# 무선 에드혹 네트워크를 위한 타이머를 이용한 CDS 구축

김보남[†]·양준모[††]

## 요 약

CDS는 다양한 라우팅과 브로드캐스팅 프로토콜을 통하여 무선 에트혹 네트워크에서 가상 백본으로 널리 사용되고 있다. 최소 CDS (minimum CDS)를 계산하는 것이 여전히 NP-hard로 알려져 있지만 sub-optimal을 구하기 위한 여러가지 방법들이 제안되고 있다. 그렇지만 대부분의 제안된 프로토콜들은 너무 복잡하거나 non-local 정보를 필요로 하고, 네트워크의 위상이 변할 때 적응하지 못한다. 뿐만 아니라 CDS로 선택된 노드와 선택되지 않은 노드들이 서로 다른양의 에너지를 소비한다는 것을 고려하지 않았다. 본 논문에서는 타이머를 이용하여 에너지를 효율적으로 사용하며 네크워크 전체의 성능을 향상시키는 Timer Based Energy Aware connected Dominating Set (TECDS) 프로토콜을 제안하였다. TECDS 프로토콜은 네트워크 위상이 변할 때 필요에 따라 CDS를 보존 또는 재구성할 수 있다. 시뮬레이션을 통한 성능 평가 결과는 제안된 TECDS 프로토콜이 다른 프로토콜보다 최적에 가까운 CDS를 구성하여 서로 다른 수준의 노드의 이동성 사이에서도 네트워크 운영이 효율적으로 연장 됨을 보여주고 있다.

**키워드 : 연결 도미네이팅셋, 무선 애드혹 네트워크, 프로토콜, 에너지, 타이머**

# TECDS Protocol for Wireless Ad Hoc Networks

Bonam Kim[†] · Junmo Yang[††]

## ABSTRACT

Connected Dominating Set (CDS) has been used as a virtual backbone in wireless ad hoc networks by numerous routing and broadcast protocols. Although computing minimum CDS is known to be NP−hard, many protocols have been proposed to construct a sub−optimal CDS. However, these protocols are either too complicated, needing non−local information, not adaptive to topology changes, or fail to consider the difference of energy consumption for nodes in and outside of the CDS. In this paper, we present two *Timer−based Energy−aware Connected Dominating Set Protocols (TECDS)*. The energy level at each node is taken into consideration when constructing the CDS. Our protocols are able to maintain and adjust the CDS when network topology is changed. The simulation results have shown that our protocols effectively construct energy−aware CDS with very competitive size and prolong the network operation under different level of nodal mobility.

**Key Words : CDS, Wireless Ad Hoc Networks, Protocol, Energy, Timer**

## 1. INTRODUCTION

The connected dominating set (CDS) has been used extensively as core or virtual backbone [1] in wireless ad hoc networks. A dominating set is a subset of nodes in a graph such that each node not in the subset has at least one direct neighbor that belongs to the subset. If the nodes in the dominating set form a connected graph, the set is called connected dominating set. It has been found extremely useful in routing [2] [3] [4], message broadcast [5] [6] [7], and collision avoidance [8].

Due to the nature of wireless ad hoc networks, it is impractical for any wireless ad hoc network protocol to assume a single node with the global view of the network acting as the coordinator. Hence, a good CDS protocol for wireless ad hoc networks should be fully distributed. In addition, it should possess the following properties.

- The resulting CDS should be as small as possible
- The CDS protocol should take into account the energy level at each
- The protocol should avoid the introduction of extra messages
- The protocol should adapt to station mobility

Most of the distributed CDS protocols, such as [2] [3] [4] [9], failed to put energy level at each node into consideration when constructing the CDS. In [10], Wu et

