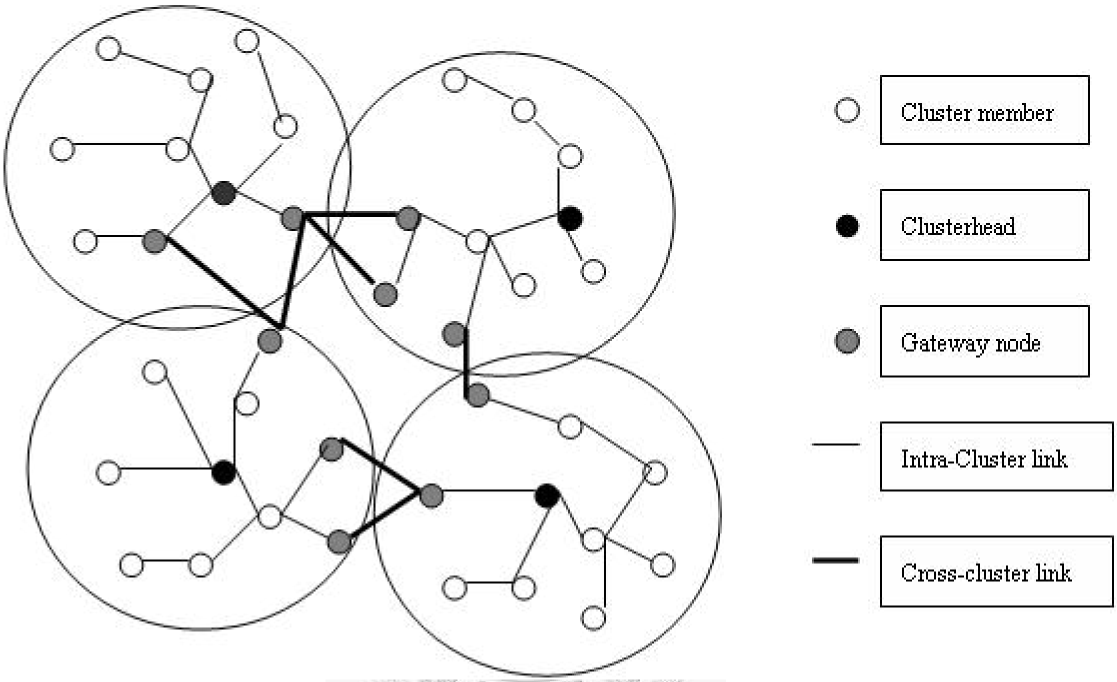

Figure 2-2 Cluster Formation

Note that the gateway nodes are not decision variables in our mathematical model, but can still be determined in the program. For a link (i, j), if node i and node j belong to different clusterheads a and b, the link is a cross link and the nodes of the cross link can be the gateway nodes of these two adjacent cluster a and b respectively.

Table 2-1 Problem Description

Problem assumption:

1. Homogeneous network, fixed transmission range and bidirectional links
2. Error-free transmission within the transmission radius
3. Three-level hierarchical architecture
4. Available geographical positioning system
5. Prediction of the link duration
6. Hierarchical cluster routing specification: A node could be the intermediate node on a routing path, only if the clusterhead of the node is also on the path. Hence, for each node on the routing path, we must pass the packet through its clusterhead.

Given:

1. The network topology includes the node set and the link set
2. The maximum hop count for cluster construction
3. The mobility information of each node
4. The predicted duration of each link
5. Capacity for each node evaluated by maximum available transmission rate (bits/sec)
6. The source node, the destination node and traffic demand for each O-D pair

Objective:

To maximize the minimum link duration of the constructed cluster topology and routing assignment

Subject to:

1. Clusterhead and cluster member relationship constraint: Each node must be belong to exactly one clusterhead and maintain an intra-cluster routing path between itself and its clusterhead. A determined clusterhead should not join other clusters
2. Intra-cluster routing assignment with the d-hop constraint
3. Inter-cluster assignment with the h-hop and hierarchical cluster routing constraints
4. End-to-end QoS requirement defined by the maximum hop count of each O-D pair
5. Nodal capacity constraint
6. Lead time limitation

To determine:

1. Clusterheads of different clusters
2. Cluster members of a cluster
3. Intra-cluster routing paths
4. Inter-cluster routing path of each O-D pair
5. The minimum link duration of the constructed topology and routing assignment

2.2 Problem notations

Table 2-2 Notations of Given Parameters

Notation	Definition
V	The set of nodes which is also the set of candidate clusterheads
L	The set of links
d	The max hop count for constructing a cluster which is also the longest distance between a cluster member and its clusterhead
Q_{uv}	The set of candidate paths between the node u and node v
r	The transmission radius of each node
W	The set of all O-D pairs
P_w	The set of candidate paths of the O-D pair w, which will be included in Q_{uv}, $P_w \in Q_{uv}$
a_w	Traffic demand of the O-D pair w, which is evaluated by traffic data rate per unit time (unit: bits/sec)
C_n	Capacity of the node n, which is evaluated by maximum transmission rate of the node n (unit: bits/sec)
H_w	Maximum hop count of each O-D pair
$\sigma_{p,(n,m)}$	1 if the link (n,m) is on the path p, and 0 otherwise. (n,m) defines that the node n and m are the outgoing node and incident node of the link respectively.
$\delta_{p\ell}$	1 if the link ℓ is on the path p, and 0 otherwise.
$x_n(t)$	The x-axis coordinate of the node n at time t
$y_n(t)$	The y-axis coordinate of the node n at time t
$v_x(t)$	The x-axis velocity of the node n at time t

$v_y(t)$	The y-axis velocity of the node n at time t
t_{ij}	The link duration between the node i and node j
t_1	Data collection time
t_2	Computation time
t_3	Decision dissemination time
M_1	The big number used in the constraint (IP 1.11). The value is set as 2.
M_2	The big number used in the constraints (IP 1.15) and (IP 1.16). The value is set as the maximum link duration of the network at the decision instance.

Table 2-3 Notations of Decision Variables

Decision variables	
Notation	**Definition**
h_g	1 if the node g is elected as a clusterhead and 0 otherwise.
b_{vg}	1 if the node v belongs to the clusterhead g and 0 otherwise
z_{pvg}	I if the node v choices the path p as its intra-cluster routing path and 0 otherwise
x_p	1 if the path $p \in P_w$ is used for the O-D pair w and 0 otherwise
T	The minimum link duration of the constructed cluster topology and routing assignment

2.3 Problem Formulation

Optimization problem:

Objective function:

$$Z_{IP1} = \max T \qquad \text{(IP1)}$$

Subject to:

Part I: Cluster Construction Constraints

$$\sum_{g \in V} b_{vg} = 1 \qquad \forall v \in V \qquad \text{(IP 1.1)}$$

$$b_{gg} = h_g \qquad \forall g \in V \qquad \text{(IP 1.2)}$$

$$b_{vg} \le h_g \qquad \forall v, g \in V \qquad \text{(IP 1.3)}$$

$$b_{vg} \le \sum_{p \in P_{vg}} z_{pvg} \qquad \forall v, g \in V \qquad \text{(IP 1.4)}$$

$$h_g = 0 \; or \; 1 \qquad \forall g \in V \qquad \text{(IP 1.5)}$$

$$b_{vg} = 0 \; or \; 1 \qquad \forall v, g \in V \qquad \text{(IP 1.6)}$$

$$\sum_{\ell \in L} \sum_{p \in P_{vg}} z_{pvg} \cdot \delta_{p\ell} \le d \qquad \forall v, g \in V \qquad \text{(IP 1.7)}$$

$$z_{pvg} = 0 \; or \; 1 \qquad \forall p \in P_{uv} \;\; u, v \in V \qquad \text{(IP 1.8)}$$

Part II: Inter-cluster Routing Constraints

$$\sum_{p \in P_w} x_p \le 1 \qquad \forall w \in W \qquad \text{(IP 1.9)}$$

$$\sum_{\ell \in L} \sum_{p \in P_w} x_p \cdot \delta_{p\ell} \le H_w \qquad \forall w \in W \qquad \text{(IP 1.10)}$$

$$\sum_{p \in P_w} x_p \cdot \delta_{pv} \le \frac{\left\{ \sum_{p \in P_w} x_p \cdot \delta_{pg} \right\} + b_{vg} + M_1 (1 - b_{vg})}{2} \qquad \forall w \in W \;\; v, g \in V \qquad \text{(IP 1.11)}$$

$$x_p = 0 \; or \; 1 \qquad \forall w \in W \;\; p \in P_w \qquad \text{(IP 1.12)}$$

$$\sum_{(i,j)\in L, i=n} \sum_{w\in W} \sum_{p\in P_w} a_w \cdot x_p \cdot \sigma_{p,(i,j)} \le C_n \qquad \forall n \in V \qquad \text{(IP 1.13)}$$

$$t_{ij} = \frac{-(ab+cd)+\sqrt{(a^2+c^2)r^2-(ad-bc)^2}}{a^2+c^2} \quad \forall (i,j)\in L \qquad \text{(IP1.14)}$$

$$a = \upsilon_{ix}(t)-\upsilon_{jx}(t)\ ,\ c = \upsilon_{iy}(t)-\upsilon_{jy}(t) \qquad b = x_i(t)-x_j(t)\ ,\ \mathrm{d} = y_i(t)-y_j(t)$$

Part III: Minimum Link Duration Constraints

$$T \le \Big(\sum_{p\in P_{vg}} z_{pvg}\cdot\delta_{p\ell}\Big)\cdot t_\ell + M_2\Big(1-\sum_{p\in P_{vg}} z_{pvg}\cdot\delta_{p\ell}\Big) \ \forall v,g\in V \ \ell\in L \qquad \text{(IP 1.15)}$$

$$T \le \Big(\sum_{p\in P_w} x_p\cdot\delta_{p\ell}\Big)\cdot t_\ell + M_2\Big(1-\sum_{p\in P_w} x_p\cdot\delta_{p\ell}\Big) \qquad \forall w\in W \ \ell\in L \qquad \text{(IP 1.16)}$$

$$M_2 \ge T \ge t_1+t_2+t_3\,. \qquad \text{(IP 1.17)}$$

Explanation of the objective function:

The objective function (IP 1) is to **maximize the minimum link duration of the constructed cluster topology and routing assignment.** In our mathematical formulation, we evaluate the minimum link duration in the constraints (IP 1.15) and (IP 1.16). Thus, our problem is a Max-Min optimization formation.

Explanation of constraints:

The clusterhead/cluster member relationship constraints:

Constraint (IP 1.1): For each node v, it must exactly belong to some node g.

Constraint (IP 1.2): (IP 1.2) confines that if the node g is elected as a clusterhead, we enforce that the node g must belong to itself by setting the decision variable b_{gg} as 1.

Constraint (IP 1.3): If some node selects the node g as its clusterhead, the node g must be determined as a clusterhead by setting the decision variable h_g as 1.

The intra-cluster routing constraints:

Constraint (IP 1.4): If the node v belongs to the clusterhead g, we must determine the intra-cluster routing path between the cluster member node v and its clusterhead g.

Constraint (IP 1.7): For the *d*-hop cluster construction, the hop count of the intra-cluster routing path must be less than *d*.

The inter-cluster routing constraints:

Constraint (IP 1.9): (IP 1.9) defines that for each O-D pair, we determine a inter-cluster routing path at most.

Constraint (IP 1.10): For the end-to-end delay QoS requirement, (IP 1.10) confines the hop count of the inter-cluster routing path of each O-D pair must be less than a predefined value H_w.

Constraint (IP 1.11): (IP 1.11) describes the hierarchical routing constraint. For each node of the O-D pair routing path, it confines that the node could be selected as a intermediate node on a inter-cluster routing path only if its clusterhead is also on the path. For this complex constraint, Table 2-4 shows an illustration of the relationship between $\sum_{p\in P_w} x_p \cdot \delta_{pv}$、$\sum_{p\in P_w} x_p \cdot \delta_{pg}$ and b_{vg}.

Table 2-4 Examination of the value table

	$\sum_{p\in P_w} x_p \cdot \delta_{pv}$	$\leq$	$\sum_{p\in P_w} x_p \cdot \delta_{pg}$	b_{vg}	$M_1(1-b_{vg})$
1. *given v g and* $b_{vg}=0$	0	$\leq$	0	0	M
2. *given v g and* $b_{vg}=0$	0	$\leq$	1	0	M
3. *given v g and* $b_{vg}=0$	1	$\leq$	0	0	M
4. *given v g and* $b_{vg}=0$	1	$\leq$	1	0	M

5. *given v g and* $b_{vg}=1$	0	$\le$	0	1	0
	$\sum_{p\in P_w} x_p \cdot \delta_{pv}$	$\le$	$\sum_{p\in P_w} x_p \cdot \delta_{pg}$	b_{vg}	$M_1(1-b_{vg})$
6. *given v g and* $b_{vg}=1$	0	$\le$	1	1	0
7. *given v g and* $b_{vg}=1$	1	$\le$	0	1	0
8. *given v g and* $b_{vg}=1$	1	$\le$	1	1	0

By enumeration, the first four rows indicate that when the decision variable b_{vg} is set as 0, there is no relationship between the node *v* and *g*. Hence we add the term $M_1(1-b_{vg})$ to eliminate the restriction of this constraint and give the freedom for choosing the node *v* or *g*. But if b_{vg} is set as 1, it requires that only when the clusterhead *g* is on the path, the node *v* could be the intermediate relay node on the path. In the sixth and eighth rows, the variable b_{vg} is set as 1 and the clusterhead *g* of the node *v* is on the selected path, we can select the node *v* as the intermediate relay node or not. In the fifth row, the clusterhead *g* and the cluster member *v* are both not on the path, so it does not violate the constraint. Finally, in the seventh row, the cluster member *v* is on the path but the clusterhead *g* not, it violates the constraint.

Note that if a destination node is also the gateway node of its cluster and receives a packet from other adjacent gateway node, this constraint defines the destination node must still forward the packet to its clusterhead. Then, the clusterhead transmits the packet to the destination node again. This scenario may be reasonable and necessary in some specific applications, such as the encoding and decoding processes on clusterheads in a military network, but wastes extra power consumption in others. To eliminate this kind of transmission, we can add a new constraint and define that there is no outgoing link on the destination node. The constraint is represented as follows:

$$\sum_{(i,j)\in L, i=n} \sum_{p\in P_w} x_p \cdot \sigma_{p,(i,j)} = 0 \qquad \forall n \in \textit{the destination node of } w,\ w \in W \qquad \text{(IP 1.18)}$$

For implementation, we set the big number M_1 as 2. The reason is that the maximum of the left-side value is at most 1; and it is enough to eliminate the restriction of the right-side when the decision variable b_{vg} is set as 0.

