國立勤益科技大學
電子工程系

碩士論文

一個新的跨層次IEEE802.15.4-Enabled路由協定使用離散調配拓樸控制演算法

A New Cross-Layer IEEE 802.15.4-Enabled Routing Protocol Using Discrete Distributed Topology-Control Algorithm

Department of Electronic Engineering, National Chin-Yi University of Technology

研究生　　：劉宏偉
指導教授：林宗宏　　博士

中華民國九十九年七月

一個新的跨層次IEEE802.15.4-Enabled路由協定使用離散調配拓樸控制演算法

A New Cross-Layer IEEE802.15.4-Enabled Routing Protocol Using Discrete Distributed Topology-Control Algorithm

研究生　：　劉宏偉
指導教授　：　林宗宏　博士

國立勤益科技大學

電子工程系

碩士論文

A Thesis Submitted to
Department of Electronic Engineering
National Chin-Yi University of Technology
in Partial Fulfillment of the Requirements
for the Degree of
Master of Engineering
in
Department of Electronic Engineering

July 2010
Taichung, Taiwan, Republic of China

中華民國九十九年七月

國立勤益科技大學

研究所碩士班

論文口試委員會審定書

本校　電子工程系　碩士班　劉宏偉　君

所提論文 A new cross-layer IEEE802.15.4-Enabled routing protocol using Discrete Distributed Topology-control Algorithm.

合於碩士資格水準，業經本委員會評審認可。

論文口試委員會：

召集人：________________

委員：________________

____

____

指導教授：________________

所長：________________

一個新的跨層次IEEE802.15.4-Enabled路由協定使用離散調配拓樸控制演算法

學生：劉宏偉　　　　　　　　　　指導教授：林宗宏　博士

國立勤益科技大學電子工程系

摘 要

我們提出一個採用離散調配拓樸控制演算法的新的跨層次 QoS 路由協定來改善 IEEE 802.15.4 為基礎網路的效能。在我們的方法中，由 Chen 和 Lin 提出的網狀叢集樹(cluster-mesh-tree, CMT)架構被我們採用離散調配拓樸控制演算法(DDTCA)來最佳化。並且，CMT 網路是基於星形網狀叢集(star-mesh-cluster, SMC)所形成，這將造成網路存在為數最多的 cluster 數量，並增加 beacon 訊框碰撞的次數與機率，也使得 CMT 路由路徑能減少的長度有限。為了解決這些問題，我們提出結合 CMT 與 DDTCA 的 "離散調配拓樸控制的 CMT 網路協定" (discrete topology-control CMT algorithm, DCMT)以及更有路由效率的 Inter-cluster-Mesh 來大大縮短路由路徑長度透過我們提出的 "更一般化的時槽租借" (Generalized Time-Slot Leasing, GTSL)方法來實現。而我們的模擬結果也驗證了 DCMT 協定確實擁有高於 CMT 網路的更佳效能。

關鍵詞：IEEE 802.15.4, 網狀叢集樹, 拓樸控制, 離散調配拓樸控制演算法, GTSL

A New Cross-Layer IEEE802.15.4-Enabled Routing Protocol Using Discrete Distributed Topology-Control Algorithm

Student： Hung-Wei Liu　　　　Advisor：Dr. Tsung-Hung Lin

Department of Electronic Engineering
National Chin-Yi University of Technology

Abstract

We propose a new cross-layer QoS routing protocol using discrete distributed topology-control algorithm to improve IEEE 802.15.4 enabled network performance. In our approach, the cluster-mesh-tree (CMT) proposed by Chen and Lin [2] is optimized using a discrete distributed topology-control algorithm (DDTCA). The CMT network is based on star-mesh-clusters, which have the most clusters, increasing the beacon frame collisions and reducing the routing path. We propose an approach for solving these problems called the "discrete topology-control CMT algorithm (DCMT)", based on the DDTCA algorithm and CMT network. This approach provides a more efficient inter-cluster-mesh link to significantly reduce the DCMT routing path length with our "generalized time-slot leasing approach" (GTSL). We also use simulation results to show that DCMT has better performance than the CMT network.

Keywords：IEEE 802.15.4, Cluster-Mesh-Tree, Topology-control, DDTCA, GTSL

誌　謝

感謝兩年來林教授宗宏的指導，在兩年的研究所生活中，林教授不只是學術研究的指導老師，更是生活上的指導教授！在做研究過程中，不論課業、生活或者是經濟方面皆受到教授諸多的照顧與指引，讓兩年的研究生活確實使我感覺到受益非凡，並且學習到做研究與學習生活其實是很相似的，在許多解決問題的過程觀念上都是相通的，讓我得以重新定義學習，對於待人處世以及解決生活問題有了新的思考方向與準則來實踐。

當然，在學術與專業上老師的幫助與意見更給予我最直接受用的幫助,我學習到很多在大學時代學習時未曾碰觸的學習議題,也體會到 「創新」 二字分別在實務與學術上所體現的不同意義，而最重要的一點是學習從看 paper 的過程中去做更多新的思考與嘗試,在研究過程中不斷發掘別人提出的新觀念與方法，從中吸收並且不忘質疑，甚而改變提出修正，藉由與老師討論的過程激盪新的想法與論述,更重要是對於思考不再天馬行空，而是找對方向以嚴謹的態度去思考可行性與實現,從而會在過程中不斷發現新問題並解決且精進，當最後可以對論文下筆時才驚覺這此中的過程想法已改變到與最初的出發點有 180 度以上的翻轉與改變，並非是拋棄了問題的本質，而是更認知問題的精神與其中的盲點，再藉由過程去挑戰與思考，形成研究必需的過程。

承蒙教授指導更感謝家人在經濟生活上的支持,特別是媽媽的辛勞，讓我得以更專心於完成學業，過程中也感謝遇到的同學與學弟妹還有特別看重我的林基源教授，讓我研究生活並非一無所獲，且自信自己得到的比其他碩士朋友要多!故感謝指導教授與所有幫助我的人，使我得以在兩年中快速蛻變，在此謹表達我微渺由衷的致謝之意，謝謝大家的照顧!

Contents

Lists of Figure

Lists of Table

Chapter 1. Introduction

A low-rate wireless personal area network (LR-WPAN) was described as a network designed for low-cost, very low-power short-range wireless communication in [2] and [8]. IEEE recently approved the 802.15.4 standard for the medium access control sub-layer (MAC) and the physical layer (PHY) for LR-WPANs. IEEE Std. 802.15.4 [8] can be used in a wide variety of low data rate applications, including health monitoring, remote controllers, toys, industrials controls and home automation. IEEE Std. 802.15.4 is also suitably applied to construct wireless sensor networks (WSNs) [4] including many small devices distributed over an area of interest where some specific tasks must be monitored. Each device has one or more sensors, embedded processors and low-power radios, and is normally battery operated.

IEEE Std. 802.15.4 defines the PHY layer and the MAC sub-layer for LR-WPANs. In IEEE Std. 802.15.4, the PHY layer is in charge of providing radio transceiver, energy detection, link quality, clear channel assessment, channel selection control and the transmission and reception of message packets through the physical medium [2]. The MAC sub-layer which sits on top of the PHY layer is an important technique that enables the successful network operation. It provides many essential functions for constructing a network such as channel scanning, collision avoidance, admission control, bandwidth reservation, and synchronization control.

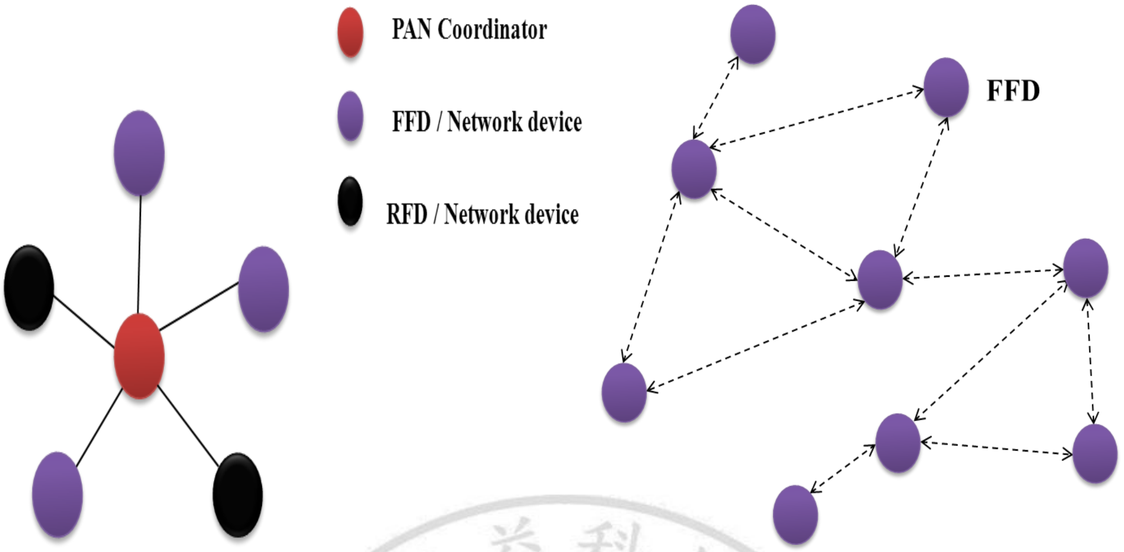
IEEE Std. 802.15.4 pertains to battery-powered wireless sensor networks. Many researches were proposed for prolonging WSN network lifetime and reducing energy consumption in each network device. Providing transmission service quality (QoS)

[2][3][4][8] has also become a primary consideration in wireless sensor networks in recent years.

The IEEE Std. 802.15.4-enabled network is comprised of two types of devices: the full-function device (FFD), and the reduced-function device (RFD) [1][2][8]. FFDs contain the complete set of the MAC services that allow it to act as a network coordinator or network device. RFDs that only contain a reduced set of MAC services can only deliver data though FFDs and cannot act as a coordinator or router.

The IEEE MAC Std. 802.15.4 provides two types of basic topologies: the star topology and the peer-to-peer topology [2][8][18], as shown in Figure 1-1. In a star topology communication is controlled by a unique PAN coordinator that operates as a network master, sending beacons for device synchronization and maintaining association management. The devices in a star topology can only communicate with the PAN coordinator. On the other hand the peer-to-peer network consists of most FFDs and allows a FFD to communicate with other FFDs within its range and have packets relayed to FFDs outside its range via a multi-hop routing path [1][11].

The cluster-tree (CT) [2][4][8] network is an essential peer-to-peer network case. It is rooted at the PAN coordinator based on the MAC "parent-child relationship" between IEEE Std. 802.15.4 devices. It divides the network into several small clusters, which have only one coordinator called a "cluster head (CH) [2][8]" to coordinate all of the devices inside its range for common tasks. In addition, the cluster-tree network employs the hierarchical address scheme [1][8][18] to form a hierarchical tree that can provide better scalability for a larger network. The typical cluster-tree network is as shown in Fig.1-2.