---

[†] 정 회 원 : 충북대학교 BK 사업단 연구원

[††] 정 회 원 : 삼성전자 통신연구소 책임연구원

논문접수 : 2007년 5월 8일, 심사완료 : 2007년 8월 6일

al proposed an energy-aware CDS protocol, but their protocol requires the introduction of extra messages. In this paper, we present two versions of the *Timer-based Energy-Aware Connected Dominating Set Protocols* (TECDS). In our TECDS protocols, the required information for the protocols is strictly obtained via beacon exchanges and the energy level at each node is used in both the initiator election and CDS construction phases, which allows the TECDS protocols to produce better energy-aware CDS. First, in the initiator election phase a node with more number of neighbors and higher energy level is elected as the initiator. In the CDS construction phase, each candidate node sets up a timer based on the number of uncovered neighbors and its own energy level, and determines whether or not to join the CDS when the timer expires. Our TECDS protocols are simple, distributed, inexpensive (i.e., introduce no extra messages and computation), and adaptive to nodal mobility. The simulation studies have shown that the proposed TECDS protocols yield better CDS than most of the existing CDS protocols in terms of CDS size and prolong the lifespan of the network.

The rest of this paper is structured as follows. In Section 2 we review the existing distributed and energy aware CDS protocols. In Section 3, we present the pseudo code of the TECDS protocols and explain how the TECDS protocols work. The simulation results and analysis are provided in Section 4. Finally, we conclude the paper in Section 5.

## 2. RELATED WORK TO CONNECTED DOMINATING SET

Constructing minimum CDS for an arbitrary graph is known to be NP-hard [14] [15]. The problem becomes more challenging when the knowledge of complete network topology is not available prior to computation, which is a practical assumption in wireless ad hoc networks. Hence, the distributed CDS protocols proposed in the past have settled with constructing a "smaller" CDS for wireless ad hoc networks based on local information available at each node.

Among them, the most noticeable protocols are [3] [10] [11]. The CDS protocol proposed in [3] obtains the CDS by eliminating unnecessary nodes from the network. It is basically a two-phase approach. In the first phase, a node exchanges the neighbor list with its neighbors. If a node finds that all its neighbors are neighbors to each other, it removes itself from the consideration of the CDS. In the second phase, some heuristic rules are applied to further reduce the size of the CDS. The protocol is simple, distributed, and most of time computes a CDS with a small size. In [10], the authors propose the algorithm that starts with a feasible and near-optimal CDS solution via marking process based on classification of neighbors, and removes vertices from this solution by redundancy elimination, until an approximate CDS is found. However, the above protocols require immediate neighbors to exchange neighbor list among one another. In addition, the protocol does not have a mechanism to maintain the CDS when network topology changes

Since energy (i.e., battery power) is a limited resource for nodes in wireless ad hoc networks, the energy-aware protocols for wireless ad hoc networks [12] [13] have always received special attention. In general, nodes in the CDS forward more packets and participate network management, so they tend to consume more energy than those outside of CDS. However, none of the above CDS protocols takes nodal energy into consideration when constructing CDS. In [11], Wu et al proposed an extended marking process that constructs energy-aware CDS for wireless ad hoc networks. This extended marking process aimed at both reducing the size of CDS and evenly distributing the energy consumption to all nodes in the network. In [11], the simulation results show that Wu's newer protocol allows the network to go through more number of CDS reconstructions, an indication that the protocol prolongs the operation of the network. However, similar to [3], this protocol also requires exchanges of extra messages between immediate neighbors. Based on this idea, we propose two versions of *T*ime-based Energy-aware Connected Dominating Set Protocols (TECDS), which is able to prolong the network operation, in this paper.

## 3. TIMER-BASED ENERGY AWARE CONNECTED DOMINATING SET PROTOCOL

Our Timer-based Energy aware Connected Dominating Set protocols (TECDS) have two phases: initiator election and CDS construction. In the first phase, a unique initiator is elected; in the second phase, the CDS is constructed rooted from the initiator. The energy level at each node is taken into consideration when electing the initiator and constructing the CDS. In Subsections III-B and III-C, we will elaborate each phase in detail.

### 3.1 Notation Definitions and Assumptions

Before introducing the TECDS protocols, we would like to present the following notations and their definition. They will be used in the following discussion as well as the protocol pseudo code.