Constraint (IP 1.13): In the constraint (IP 1.13), for each node, we aggregate the traffic working load of different routing paths which uses the node *n* as their intermediate relay node; and ensure that the total traffic load does not exceed its maximum available transmission rate.

The minimum link and node duration constraints:

Constraint (IP 1.14): In the constraint (IP 1.14), we compute the duration of each link by adopting the predicted mobility information of each node and the formula of **[7]**.

Constraint (IP 1.15) and (IP 1.16): In the constraints (IP 1.15) and (IP 1.16), we calculate the minimum link duration of the intra-cluster and inter-cluster routing paths respectively.

Constraint (IP 1.17): (IP 1.17) defines the low bound and the upper bound of the decision variable T. The lower bound is the lead time, which is the sum of data collection time, processing time, and decision dissemination time. The upper bound is the maximum link duration of the entire network at the decision instance.

The integer constraints:

Constraint (IP 1.5)、(IP 1.6)、(IP 1.8) and (IP 1.12) are the integer constraints of decision variables.

2.4 Extension of the Objective Function

In some wireless applications, the transmission holding time t_w of each O-D pair w may be different. Considering with maximizing the total revenue of the served O-D pairs, we extend the objective function of (IP 1) as follows:

$$Z_{IP2} = \max \Psi(\sum_{w \in W} \sum_{p \in P_w} a_w \cdot x_p \cdot Min\{t_w, T\}) \qquad \text{(IP2)}$$

where Ψ is the reward function, and a_w is the transmission rate (bits/sec) of the O-D pair w. For each O-D pair w, because we are not sure the minimum duration T is longer than the transmission holding time t_w or not, we calculate the minimum of these two values.

Chapter 3 Solution Approach

3.1 Introduction to Lagrangean Relaxation Method

Lagrangean relaxation method was widely used for scheduling and solving integer programming problems in the 1970s **[20]**, because it is flexible and provides good solutions for these problems. Recently, it has become one of the best tools for solving optimization problems, such as integer programming, linear programming with a combinatorial objective function, and non-linear programming.

By adopting Lagrangean relaxation method, there are several advantages. For example, we could relax the complicated constraints of the primal mathematical formulation and design a new Lagrangean relaxation problem in many different ways. By relaxing the complicated constraints and putting them in the objective function with the corresponding Lagrangean multipliers, we can divide the original problem into several easily-solved and independent subproblems. Then, for each subproblem, we explore the underlying structure and property and solve it optimally in some well-known algorithms **[16]**.

In addition, we can get a reasonable boundary to the objective function of the original formulation. The result of the Lagrangean relaxation problem is always a lower bound to the original minimization problem (or an upper bound to a maximization problem). Then, we use the boundary to design a heuristic approach to get a primal feasible solution. To solve the original problem optimally and reduce the gap between the primal problem and the Lagrangean relaxation problem, we improve the lower bound by solving the decomposed subproblem optimally and using the subgradient method to adjust the multipliers at each iteration **[16].** In Figure 3.1, we explain Lagrangean relaxation in a straightforward way. Figure 3.2 shows the detailed procedure for Lagrangean relaxation.

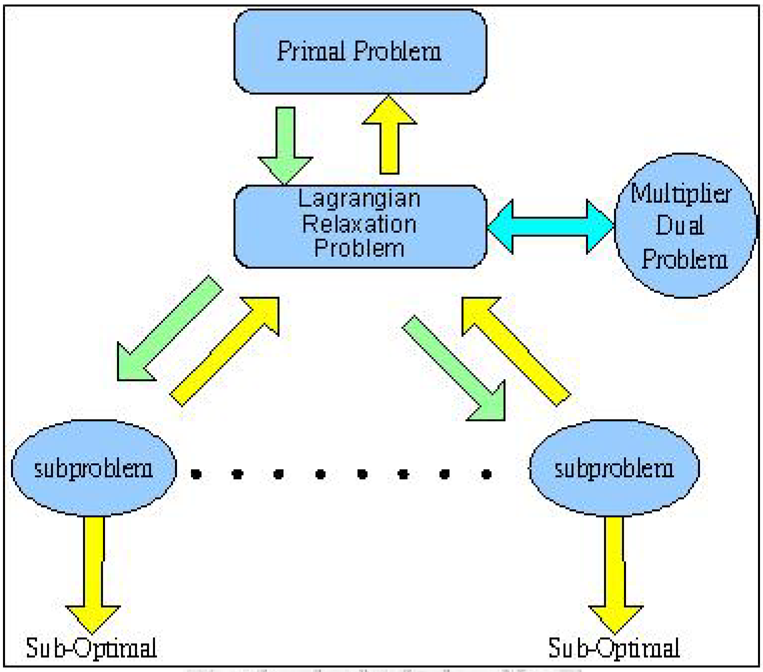

Figure 3-1 Illustration of Lagrangean Relaxation Method

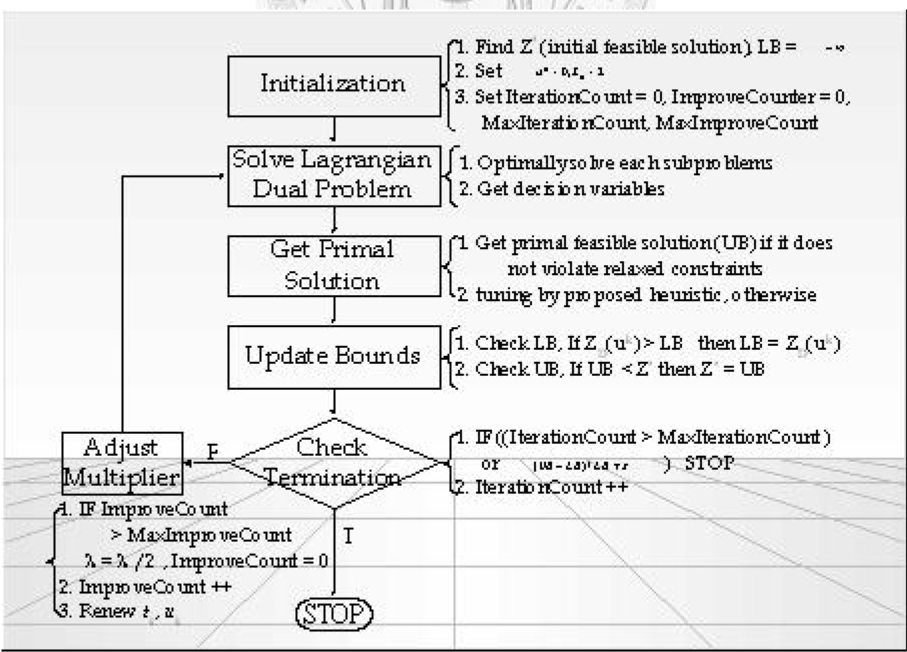

Figure 3-2 Procedures of Lagrangean Relaxation Method

3.2 Lagrangean Relaxation

By using Lagrangean relaxation method, we can transform the primal problem into the following Lagrangean relaxation problem (LR) where constraints (IP 1.2), (IP 1.3), (IP 1.11), (IP 1.13), (IP 1.15), and (IP 1.16) are relaxed. As a convention, we first multiple the objective function of the primal problem with minus one and transform it into a minimization problem. For a vector of non-negative multipliers, we represent the Lagrangean relaxation problem as follows:

$$\begin{aligned}
Z_d(\mu_1,\mu_2,\mu_3,\mu_4,\mu_5,\mu_6) = Min\text{-}T &+ \sum_{g\in V}\mu_{1g}(b_{gg}-h_g) + \sum_{v\in V}\sum_{g\in V}\mu_{2vg}(b_{vg}-h_g) \\
&+ \sum_{v\in V}\sum_{g\in V}\sum_{w\in W}\mu_{3vgw}\Big(\sum_{p\in P_w} x_p\cdot\delta_{pv} - \frac{\sum_{p\in P_w} x_p\cdot\delta_{pg}}{2} - \frac{b_{vg}}{2} - \frac{M_1(1-b_{vg})}{2}\Big) \\
&+ \sum_{n\in V}\mu_{4n}\Big(\sum_{(i,j)\in L, i=n}\sum_{w\in W}\sum_{p\in P_w} a_w\cdot x_p\cdot\sigma_{p(i,j)} - C_n\Big) \\
&+ \sum_{v\in V}\sum_{g\in V}\sum_{\ell\in L}\mu_{5vg\ell}\Big(T - \big(\sum_{p\in Q_{vg}} z_{pvg}\cdot\delta_{p\ell}\big)\cdot t_\ell - M_2\big(1-\sum_{p\in Q_{vg}} z_{pvg}\cdot\delta_{p\ell}\big)\Big) \\
&+ \sum_{w\in W}\sum_{\ell\in L}\mu_{6w\ell}\Big(T - \big(\sum_{p\in P_w} x_p\cdot\delta_{p\ell}\big)\cdot t_\ell - M_2\big(1-\sum_{p\in P_w} x_p\cdot\delta_{p\ell}\big)\Big)
\end{aligned}$$

(LR)

Subject to

$\sum_{g\in V} b_{vg} = 1$	$\forall v\in V$	(LR 1.1)
$b_{vg} \le \sum_{p\in Q_{vg}} z_{pvg}$	$\forall v,g\in V$	(LR 1.2)
$h_g = 0\ or\ 1$	$\forall g\in V$	(LR 1.3)
$b_{vg} = 0\ or\ 1$	$\forall v,g\in V$	(LR 1.4)
$\sum_{\ell\in L}\sum_{p\in Q_{vg}} z_{pvg}\cdot\delta_{p\ell} \le d$	$\forall v,g\in V$	(LR 1.5)
$z_{pvg} = 0\ or\ 1$	$\forall p\in Q_{uv}\ \ u,v\in V$	(LR 1.6)
$\sum_{p\in P_w} x_p \le 1$	$\forall w\in W$	(LR 1.7)

$$\sum_{\ell\in L}\sum_{p\in P_w} x_p \cdot \delta_{p\ell} \le H_w \qquad \forall W \in W \qquad \text{(LR 1.8)}$$

$$x_p = 0 \ or\ 1 \qquad \forall w \in W\ p \in P_w \qquad \text{(LR 1.9)}$$

$$t_{ij} = \sqrt{\frac{-(ab+cd)+(a^2+c^2)r^2-(ad-bc)^2}{a^2+c^2}} \qquad \forall (i,j) \in L \qquad \text{(LR 1.10)}$$

$$a = \upsilon_{ix}(t) - \upsilon_{jx}(t)\ ,\ c = \upsilon_{iy}(t) - \upsilon_{jy}(t) \qquad b = x_i(t) - x_j(t)\ ,\ \mathrm{d} = y_i(t) - y_j(t)$$

$$M_2 \ge T \ge t_1 + t_2 + t_3\,. \qquad \text{(LR 1.11)}$$

where (μ_1, μ_2, μ_3, μ_4, μ_5, μ_6) is the non-negative vector of the Lagrangean multipliers $\{\mu_{1g}\}$, $\{\mu_{2vg}\}$, $\{\mu_{3vgw}\}$, $\{\mu_{4n}\}$, $\{\mu_{5vg\ell}\}$, and $\{\mu_{6w\ell}\}$. To solve the Lagrangean relaxation problem, we decompose the problem into the following four independent and easily solved optimization subproblems with different decision variables.

3.2.1 Subproblem 1 (related to decision variable h_g)

$$Z_{LR-sub1} = \min \sum_{g\in V}(-\mu_{1g} - \sum_{v\in V}\mu_{2vg}) \cdot h_g \qquad \text{(sup1)}$$

Subject to:

$$h_g = 0 \ or\ 1\,. \qquad \forall g \in V \qquad \text{(sup1 1.1)}$$

This problem can be decomposed into $|V|$ independent subproblems and solved optimally by a simple algorithm. For each node v, we first compute the coefficient $-\mu_{1g} - \sum_{v\in V}\mu_{2vg}$. Then, for each node g, if the coefficient is less than zero, we set h_g as 1; otherwise we set h_g as 0. Because for each node v, it takes $|V|$ iterations to compute its coefficient, the complexity of Subproblem 1 is $|V|^2$, where $|V|$ is the number of nodes.

3.2.2 Subproblem 2 (related to decision variable b_{vg} and z_{pvg})

$$Z_{LR-sub2} = \min \sum_{g\in V} \mu_{1g} \cdot b_{gg} + \sum_{v\in V}\sum_{g\in V} \mu_{2vg} \cdot b_{vg} - \sum_{v\in V}\sum_{g\in V}\sum_{w\in W} \mu_{3vgw} \left(\frac{b_{vg} + M_1(1-b_{vg})}{2}\right)$$

$$- \sum_{v\in V}\sum_{g\in V}\sum_{\ell\in L} \mu_{5vg\ell} \left\{ \left(\sum_{p\in Q_{vg}} z_{pvg} \cdot \delta_{p\ell} \right) \cdot t_\ell + M_2 \left(1 - \sum_{p\in Q_{vg}} z_{pvg} \cdot \delta_{p\ell}\right) \right\}$$

$$= \min \sum_{g\in V} \left(\mu_{1g} + \mu_{2gg} + \frac{(M_1-1)}{2} \sum_{w\in W} \mu_{3vgw}\right) b_{gg} + \sum_{v\in V} \sum_{g\in V, v\neq g} \left(\mu_{2vg} + \frac{(M_1-1)}{2} \sum_{w\in W} \mu_{3vgw}\right) b_{vg}$$

$$+ \sum_{v\in V}\sum_{g\in V}\sum_{p\in P_{vg}} \left(\sum_{\ell\in L} \mu_{5vg\ell} \cdot (M_2 - t_\ell) \right) \cdot z_{pvg} - \sum_{v\in V}\sum_{g\in V}\sum_{w\in W} \frac{\mu_{3vgw} \cdot M_1}{2} - \sum_{v\in V}\sum_{g\in V}\sum_{\ell\in L} \mu_{5vg\ell} \cdot M_2$$

(sup2)

Subject to:

$$\sum_{g\in V} b_{vg} = 1 \qquad \forall v \in V \qquad \text{(sup2 2.1)}$$

$$b_{vg} \le \sum_{p\in Q_{vg}} z_{pvg} \qquad \forall v, g \in V \qquad \text{(sup2 2.2)}$$

$$b_{vg} = 0 \text{ or } 1 \qquad \forall v, g \in V \qquad \text{(sup2 2.3)}$$

$$\sum_{\ell\in L} \sum_{p\in Q_{vg}} z_{pvg} \cdot \delta_{p\ell} \le d \qquad \forall v, g \in V \qquad \text{(sup2 2.4)}$$

$$z_{pvg} = 0 \text{ or } 1 \qquad \forall p \in P_{uv}\ \ u, v \in V \qquad \text{(sup2 2.5)}$$

$$t_{ij} = \sqrt{\frac{-(ab+cd) + (a^2+c^2)r^2 - (ad-bc)^2}{a^2+c^2}} \qquad \forall (i,j) \in L \qquad \text{(sup2 2.6)}$$

$a = \upsilon_{ix}(t) - \upsilon_{jx}(t)$, $c = \upsilon_{iy}(t) - \upsilon_{jy}(t)$ $\qquad$ $b = x_i(t) - x_j(t)$, $\mathrm{d} = y_i(t) - y_j(t)$.