(a) Star topology (b) peer-to-peer topology

Fig. 1-1.Two types of basic topologies of IEEE Std. 802.15.4

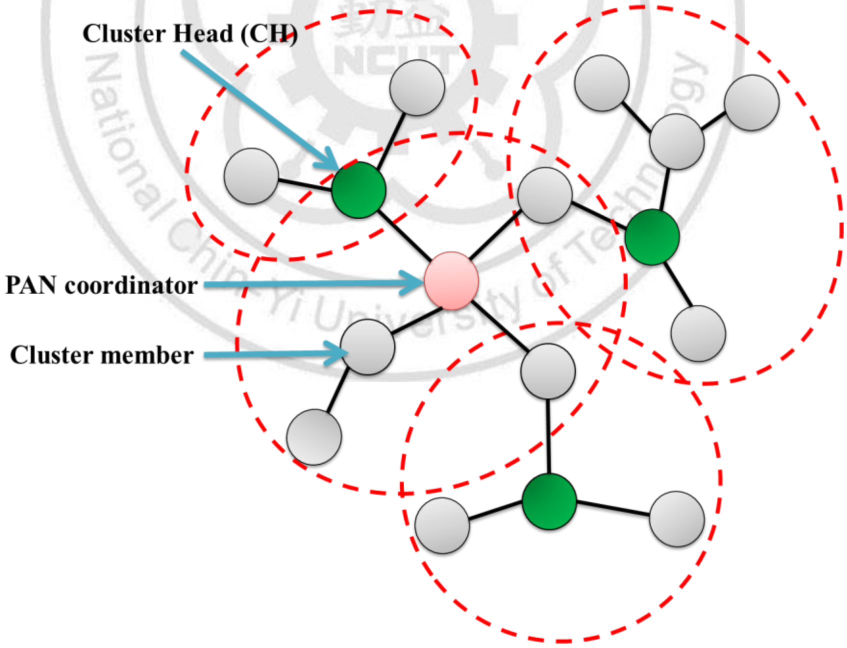


Fig. 1-2.Cluster-tree network

The cluster-tree network is suitable for wireless sensor networks, although it is not specified in detail in IEEE Std. 802.15.4, it is still adopted as a formal topology by the ZigBee [1][10][17][18] standard, which is a wireless technology built upon IEEE Std. 802.15.4. The entire cluster-tree network very easily breaks down when a only few nodes break down [2][4][5]. To our best knowledge, a cluster-tree network will lead to the “transmission holding [3][16]” problem. This is because the cluster-tree network uses the “master-slave [3][9]” mode through a parent-child relationship. The cluster-tree only allows any child node to communicate with its parent node. Therefore, if a child node wants to deliver data to another child node belonging to the same parent node, it must transmit that data to its parent node to relay the data down to the destination node. The order for this transmission is “child-parent-child [2]” or “slave-master-slave [3]”.

The slave-master-slave communication mode produces a heavier workload on the “master” than the “slaves”. This excessive workload decreases the master node lifetime and generates non-uniform energy consumption in the network. It will reduce the network lifetime. On the other hand, the CH of each cluster has to coordinate all devices within its range for common tasks and collect data from these devices to relay to other clusters.

The transmission holding problem is the tree routing bottleneck in the cluster-tree network because of the workload imbalance of the master device, such as CHs, PAN coordinator. Many methods have been proposed to solve such an energy-inefficiency problem.

A clustering protocol, called LEACH [14], is an adaptive clustering routing protocol that elects a CH from a cluster of homogeneous nodes to act as the router for the cluster for a given time interval. A new CH will be elected to replace the original CH after the interval. This protocol uses a mechanism which periodically elects a new

CH that has a higher remaining energy level than the other devices in its cluster to spread the workload of each CH to the other devices. It assumes that all CHs can directly reach a PAN coordinator, preventing the network from expanding, and hence, it cannot cover large regions.

The tree routing protocol cannot avoid the single points of failure (SPOFs) [5] problem because it does not offer a backup path mechanism for routing tree failure discovery. This results from the cluster-tree network allowing only child-parent-child communications and prohibiting direct child-to-child communications.

The SPOF problem decreases both the reliability and connectivity of a cluster-tree network, and it reduces the network lifetime. Chen and Lin [2] designed a routing redundancy protocol called the cluster-mesh-tree (CMT) routing protocol to alleviate the impact from SPOFs and the transmission holding problem.

CMT utilizes an additional mesh-link that provides child-to-child communications through a “time-slot-leasing” (TSL) [16] mechanism that offers backup paths for cluster-tree networks. This method provides QoS to assure that any routing path has enough bandwidth to deliver data from sources to destinations. However, CMT adopts star-based mesh clusters (SMCs) without considering topology control based on energy-efficiency. F. Cuomo et al. [4] analyzed that the cluster-tree formation in IEEE Std. 802.15.4 should impose constraints on the topology by setting some important parameters to increase network performance. The parameters include the maximum depth of a cluster-tree and the maximum number of end-devices that a router may have as children, and so on.

The aforementioned perspectives are not discussed by the CMT routing protocol. To improve CMT performance, we adopted a new cluster-tree formation architecture based on the “discrete distributed topology-control algorithm (DDTCA)” [7] to replace the star-based mesh cluster formation. The proposed method will generate fewer clusters than the CMT because its cluster size is not fixed to 1-hop.

The DDTCA algorithm is based on energy-efficiently, designed to divide a network into several wireless sensor network clusters. We utilized DDTCA to form a new cluster-tree structure and replaced CMT formation. We propose a modified CMT protocol called the “discrete distributed cluster-mesh-tree (DCMT)” routing protocol. DCMT can provide a cluster-to-cluster path reservation mechanism [2] through the “Generalized Time-Slot Leasing (GTSL)” approach. This will reduce the routing path length further with an inter-cluster-mesh link between two sibling clusters.

The remainder of this thesis is organized as follows. Chapter 2 will describe TSL, CMT, DDTCA, and our basic idea in detail. Chapter 3 explains the DCMT routing protocol and the more generalized TSL analysis. Chapter 4 shows the simulated results to prove DCMT performance. Chapter 5 concludes this thesis.

Chapter 2. Related Works

In this chapter we describe the basic DCMT protocol idea and some basic concepts including time-slot-leasing, the cluster-mesh-tree routing protocol and discrete distributed topology-control algorithm. All of these concepts are basic components in the DCMT protocol.

Zhang et al. [16] proposed a mechanism, called the “time-slot leasing (TSL)” approach, to address the problems associated with the slave-master-slave Bluetooth WPAN model. The TSL approach makes slave-to-slave communication possible. Chen and Lin [2] incorporated it into a cluster-tree network based on IEEE Std. 802.15.4 to provide a backup routing mechanism through the child-to-child communication mode.

CMT formation consists of SMC clusters. SMC is a kind of 1-hop clustering formation, shown in Figure 2-1. It contains two types of links: the tree-link and the mesh-link [2] which is applied to provide child-to-child communications. However, the SMC is produced only in a network with the largest number of clusters. The 1-hop clustering scheme is adopted to reduce the time a slave node (or cluster member) leases the master node time-slot (or CH) for direct slave-to-slave communications. If there a length of more than one hop exists between a slave node and its master node, more time is necessary to set time-slot leasing operations. SMC may only offer the child-to-child mesh link within its range. It provides a backup link when a tree link cannot offer enough bandwidth to assure that the routing path corresponds with QoS. It is still possible for the CMT routing protocol to cause a very long routing path.

If there are too many clusters in a cluster-tree/cluster-mesh-tree network, a higher probability for beacon frame collisions will occur [10]. This will decrease network performance and increase energy consumption. To alleviate this effect, we modified the CMT protocol to generate a fewer number of clusters and further reduced the tree routing path length through a more generalized TSL mechanism (GTSL) which utilizes an inter-cluster-mesh link between two adjacent sibling clusters. We used DDTCA to construct an energy-efficiency cluster-tree network to form a set of clusters that have various cluster sizes. We called the new CMT architecture the “discrete distributed cluster-mesh-tree (DCMT)” routing protocol.

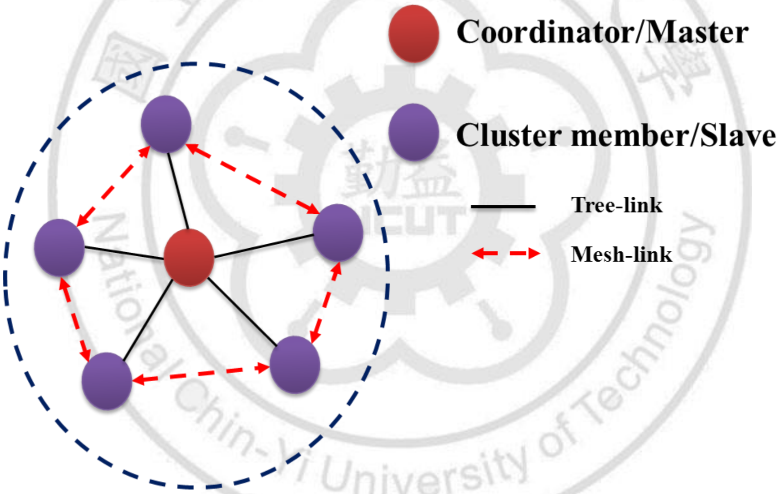


Fig. 2-1.Star-based mesh cluster

2.1 Time-Slot Leasing approach (TSL)

Both the Bluetooth and the IEEE 802.15.4-enabled WSN have a common drawback, or transmission holding problem. This means that the master node is a communications bottleneck for each slave-master-slave communication.

Zhang et al. [16] proposed a TSL approach that supports slave-to-slave communication in the Bluetooth piconet [9] network to solve the transmission holding problem. The TSL approach does not permanently change the basic piconet structure and has no negative effects on inter-piconet communications. As illustrated in Fig. 2-2, if slave node S_1 needs to deliver a lot of data to another slave node S_2, S_1 will request master M to lease time slots 1 and 2 for direct communication between slave S_1 and S_2. In the example in Fig. 2-2, QoS requests 2 time slots for each delivery. Time slots 3 and 4 for slaves S_1 and S_2 are busy, but time slot 3 and 4 for master M are free. This makes slave-master-slave delivery impossible. Because neither slave S_1 nor S_2 has enough common free time slots to deliver data through Master M, Slave S_1 or S_2 must enable TSL to lease time slots 1 and 2 from master M. The slaves can therefore use the two time slots to communicate with each other, as shown in Fig. 2-2.

Figures 2-3 show a TSL setup procedure. It contains 8 steps to set the TSL approach for direct communication between slaves S_1 and S_2. Each step requires a time slot.