A wireless ad hoc networks is represented as an undirected graph $G = (V, E)$, where $V$ is the set of all stations in the wireless ad hoc networks and $E$ is the edge set with $(u, v) \in E$ *if and only if* $u$ and $v$ are within each other's transmission range.

If $G$ is connected, a set $DS \subset V$ is called a *dominating set* if for every vertex $v \in V - DS$, there exists a vertex $w \in DS$ such that $(v, w) \in E$. A dominating set is said to be *connected* if its induced graph in $G$ is connected.

A node $u \in V$ is said to be in the state of *inDS*, covered (by *DS*), or uncovered (by *DS*) according to the following:

- *inDS*: if $u \in DS$;
- covered: if $u \notin DS$ and there is an edge $(u, v) \in$ E for some $v \in DS$;
- uncovered: if $u \notin DS$ and there is no edge joining u to any node in *DS*;

We assume that each node in the network has the same transmission range. Like every wireless network system, we assume each node to periodically broadcast beacon signal. Two types of beacon signals are used in the protocols: the regular beacon and the announce beacon. In the regular beacon, a node's MAC address, the status (i.e., uncovered, covered, or *inDS*), and a color value (used to detect if the initiator is still active) are includes in the header of the beacon. In the "announce" beacon, a node encodes those included in the regular beacon as well as the energy level and number of neighbors for its initiator (for possible election of new initiator) in the header of the beacon. Notice that a *broadDS* message is actually a regular beacon encoded with *inDS* status. The reason to introduce two different beacon formats is to reduce the overhead of the protocols. Since the announce beacon carries more information, it is larger than the regular beacon. The protocols is carefully designed so that the announce beacon is sent every *initMax* regular beacon periods.

### 3.2 Initiator Election

In TECDS, we are interested in creating the CDS with smaller size and containing nodes with higher energy level, so two criteria are used when the protocol picks the initiator: the number of neighbors and the energy level. Depending on the order of consideration for these two criteria, two different versions of TECDS, namely TECDS1 and TECDS2, are introduced. In TECDS1, the node with the most energy is picked as the initiator. In case when multiple nodes have the same energy level, the one with the most neighbors is picked as the initiator.

In TECDS2, the node with the most neighbors is picked as the initiator. When multiple nodes are found to have the most neighbors, the one with the highest energy level is then elected as the initiator. In both protocols, when multiple nodes have the same number of neighbors and the same energy level, the node with the minimum MAC address among them is picked as the initiator to break the tie.

The following is the pseudo code for the initiator election phase of the TECDS1 protocol.

The timers used in this protocol have initial value −1. A positive integer is assigned when a timer is started. The timer value will go down each beacon period. When the value reaches 0, the timer expires and the value stops at 0.