This problem can be decomposed into $|V|$ independent subproblems and optimally solved by Algorithm 1. We represent Algorithm 1 as follows:

Step1: Compute the two-dimensional cost matrix B. For each coefficient β_{ij}, it consists of three parts: (1) $\mu_{2ij} + \frac{(M_1-1)}{2} \sum_{w\in W} \mu_{3ijw}$, (2) μ_{1j} , if $i = j$, and (3)

$\sum_{i \in V} \sum_{j \in V} \sum_{p \in P_{ij}} (\sum_{\ell \in L} \mu_{5ij\ell} \cdot (M_2 - t_\ell)) \cdot z_{pij}$ which is the cost of the hop-count constrained shortest path between node i and node j. The path is determined by adopting Bellman-Ford algorithm, where d is the hop count, and $\mu_{5ij\ell} \cdot (M_2 - t_\ell)$ is the link cost of ℓ.

Step 2: For each node v, we select the $g-th$ column with the smallest coefficient β_{vg} at the $v-th$ and set the decision variable b_{vg} as 1.

Step 3: For the selected decision variable b_{vg}, we record the path computed in Step 1 as the intra-cluster routing path between node v and node g.

To demonstrate that we solve Subproblem 2 optimally, we propose Proposition 1 and prove it by using a direct proof.

Proposition 1: Subproblem 2 can be solved optimally by adopting Algorithm 1.

Proof:

1. For the constraint (sub2 2.1), Subproblem 2 can be decomposed into |V| row-wise independent subproblems. Thus, for each node v at $v-th$ row, we select one column g exactly and set the decision variable b_{vg} as 1.

2. For the constraints (sub2 2.2) and (sub2 2.4), if the node v selects the $g-th$ column, the intra-cluster shortest path z_{pvg} with d-hop constraint should be determined simultaneously. Hence, the cost coefficient of the pair of the node v and g consists of two parts: (1) the coefficient of b_{vg} which is calculated by multipliers; and (2) the cost of the intra-cluster path z_{pvg}.

3. By adopting Bellman-Ford algorithm, we can determine all pair intra-cluster

routing paths of node v and g optimally and minimize the cost coefficient of each node pair. Hence, by selecting the $g-th$ column with smallest coefficient β_{vg}, we can minimize the cost of each node v subproblem and solve Subproblem 2 optimally.

Since the complexity of Bellman-Ford algorithm is $|d|\times|E|$, where $|d|$ is the hop count and $|E|$ is the number of the edges, the complexity of computing coefficient β_{ij} is $|W|+|d|\times|E|$. Hence, the complexity of Subproblem 2 is $|V|^2\times(|W|+|d|\times|E|)$.

3.2.3 Subproblem 3 (related to decision variable x_p)

$$Z_{LR-sub3}=\min\sum_{v\in V}\sum_{g\in V}\sum_{w\in W}\mu_{3vgw}\Big(\sum_{p\in P_w}x_p\cdot\delta_{pv}-\frac{\sum_{p\in P_w}x_p\cdot\delta_{pg}}{2}\Big)+\sum_{n\in V}\mu_{4n}\Big(\sum_{(i,j)\in L,i=n}\sum_{w\in W}\sum_{p\in P_w}a_w\cdot x_p\cdot\sigma_{p(i,j)}-C_n\Big)$$
$$-\sum_{w\in W}\sum_{\ell\in L}\mu_{6w\ell}\{(\sum_{p\in P_w}x_p\cdot\delta_{p\ell})\cdot t_\ell+M_2(1-\sum_{p\in P_w}x_p\cdot\delta_{p\ell})\}$$
$$\min\sum_{w\in W}\sum_{p\in P_w}\{\sum_{\ell\in L}\mu_{6w\ell}\cdot(M_2-t_\ell)\cdot\delta_{p\ell}+\sum_{v\in V}(\sum_{g\in V}\mu_{4vgw}-\sum_{v\in V}\frac{\mu_{4gvw}}{2})\cdot\delta_{pv}$$
$$+\sum_{n\in V}\mu_{4n}(\sum_{(i,j)\in L,i=n}a_w\cdot\sigma_{p(i,j)})\}\cdot x_p-\sum_{v\in V}\mu_{4v}\cdot C_v-\sum_{w\in W}\sum_{\ell\in L}\mu_{6w\ell}\cdot M_2$$

(sup3)

Subject to:

$$\sum_{p\in P_w}x_p\le 1 \qquad \forall w\in W \qquad \text{(sup3 3.1)}$$

$$\sum_{\ell\in L}\sum_{p\in P_w}x_p\cdot\delta_{p\ell}\le H_w \qquad \forall W\in W \qquad \text{(sup3 3.2)}$$

$$x_p=0\ or\ 1 \qquad \forall p\in P_w \qquad \text{(sup3 3.3)}$$

$$t_{ij}=\sqrt{\frac{-(ab+cd)+(a^2+c^2)r^2-(ad-bc)^2}{a^2+c^2}} \qquad \forall(i,j)\in L \qquad \text{(sup3 3.4)}$$

$a=\upsilon_{ix}(t)-\upsilon_{jx}(t)$, $c=\upsilon_{iy}(t)-\upsilon_{jy}(t)$ $b=x_i(t)-x_j(t)$, $\mathrm{d}=y_i(t)-y_j(t)$.

This problem is related to the existing O-D pairs and can be decomposed into $|W|$ independent shortest path subproblems with the h-hop count constraint, link cost and nodal cost. The hop count constraint is predefined as H_w, the link cost of ℓ is calculated by $\mu_{6w\ell} \cdot (M_2 - t_\ell)$, and the nodal cost of node i is the sum of $(\sum_{j \in V} \mu_{4ijw} - \sum_{j \in V} \frac{\mu_{4jiw}}{2})$ and $a_w \cdot \mu_{4i}$. Again, Subproblem 3 can be solved by adopting Bellman-Ford algorithm. Since the complexity of Bellman-Ford algorithm is $|H_w| \times |E|$, where $|H_w|$ is the hop count and $|E|$ is the number of the edges, the complexity of Subproblem 3 is $|W| \times |H_w| \times |E|$.

3.2.4 Subproblem 4 (related to decision variable T)

$$Z_{LR-sub4} = \min(-1 + \sum_{v \in V} \sum_{g \in V} \sum_{\ell \in L} \mu_{5vg\ell} + \sum_{w \in W} \sum_{\ell \in L} \mu_{6w\ell}) \cdot T$$

Subject to:

$$M_2 \ge T \ge t_1 + t_2 + t_3 . \qquad \text{(sup4 4.1)}$$

This problem can be optimally solved by a simple algorithm. By computing the coefficient $-1 + \sum_{v \in V} \sum_{g \in V} \sum_{\ell \in L} \mu_{5vg\ell} + \sum_{w \in W} \sum_{\ell \in L} \mu_{6w\ell}$, if the coefficient is less than zero we set T as the upper bound M_2. Otherwise, we set T as the lower bound $t_1+t_2+t_3$. The complexity of Subproblem 4 is $(|V|^2 + |W|) \times |E|$, which is the number of iterations for computing the coefficient.

3.3 The Dual Problem and the Subgradient Method

According to the weak Lagrangean duality theorem, for any $\mu_{1g}, \mu_{2vg}, \mu_{3vgw}, \mu_{4n}$, $\mu_{5vg\ell}, \mu_{6w\ell} \geq 0$, $Z_d(\mu_1, \mu_2, \mu_3, \mu_4, \mu_5, \mu_6)$ is a lower bound of (IP 1). The following dual problem (D2) is then constructed to calculate the tightest lower bound.

Dual Problem (D2)

$$Z_{D2} = \max \ Z_d(\mu_1, \mu_2, \mu_3, \mu_4, \mu_5, \mu_6) \tag{D2}$$

Subject to:

$\mu_{1g} \geq 0$	$\forall v \in V$	(D 2.1)
$\mu_{2vg} \geq 0$	$\forall v, g \in V$	(D 2.2)
$\mu_{3vgw} \geq 0$	$\forall v, g \in V \ w \in W$	(D 2.3)
$\mu_{4n} \geq 0$	$\forall n \in V$	(D 2.4)
$\mu_{5vg\ell} \geq 0$	$\forall v, g \in V \ \ell \in L$	(D 2.5)
$\mu_{1g} \geq 0$	$\forall w \in W \ \ell \in L$	(D 2.6)

The most popular method to solve the dual problem (D2) is the subgradient method. Let g be a subgradient of $Z_d(\mu_1, \mu_2, \mu_3, \mu_4, \mu_5, \mu_6)$. Then, in iteration k of the subgradient optimization procedure, the multiplier $\pi = (\mu_1, \mu_2, \mu_3, \mu_4, \mu_5, \mu_6)$ is updated by $\pi^{k+1} = \pi^k + t^k \cdot g^k$. The step size t^k is determined by $t^k = \delta \times \frac{Z_{IP1}^h - Z_{D2}(\pi_k)}{\| g^k \|^2}$. Thus, Z_{IP1}^h is the primal objective function value for a heuristic solution. δ is the constant between 0 and 2.

Chapter 4 Getting Primal Feasible Solution

4.1 Lagrangean Relaxation Results

By using Lagrangean Relaxation method and the subgradient method to solve our complicated problem, we obtain not only a theoretical lower bound, but also some hints for us to get a feasible solution to the primal problem per iteration.

Since some difficult constraints of the primal problem are relaxed by using Lagrangean relaxation method, we can not guarantee that the consolidated result of the Lagrangean relaxation problem is feasible to the primal problem. A feasible solution is found, if the decision variables of the result are satisfied with all constraints of the primal problem. Otherwise, to get a primal feasible solution, a modification to this kind of infeasible result is necessary.

The modification is focus on adjusting the decision variables of the Lagrangean relaxation problem to be satisfied with the relaxed constraints of the primal problem, because other constraints are considered in the divided subproblems. To get a primal feasible solution, we use the result of the Lagrangean relaxation problem at each iteration as a starting point and manipulate our proposed heuristic procedure. Considering with different decision variables and the corresponding relaxed constraints, we divide the heuristic procedure into three parts: clusterhead/cluster member adjustment, inter-cluster routing adjustment and minimum link duration adjustment. We discuss each heuristic adjustment in detail in the following sections.

4.2 Getting Primal Feasible Heuristics

To get a feasible solution to the primal problem, we consider the consolidated result of the Lagrangean relaxation problem and adopt the following heuristic adjustments.

4.2.1 Clusterhead/Cluster Member Relationship Adjustment

To adjust the relaxed constraints (IP 1.2) and (IP 1.3), we consider the decision variables h_g and b_{vg} of Subproblem 1 and Subproblem 2 respectively. Then, for each node v, we adjust the decision variables by the following conditions them if necessary.

Step 1: If the node v is determined as a clusterhead by setting h_v as 1, the constraint (IP 1.2) defines that the node v must be belong to itself and set the decision variable b_{vv} as 1. If not, go to step 3.

Step 2: If the node v is not a clusterhead, we check the selected decision variable b_{vg} by the constraint (IP 1.3). If the node v selects the node g which is not a clusterhead as its clusterhead by setting b_{vg} as 1 and h_g as 0, we simply adjust the node g to be a new clusterhead in step 3.

Step 3: For the adjusted node g, we simply set the decision variable h_g as 1, refresh the $g-th$ row of the b_{vg} matrix, and set the decision variable b_{gg} as 1.

4.2.2 Inter-Cluster Routing Adjustment

To adjust the relaxed constraints (IP 1.11) and (IP 1.13), we consider the decision variable x_p of Subproblem 3. For each O-D pair w, we adjust each node of the routing path x_p by the following procedure.

Step 1: First, we route the traffic to the clusterhead g of the source node s by using the intra-cluster routing path z_{psg}. Second, we select a gateway node whose minimum link duration of its intra-cluster routing path is max from the current cluster. By using the intra-cluster routing path, we continue on routing the traffic to the next node of the path x_p.

Step 2: Then, we continue on adjusting the next node of the path x_p. If the node is in the cluster of the last adjusted node, we simply route the traffic to it. Otherwise, it is a gateway node of the adjacent cluster. We route the traffic to the gateway node, adjust the path to the new clusterhead, and then determine a gateway node to route the traffic to the next node of the path x_p like step 1.

Step 3: Repeat Step 2 until reaching the destination node of the O-D pair w.

Step 4: During adjusting each node of the path x_p, we increase the hop count and subtract the traffic demand a_w of the O-D pair w from its available transmission rate. If the hop count exceeds the upper bound H_w or the remaining available transmission rate of an intermediate relay node is less than the traffic demand a_w, we terminate the getting primal feasible procedure and skip to the next iteration, since we cannot get a primal feasible solution in this iteration.

4.2.3 Minimum Link Duration Adjustment

To adjust the relaxed constraints (IP 1.15) and (IP 1.16), we consider the decision variable T of Subproblem 4 and try to compute the minimum link duration of the constructed cluster topology and the routing assignment. Initially, we set T as a big number and then adjust it by the links of the intra-cluster and inter-cluster routing paths.

Step 1: For each link of the intra-cluster routing path z_{pvg}, if the link duration is less than T, we update T by this value.

Step 2: For each link of the inter-cluster routing path x_p, if the link duration is less than T, we update T by this value.

Chapter 5 Computational Experiments

Since the clustering problem is proved as NP-Complete, it is not easy for us to get a tight theoretical lower bound to our primal problem by solving the Lagrangean relaxation problem iteration by iteration. To evaluate the solution quality of our best primal feasible solution, we also implement four heuristic algorithms **[2][4]**. By comparing the computational result of the Lagrangean relaxation based algorithm with those of the four algorithms, we can demonstrate the difference of the solution quality between them.