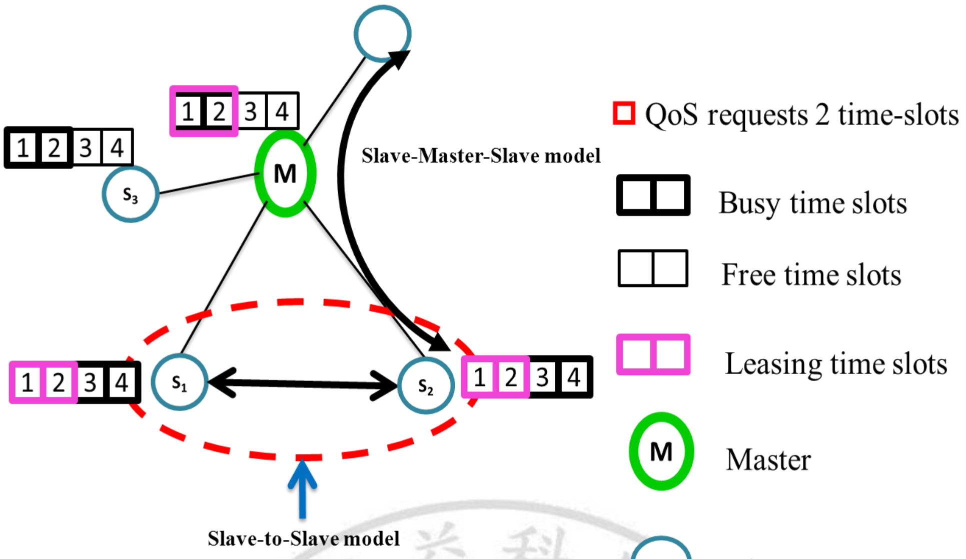


Fig. 2-2.A TSL approach example

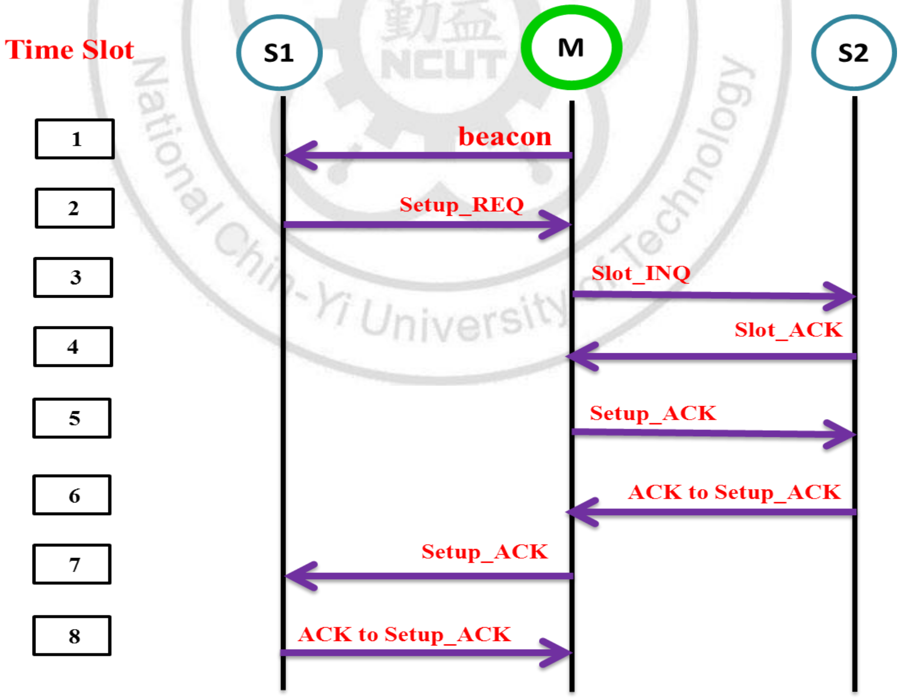


Fig. 2-3.TSL setup procedure

2.2 Cluster-Mesh-Tree Routing Protocol

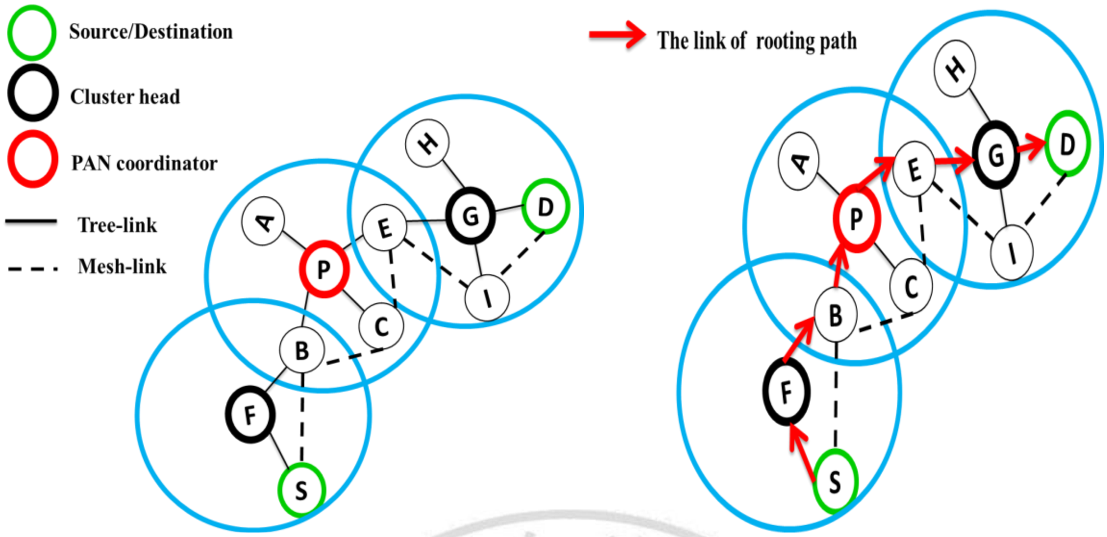
The Cluster-Mesh-Tree (CMT) [2] is also a kind of cluster-tree network. It has two types of links: the tree-link and mesh-link. CMT provides a backup routing mechanism which supports reconstruction of a failed routing path using tree-link and mesh-link collaboration. A cluster-tree network only permits construction of one path to transfer data. If some devices in this unique path are busy or failed, the data cannot be transmitted. This is because the “transmission holding” problem occurs when CH nodes carry too much load. Excessive load causes them to be busy or fail easily. The network energy-efficiency and lifetime of a CT network often decreases rapidly. The transmission holding problem is alleviated by a backup routing mechanism in CMT networks. This is because each child-parent-child communication can be changed into child-to-child temporarily by TSL, as shown in Fig. 2-2. Consequently, CMT is more efficient than CT. This is because CMT provides a backup routing path mechanism to alleviate the transmission holding problem, allowing data to be transferred by other backup paths to decrease delay time. This process increases network energy-efficiency, reduces power consumption and extends the lifetime of the entire network.

However, CMT formation is based on the SMC cluster, which belongs to the 1-hop clustering architecture. Many SMC clusters are formed in a CMT network. In a TDMA synchronized wireless sensor network, the more clusters it owns, the more beacon frame collisions occur [10].

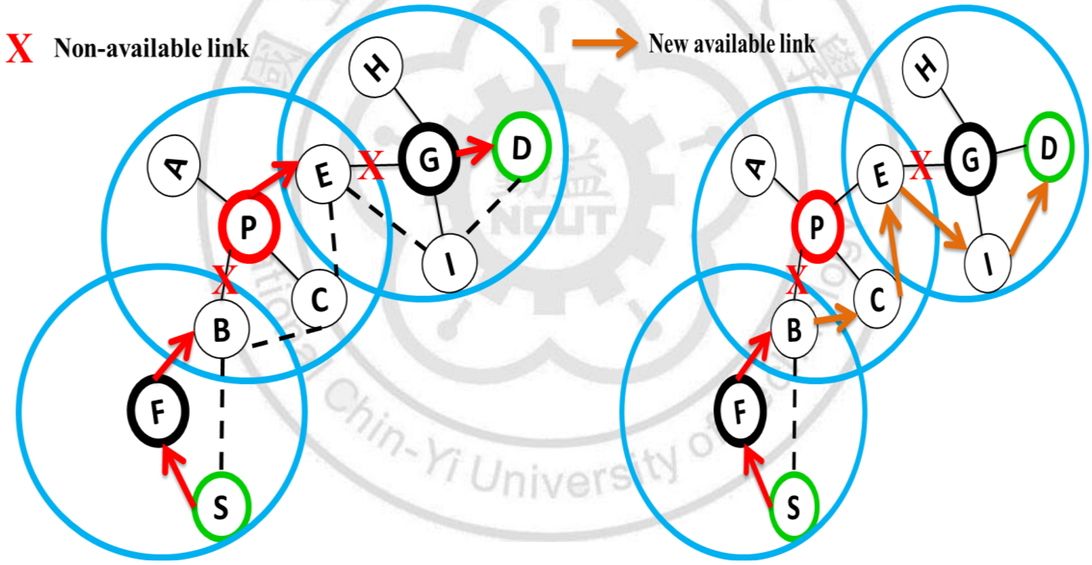
The 1-hop clustering architecture is adopted by CMT to allow cluster members requesting lease time slots from the cluster head more rapidly for direct communication with other cluster members. This feature is the same as the SMC cluster. However, the distance between the cluster member and cluster head is only 1 hop.

According to the above statements we can infer that CMT can only reduce the tree routing path length and this is associated with two factors: each SMC cluster can decrease any tree routing path by one hop length at most, and that it is still possible for the CMT routing path to be too long because unbalanced cluster-tree WSNs may produce a long routing path. Therefore, to improve the above drawbacks, we modified the CMT method to generate a network that has fewer clusters using DDTCA, reducing the tree routing path length further using the generalized TSL (GTSL) for inter-cluster-mesh reservation. We will discuss the GTSL model in detail in Chapter 3.

The CMT routing protocol procedure is shown in Fig 2-4. In this example, node S wants to deliver data to node D, and it sends a QoS_REQ [2] routing discovery packet to build a routing path in the CMT network. Figure 2-4(a) indicates this case has a transmission from node S to D and three SMC clusters. The only tree routing path: S->F->B->P->E-G->D is shown in Fig. 2-4(b). In Fig. 2-4(c), if link $\overrightarrow{BP}$ and $\overrightarrow{EG}$ break down or do not have enough bandwidth to assure QoS transmission, node B and E will enable TSL and mesh-links to find a backup routing path. The backup routing path is shown in Fig. 2-4(d), if B->C->E and E->I->D can replace link $\overrightarrow{BP}$ and $\overrightarrow{EG}$ respectively without going through Cluster head P and G, the new path S->F->B->C->E->I->D will be used to replace the failed tree routing path.



(a) CMT formation (b) Tree routing path



(c) Link $\overrightarrow{BP}$ and $\overrightarrow{EG}$ are unavailable (d) CMT backup routing path

Fig. 2-4.A CMT routing protocol example

2.3 Discrete Distributed Topology-control Algorithm

X. Huang and P. Liu [7] described the WSN topology-control algorithm as minimizing energy consumption and increasing network lifetime. Topology control has been an important issue in the WSN field. Recently, Huang Xuyong and Liu Pei [7] proposed a localized discrete distributed topology-control algorithm (DDTCA) to obtain optimal WSN energy consumption and increase network performance. The DDTCA can strongly decrease the number of redundant WSN links to achieve minimal energy consumption.

DDTCA adopts the distance (denoted as $d\ (u, v)$) and maximum transmission power (denoted as P_{max}) between two nodes of link $\overleftrightarrow{uv}$ to estimate the quality of link $\overleftrightarrow{uv}$. The idea is to use $d\ (u, v)$ to get a value, called a weight (denoted as $W\ (u, v)$), which is proportional to $d\ (u, v)$. The power consumption for each transmission between two nodes (denoted as $P\ (u, v)$) of link $\overleftrightarrow{uv}$ can then be acquired with the formula (1) [7] as below:

$$P\big(d(u,v)\big) = k \times W(u,v) \times P_{max} \tag{1}$$

Using formula (1) we can know that the quality of link $\overleftrightarrow{uv}$ must be inversely proportional to the weight value $W\ (u, v)$.

X. Huang and P. Liu [7] used Fig. 2-5 to show a general DDTCA network case. As shown in Fig.2-5 (a), all of the WSN nodes construct a competed graph first. As shown in Fig.2-5 (b) the DDTCA divides the network into a set of isolated network islands. Figures 2-5 (c) and (d) apply the inter-sub-graph algorithm with *Inter-Link*=1 and *Inter-Link*=2 respectively to finish a DDTCA network.

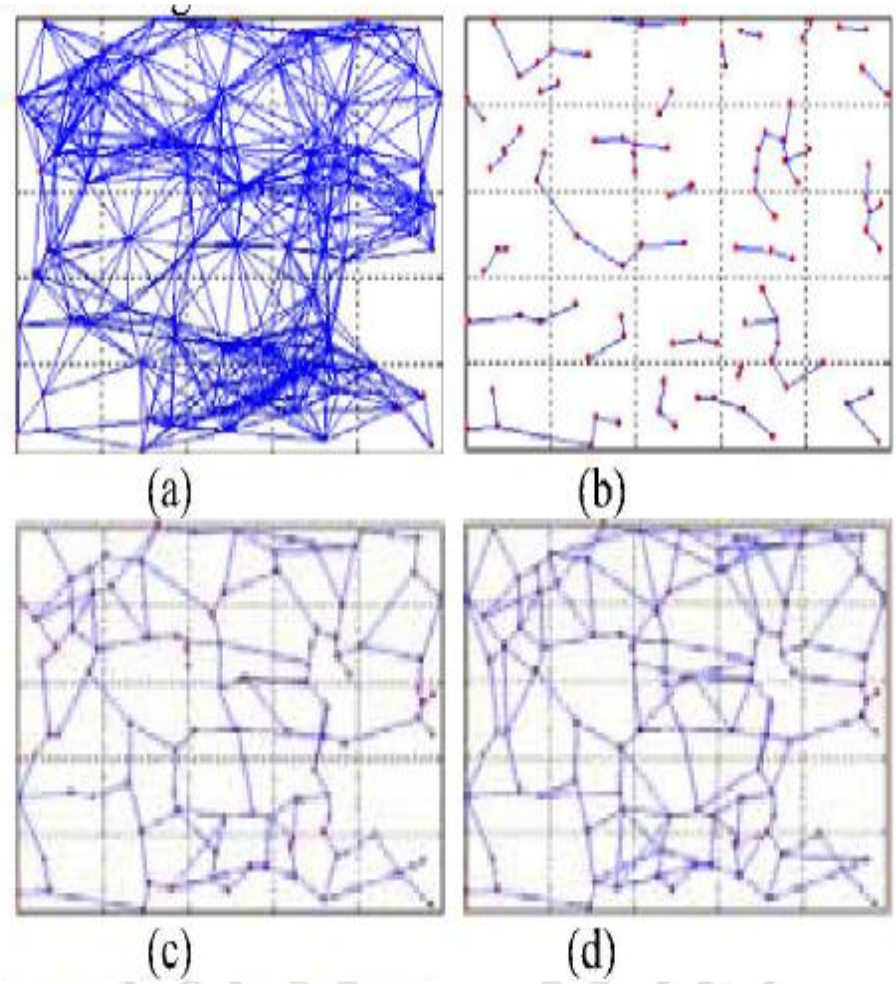


Fig. 2-5.The procedure of DDTCA [7]

The DDTCA procedure shown in Fig. 2-5 is briefly described below. First the WSN links are divided and formed into a number of isolated network island topologies that can be used to act as clusters in the cluster-tree network. All of the links in these isolated island topologies have the best quality and are also bounded. All isolated network island topologies are linked by the Inter-Sub-Graph [7] algorithm which minimizes the power consumption between the isolated network islands.