The initiator sends out *announce* message every *initMax* beacon periods. It will refresh *InitTimer* of other nodes which expires after *2*initMax* beacon periods. It is a soft state protocol in the sense that the expiration of *InitTimer* for nodes other than the initiator implies that the initiator leaves the WIRELESS AD HOC NETWORKS. The nodes will wait *2*initMax* to make sure all the *InitTimers* expire, then the initiator election process starts again. Additionally, the color at each node indicates Notice that the pseudo code of the initiator

```
/* node i in wireless ad hoc networks executes the following: */
Initialization :
                initiator(i) ←  MAXINIT
                status(i) ←  uncovered
                color(i) ←  0
                DSTimer(i) ← −1
                ODSTimer(i) ←  −1
                energy(i) ←  energy level of node i
                nbrNum(i) ←  number of neighbors for node i
                InitTimer(i) ←  initMax, start InitTimer

InitTimer expires
        if initiator(i) = MAXINIT then /* initial announcement */
                initiator(i) ← i
                announce(i, i, color(i), energy(i), nbrNum(i))
                InitTimer(i) ←  2 * initMax, start InitTimer
        else if initiator(i) = i then /* initiator is selected */
                color(i) ← color(i) + 1
                announce(initiator(i), i, color(i), energy(i), nbrNum(i))
                InitTimer(i) ←  initMax, start InitTimer
                /* refresh message*/
        else if initiator(i) ≠ i then /* re-elect initiator */
                status(i) ←  uncovered
                initiator(i) ←  MAXINIT
                color(i) ← 0
                wait for 2 * initMax
I               nitiator(i) ← i
                announce(initiator(i), i, color(i), energy(i), nbrNum(i))
                InitTimer(i) ← 2 * initMax, start InitTimer

Node i receiving announce(j, k, c, e, r) /* i ≠ j */
/*Compare energy, neighbor numbers, and MAC ID*/
        if energy(i) < e then
                initiator(i) ←  j
                energy(i) ←  e
                nbrNum(i) ←  r
                color(i) ←  c
                announce(initiator(i), i, color(i), energy(i), nbrNum(i))
                InitTimer(i) ←  2*initMax, start InitTimer
        else if energy(i) = e and nbrNum(i) < r then
                initiator(i) ←  j
                energy(i) ← e
                nbrNum(i) ←  r
                color(i) ←  c
                announce(initiator(i), i, color(i), energy(i), nbrNum(i))
                InitTimer(i) ←  2*initMax, start InitTimer
        else if energy(i) = e and nbrNum(i) = r and Initiator(i) > j then
                initiator(i) ←  j
                energy(i) ←  e
                nbrNum(i) ←  r
                color(i) ←  c
                announce(initiator(i), i, color(i), energy(i), nbrNum(i))
                InitTimer(i) ← 2*initMax, start InitTimer
```

Initiator Election Phase

election phase for TECDS2 is basically the same as the above except that the order of the criteria used for picking the initiator (i.e., number of neighbors and energy level) is reversed. In other words, the first two if statements after the Compare energy, neighbor numbers, and MAC ID" comment are changed to the following:

if nbrNum(i) < r
and
else if nbrNum(i) = r and energy(i) < e then

### 3.3 Time based Energy aware Connected Dominating Set Construction

After the election of the initiator, the initiator enters *DS* first. It broadcasts to its neighbor about its *inDS* status. A neighboring uncovered node becomes *covered* after receiving the message. Then the *covered* node calculates the $\Delta T$ according to the following formula if it still has *uncovered* neighbors and starts its *DSTimer*.

$$\Delta T = T_{max} \cdot \frac{1}{N_{uncovered}} \cdot \frac{1}{E} \qquad (1)$$

In Equation 1, the $N_{uncovered}$ is the number of uncovered neighbors and E is the energy level. This equation uses both the number of uncovered neighbors and *the energy level* at each node to compute $\Delta T$. It is obvious that nodes with more *uncovered* neighbors or higher energy level results in shorter defer time compared with nodes with fewer *uncovered* neighbors and lower energy level.

When the *DSTimer* expires, the node enters *DS* and broadcasts to its neighbor about its *inDS* status. For *inDS* node, the broadDS message is sent every beacon period. Again we utilize soft state technique to maintain the status of covered nodes. If a covered node does not receive a broadDS message for $\delta$t beacon periods, it implies that the dominator(s) have left. Notice that, since the size of the necessary information for our protocols is fixed, there is no need for any new control messages for the TECDS protocols. The entire messages mentioned in the pseudo code are in fact various types of beacon signal.