5.1 Simple Algorithms

In Chapter 3, the problem is decomposed into three parts: the cluster construction subproblems (Subproblem 1 and Subproblem 2), the inter-cluster routing assignment subproblem (Subproblem 3), and the minimum link duration computation subproblem (Subproblem 4). The major difference between the Lagrangean relaxation based algorithm and the four algorithms is in the criteria of the cluster construction. Since there is no information about determining the inter-cluster routing assignment in the four algorithms, we apply Bellman-Ford algorithm with H_w hop-count constraint to solve the subproblem as the Lagrangean relaxation based algorithm. Finally, we use the minimum link duration adjustment to calculate the objective value of these five algorithms.

In the following sections, we describe the four algorithms, namely, lowest identifier based algorithm (LID), highest degree based algorithm (HD), MaxMin algorithm, MHMR algorithm and Lagrangean relaxation based algorithm (LR) in detail.

5.1.1 Lowest Identifier Based Algorithm (LID)

The heuristic criterion of lowest identifier based algorithm is electing the node which has minimum identifier in its d-hop neighboring nodes as a clusterhead. Thus, a non-clusterhead node selects the node which has minimum identifier in its d-hop neighboring nodes as its clusterhead.

Each node maintains two values, Winner and Sender. The Winner is the selected identifier of each node at each round and initialized by its own identifier. The Sender is the node identifier for a particular round and used to determine the shortest path back to the selected clusterhead. The detail procedure is represented as follow:

Step 1: Each node locally broadcasts its Winner value to all its 1-hop neighboring nodes. After all neighboring nodes have heard from, for a single round, the node updates its Winner value by the smallest value among all its received Winner values and Sender value by the identifier of the corresponding neighboring node. This process continues for d rounds.

Step 2: After the information exchange, if the Winner value of a node is the same as its identifier, it declares itself as a clusterhead. Otherwise, the node uses its Sender value to construct the shortest path back to the clusterhead elected by its Winner value.

Step 3: The clusterhead of each cluster determines the cross links and gateway nodes. As mentioned above, the nodes of a cross link belong to two different clusterheads and could be the gateway nodes of these two clusters respectively.

Step 4: Execute Bellman-Ford algorithm with H_w hop count constraint to determine the shortest path for each O-D pair w. Then, to be satisfied with the nodal capacity and hierarchical routing constraint, we adjust the path by adopting the adjustment proposed in section 4.2.2.

Step 5: We calculate the minimum link duration by adopting the adjustment proposed in section 4.2.3.

5.1.2 Highest Degree Based Algorithm (HD)

The difference between highest degree based algorithm and lowest identifier based algorithm is that we elect the node which has highest degree in its d-hop neighboring nodes as a clusterhead. Again, a non-clusterhead node selects the node which has highest degree in its d-hop neighboring nodes as its clusterhead.

Besides Winner and Sender, this algorithm needs an extra data structure Degree to record the highest degree heard from its neighboring nodes at each round. The Degree value of each node is initialized its own connectivity degree. The algorithm procedure is the same as lowest identifier based algorithm, but, for a single round, each node updates its Degree value by the largest value among all its received Degree values, Winner value by the identifier of the node which has the highest Degree value, and Sender values by the identifier of the corresponding neighboring node.

5.1.3 MaxMin Algorithm

MaxMin algorithm is a variation of lowest identifier based algorithm for load balance **[4]**. For recording the exchanged information per round, each node maintains two arrays of size 2d, Winner and Sender, which are initialized by its own identifier. For cluster construction, the algorithm includes the Floodmin phase and the Floodmax phase and is represented as follow:.

Step 1: This step is called as Floodmin phase and the same as step 1 of lowest identifier based algorithm. However, for the d-th round, we record the selected value in the d-th value of the arrays: Winner and Sender.

Step 2: (Floodmax) This step is similar to Floodmin phase, whereas a node chooses the largest value as its new Winner instead of the smallest value.

Step 3: (Clusterhead Election and Node Pair) First, each node checks to see if it has received its own original node id during the 2d rounds of flooding, it declares itself as a clusterhead and skip the rest phases of the heuristic. Otherwise, each node looks for a minimum node pair which is a node identifier that occurs at least once as a WINNER value in both the 1st (Floodmax) and 2nd (Floodmin) d rounds. Then, the node selects the node pair identifier as its clusterhead and uses the corresponding Sender value to construct the intra-cluster routing path. If a node pair does not exist, the node selects the minimum node id in the 1st d rounds of flooding as its clusterhead.

Step 4~6: The following steps are the same as the steps 3~5 of lowest identifier based algorithm.

5.1.4 MHMR Algorithm

MHMR algorithm is proposed in **[2]** for organizing mobile nodes into non-overlapping clusters which has adaptive variable size according to their respective mobility. We show the main steps of this algorithm as follows.

Step 1: Each node locally broadcasts its velocity information ($\upsilon(n,t_i),\ i=1,2...$)

to all its neighboring nodes periodically.

Step 2: Upon reception of neighboring nodes' velocity information, each node calculates the relative velocity $V(m,n,T)$ of each pair node and exchange it periodically. Then, each node m calculates the average relative mobility $M_{m,n,T} = \frac{1}{N}\sum_{i=1}^{N}|V(m,n,T)|$ between itself and node n from the n intervals of the T period.

Step 3: For each node m, it uses all received mobility information to calculate the mean value m_{mob}, the standard deviation δ_{mob}, and set the threshold value Th_{mob} as $Th_{mob} = m_{mob} + k \cdot \delta_{mob}$. The parameter k was chosen as 1.5 based on experimentation **[2]**. The node among all neighboring nodes of node m which has the lowest identifier and satisfies with the condition $TCH = Least_{i \in S_m}\{ID \mid M_{m,i,T} < Th_{mob}\}$, is selected as a tentative clusterhead. S_m is the neighbor set of node m.

Step 4: (Cluster Merging) According to step3, a parent clusterhead can include other child clusterheads as long as satisfying the TCH criteria and d-hop count constraint.

Step 5~7: The following steps are the same as the steps 3~5 of lowest identifier based algorithm.

5.2 Lagrangean Relaxation Based Algorithm (LR)

This algorithm is based on the mathematical formulation described in Chapter2. The Lagrangean relaxation problem is solved optimally as described in Chapter 3 for getting a lower bound to the primal problem. We adopt the heuristic procedure proposed in Chapter 4 to get a primal feasible solution at each iteration. Thus, we use the subgradient method to update the Lagrangean multipliers. We summarize the

Lagrangean relaxation based algorithm into the following steps:

Step 1: (Initialization)

1. Generate a random connected network topology, O-D pairs and the traffic demand of each O-D pair.
2. Initialize all multipliers to an infinitesimal value, Epsilon. Set upper bound (UB), and iteration count as 0 and delta factor as 2.

Step 2: (Termination Criteria)

1. The gap between the upper bound (UB) and the lower bound (LB) is less than Epsilon, 10^{-4}. (Convergence case 1)
2. The number of iterations exceeds the maximum iteration count.
3. Step size is less than 10^{-7}. (Convergence case 2)

Step 3: (Calculating Lower Bound)

With the given Lagrangean multipliers per iteration, we solve the subproblems optimally as described in Chapter 3 to get the value Z_d .

Step 4: (Getting Primal Feasible Solution)

Apply the heuristic procedure proposed in Chapter 4 to calculate the value Z_{IP} .

Step 5: (Updating Lower Bound, Upper Bound and Lagrangean Multipliers)

1. If $Z_d > LB$, update LB by Z_d .
2. If $Z_{IP} < UB$, update UB by Z_{IP} .
3. Calculate the step size and update Lagrangean multipliers by using the subgradient method as described in section 3.3.
4. Increase the iteration count *i* and go to Step 2 if no mating with the termination criteria.

5.3 Parameters and Cases of Experiment

Number of iteration	1000
Improvement counter	10
Begin to get primal feasible solution	1
Initial upper bound	0
Initial scalar of step size	2
Stopping step size	10^{-7}
Hop count of cluster construction	3

Table 5-1 Parameters of Lagrangean relaxation based algorithm

Number of nodes	20~100 (depend on each case)
Region	100 x 100 unit square
Transmission radius	30,33,35 unit (depend on each case)
Geographical position x-axis	0~100 unit
Geographical position y-axis	0~100 unit
Maximum velocity	0~10 unit/sec
Maximum direction	0~2 π
Nodal capacity	10~20 bits/sec
O-D pair number	2~50 (depend on each case)
O-D pair traffic demand	1~5 bits/sec

Table 5-2 Parameters of testing cases

The parameters listed in Table 5-1 and Table 5-2 are used for all cases of experiment. The network topologies with different number of nodes used in our experiment are generated by the fixed random seed 100. Considering with connectivity, we set the transmission radius of the small network as a larger value.

Thus, we increase the number of hop count H_w as the size of the network grows.

First, we experiment different number of nodes with different number of O-D pairs to evaluate the gap (%) of the result of Lagrangean relaxation based algorithm. The experiment result is summarized in Table 5-3. Second, we compare the result of Lagrangean relaxation based algorithm with those of the four heuristic algorithms as described in section 5.1. The experiment result is summarized in Table 5-4. Finally, we design a mobility analysis and evaluate the minimum link duration at different mobility level and network size with medium traffic load. The experiment result is summarized in Table 5-5.

5.4 Experiment Results

To make the comparison easier, solutions to the minimization problem are transformed into solutions to the original maximization problem. The Gap (%) is calculated by (UB-LB)*100/LB.

node		OD -pair	Lower Bound	Upper Bound	Gap (%)
20	radius=35 Hw=10	2	35.1628	34.021	3.247182
		5	35.0117	31.9614	8.71223
		10	34.8269	31.9614	8.227835
		15	25.2112	21.1255	16.20589
30	radius=35 Hw=10	2	35.1779	34.8084	1.050375
		5	35.1777	34.7818	1.125429
		10	35.1707	33.7561	4.022098
		20	35.1784	30.1026	14.42874

40	radius=33 Hw=12	2	33.189	32.8476	1.028654
		5	33.1876	32.6167	1.720221
		10	33.1852	32.4632	2.175669
		20	33.1865	24.9667	24.76851
50	radius=30 Hw=12	2	30.1959	28.674	5.040088
		5	30.1929	28.674	5.030653
		10	30.1734	28.674	4.969278
		20	30.1874	28.2976	6.260228
60	radius=30 Hw=15	2	30.2078	29.7762	1.42877
		5	30.2056	29.7716	1.43682
		10	30.2042	29.2228	3.249217
		20	30.2	29.123	3.566225
		30	30.2061	28.9754	4.074343
70	radius=30 Hw=15	2	30.2077	29.3822	2.732747
		5	30.2073	28.9441	4.181771
		10	30.2061	28.9441	4.177964
		20	30.1917	28.9441	4.132262
		30	30.1986	26.4613	12.37574
80	radius=30 Hw=15	2	30.2079	29.6624	1.805819
		5	30.2077	29.2904	3.036643
		10	30.2076	29.2904	3.036322
		20	30.2075	28.9754	4.078788
		30	30.207	28.7366	4.867746
		40	30.2065	28.7366	4.866171

90	radius=30 Hw=18	2	30.2079	29.6042	1.998484
		5	30.2075	29.6042	1.997186
		10	30.2079	29.6042	1.998484
		20	30.2065	29.2586	3.138066
		30	30.2076	28.6044	5.307274
		40	30.2077	28.6044	5.307587
100	radius=30 Hw=18	2	30.2079	29.8697	1.119574681
		5	30.2071	29.6532	1.833674865
		10	30.2075	29.4654	2.456674667
		20	30.2079	28.874	4.415732308
		30	30.2079	28.8366	4.539540981
		40	30.2052	25.7867	14.62827593
		50	30.2079	28.8366	4.539540981

Table 5-3 Evaluation of Gap (%) by given different number of nodes and O-D pairs

Then, we summarize the above experiment results into diagrams and present them in Figure 5-1, Figure 5-2, and Figure 5-3.

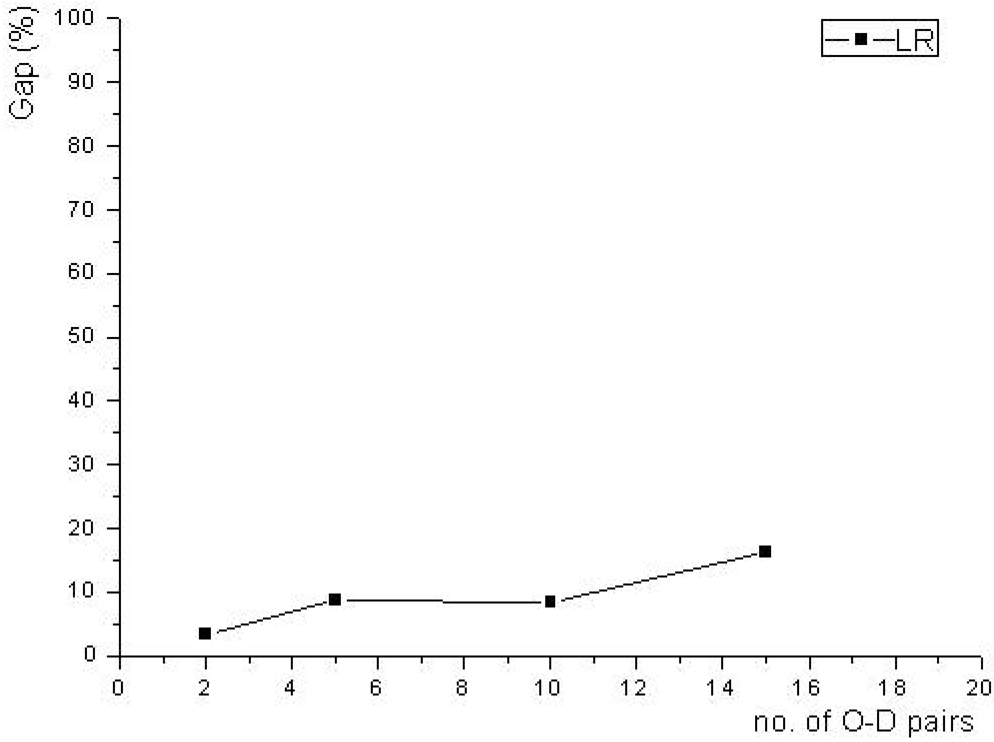

Figure 5-1 Evaluation of Gap (%) (Small Network, No. of Nodes=20)