To describe the DDTCA algorithm, WSN is simply denoted as a graph G (V, E) in

[7], where V is the set of nodes and E represents the set of communication links. The DDTCA algorithm is separated into three parts shown below:

1. Fun (u, v) [7]: this part is a function used to determine whether link (u, v) has the highest link quality. It is utilized to determine whether if a link existing between a set of shared neighbors for the nodes u and v and u, has better link quality than link (u, v). If there is any link that has higher quality than link (u, v), Fun (u, v) will return true to the Discrete algorithm (u).

2. Discrete algorithm (u) [7]: This part deletes all of the links that have worse link quality with Fun (u, v). The network graph G is then separated into many independent sub-graphs, denoted as INI or G_{ini}. We call the graph $G_{dis}(V, E_{dis})$ that is composed of all of these sub-graphs. Let graph G_{dis} be symmetrical, each node of G_{dis} must exchange its neighbor information though broadcasting a “hello message” to all of its neighbors in the next phase. Each pair of nodes that have a link that is the best link quality in G_{dis} must be neighbors to each other. That is, one node is a neighbor of another node in G_{dis} if and only if another node is a neighbor of that node.

3. Inter-sub-graph () [7]: This algorithm will associate with all G_{ini} to form a final graph G_{ddtc}. We can use a value *Inter-Link* for G_{ddtc} to build *Inter-Link* links between any two adjacent G_{ini}. This causes the network based on DDTCA to preserve connectivity. We can also use the links between all G_{ini} to generate inter-cluster-mesh links to enable “generalized time-slot leasing (GTSL)” across clusters in our DCMT protocol.

Chapter 3. Our DCMT Protocol

Natural bottlenecks exist in the cluster-mesh-tree routing protocol because it is based on the SMC cluster. It generates so many 1-hop clusters that CMT may increase the number of beacon frame collisions, causing the tree-based routing path length to increase. To solve these problems, we adopted the DDTCA (using *Inter-Link*=1) to build a new clustering scheme that may have various cluster sizes to decrease the number of clusters in the cluster-tree network. In addition this scheme can provide inter-cluster mesh-links to further reduce the tree-based routing path length.

The new clustering scheme based on the DDTCA and CMT must provide the generalized TSL (GTSL) approach that can operate using multi-hop clusters for the inter-cluster-mesh mechanism. This chapter contains three parts. First we introduce and analyze the GTSL performance to understand why it can realize communication between two end nodes in the inter-cluster-mesh link as well as increase the transmission performance. Secondly, we describe how DCMT formation with DDTCA is constructed. Finally, we propose a new QoS routing protocol based on the GTSL approach and CMT routing protocol.

3.1 The analysis of the generalized TSL (GTSL)

Zhang et al. [16] proposed the TSL approach to provide a slave-to-slave model in a Bluetooth environment. TSL can only be used for 1-hop piconet, meaning that any two slave nodes cannot communication directly. This feature fits only for the SMC cluster in the CMT network and can reduce the TSL set time. The traditional TSL also causes the network to form many 1-hop clusters to contend with WSN scalability.

The traditional TSL approach does not fit our DCMT protocol because the DCMT clusters are not a 1-hop clustering network. They have a larger cluster size generally and this property will obviously decrease the number of network clusters.

To extend the TSL approach to fit the DCMT network, we modify the TSL approach to make any pair of slave nodes communicates directly and they can have individual parents. Therefore, if any two slave nodes in the DCMT network can communicate directly with each other. They can be in different clusters and the DCMT allows many ancestor nodes between them. We call this modified TSL “Generalized TSL (GTSL)” approach. It realizes the inter-cluster-mesh mechanism of the DCMT network. Figure 3-1 and 3-2 show the difference between the TSL model and the GTSL model.

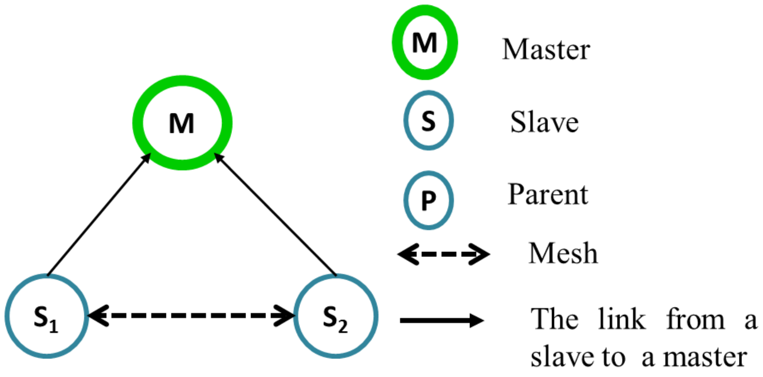


Fig. 3-1.TSL model

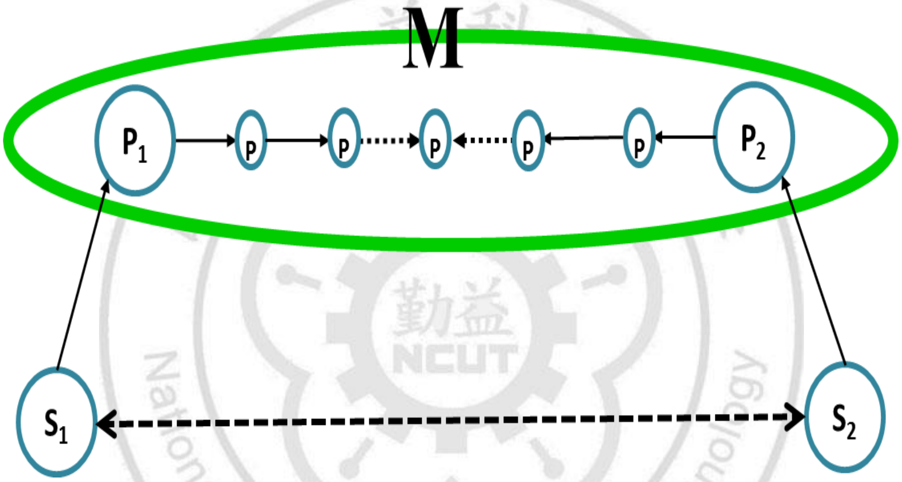


Fig. 3-2.GTSL model

In the remainder of this section we analyze the set time for the GTSL and illustrate why it can increase the cluster-tree network performance when the cluster- tree network has a larger amount of data to deliver.

First we analyze the GTSL set time. The TSL set time is shown in Fig. 2-3. Obviously, we can determine that the TSL requires 8 time slots to setup. In the other words if a slave node in the 1-hop network (piconet or cluster) wants to lease time slots from its master to communicate directly with another slave node, it must spend 8 time slots to set the TSL and then begin to deliver data directly to another slave node. An example is shown in Fig 2-2.

The GTSL may have more than one hop between two slave nodes as shown in Fig 3-2. There are many parent nodes between slave nodes S_1 and S_2 We can logically consider these nodes as the master set for nodes S_1 and S_2. To understand the relationship between the master and number of hops between two slave nodes (denoted as *hops*) we illustrate an example shown in Fig. 3-3. The example shows a simple 2-hop network (*hops*=2). If node S wants to communicate directly with node D the set time for the GTSL (denoted as *GTSL Set Time*) is 11 time slots from Fig. 3-4.

The above observations show that when *the number of hops* becomes greater the *GTSL Set Time* is also longer we can introduce this relationship into formula (2) as below.

$$GTSL\ Set\ Time = 8 + 3 \times [hops - 1] \text{ time slots} \tag{2}$$

Formula (2) is used when the number of *hops* is equal to 1 the *GTSL Set Time* will be equal to 8 time slots. This result is equal to the set time for the TSL approach. This is because the TSL must have 8 time slots to enable a slave-to-slave communication and whenever the number of *hops* increases by 1 it also adds 3 time slots to set each slave-to-slave communication. Therefore, we can use formula (2) to calculate the *GTSL Set Time* for any case of *hops*.

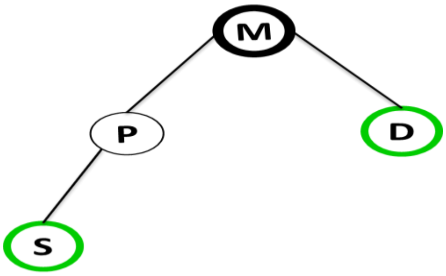


Fig. 3-3.A 2-hop network model

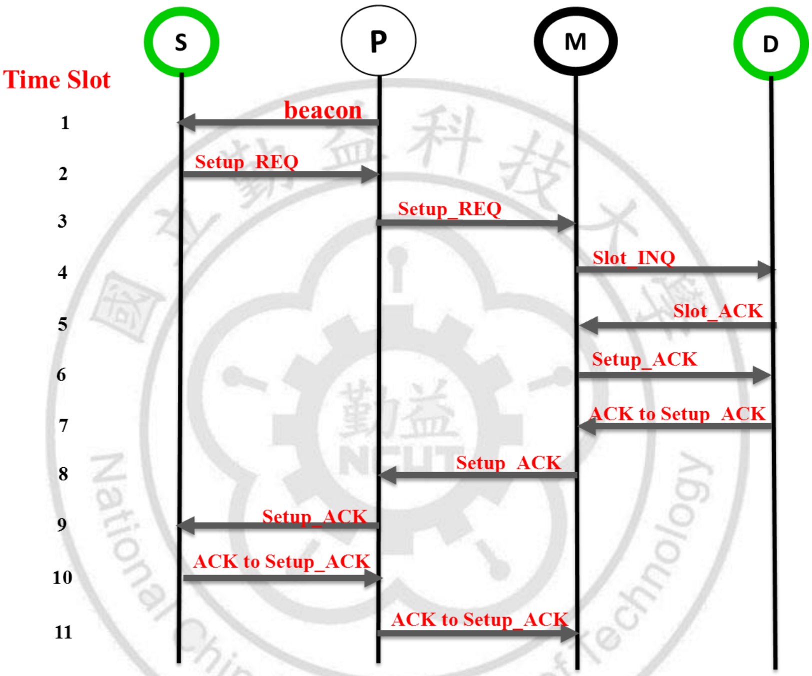
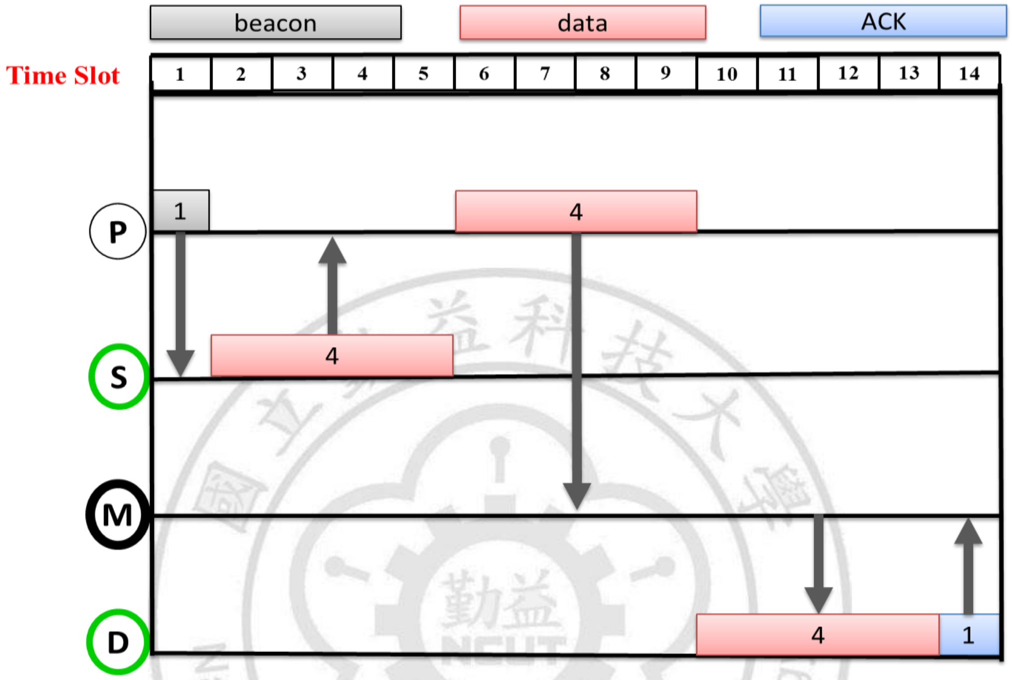