The pseudo code for the CDS construction phase of the TECDS protocol is presented in the following.

```
/* node i executes the following */
        Node i detects itself as initiator :
                status(i)  ←  inDS
                color(i)  ←  color(i) + 1
                broadDS(i, color(i))

        Receiving broadDS(j,c) :
                if color(i) < c then
                color(i)  ←  c
                if status(i) = uncovered then
                status(i)  ←  covered
                start CoveredTimer with $\delta$ t

    Node i in covered state :
            if ( i hears from two neighbors with large color difference )
                    status(i)  ←  inDS
                    broadDS(i)
            if ( i does not have any uncovered neighbor )
                    DSTimer(i) = -1
            else if ODSTimer(i) = −1 then
                    DSTimer(i)  ←  $\Delta$T, start DSTimer
                    ODSTimer(i)  ←  $\Delta$T
            else if ODSTimer <  $\Delta$T then
                    DSTimer(i)  ←  $\Delta$T, start DSTimer
                    ODSTimer(i)  ←  $\Delta$T
Node i in inDS state :
        if ( i does not have any covered neighbor ) and (i has at least one inDS neighbor) then
        status(i)  ←  covered
DSTimer expires :
        status(i)  ←  inDS
        broadDS(i)
CoveredTimer expires :
        status(i)  ←  uncovered
```

Connected Dominating Set Construction Phase

## 4. SIMULATION RESULTS AND ANALYSIS

### 4.1 Simulation environment and metric selection

We assume a link between two nodes only if their geometric distance is less than the wireless transmission range. In our simulation, the transmission range of a single station is normalized to 1 unit of distance. Random network topologies are generated by randomly placing nodes in 4 × 4 and 8 × 8 square grids of a two-dimensional simulation area. The value of x and y coordinate is uniformly distributed. The value of *Tmax*, *initMax*, and $\delta$t is chosen to be 100, 20, and 4 time units respectively.

20 different network topologies are randomly generated. The energy level at each node is generated in normal distribution with the average 7.0 and the variance 2.0. The performance of protocols is assessed by the average size of CDS, the average energy level of nodes in CDS, the minimum energy level of node in CDS, and the variance of minimum energy level of nodes in the CDS for both 4 × 4 and 8 × 8 with various nodal densities. Notice that when network topology is static, the minimum and average energy level of nodes in the CDS can be considered an indication of the lifespan of the CDS. Other than TECDS protocols, we implemented three other CDS protocols, namely Wu1 and Wu2's protocols in [2] [3], and Wan's protocol in [9]. These protocols are also implemented in the same environment with the same matrix which TECDS protocol has.

### 4.2 Performance evaluation

In Figure 1 and Figure 2, the x-axis is the size of network and the y-axis is the size of the resulting CDS from five different protocols. It is clear that TECDS1 and TECDS2 constantly generate the smallest CDS and Wu1 constantly generates the largest CDS among all protocols for both 4 × 4 and 8 × 8 grids. When the scale of the network size increases from 4 × 4 to 8 × 8 grids the performance difference of our energy−aware TECDS1 and TECDS2 is getting thinner. Although Wu2 and Wan significantly reduced the size of the CDS as oppose to Wu1, its CDS size is still 40 to 50% larger than the CDS generated by TECDS1 and TECDS2.

In Figure 3 and Figure 4, the x−axis is the size of network and the y−axis is the average energy level of nodes in the resulting CDS from five different protocols. We find that TECDS1 and TECDS2 are able to achieve approximately 20% higher average energy level of nodes in CDS than the others for both 4 × 4 and 8 ×8 grids by slightly increasing the CDS size from Wu2. On the other hand, Wu1 and Wan have the lowest average energy level among all protocols for both 4 × 4 and 8 × 8 grid. It is surprised to find that even though Wu2 considers energy level at each node, it does not improve much the average energy level as shown in Figure 3 and Figure 4 (merely 5% from Wu1).

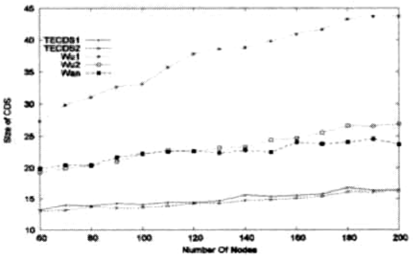
In Figure 5 and Figure 6, the x−axis is the size of network and the y−axis is the minimum energy level of nodes in the resulting CDS from five different protocols. From these figures we can see that our energy−aware TECDS1 and TECDS2 select the nodes with higher minimum energy level than others.