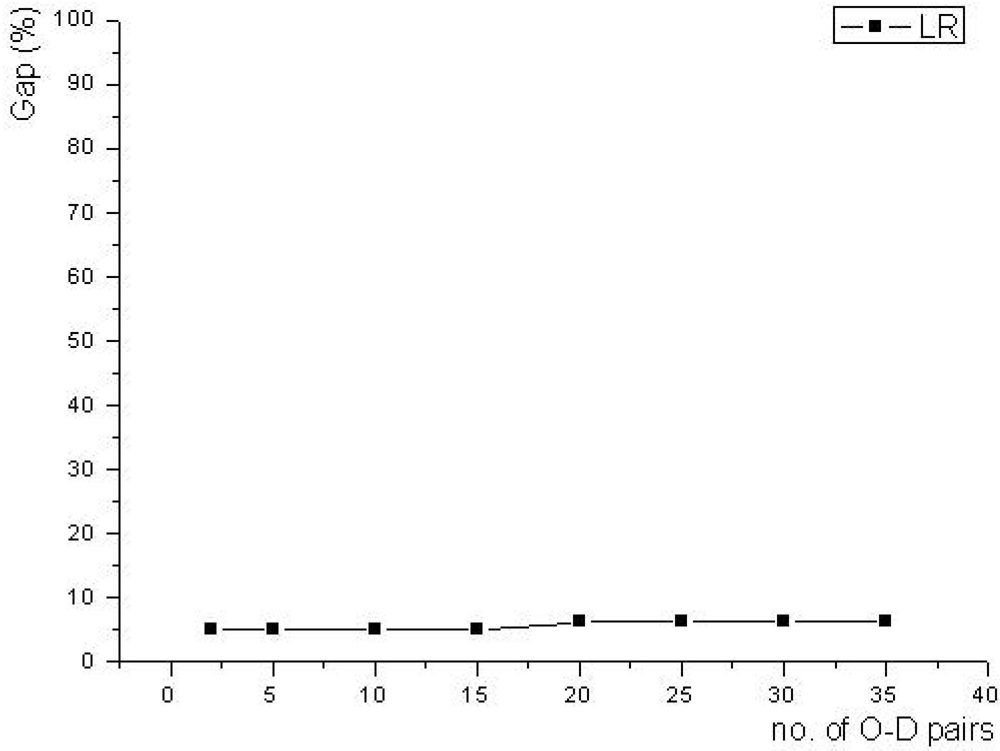

Figure 5-2 Evaluation of Gap (%) (Medium Network, No. of Nodes=50)

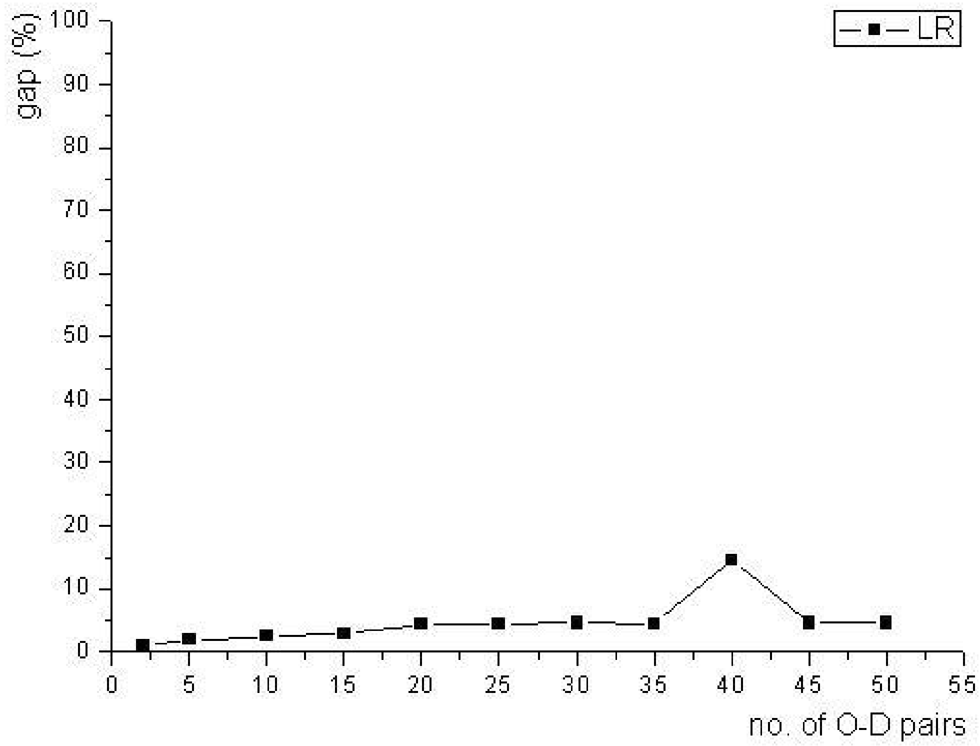

Figure 5-3 Evaluation of Gap (%) (Large Network, No. of Nodes=100)

In Table 5-4, the values are experiment results of Lagrangean relaxation based algorithm and four simple algorithms. We calculate the minimum link duration of all O-D to evaluate the solution quality. The improvement ratio is calculated by (LR-SA)*100/SA. The "x" indicates no feasible solution for that case.

node		O-D pair	LR	LID	Impro. Ratio(%)	HD	Impro. Ratio(%)	MaxMin	Impro. Ratio(%)	MHMR	Impro. Ratio(%)
20	radius=35 Hw=10	2	34.021	31.303	8.681	22.435	51.643	28.874	17.826	31.304	8.678
		5	31.961	22.192	44.024	18.665	71.234	22.192	44.024	22.192	44.024
		10	31.961	9.424	239.132	9.424	239.132	x	x	x	x
30	radius=35 Hw=10	2	34.808	22.192	56.853	25.149	38.409	28.814	20.805	26.963	29.097
		5	34.782	9.424	269.059	22.519	54.454	22.519	54.454	x	x
		10	33.756	x	x	x	x	x	x	x	x

40	radius=33 Hw=12	2	32.848	18.881	73.976	22.674	44.869	28.459	15.420	18.881	73.976
		5	32.617	14.936	118.384	22.674	43.851	26.404	23.532	14.936	118.384
		10	32.463	14.936	117.356	x	x	22.883	41.866	x	x
		20	24.967	x	x	x	x	x	x	x	x
50	radius=30 Hw=12	2	28.674	9.744	194.271	13.346	114.844	9.744	194.271	12.941	121.573
		5	28.674	9.744	194.271	x	x	9.744	194.271	9.744	194.271
		10	28.674	6.518	339.915	x	x	x	x	6.518	339.915
		20	28.298	x	x	x	x	x	x	x	x
60	radius=30 Hw=15	2	29.776	19.674	51.347	11.374	161.799	21.166	40.683	20.050	48.510
		5	29.772	8.656	243.933	11.374	161.758	18.147	64.062	19.460	52.993
		10	29.223	6.518	348.334	x	x	x	x	x	x
		20	29.123	x	x	x	x	x	x	x	x
70	radius=30 Hw=15	2	29.382	12.941	127.046	11.577	153.800	19.992	46.967	15.820	85.734
		5	28.944	7.377	292.344	11.153	159.523	19.619	47.534	15.820	82.965
		10	28.944	7.377	292.344	x	x	x	x	x	x
		20	28.944	x	x	x	x	x	x	x	x
80	radius=30 Hw=15	2	29.662	6.158	381.683	24.051	23.332	17.568	68.844	15.313	93.710
		5	29.290	6.158	375.642	x	x	17.568	66.727	15.313	91.280
		10	29.290	x	x	x	x	x	x	x	x
		20	28.975	x	x	x	x	x	x	x	x
90	radius=30 Hw=18	2	29.604	18.110	63.465	12.198	142.691	24.985	18.487	11.838	150.084
		5	29.604	12.941	128.761	12.198	142.691	14.763	100.531	11.838	150.084
		10	29.604	9.744	203.817	x	x	x	x	x	x
		20	29.259	x	x	x	x	x	x	x	x

100	radius=30 Hw=18	2	29.870	7.377	304.891	22.149	34.856	12.446	139.994	17.725	68.518
		5	29.653	7.377	301.956	x	x	12.446	138.255	17.725	67.296
		10	29.465	x	x	x	x	x	x	x	x
		20	28.874	x	x	x	x	x	x	x	x
		30	28.837	x	x	x	x	x	x	x	x

Table 5-4 Comparison of different algorithms

Again, we summarize the above experiment results into diagrams and present them in Figure 5-4, Figure 5-5, and Figure 5-6.

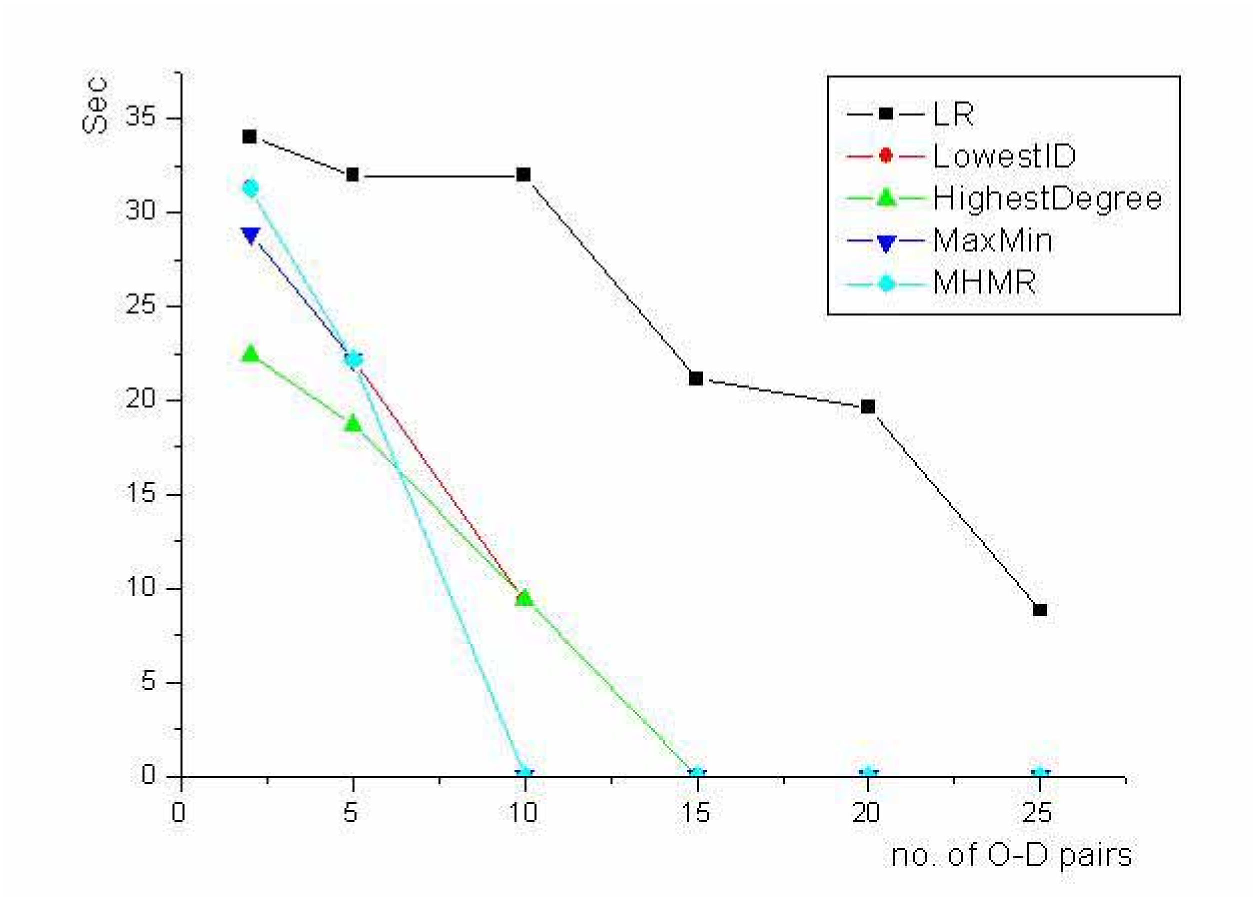

Figure 5-4 Minimum Link Duration (Small Network, No. of nodes=20)

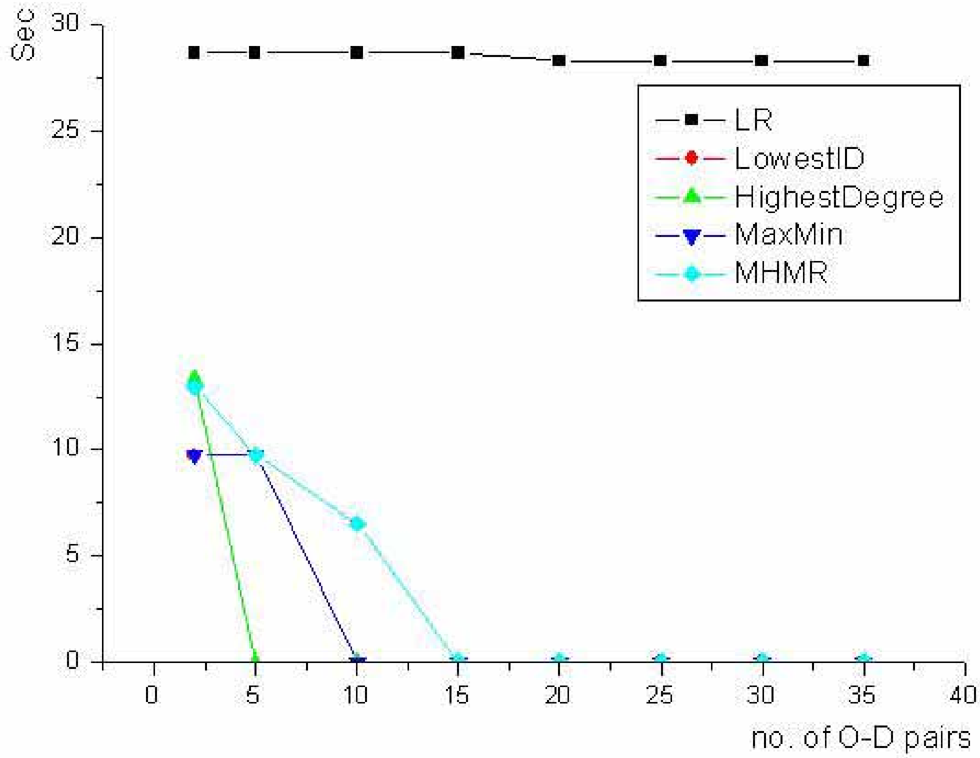

Figure 5-5 Minimum Link Duration (Medium Network, No. of Nodes=50)