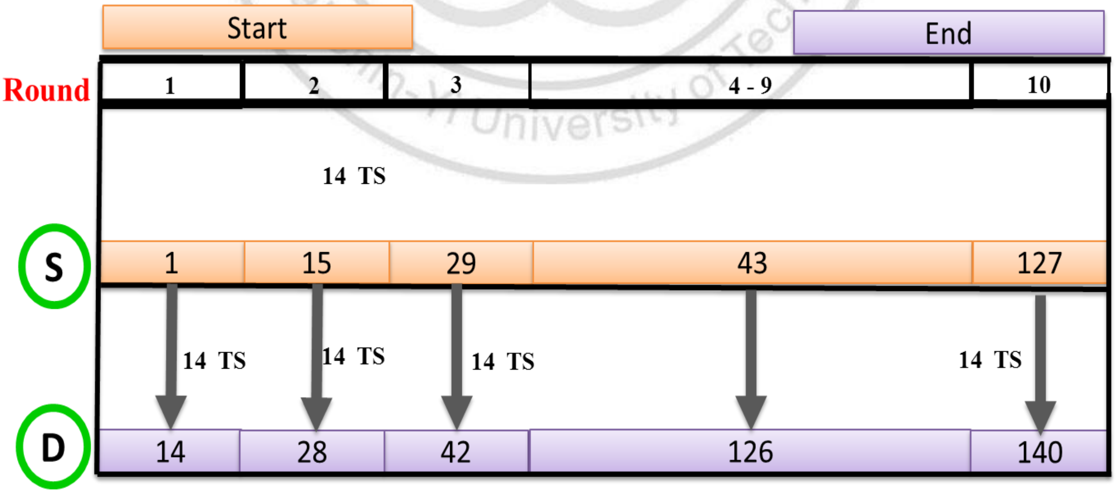
Fig. 3-4.The setting procedure of GTSL

We utilize formula (2) to prove that GTSL can reduce the transmission time shown Fig 3-5. It uses the same example shown in Fig. 3-3 and assumes that it has an amount of data requiring 40 time slots to deliver data from node S to node D. Furthermore it also requests the QoS bandwidth to be equal to a least 4 time slots. In Figs 3-5, 3-6 and 3-7, we show the required transmission time for slave-master-slave, TSL and 2-hop GTSL respectively.

(a) Slave-master-slave transmission model.



(b) The transmission time of 1th round in slave-master-slave model.



(c) The total transmission time of slave-master-slave model.

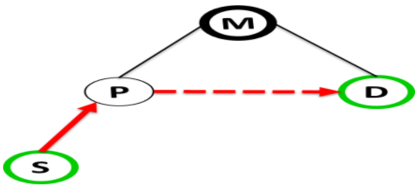
Fig. 3-5.The slave-master-slave model transmission time

Figure 3-5 uses a slave-master-slave path S->P->M->D to accomplish the transmission task. In this case, 140 time slots are necessary to deliver data. The transmission data times are first divided into 10 rounds because the data has 40 time slots. Each node can only spend 4 time slots at most to deliver data in each round. Similarly, the cases in Figs. 3-6 and 3-7 also need 10 rounds to deliver the entire amount of data. The case in Fig 3-5 spends 140 time slots to deliver all of the data. This is because each node must spend 1 time slot to acquire a beacon from its parent node and 4 time slots to transmit data to the next hop. If the relay nodes are greater than one (*hops* is not equal to 1), 4×(*hops* + 1) time slots are required to forward the data in each round. When the data for each round is successfully transmitted to the destined node, 1 time slot is required for ACK to tell the parent node of the destination node that the delivery round is completed. The next round will start after this round. The total amount of transmission time for this case is 140 time slots, as shown in Fig. 3-5(c).

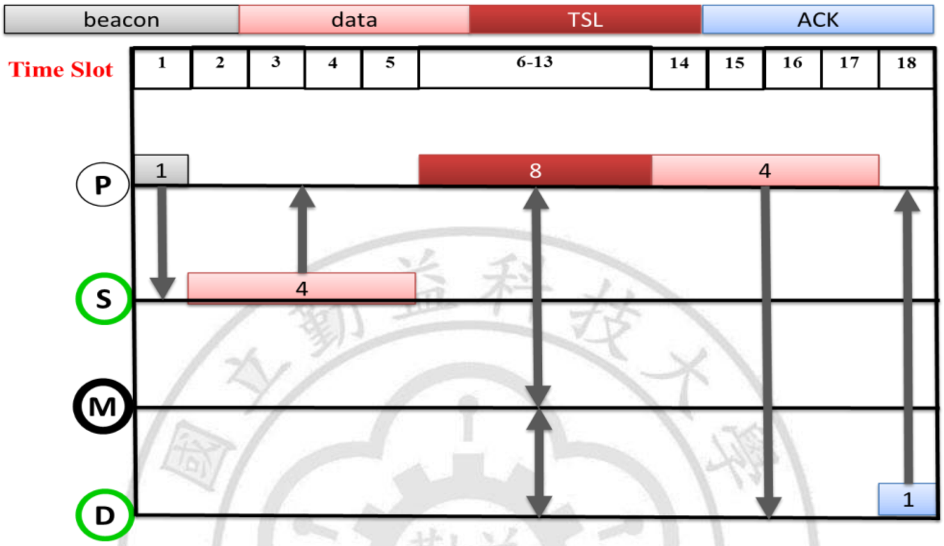
Figure 3-6 adopts the TSL model to transmit data from node S to node D. We use the TSL approach to deliver the data without going through node M and reduce the routing path length between node S and node D in Fig.3-5 to 1 hop. The path in this case is S->P->D. In this case, the first round will spend more time slots than the other rounds because the first round includes time to set TSL between node P and node D. Eight time slots are required to enable direct slave-to-slave communication. The time the first data is transmitted to node P by node S and the forwarding time from node P to node D must be included, as shown in Fig. 3-6(b). The transmission time for the first round in the TSL model is 18 time slots. The remaining rounds in this case will spend the same time to transmit the remaining data. These rounds do not need to set TSL again and again. Each of the remaining rounds must spend 10 time slots to deliver data from node S to P and then forward the data from node P to D, as shown in Fig. 3-6 (c).

The transmission time for the TSL case is 108 time slots with the first round time (18 time slots) added to the total time for the remaining rounds (10*9 = 90 time slots). The result is shown in Fig. 3-6(c).

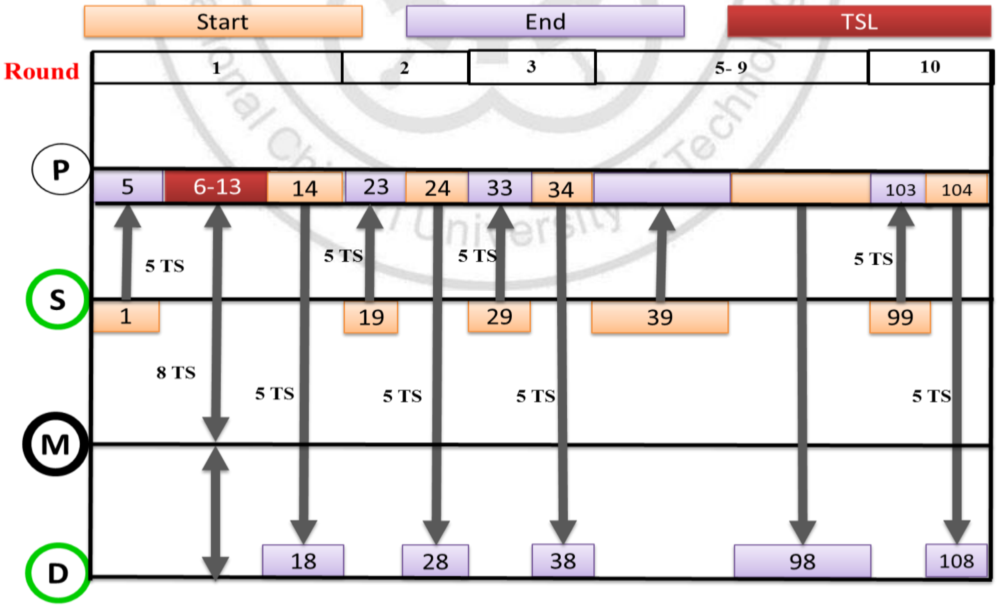
We discuss the 2- hop GTSL case in Fig 3-7. In this case, node S utilizes the GTSL approach to send data directly to the destination node D and reduce the routing path length to 0 *hops*. Therefore, only the *GTSL Set Time* and the data transmission time from S to D are involved. Formula (2) determines the *GTSL Set Time* as 11 time slots. After setting GTSL successfully, the data can be sent directly from node S to D and it must repeat ten rounds. The transmission time for each round is 5 time slots, as shown in Fig. 3-7(b). Four time slots are needed for forwarding data from node S to D and 1time slot for the ACK control packet. Therefore, the transmission time for the 2-hop GTSL case is 61 time slots. This case contains the *GTSL Set Time* (11 time slots) and the total transmission time for all rounds (5*10 = 50time slots), as shown in Fig. 3-7(b).



(a) TSL transmission model.



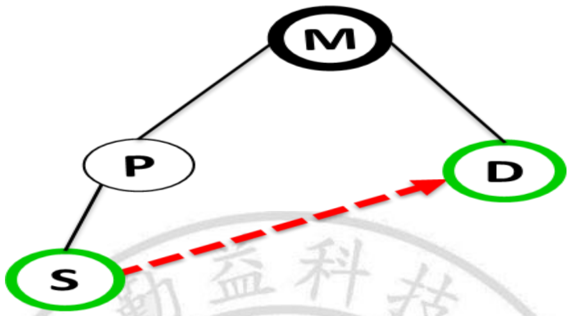
(b) The transmission time for the 1th round in TSL model.