Wu1 selects the nodes having lowest minimum energy level among all others. These figures indicate that the CDS created by our energy−aware TECDS1 and TECDS2 live longer than any other protocols under static network. In 4 × 4 grids, the minimum energy level of nodes in CDS generated by TECDS2 and Wu1 are about 5.1 and 2.6 respectively. It means the minimum energy level of nodes in CDS generated by TECDS2 is mostly 50% higher than that by Wu1. For 8 × 8 grids, the performance of the minimum energy level for different protocols show similar trend.

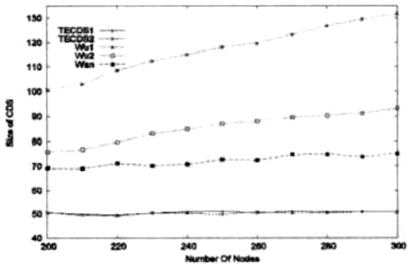
In closing, the results have shown that the proposed energy aware TECDS1 and TECDS2 produce better CDS in terms of the average/minimum energy level of nodes in CDS, and the number of rounds that represents the lifespan of the CDS and the network. In addition, the size of the CDS generated by TECDS1 and TECDS2 is constantly smaller than that by Wu1 and Wu2.

## 5. CONCLUSION

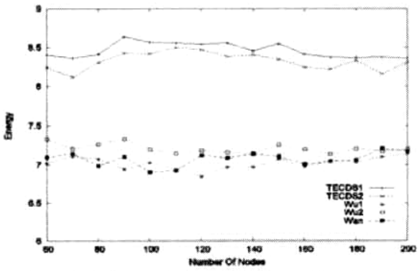
In this paper, we present two versions of *Timer−based Energy−aware Connected Dominating Set Protocols* (TECDS). Our protocols utilizes defer timer at each node to compute the CDS for wireless ad hoc networks. The energy level at each node is taken into consideration when constructing the CDS. The TECDS protocols effectively construct energy aware CDS that prolong the network operation under different level of nodal mobility. The simulation results have shown that our protocols constantly generate significantly smaller CDS for virtual backbone in wireless ad hoc networks without introducing extra messages than those proposed in [3] and [11]. Additionally, the CDS generated by our TECDS protocols consists constantly nodes with higher energy level which implies a longer lifespan of the CDS when network is static.



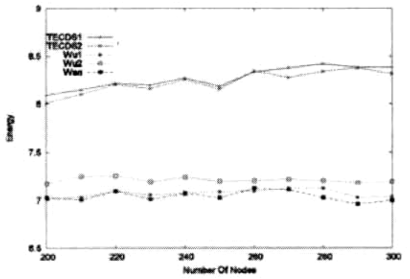
(Fig. 1) CDS size for 4 × 4 grids



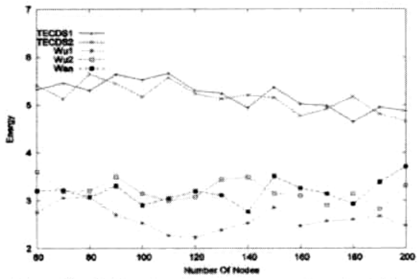
(Fig. 2) CDS size for 8 × 8 grids



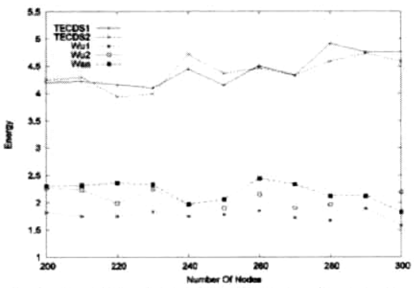
(Fig. 3) CDS Average Energy for 4 x 4 grids



(Fig. 4) CDS Average Energy for 4 x 4 grids



(Fig. 5) CDS Min Energy a in 4 x 4 grids



(Fig. 6) CDS Average Energy a in 4 x 4 grids

## REFERENCES

[1] S. Butenko, X. Cheng, D.Z. Du, and P. Pardalos, "On the construction of virtual backbone for ad hoc wireless networks," In Proc. 2nd Conference on Cooperative Control and Optimization.