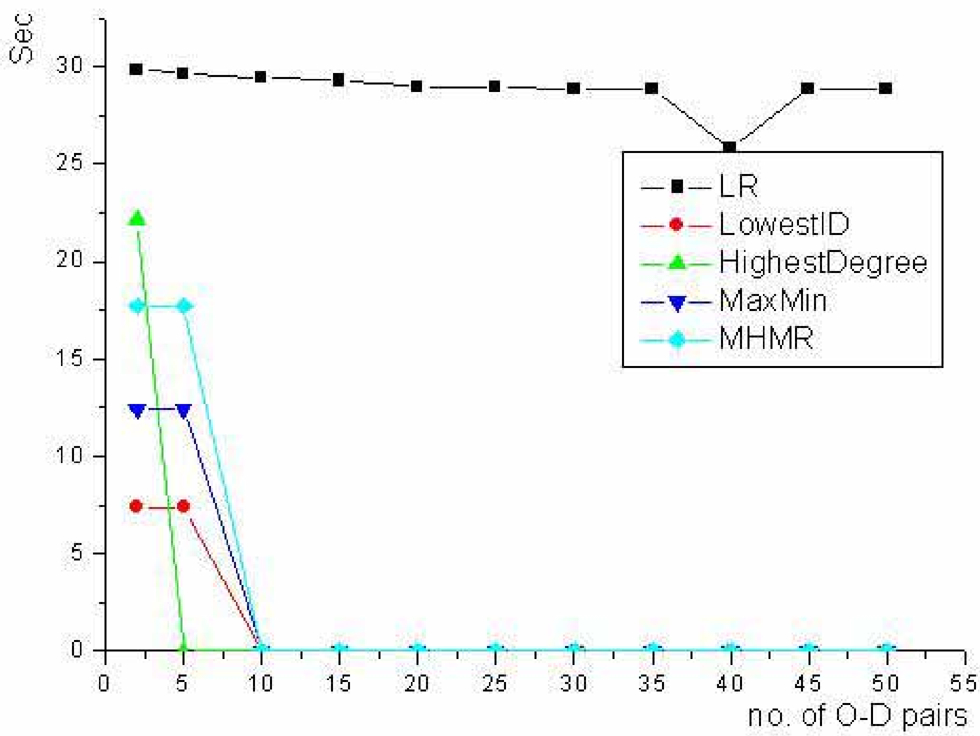

Figure 5-6 Minimum Link Duration (Large Network, No. of Nodes=100)

In Table 5-5, the values are experiment results of Lagrangean relaxation based algorithm at different mobility level. We calculate the minimum link duration of all O-D pairs to evaluate the solution quality. There are three design cases, the small network (node size=20), the medium network (node size=50), and large network (node size=100) with medium traffic load and different mobility level.

Network size	**Parameters**	**Mobility Level**	**Lowe Bound**	**Upper Bound**	**Gap (%)**
nodes=100	radius=30 Hw=18 No. of O-D pairs=30	1	59.6184	45.709	23.33072
		3	41.922	36.9618	11.83197
		5	36.287	34.2077	5.730151
		7	33.1219	31.6747	4.369315
		10	30.2079	28.8366	4.539541
		15	26.7636	25.5822	4.414204
		20	22.1038	21.2985	3.643265
		25	14.4579	13.6751	5.414341
		30	3.06906	2.30117	25.02036
nodes=50	radius=30 Hw=12 No. of O-D pairs=20	1	59.5688	45.4467	23.70721
		3	41.4452	36.2707	12.48516
		5	36.1443	34.0185	5.881425
		7	33.1039	31.3164	5.399666
		10	30.1734	28.674	4.969278
		15	26.7529	25.3216	5.350074
		20	22.0946	20.7079	6.276194
		25	14.4487	13.0897	9.40569
		30	3.05513	1.71918	43.72809

nodes=20	radius=35 Hw=10 No. of O-D pairs=10	1	65.3501	55.6161	14.89516
		3	48.0922	41.6076	13.48368
		5	42.0324	36.6175	12.88268
		7	38.5153	33.6626	12.59941
		10	34.8269	31.9614	8.227835
		15	31.7275	27.4151	13.59199
		20	27.0724	22.7899	15.81869
		25	18.415	15.1647	17.65029
		30	7.96189	5.0116	37.05515

Table 5-5 Mobility analysis

Note that at the high mobility level (eg. 30units/sec), the network topology and link state change dramatically and make it difficult to get a primal feasible solution. We also summarize the above experiment results into diagrams and present them in Figure 5-7, Figure 5-8, and Figure 5-9.

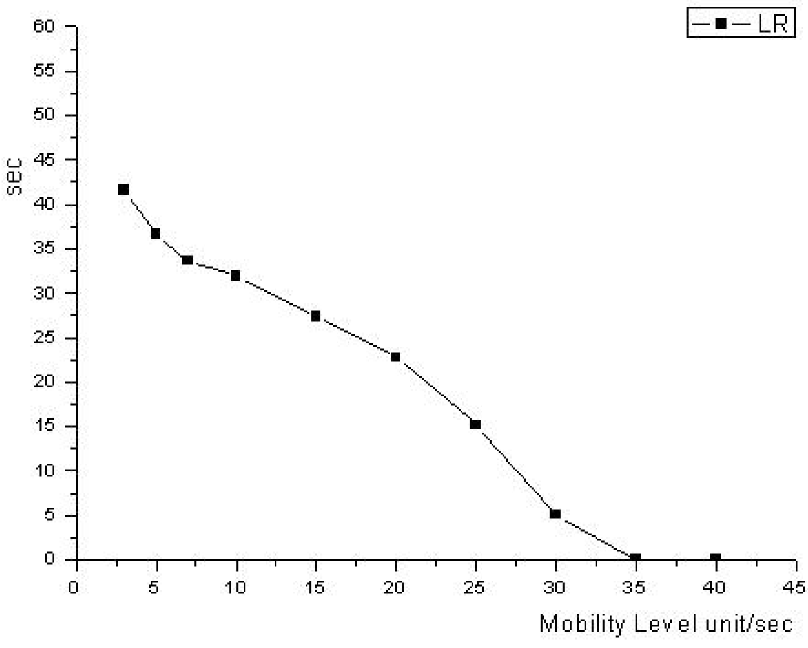

Figure 5-7 Mobility Analysis (Small Network, No. of O-D pairs=10)

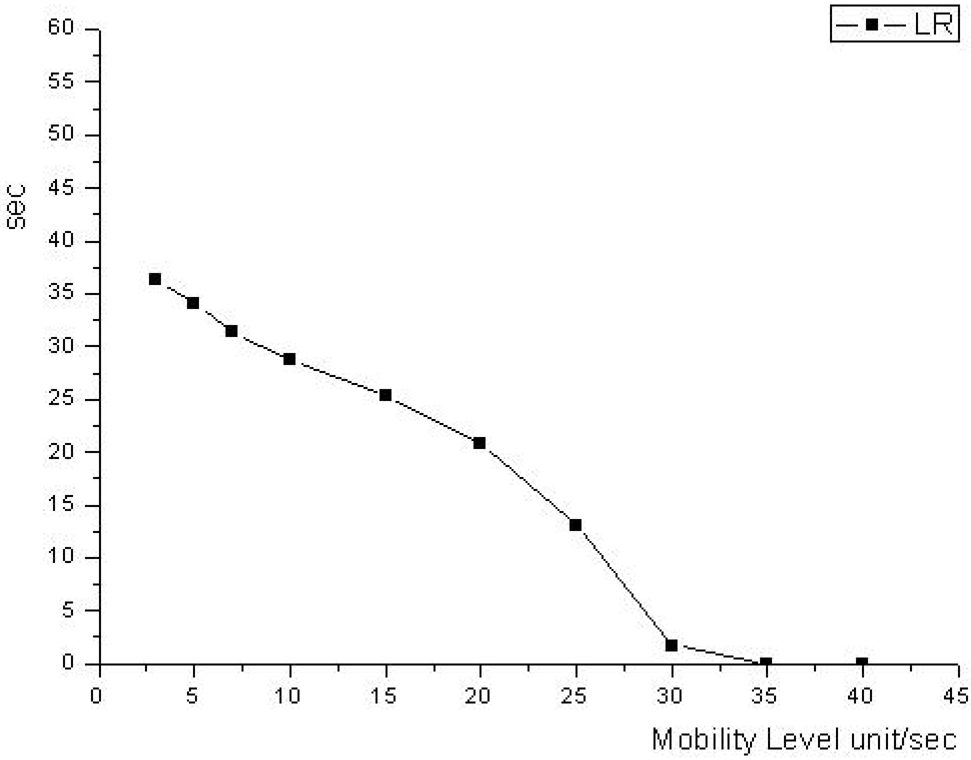

Figure 5-8 Mobility Analysis (Medium Network, No. of O-D pairs=20)

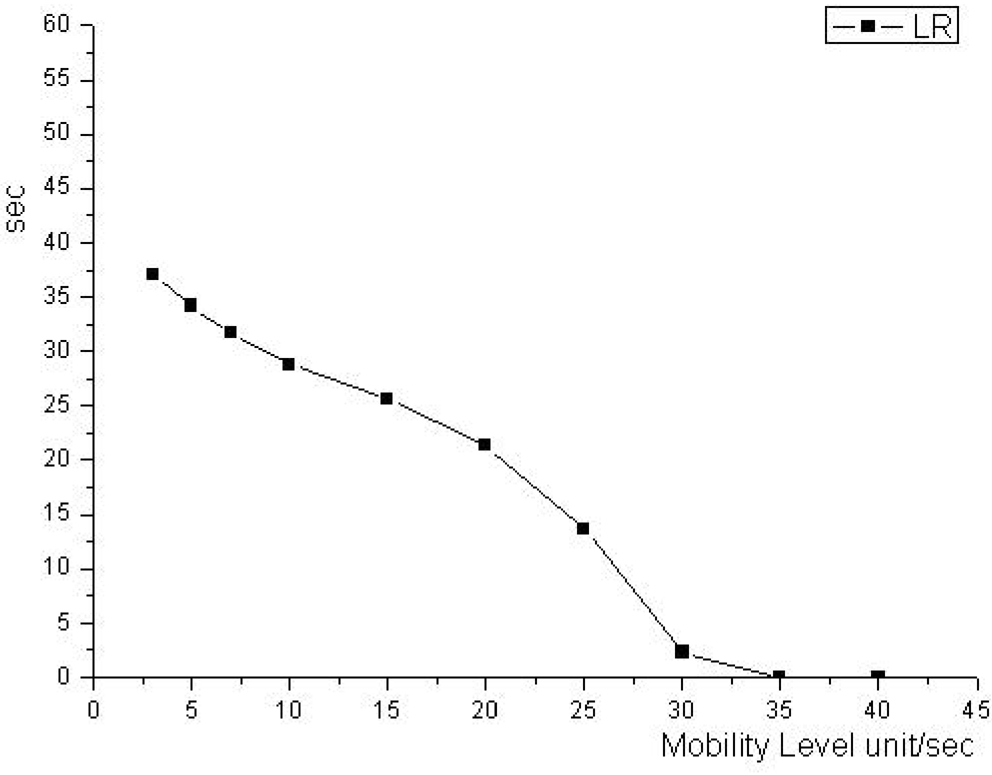

Figure 5-9 Mobility Analysis (Large Network, No. of O-D pairs=30)

5.5 Result Discussion

According to our experiment results, in Table 5-4 Comparison of different algorithms, we can observe that the solution quality of the LR based algorithm is much better than that of the other four simple heuristic algorithms. The reason is that the four simply algorithms just use node identity, connectivity degree, and relative mobility as heuristics, and do not quantities the duration of the constructed cluster topology and routing assignment. In contrast, by adopting the Lagrangean relaxation approach, we consider the multipliers of the corresponding relaxed constraints as the part of the link cost function. The multipliers are adjusted by the subgradient method iteration by iteration. To minimize the objective value of the Lagrangean relaxation problem, we attempt to adjust the heavily-loaded clusterheads and reroute the congested routing paths. Hence, we can still get a good primal solution in a heavily-loaded, large scale network. In Table 5-3, we show that as the network size and the number of O-D grow, the convergence property (gap %) and solution quality of our proposed algorithm become better and better.

In the mobility analysis, we find that links are fragile and network topology change frequently at a high mobility level. These problems cause the routing path is not stable enough to finish the transmission. Although it is difficult to get a good primal solution at a high mobility level, by adopting Lagrangean relaxation method and our proposed algorithms, the solution quality we obtained is still not bad and has significant improvement ratio than those of the other algorithms. Based on our experiment results, our algorithm has the best convergence property and solution quality in large scale networks with the medium traffic load and mobility pattern.

5.6 Computational Time

No. of Nodes	LR	Average Time
20	1	0.001
30	1.5	0.0015
40	5	0.005
50	10	0.01
60	24	0.024
70	79	0.079
80	135	0.135
90	284	0.284
100	489	0.489

Table 5-6 Computation Time with No. of O-D Pair =10

The test platform is a PC with Pentium4 2.4G CUP and 1024MB DRAM. We execute our program on Windows 2000 and Virtual C++ 6.0. The program is written in programming language C++.

Because the complexity of our proposed Lagrangean relaxation based algorithm is dominated by the number of edges, as the network size grows, the computational time increases quickly. However, the growth of the computational time is still under an acceptable level.

Chapter 6 Real-Time Reliable Cluster Construction and QoS-Constrained Routing Assignment

As mentioned as Section 2.1, it takes the lead time, which consists of data collection time, processing time, and decision dissemination time, to make a decision. To design a good network management mechanism in, we should compute a new cluster topology before the duration of the cluster topology and the routing assignment, computed during the last decision cycle, is expired. In the previous section, we regard the reliable cluster construction (RCC) problem as a network planning problem and do not consider the lead time. Hence, in this section, we design a real-time cluster construction mechanism with the QoS-constrained routing assignment which should report a decision in a limited time interval.

In a general distributed environment, since the transmission occasion of each node is infinite and each has infinitesimal probability, we can assume the number of new O-D pairs is a Poisson distribution with mean rate λ. The transmission holding time is modeled as an exponential distribution with the mean value μ. Hence, for a decision time interval T, the number of new O-D pairs, which would arrived during this decision cycle, is calculated by $\lambda \times T$. In addition, if the transmission holding time of an old O-D pair is longer than the decision time interval, the residual traffic demand of the O-D pair would be served during the next decision cycle. Figure 6-1 shows an illustration of the number of O-D pairs at different decision cycle. T_{n-1} and T_n are the decision time intervals of the n-1-th and the n-th decision cycle respectively. Thus, ε_{n-1} and γ_{n-1} are the number of existing O-D pairs and the number of remaining O-D pairs respectively. The total number of O-D pairs of the n-th decision cycle ε_n is calculated by $\gamma_n + \lambda \times T_{n-1}$, where $\lambda \times T_{n-1}$ is the number of new O-D pairs at the n-1-th decision cycle.