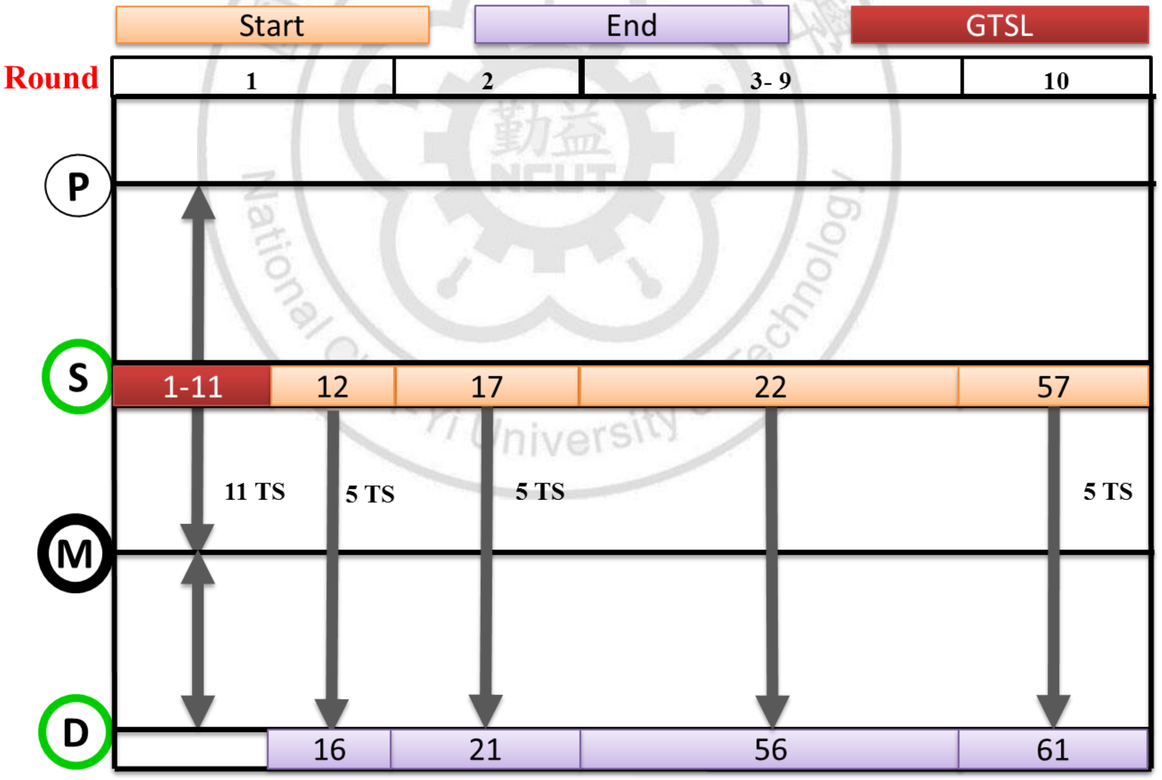
(c) The total transmission time for the TSL model.

Fig. 3-6.The transmission time for TSL model

Obviously, we determine the total transmission time for the 2-hop GTSL model is the least of all transmission models (61<108<140 time slots). This result proves that the GTSL approach can significantly improve the transmission performance of WSN networks when the number of transmissions in a WSN network is larger.



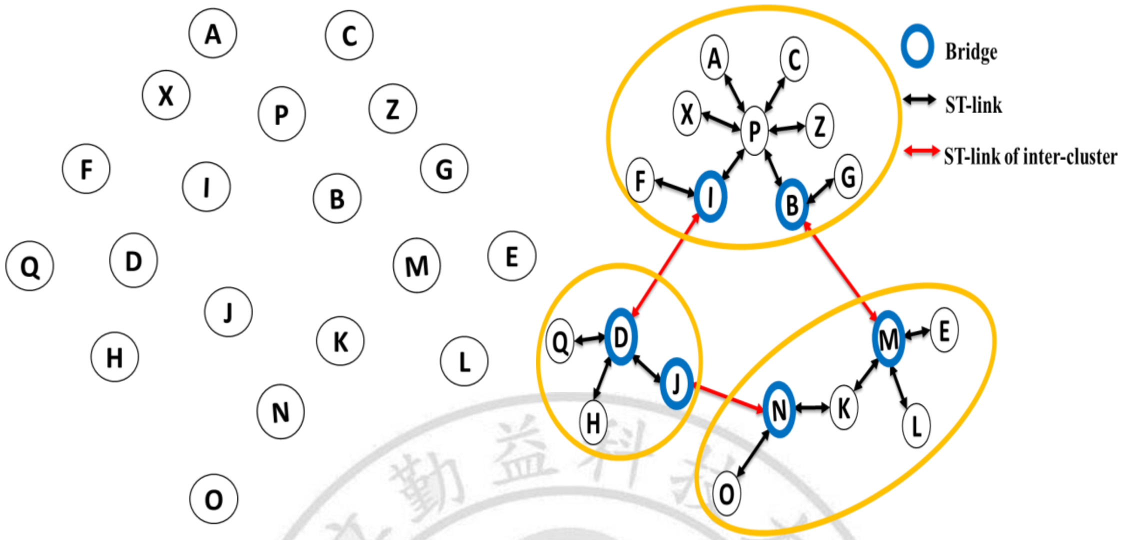
(a) 2-hop GTSL transmission model



(b) The total transmission time for 2-hop GTSL

Fig. 3-7.The transmission time for 2-hop GTSL model

3.2 Formation of DCMT



(a) A set of the discrete WSN nodes (b) DDTCA (using *Inter-Link*=1).

Fig. 3-8.A DCMT example

We use a simple case to explain the DCMT formation procedure in this section. To construct a DCMT network, a group of discrete nodes that enable neighboring node detection are used to form a full-connected network topology. The network is then optimized using DDTCA (using *Inter-Link*=1) to obtain the best and least number of links. Figure 3-8 shows an example to explain that a WSN is optimized by DDTCA (using *Inter-Link*=1). We will adopt consecutive examples to describe the DCMT formation flow.

After the above steps, we can obtain some powerful information for DCMT construction, defined as below:

Defintion 1. The suggested tree-link (ST-link): DDTCA preserves only the best link based on energy-efficiency for any pair of adjacent nodes. It deletes other non-optimal WSN links and therefore, we should use these links as the tree-links for the energy-efficient DCMT.

Defintion 2. Useful-link (U-link): although DDTCA deletes some non-optimal links, we can record them into each node to perform as backup links to implement redundant paths.

Defintion 3. Degree: we also record the number of ST-links for each node. The node with the most degrees will be the possible coordinator of its cluster.

Defintion 4. Cluster: DDTCA can form one or more isolated network island topologies. These "network islands" are connected with *Inter-Link* links between two adjacent islands. We can think of them as DCMT clusters.

Defintion 5. Bridge: the WSN formed through DDTCA will obtain some nodes that have links that can connect with other clusters. The nodes on these links can be used as bridges for relaying data to other clusters.

We briefly describe the DCMT formation procedure as follows:

Step 1. Construct a full connected network.

Step 2. Reduce the full connected network with DDTCA (using *Inter-Link*=1).

Step 3. Select a PAN coordinator C_P and to denote C_P's cluster as cluster 0 (C_0).

Step 4. Cluster C_0 begins association procedure [4] to construct the tree-links and mesh-links within its range.

Step 5. If C_0 determines a node belongs to other network islands. It then gives the new cluster a name and records the new cluster as its child-cluster.

Step 6. All new found clusters begin association procedure.

Step 7. If a cluster finds an adjacent cluster and they belong to the same parent-cluster, the two clusters become sibling-clusters to each other. The ST-link between their bridge nodes is changed into a inter-cluster-mesh link.

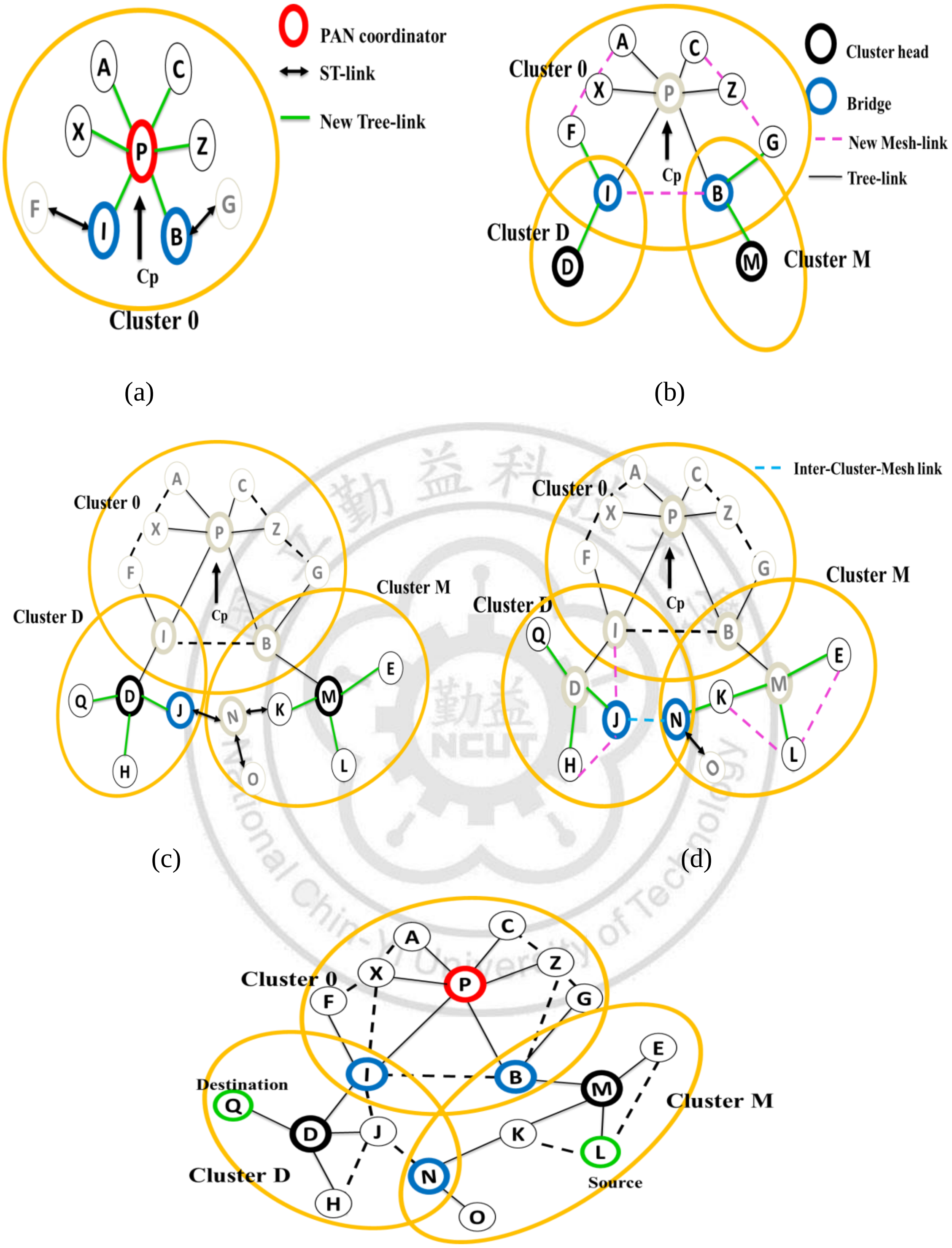
Step 8. If there is any cluster that does not complete the association procedure, that cluster will continue the association procedure until all of the nodes inside its range complete the network device searches within their radio propagation area.

Step 9. When all clusters have completed the association procedure, the DCMT formation procedure ends.

Figure 3-9 shows the DCMT formation procedure. First, a node that is as close as possible to the center position of the entire network is chosen as the PAN coordinator. This node should have the greatest number of ST-links. As shown in Fig. 3-9(a), node P is chosen as the PAN coordinator (C_P) and its cluster is denoted Cluster C_0. The PAN enables the association procedure to search its neighbors and join them into the network. Nodes A, B, C, X, I and Z have ST-links that connect directly with PAN. The PAN coordinator P allows these nodes to join the cluster C_0 and transform ST-links among them into tree-links. After the PAN node association procedure, the PAN coordinator child nodes also start association procedures to finish the topology of cluster C_0 and find new clusters. As shown in Figure 3-9(b), all of the nodes in cluster C_0 have individual tree-links and mesh-links. The find mesh-links method refers to the CMT formation method. Figure 3-9(b) shows Bridges I and B of cluster C_0 finding new clusters C_D and C_M through Bridge D and M discovery. We join nodes I and B to clusters C_D and C_M respectively, to balance the cluster head workload of cluster C_0 (C_P) because cluster C_0 has more nodes than the other clusters. If new cluster members D and M are added, the cluster head C_P will have to spend more resources to coordinate the new cluster member. Clusters C_D and C_M record cluster C_0 as their parent-cluster because the parent nodes of node D and M are cluster members of cluster C_0.