[2] Jie Wu, Fei Dai, Ming Gao, and Ivan Stojmenovic, "On Calculating Power-Aware Connected Dominating Sets for Efficient Routing in Ad Hoc Wireless Networks," Journal of Communications and Networks, Vol 4, No 1, March 2002.

[3] J. Wu and H. Li, "A Dominating-Set-Based Routing Scheme in Ad Hoc Wireless Networks," special issue on Wireless Networks in the Telecommunication Systems Journal, Vol. 3, 2001, 63-84. 7.

[4] J. Wu, "Extended Dominating-Set-Based Routing in Ad Hoc Wireless Networks with Unidirectional Links," IEEE Transactiosn on Parallel and Distributed Computing, 22, 14, 2002, 327-340.

[5] Julien Cartigny, David Simplot, and Ivan Stojemenovic, "Localized minimum-energy Broadcasting in ad-hoc networks," IEEE INFOCOM 2003.

[6] M. Sun, W. Feng and T.H. Lai, "Location Aided Broadcast in Wireless Ad Hoc Networks," Proc. IEEE GLOBECOM 2001, pp.2842-2846, San Antonio, TX, November 2001.

[7] Mario Cagalj, Jean-Pierre Hubaux, and Christian Enz, "Minimum-Energy Broadcast in All-Wireless Networks: NP-Completeness and Distribution Issuses," Mobicom 2002,September,Atlanat,GA,USA.

[8] Y. Wang J. J. Garcia-Luna-Aceves, "Collision avoidance in multi-hop ad hoc networks," IEEE MASCOTS 2002, pp.145-154.

[9] P.J. Wan, K.M.Alzoubi and O.Frieder, "Distributed Construction of Connected Dominating Set in Wireless Ad Hoc Networks," IEEE INFOCOM 2002, pp.1597-1604.

[10] J. Wu, Fei Dai, Ming Gao, and Ivan Stojmenovic, "On Calculating Power-Aware Connected Dominating Sets for Efficient Routing in Ad Hoc Wireless Networks," Journal of Communications and Networks, Vol 4 ,No 1, March 2002.

[11] Hui Liu, Yi Pan, Ivan Stojmenovic: A Pure Localized Algorithm for Finding Connected Dominating Set in MANETs by Classification of Neighbors. WASA 2006: 371-381

[12] C.-K. Toh, "Maximum battery life routing to support ubiquitous mobile computing in wireless ad hoc networks," IEEE Commun. Maga., pp.138. 147, June 2001.

[13] N. Bambos, "Toward power-sensitive network architectures in wireless communications: Concepts, issues, and design aspects," IEEE Personal Commun., pp.50.59, June 2000.

[14] M. R. Garey and D. S. Johnson, "Computers and Intractability-A Guide to the Theory of NP-Completeness," San Francisco, CA: Freeman, 1979.

[15] Dorit S. Hochbaum, "Approximation Algorithms for NP-Hard Problems," PWA Publishing Company, Boston, MA, 1995.

### 김보남

e-mail : kimbona@chungbuk.ac.kr
1991년 단국대학교 전산통계학과(학사)
2003년 Auburn University Computer Science & Software Engineering (공학석사)
2006년 Auburn University Computer Science & Software Engineering (공학박사)
2006년~2007년 마음넷 주식회사 연구원
2007년~현 재 충북대학교 BK 사업단 연구원
관심분야 : MANET, WSN, Network Security 등

### 양준모

e-mail : junmo2.yang@samsung.com
1996년 경희대학교 경영학과(학사)
2001년 Auburn Univ. Computer Science & Software Engineering (공학석사)
2006년 Auburn Univ. Computer Science & Software Engineering(공학박사)
2007년~현 재 삼성전자 통신연구소 책임연구원
관심분야 : Ad hoc Network, Wireless Sensor Network (WSN), 광대역통신 등