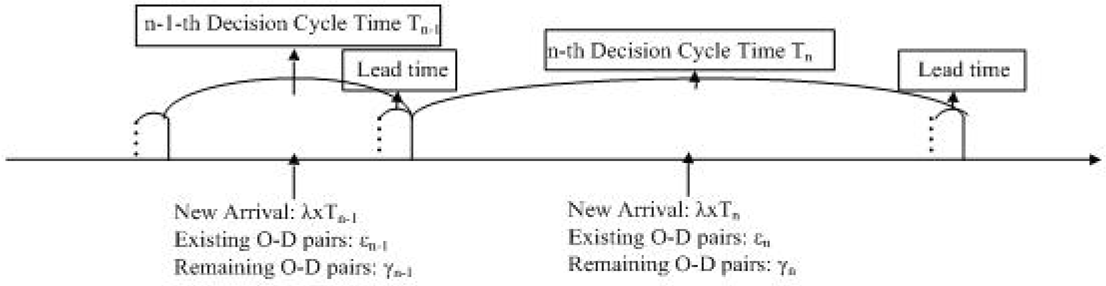

Figure 6-1 Time Diagram of the Real-Time Cluster Construction

Since the transmission holding time is modeled as an exponential distribution with mean value μ, $\gamma_n = \left\lfloor \varepsilon_{n-1} \cdot e^{-\frac{T_{n-1}}{u}} \right\rfloor$ represents the number of remaining O-D pairs at the n-th decision cycle, where $\lfloor \ \rfloor$ is a floor function, T_{n-1} is the decision cycle time, ε_{n-1} is the number of O-D pairs, and $e^{-\frac{T_{n-1}}{u}}$ is the probability *Prob{transmission holding time>T_{n-1}}*. By the way, if the capacity of the cluster topology is insufficient for serving the aggregate traffic load of all O-D pairs, some O-D pairs may be drop and generate new requests at the next decision cycle.

6.1 Modified Problem Formulation

6.1.1 Problem Description

Based on the formulation proposed in Section 2.3, before the decision cycle is expired, we solve the Lagrangean dual problem proposed in Section 3.2 and adopt the getting primal feasible procedure proposed in Chapter 4 in a predefined lead time interval $t_1 + t_2 + t_3$. Actually, the length of each decision cycle is different and depends on the minimum link duration of the constructed cluster topology and the routing assignment. After information collecting, decision making and disseminating, each wireless device executes the new decision and starts a new decision cycle.

However, different decision cycles are not independent of each other. Some decision variable and information could be reused at the next decision cycle. For example, if the duration of some cluster is still stable and satisfied with some criteria at the next decision cycle, the clusterhead/cluster member relationship could be reserved. Because the mobility of wireless devices causes the links fragile and change frequently, for this kind of clusters, we still need to compute the intra-cluster routing assignment, the inter-cluster routing assignment and the minimum link duration of the next decision cycle.

The criterion, which we use to evaluate the quality of a retained cluster, is whether the capacity of the cluster is enough to serve the traffic demand of the O-D pairs of the cluster or not. As mentioned above, the traffic demand of an O-D pair is characterized by its transmission rate and transmission holding time. Hence, if the duration of a cluster is longer than the maximum holding time of all O-D pairs and the maximum available transmission rate of the clusterhead is enough to serve the aggregate traffic load of all O-D pairs, we can retain this cluster and its cluster/cluster member relationship at the next decision cycle. Although it takes some overheads to compute the retained clusterheads and cluster members in advance, we can reduce the problem size of the Lagrangean relaxation problem, slash its complexity and make the Lagrangean relaxation based algorithm more effective and efficient.

In this section, we modify the original formulation propose in Section 2.3 into a more general form to solve the real-time cluster construction and QoS constrained routing assignment iteratively. The problem description, problem assumption, objective function and constraints of the modified formulation are the same as those of the original one. The major difference in the modified formulation is the given parameters. There are two node sets, the set of the retained clusterheads and the set of the new candidate clusterheads. We discuss the modified formulation in detail later.

Table 6-1 Problem Description

Problem assumption:
The same as Table 2-1.
Given:
The same as Table 2-1.
Objective:
To maximize the minimum link duration of the cluster construction and the routing assignment
Subject to:
The same as Table 2-1.
To determine:
The same as Table 2-1.

6.1.2 Problem Notation

Table 6-2 Notations of Given Parameters

Notation	**Definition**
V_n	The set of nodes which would join the decision process of the clusterhead/cluster member relationship at the next decision cycle. The set is also the set of the candidate clusterheads.

V_r	The set of nodes whose clusterhead/cluster member relationship has been determined by the retained cluster of the last decision cycle.
V	The set of nodes which is the union of the set V_n and the set V_r.
T_{n-1}	The duration time of the n-1-th decision cycle.
λ	Poisson arrival rate of the O-D pairs
μ	Mean value of the exponential transmission holding time
W_{n-1}	The set of all O-D pairs at the n-1-th cycle.
W_n	The set of all O-D pairs at the n-th decision cycle.
d	The max hop count to construct a cluster which is the longest distance between a cluster member node and its clusterhead
Q_{uv}	The set of candidate paths from the node u to the node v
r	The transmission radius of each node
L	The set of links
P_w	The set of candidate paths for the O-D pair w, which will be included in Q_{uv}, $P_w \in Q_{uv}$
a_w	The traffic demand of the O-D pair w, which is evaluated by its traffic data rate per unit time (unit bits/sec)
C_n	Capacity of the node n, which is evaluated by its maximum transmission rate (unit: bits/sec)
H_w	Maximum hop count of each O-D pair
$\sigma_{(n,m)}$	1 if the link (n,m) is on the path p, and 0 otherwise. (n,m) defines that the node n and node m are the outgoing node and incident node of the link respectively.
$\delta_{p\ell}$	1 if the link ℓ is on the path p, and 0 otherwise.
$x_n(t)$	The x-axis coordinate of the node n at time t

$y_n(t)$	The y-axis coordinate of the node n at time t
$v_x(t)$	The x-axis velocity of the node n at time t
$v_y(t)$	The y-axis velocity of the node n at time t
t_{ij}	The link duration between node i and node j
t_1	Data Collection time
t_2	Computation time
t_3	Decision dissemination time
M_1	The big number used in the constraint (IP 1.11). The value is set as 2.
M_2	The big number used in the constraint (IP 1.15) and (IP 1.16). The value is set as the maximum link duration of the entire network at the decision instance.

Table 6-3 Notations of Decision Variables

Decision variables	
Notation	**Definition**
h_g	1 if the node g is elected to be a clusterhead and 0 otherwise.
b_{vg}	1 if node v belongs to the node g and 0 otherwise
z_{pvg}	I if the node v choices the path p connect to the clusterhead g and 0 otherwise
x_p	1 if the path $p \in P_w$ is used for transmitting the packet of the O-D pair w and 0 otherwise
T_n	The minimum link duration of the constructed cluster topology and routing assignment at the n-th decision cycle

6.1.3 Problem Formulation

Optimization problem:

Objective function:

$$Z_{IP1} = \max T_n \qquad \text{(IP2)}$$

Part I: Cluster Construction Constraints

$$\sum_{g \in V} b_{vg} = 1 \qquad \forall v \in V_n \qquad \text{(IP 2.1)}$$

$$b_{gg} = h_g \qquad \forall g \in V \qquad \text{(IP 2.2)}$$

$$b_{vg} \le h_g \qquad \forall v, g \in V \qquad \text{(IP 2.3)}$$

$$b_{vg} \le \sum_{p \in P_{vg}} z_{pvg} \qquad \forall v, g \in V \qquad \text{(IP 2.4)}$$

$$h_g = 0 \text{ or } 1 \qquad \forall g \in V_n \qquad \text{(IP 2.5)}$$

$$b_{vg} = 0 \text{ or } 1 \qquad \forall v \in V_n, g \in V \qquad \text{(IP 2.6)}$$

$$\sum_{\ell \in L} \sum_{p \in P_{vg}} z_{pvg} \cdot \delta_{p\ell} \le d \qquad \forall v, g \in V \qquad \text{(IP 2.7)}$$

$$z_{pvg} = 0 \text{ or } 1 \qquad \forall p \in P_{uv} \ u, v \in V \qquad \text{(IP 2.8)}$$

Part II: Inter-cluster Routing Constraint

$$\sum_{p \in P_w} x_p \le 1 \qquad \forall w \in W_n \qquad \text{(IP 2.9)}$$

$$\sum_{\ell \in L} \sum_{p \in P_w} x_p \cdot \delta_{p\ell} \le H_w \qquad \forall W \in W_n \qquad \text{(IP 2.10)}$$

$$\sum_{p \in P_w} x_p \cdot \delta_{pv} \le \frac{\left\{ \sum_{p \in P_w} x_p \cdot \delta_{pg} \right\} + b_{vg} + M_1 (1 - b_{vg})}{2} \qquad \forall w \in W_n \ v, g \in V \qquad \text{(IP 2.11)}$$

$$x_p = 0 \text{ or } 1 \qquad \forall w \in W_n \ p \in P_w \qquad \text{(IP 2.12)}$$

$$\sum_{(i,j) \in L, i=n} \sum_{w \in W} \sum_{p \in P_w} a_w \cdot x_p \cdot \sigma_{p(i,j)} \le C_n \qquad \forall n \in V \qquad \text{(IP 2.13)}$$

$$t_{ij} = \frac{-(ab+cd)+\sqrt{(a^2+c^2)r^2-(ad-bc)^2}}{a^2+c^2} \quad \forall (i,j) \in L \qquad \text{(IP2.14)}$$

$$a = \upsilon_{ix}(t)-\upsilon_{jx}(t) \text{ , } c = \upsilon_{iy}(t)-\upsilon_{jy}(t) \qquad b = x_i(t)-x_j(t) \text{ , } \mathrm{d} = y_i(t)-y_j(t)$$

Part III: Minimum Link Duration Constraints

$$T \le (\sum_{p\in P_{vg}} z_{pvg}\cdot\delta_{p\ell})\cdot t_\ell + M_2(1-\sum_{p\in P_{vg}} z_{pvg}\cdot\delta_{p\ell}) \quad \forall v,g \in V \;\; \ell \in L \qquad \text{(IP 2.15)}$$

$$T_n \le (\sum_{p\in P_w} x_p\cdot\delta_{p\ell})\cdot t_\ell + M_2(1-\sum_{p\in P_w} x_p\cdot\delta_{p\ell}) \quad \forall w \in W_n \;\; \ell \in L \qquad \text{(IP 2.16)}$$

$$M_2 \ge T_n \ge t_1+t_2+t_3\,. \qquad \text{(IP 2.17)}$$

$$|W_n| = |W_{n-1}|\cdot e^{-\frac{T_{n-1}}{\mu}} + \lambda \times T_{n-1} \qquad \text{(IP 2.18)}$$

Objective function explanation:

The objective function (IP 2) is the same as that of (IP 1). The purpose is to **maximize the minimum link duration or node duration of the constructed cluster topology.** The minimum link duration of the constructed cluster topology and the routing assignment is evaluated by constraints (IP 2.15) and (IP 2.16). Thus, the problem is a Max-Min optimization formation.

Constraints explanation:

Constraints (IP 2.1) ~ (IP 2.17): The explanation of the constraints (IP 2.1) ~ (IP 2.17) is the same as that of the constraints (IP 1.1) ~ (IP 1.17).

Constraint (IP 2.18): Constraint (IP 2.18) represents the new arrival O-D pairs at the n-1-th decision cycle. As mentioned above, $|W_{n-1}|$ and $|W_n|$ is the cardinalities of the number of O-D pairs at the n-1-th and the n-th decision cycle respectively. Thus, $e^{-\frac{T_{n-1}}{\mu}}$ is the probability *Prob{transmission holding time>T_{n-1}}*, and $\lambda \times T_{n-1}$ is the number of new O-D pairs arrived at the n-1-th decision cycle.

6.2 Real-time Lagrangean based algorithm

Again, by using the Lagrangean relaxation (LR) method, we can transform the primal problem (IP 2) into the following Lagrangean relaxation problem (LR 2) where constraints (IP 2.2), (IP 2.3), (IP 2.11), (IP 2.13), (IP 2.15), and (IP 2.16) are relaxed. The structure and subproblems of the Lagrangean relaxation problem (LR 2) are similar to those of the Lagrangean relaxation problem (LR) in Section 3.2, whereas the problem size of Subproblem 1 and Subproblem 2 is modified as $|V_n|$, and the problem size of Subproblem 3 is modified as $|W_n|$.

To get a primal feasible solution to the modified formulation (IP 2), we adopt the procedure proposed in Chapter 4. In a real-time scenario, because we adjust the cluster topology and routing assignment base on that of the last decision cycle, some information, such as multipliers, could be reuse at the next decision cycle. Because routing paths, the link state and duration may change due to the mobility of wireless devices, the multipliers which are related to links, such as constraints (IP 2.11), (IP 2.13), (IP 2.15) and (IP 2.16), would not be suitable for initialization. Since the duration the routing path is sensitive to the link duration, we can initialize the multiplier of each link as an infinitesimal value. However, the multipliers of the retained clusterheads and cluster members, such as the set V_r and the corresponding constraints (IP 2.2) and (IP 2.3), can be the initial value of the n-th decision cycle. By appropriately assigning the initial value and adjusting the step scalar, we can reduce the complexity, computation time, and make the real-time based Lagrangean based algorithm more effectively and efficiently. The procedure of the real-time LR based algorithm is illustrated in Figure 6-2 The Procedure of the Real-Time LR based RCC algorithm.