Figures 3-9 (c) and (d) show clusters C_D and C_M enabling the topology procedure. This procedure is similar to cluster $C_{0.}$ Clusters C_D and C_M identify that the Bridges J and N have a ST-link between them. This link is changed into an inter-cluster-mesh link between Cluster C_D and C_M. This is because clusters C_D and C_M have the same parent-cluster C_0. Cluster C_D and C_M record the inter-cluster-mesh link $\overleftrightarrow{JN}$ and the sibling cluster relationship between them. If there is a node that has more ST-links than the cluster head in a cluster, that node will become the new cluster head for that cluster. The DCMT formation result is shown in Figure 3-9(e).

Figure 3-10 shows the CMT formation in Figure 3-8. We compare the DCMT formation in Figure 3-9(e) with the CMT formation and see that the number of DCMT and CMT clusters are 3 and 4, respectively. This is because the CMT cluster is based on the SMC structure. The number of DCMT clusters is less than the number of CMT clusters.



(a) (b)

(c) (d)

(e)

Fig. 3-9.The DCMT formation procedure

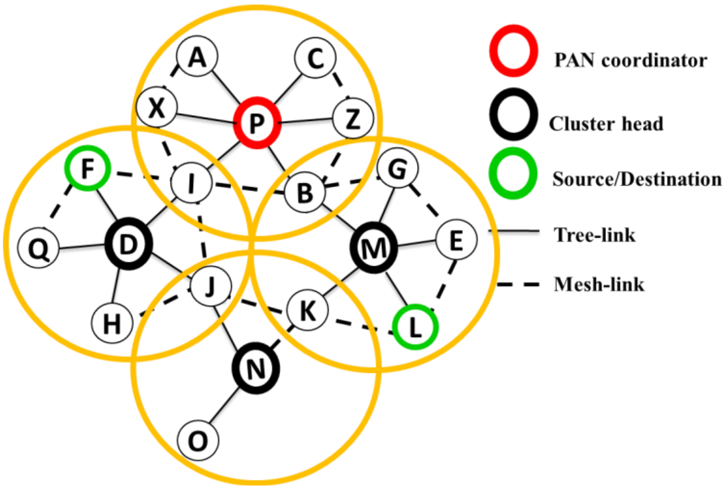


Fig. 3-10.The CMT formation

Now, we not only complete DCMT formation but also obtain the relationship of all clusters and nodes, as well as all tree-links, mesh-links and inter-cluster-mesh links in the network. We utilize this information to build the DCMT routing protocol.

3.3 DCMT QoS routing protocol

The DCMT routing protocol concept is not complex. DCMT contains the relationship between adjacent clusters and indicates the parent-cluster, child-cluster and sibling-cluster of each cluster. We use this information to build the inter-cluster routing discovery strategy and replace the typical tree routing strategy. That means our DCMT routing protocol searches inter-cluster paths first and then delivers the data through this inter-cluster path to the designed cluster from the source cluster. Each cluster must select its parent-cluster or child-cluster as members of its inter-cluster path. If the link between the cluster and its parent-cluster or child-cluster has been corrupted, it will search for an existing inter-cluster-mesh link between it and its sibling-cluster to replace the unavailable link between it and its parent-cluster. If an inter-cluster-mesh link is found with its sibling-cluster, this link will be added to the routing path to replace the original inter-cluster path.

After determining the inter-cluster path, we must still find the intra-cluster paths in each cluster on the inter-cluster path. The intra-path is found in a way similar to that for the inter-cluster path. Each node sends data to its parent in a cluster first, but if the link is corrupt, that node will then search for an available mesh-link to replace this link. This enables the GTSL approach to routing data through the mesh-link.

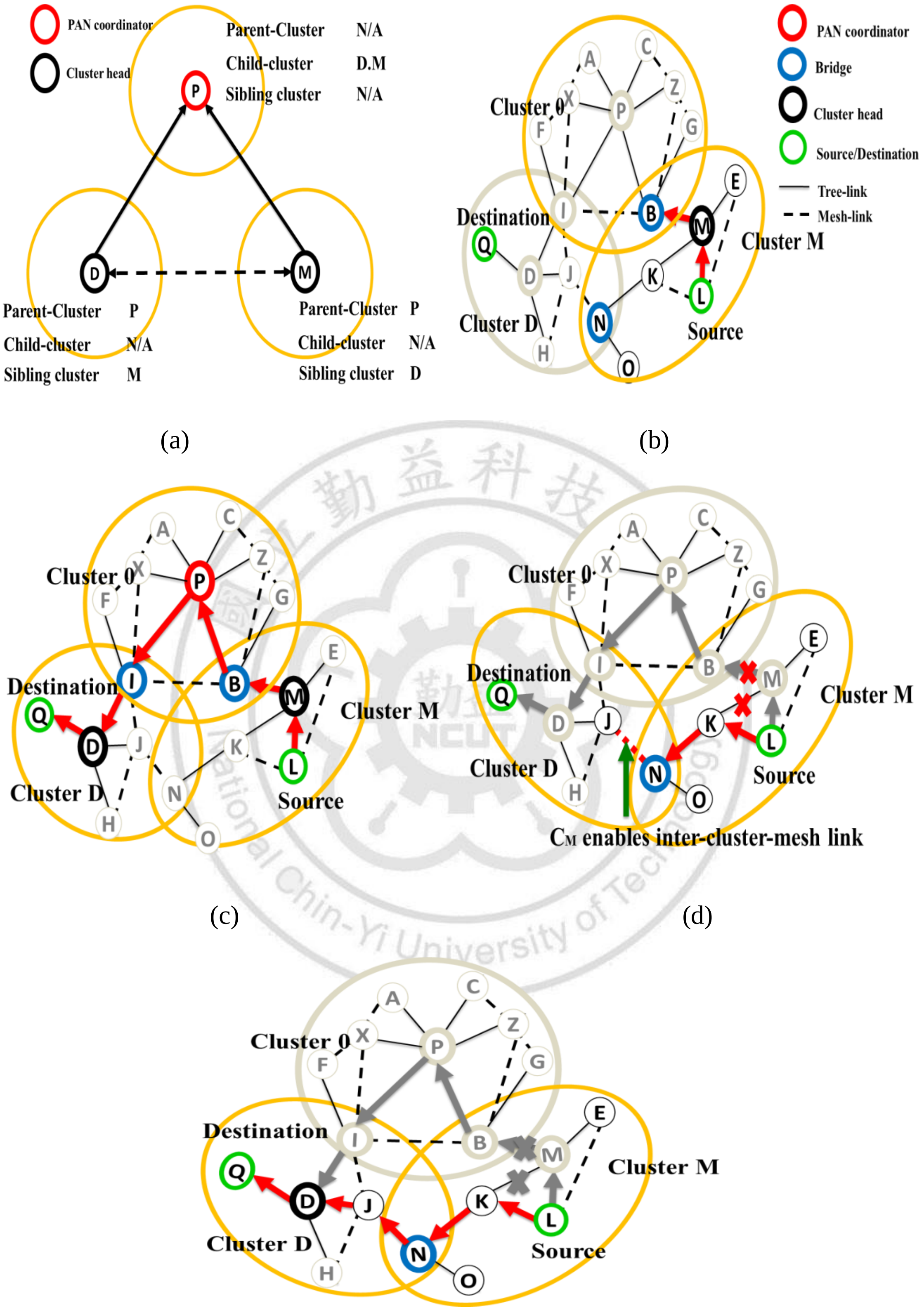
To be brief, the DCMT routing path can be denoted with inter-cluster + intra-cluster paths, and if the inter-cluster path cannot be built, the DCMT routing fails. The DCMT routing discovery flow is similar to that for CMT. The source node sends a DCMT_REQ packet to perform the routing discovery. The DCMT_REQ as shown in Table 3-1 modifies the QoS_REQ [2] CMT packet to contend with the DCMT routing protocol requirement. The DCMT routing discovery in Fig. 3-9(e) is shown as Fig. 3-11.

Figure 3-11 (a) shows the relationships between all of the clusters. It indicates that cluster C_0 is the parent-cluster of clusters C_M and C_D, and the relationship between clusters C_M and C_D is sibling. In Fig. 3-11(b), the source cluster C_M forwards DCMT_REQ toward to its parent-cluster C_P and record the inter-cluster path C_M->C_P as well as intra-cluster path L->M->B. In Fig. 3-11(c), the intermediate cluster C_0 routes the DCMT_REQ to its another child-cluster C_D and record it into the inter-cluster path to finish the successful inter-cluster path C_M->C_P->C_D, at the same time, if the inter-cluster link $\overrightarrow{ID}$ as well as both the intra-cluster paths of the C_P and C_M are available to route the data to the destination node Q. The successful routing path L->M->B->P->I->D->Q will be returned to source node L using a DCMT_RREP packet from destination node Q.

In Fig. 3-11(d), if the inter-cluster link $\overrightarrow{MB}$ is unavailable without QoS bandwidth, the inter-cluster path C_M->C_P->C_D fails, and then, the DCMT_REQ is dropped and the Source node will enable inter-cluster-mesh link to recover a new inter-cluster path. In Fig. 3-11(e), if the inter-cluster-mesh is available and both the intra-cluster paths of C_M and C_D can be found, then the DCMT_RREP will be returned to source node L from destination node Q through the inter-cluster path C_M->C_D and the set of intra-cluster paths L->K->N->J->D->Q.

Table 3-1.The DCMT_REQ Packet

Packet field	Field description
DS_Cluster	The cluster name of the source node
DS_Node	The name (or address) of the source node
DD_Cluster	The cluster name of destination node
DD_Node	The name (or address) of the destination node
Inter_Cluster_List	List of clusters that records the path from source to the current traversed cluster
Temp_Intra_Path_List	The intra-cluster path on the each cluster of Inter_Cluster_List
Temp_Path_List	List of nodes that records the path from source to the current traversed node
Path_List	The completed Temp_Path_List that is copied to the List and this list becomes the successful routing path from source node to destination node
Cur_Free_TS	The free time slots for the current traversed node
QoS_RQ	QoS requirement
Min_FTP	Minimum available free time slot packets among the Path_List



(a)

(b)

(c)

(d)

(e)

Fig. 3-11.The DCMT routing procedure

Chapter 4. Experimental Results

In our simulation, we compare DCMT (using *Inter-Link*=1) with DCMT Without GTSL (using *Inter-Link*=1), CMT and CT (cluster-tree) in the larger network. We adopt some performance metrics to archive our goal in this simulation. The performance metrics are shown below:

1. Success rate: Chen and Lin [2] define that Success rate is the ratio of successful requests/packages to the total requests/packages.
2. Throughput: Chen and Lin [2] define that Throughput is the value of all data bytes received by all devices from the starting to the designated cycle time.
3. Bandwidth utilization (BW): Chen and Lin [2] define that Bandwidth utilization is the ratio of all used time slots of successful QoS requests to the total time slots for each cycle time.
4. Life time: the network lifetime is the time that network can deliver data from any node to another. In this metric, we can prove that DCMT has the better energy-efficiency.

.

We use C++ to simulate the performance metrics as presented above. All of the nodes in the network are assumed static nodes in the environment shown in Table 2 simulation parameters. We request that each transmission quadrant 4 time slots for the QoS bandwidth requirement and its data size be 30 -80 time slots. This is because the DCMT protocol performance will be much better than CMT and CT protocol performance when the network has a great deal of transmission.

Table 4-1.Simulation parameters

Parameter	Value
N: Number of nodes	50 - 100 nodes
A: Side of the square area	250m *250 m
R: Radio propagation range	$\sqrt{2}$ m
Ct: Cycle time	16 time slots
PL: Package length	1 time slots
M: Mobility	No
QL: QoS request length	5-20 packages
BW: QoS bandwidth	4 time slots
QR: QoS request frequency	2 requests/cycle

The comparison result is shown in Figures 4-1, 4-2, 4-3 and 4-4. The performance of our DCMT protocol is superior to that of other protocols in the larger network, regardless of throughput, success rate, bandwidth utilization, or average network lifetime. This is because CMT does not have a suitable topology-control mechanism in the larger network, and it is a network based on SMC, therefore, there are more clusters than DCMT and this incurs more beacon frame collisions as well as excessive routing path length. The network lifetime of CMT is less than that of DCMT because its energy consumption is higher than DCMT. DCMT can use the inter-cluster-mesh to efficiently reduce the routing path length. This does decreases energy consumption for each node significantly, and also increases network throughput and success rate. Although more time is needed to set GTSL, the great deal of transmission will compensate for the negative effect.