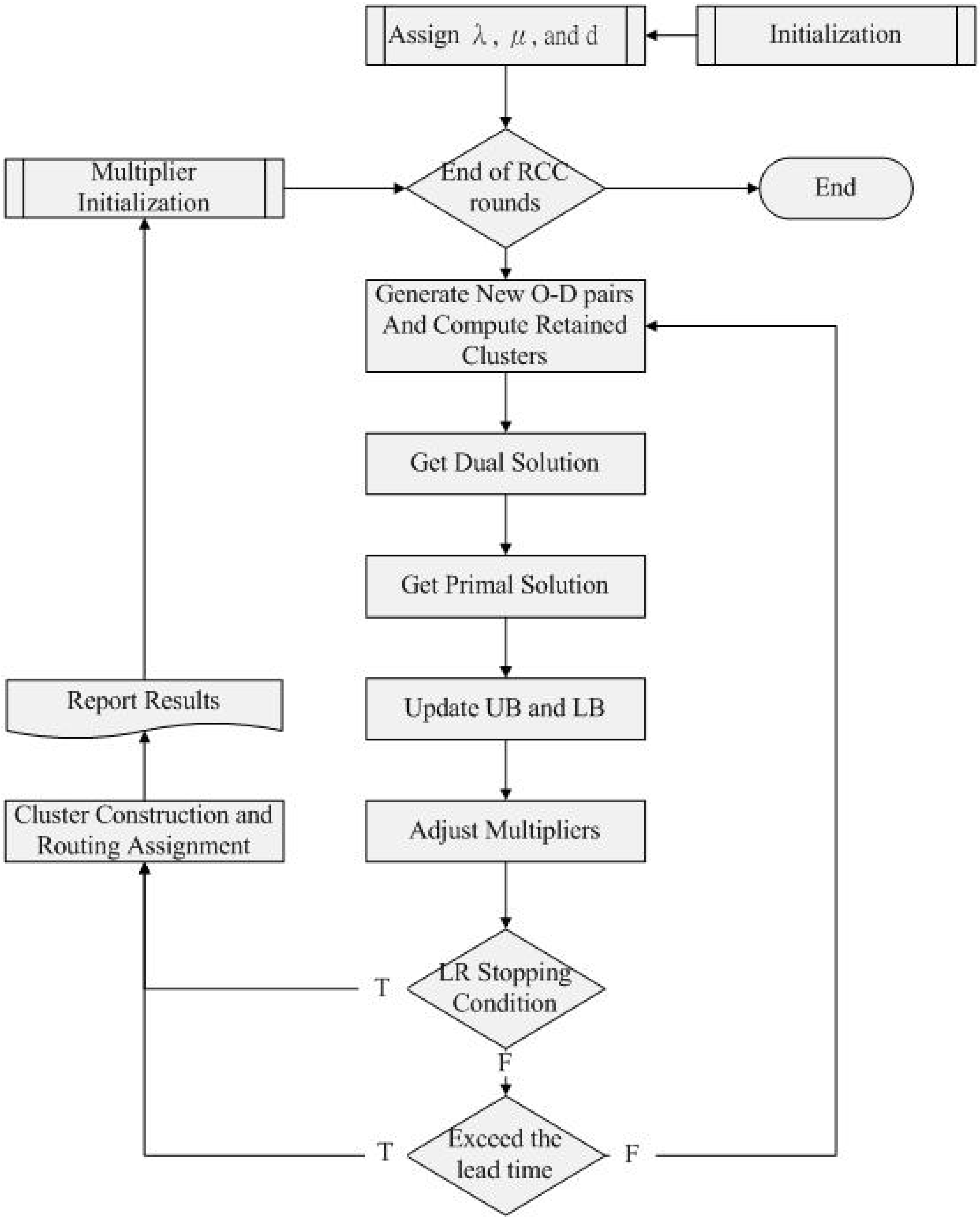

Figure 6-2 The Procedure of the Real-Time LR based RCC algorithm

The detail of the real-time LR based algorithm is represented as follow:

Step 1: (General Initialization)

1. Generate a random connected network topology.

Step 2: (Initialization of each round)

1. Generate new O-D pairs and the traffic demand of each O-D pair.

2. Update the node position and link duration at the decision instance.
3. If it is not the first round, compute the retained O-D pairs, clusterheads and cluster members.
4. If it is not the first round, initialize all multipliers based on those of the last round. Otherwise, set all multipliers as an infinitesimal value, Epsilon.
5. Initialize upper bound (UB), and iteration count as 0 and delta factor as 2.

Step 3: (LR Termination Criteria)

1. The difference between upper bound (UB) and lower bound (LB) is less than Epsilon 10^{-4}. (Convergence case 1)
2. The number of iterations exceeds maximum iteration count.
3. Step size is less than 10^{-7}. (Convergence case 2)
4. If the computation time exceeds the lead time, terminate the LR procedure and report the current best solution.

Step 4: (Calculating Lower Bound)

1. With the given Lagrangean multipliers per iteration, we optimally solve the subproblems to get the value Z_d.

Step 5: (Getting Primal Feasible Solution)

1. Apply the heuristic procedure proposed in Chapter 4 to calculate the value Z_{IP}.

Step 6: (Updating Lower Bound, Upper Bound and Lagrangean Multipliers)

1. If $Z_d > LB$, set $LB = Z_d$.
2. If $Z_{IP} < UB$, set $UB = Z_{IP}$.
3. Calculate the step size and update Lagrangean multipliers by using the subgradient method as described in section 3.3.

4. Increase the iteration count *i* and go to Step 3 if no mating with the termination criteria.

Step 7: (Reliable Cluster Construction)

1. According to the decision variables of UB, construct the cluster topology and routing assignment.
2. If reach the maximum number of simulation rounds, terminate the simulation. Otherwise, go to Step 2 and restart the next round.

6.3 Performance Metrics

To evaluate the solution quality of the real-time LR based algorithm, we consider five performance metrics, namely, minimum link duration, blocking ratio, packet delivery rate, average number of iteration in a predefined lead time, and average error gap. We describe these metrics in detail as follows.

- **Minimum link duration** The objective of the real-time RCC mechanism is to maximize the minimum link duration of the constructed cluster topology and routing assignment. The simulation can design by three aspects, network size, mobility level, and the number of O-D pairs to calculate the minimum link duration.
- **Blocking ratio** Because of the limited nodal capacity, it may be infeasible to admit all traffic demands including the retained and the new O-D pairs. If the aggregated transmission rate of a clusterhead exceeds its maximum available transmission rate, some O-D pairs would be blocked and retransmit at the next decision cycle. Note that the retained O-D pairs have higher priority to the new O-D pairs. The blocking ratio is calculated by the admitted number of O-D pairs/the number of all O-D pairs.

- **Packet delivery rate** Since each node moves during the decision cycle, the link and routing path may fail and breakdown. Because we describe the traffic demand of an O-D pair by its transmission holding instead of the number of transmission packets, we compute the packet delivery rate by the transmission interval/the required transmission time. The transmission interval begins at the start point of the decision cycle and ends on the failure of the routing path.
- **Average number of iteration in a predefined lead time** By increasing the computation time, the LR based algorithm can execute more iterations to improve its solution quality. The complexity increases as the problem size grows. Hence, this performance metric is a simple indicator to evaluate the effectiveness of our proposed algorithm.
- **Average Error Gap** This performance metric is calculated by (UB-LB)*100/LB (%) and illustrates the optimality of the solution. The smaller gap indicates the better solution quality.

Chapter 7 Summary and Conclusion

7.1 Summary

In this thesis, we consider the problem of reliable cluster construction and QoS constrained routing assignment. To evaluate the solution quality we calculate the minimum link duration by adopting the formula of ODMRP **[7]** and the Gauss-Markov mobility model **[6]**.

We model the problem as a mathematical optimization formulation, where the objective function is to maximize the minimum link duration of the constructed cluster topology and routing assignment. For a given wireless network, we jointly determine the clusterhead/cluster member relationship, intra-cluster routing assignment and inter-cluster routing assignment. Then we use Lagrangean relaxation and the subgradient method to solve the problem. While applying this methodology, we relax some complicated constraints in our objective function combined with corresponding Lagrangean multipliers and divide the original problem into four subproblems that are easier to solve. We analyze these subproblems, solve them optimally and adopt our proposed heuristic to obtain a primal feasible solution.

We implement the algorithms in C++ code, and test them on networks generated by fixed random seeds. Our experiment results indicate a significant improvement over other algorithms because, by applying our algorithms, the minimum duration of the used link is maximized. Note that by ensuring that the aggregated traffic load of an intermediate node does not exceed its maximum available transmission rate and by rerouting congested paths, we can achieve load balance and improve the utilization and stability of the constructed network.

7.2 Conclusion

Our contribution in this thesis is that we model the problem of reliable cluster construction and QoS constrained routing assignment as a mathematical formulation and propose an efficient algorithm to solve it. Our algorithm is easily implemented, because the most complex parts of our subproblems are Bellman-Ford shortest path problems with hop-count constraints. By adopting Lagrangean relaxation method, we can consider more QoS constraints, adjust our algorithm to a more generalized situation without a major modification of our proposed formulation and structure, and ensure that the routing assignment is close to the real-world wireless environment.

Since we adjust the cluster topology and reroute the congested paths by using the multipliers of the nodal capacity and end-to-end delay constraints as link cost functions, we can allocate and schedule the traffic load of O-D pairs on the entire network efficiently. The multipliers adjusted according to the decision variables and corresponding relaxed constraints per iteration also help our proposed algorithm get a good primal feasible solution and converge to an optimal solution. Compared with other algorithms, our algorithm can obtain a significantly better solution than others in heavily-loaded and large scale networks.

7.3 Future work

In this paper, we only consider end-to-end delay and nodal capacity as our QoS constraints. However, there are still many aspects for QoS constraints, such as packet delivery rate and bandwidth allocation. We can generalize our mathematical formulation by including other QoS constraints without a major modification of the proposed algorithm.

Another, there are different criteria for the reliable cluster construction and routing assignment, such as communication and processing overheads, the number of clusterheads, cluster size, and energy consumption of the intermediate nodes on the routing path. Although we just consider the minimum link duration as our objective function, one can try to design a new formulation with combinatorial objective function.

Finally, to be close to the real-world environment, we should consider many natural and physical phenomena. Thus, the link cost function of the routing path not only includes the cost of end-to-end delay and nodal capacity, but also the cost of the controlling overheads, the transmission collision, and the transmission fading. Other network management issue, such as channel allocation, should also be taken into consideration. The model and formulation will become more difficult and complex, but also more interesting and practical.

For the real-time reliable cluster construction (RCC) mechanism, although we do not implement the real-time Lagrangean relaxation (LR) based algorithm, we still modified the original model and formulation into a more real-time based formation.

Reference

[1] B. An and S. Papavassiliou, "A mobility-based clustering approach to support mobility management and multicast routing in mobile ad-hoc networks", International Journal of Network Management 200111 page 387-395.

[2] B. An and S. Papavassiliou, "MHMR: mobility-based hybrid multicast routing protocol un mobile ad hoc wireless network", Wireless Communication and Mobile Computing 2003 3 page 255-270.

[3] B. An and S. Papavassiliou, "Geomulticast: architecture and protocols for mobile ad hoc wireless networks", http://www.ComputerScienceWeb.com received 17 October 2002.

[4] A. D. Amis, R. Prakash, T. H. P. Vuong, and D. T. Huynh, "Max-min D-cluster formation in wireless ad hoc networks", Inforcom 2000.

[5] J.J. Garcai-Luna-Aceves and E. L. Madruga, " A multicast routing protocol for ad hoc networks", Processing of Inforcom99 March 99 page 784-792.

[6] T. Camp, J. Boleng, and V. Davies, "A survey of mobility models for ad hoc network research", Wireless Communication and Mobile Computing 2002 2 page 483-502 .

[7] S. J. Lee, W. Su, and M. Gerla, "ODMRP On-demand multicast Routing Protocol in Multihop Wireless Mobile Networks", Mobile Network and Application Dec 2002 7 6 ABI/INFORM Global.

[8] Das, A. K. Marks, R. J. El-Sharkawi, M. Arabshahi, and P. Gray, "Minimum power broadcast trees for wireless networks: Integer programming formulations", INFOCOM 2003, 30 March-3 April 2003, p1001- 1010 vol.2.

[9] E. L. JOHNSON, A. MEHROTRA, and G. L. NEMHAUSER, "Cliques and clustering: A combinatorial approach", Mathematical Programming, 62,

133-151.

[10] Alan, D. Amis, and R. Prakash, “Load-balancing clusters in wireless ad hoc networks”, Proceedings of the 3rd IEEE Symposium on Application-Specific Systems and Software Engineering Technology 2000, P.25.

[11] Jamal N. ,Al-Karaki Ahmed E. ,and K. Raza, “On the optimal clustering in mobile ad hoc networks”, Proceedings of IEEE Consumer Communications and Networking, Las Vegas, Nevada USA/ January 5-8, 2004.

[12] JEFFREY E. , WIESELTHIER, and GAM D. NGUYEN, “Algorithms for energy-efficient multicasting in static ad hoc network”, Mobile Networks and Applications 6,251-263,2001.

[13] W. Lou, Student Member, IEEE, and J. Wu Senior Member, “On reducing broadcasting redundancy in ad hoc network”, IEEE transactions on mobile computing April-June 2002.

[14] J. Aslam, Q. Li*, and D. Rus, “Three power-aware routing algorithms for sensor networks”, WIRELESS COMMUNICATIONS AND MOBILE COMPUTING Wirel. Commun. Mob. Comput. 2003; **3**:187–208 (DOI: 10.1002/wcm.111).

[15] X.-Y. Li, ‘Algorithmic, geometric and graphic issues in wireless networks”, WIRELESS COMMUNICATIONS AND MOBILE COMPUTING *Wirel. Commun. Mob. Comput.* 2003; **3**:119–140 (DOI: 10.1002/wcm.107).

[16] Y. X. FENG, L. L. ZHAO, G. X. Wang, “A Clustering Algorithm Applied to Network Management in MANETS”, Info-tech and Info-net, 2001. Proceedings. ICII 2001-Beijing. 2001 International Conferences on Publication Date: 2001 Volume: 2, p. 626-631 vol.2.

[17] K. Liu, J. D. Li, “Mobile cluster protocol in wireless ad hoc networks”, Proc. Intl. Conf. on Commun. Tech. (WCC - ICCT 2000), Vol. 1, 2000.

[18] C. Y. Chiu, G. H. Chen, and Eric H. K. Wu, “A stability aware cluster routing protocol for mobile ad hoc networks”, Wireless Communications and Mobile Computing Volume 3, Issue 4 , Pages 503 – 515.

[19] S. Harada and H. Higaki (Japan), “Sporadic communication protocol between clusters of mobile computers”, wireless networks and engineering technologies WNET 2004 Track - Wireless Technologies II 424-050.

[20] M. L. Fisher, “The Largragian relaxation method for solving integer programming problems’, Management Science Vol. 27. N0. 1. January 1981 Printed in U.S.A.

簡　　歷

姓　名：林明源

出生地：台灣省台北市

生　日：民國七十年八月二十五日

學　歷：民國八十八年九月至九十二年六月

國立台灣大學資訊管理系學士

民國九十二年九月至九十三年七月

國立台灣大學資訊管理研究所碩士