The DCMT without GTSL has worse network performance. Because it uses TSL to provide general slave-to-slave communication, it cannot adapt some mesh and inter-cluster-mesh links that have no common parent node. The DCMT without GTSL is not an efficient protocol. It and CMT have the same drawback, or, their average routing path length is higher than that for DCMT.

Finally, we see that the CT network has the least performance because CT does not provide a backup routing path mechanism and topology-control approach. This incurs the transmission holding problem more seriously than for the other network protocols in our simulation.

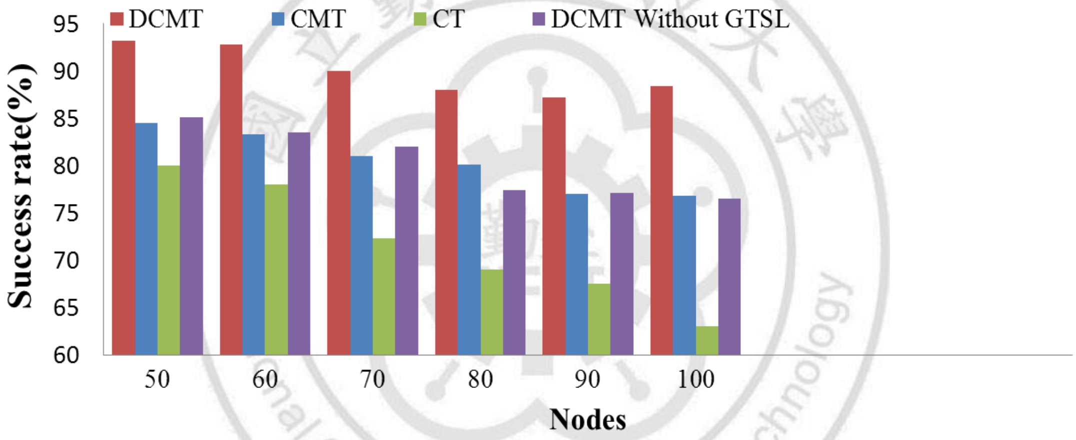


Fig. 4-1.Success rate

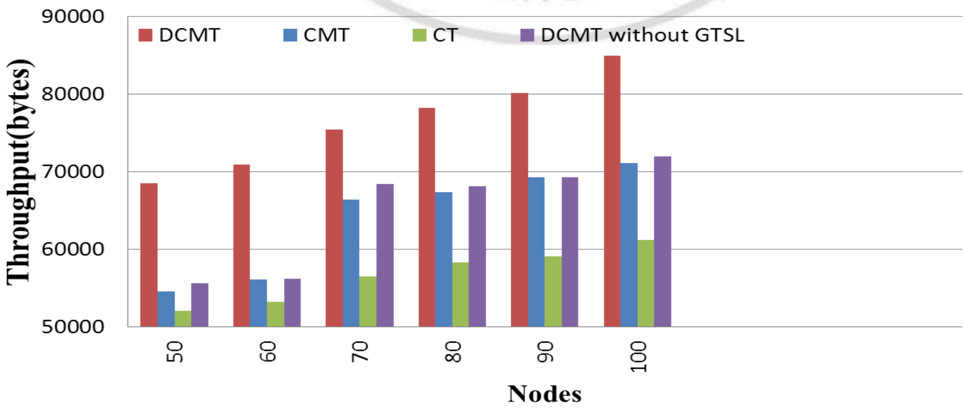


Fig. 4-2.Throughput

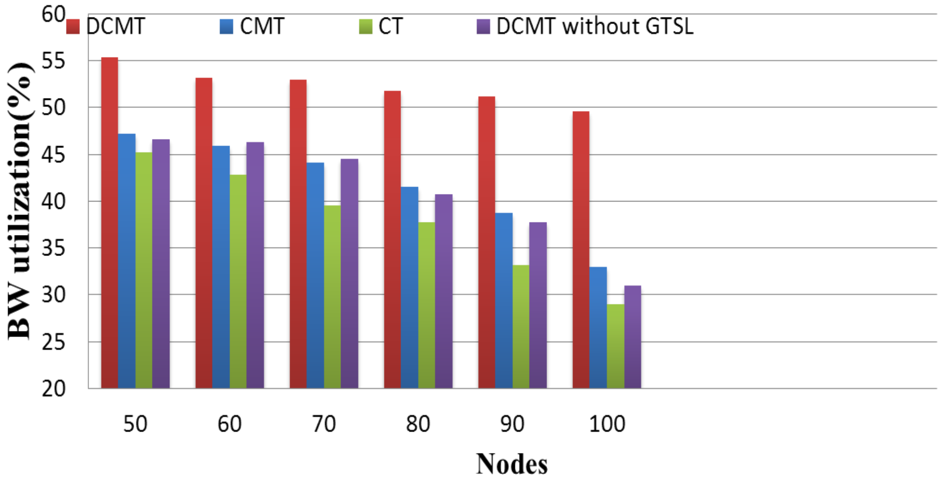
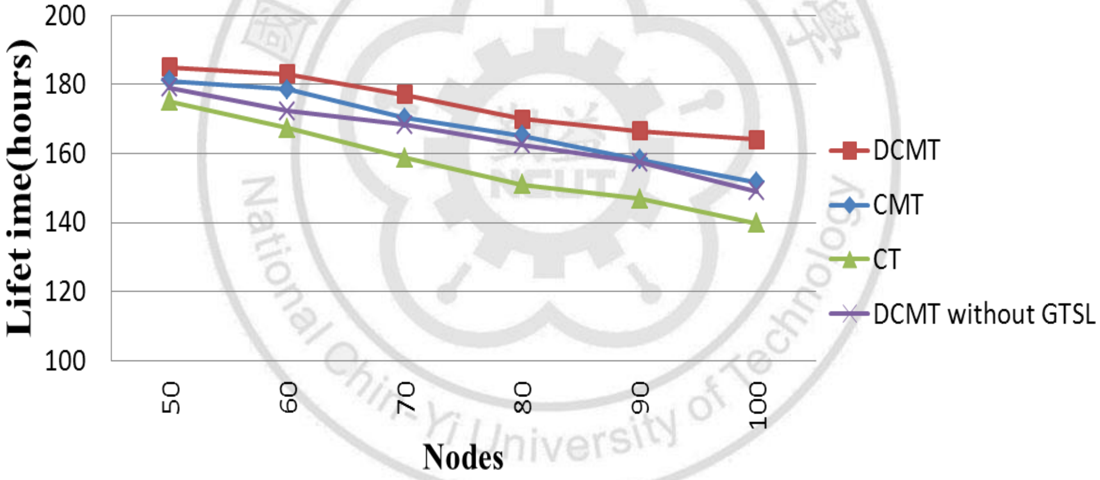


Fig. 4-3.Bandwidth utilization



.

Fig. 4-4.Average life time

Chapter 5. Conclusions

This thesis presented a new network scheme based on the CMT routing protocol and DDTCA (using *Inter-Link*=1), called the DCMT QoS routing protocol. In the DCMT routing protocol, DDTCA is used to decrease the number of clusters and provide an inter-cluster-mesh with the GTSL mechanism to reduce the routing path length. The DCMT protocol has higher energy-efficiency because it adopts DDTCA to build an energy-efficient network topology and decrease the average degrees for each node. This results in prolonged network lifetime and alleviates energy consumption. Our simulation results show that the network lifetime, data throughput, and transmission success rate exhibit better performance than the CMT routing protocol. Our concept which uses DDTCA to decrease the number of network clusters and increase the network energy-efficiency assure that DCMT has better performance than CMT.

References

[1] A. Bhatia, and P. Kaushik, “A cluster based minimum battery cost AODV routing using multipath route for ZigBee”. *International Conference on Network (ICON)*, Dec. 2008.

[2] Y. S. Chen and T. H. Lin, "A QoS Routing Protocol using Cluster-Mesh-Tree for IEEE 802.15.4 Low-Rate WPANs". *The Proceeding of the Workshop on Wireless, Ad Hoc and Sensor Networks (WASN 2005),* Chung-Li, Taiwan, Aug. 1-2. 2005.

[3] Y. S. Chen and T. H. Lin, "A Time-Slot Leasing-Based QoS Routing Protocol over Bluetooth WPANS," *International Journal of Ad Hoc and Ubiquitous Computing*, Vol. 2, Nos. 1/2, pp. 92-108. 2007.

[4] F. Cuomoa, S. D. Lunaa, E. Cipollonea, P. Todorovab and T. Suihkoc. "Topology formation in IEEE 802.15.4: cluster-tree characterization". *Sixth Annual IEEE International Conference on Pervasive Computing and Communications*, pp. 276-282, 2008.

[5] H. Fariborzi and M. Moghavvemi, "EAMTR: energy aware multi-tree routing for wireless sensor networks". *IET Commun* , Vol. 3, Issue 5, pp. 733–739, May 2009.

[6] P.J.M. Havinga and G.J.M. Smit. "Energy-efficient TDMA medium access control protocol scheduling". In: *Asian International Mobile Computing Conference, AMOC*, Penang, Malaysia, Nov. 2000.

[7] X. Huang and P. Liu. “A Discrete Distributed Topology-control Algorithm of Wireless Sensor Network”. *Third Annual International Conference on Mobile and Ubiquitous Systems: Networking & Services*, Page(s):1 – 6, July 2006.

[8] IEEE 802 Working Group. “Standard for Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low Rate Wireless Personal Area Networks (LR-WPANs)”, *ANSI/IEEE 802.15.4*, Oct. 2003.

[9] J. Kardash, “Bluetooth Architecture Overview”, *Intel Technology Journal Q2*, Mobile Computing Group, Intel Corporation, 2000.

[10] A. Koubaa, A. Cunha and M. Alves, “A Time Division Beacon Scheduling Mechanism for IEEE 802.15.4/ZigBee Cluster-Tree Wireless Sensor Networks”. *Real-Time Systems, 2007. ECRTS '07. 19th Euromicro Conference on 4-6 July 2007*, pp. 125-135, Aug. 2007.

[11] C.E. Perkins and E.M. Royer, “Ad hoc on-demand distance vector routing”, in *Proc. 2nd IEEE Workshop on Mobile Computer Systems and Application*, New Orleans, LA, Vol. 1, pp. 90-100, Feb. 1999.

[12] R.A. Rashid and W.M.A.E.W. Embong. "Development of Energy Aware TDMA-Based MAC Protocol for Wireless Sensor Network System". *European Journal of Scientific Research*, Vol.30 No.4, pp.571-578, 2009.

[13] V. Rodoplu and T.H. Meng. "Minimum energy mobile wireless networks," in *Proc. of the 1998 IEEE International Conference on Communications(ICC'98)*, Vol.3, pp. 1633-1639, June 1998.

[14] A.H. Shuaib and A.H. Aghvami. "A Routing Scheme for the IEEE-802.15.4-Enabled Wireless Sensor Networks". *IEEE Transactions on Vehicular Technology*, Vol. 58 No. 9, November 2009.

[15] W. Ye, J. Heidemann, and D. Estrin. “An energy-efficient MAC protocol for wireless sensor networks”. *21st Conference of the IEEE Computer and*

Communications Societies (INFOCOM), Vol. 3, pages 1567-1576, June 2002.

[16] W. Zhang, H. Zhu and G. Cao. “Improving Bluetooth Network Performance through a Time-Slot Leasing Approach”. *Proceeding of the 2002 IEEE Wireless Communications and Networking Conference (WCNC2002)*, Orlando, Florida, March 2002.

[17] J. Zheng and M.J. Lee, "NS2 Simulator for IEEE802.15.4". Available online: http://ees2cy.engr.ccny.cuny.edu/zheng/pub/. 2004.

[18] ZigBee-Alliance, "ZigBee specification". http://www.zigbee.org/. 2005.