Analysis of Link Error Effects in MANET Address Autoconfiguration Protocols

Sang-Chul Kim and Jong-Moon Chung

Abstract: **This paper focuses on message complexity performance analysis of mobile ad hoc network (MANET) address autoconfiguration protocols (AAPs) in reference to link errors generated by mobile wireless nodes. An enhancement was made using a proposed retransmission limit, S, to be computed for error recovery (based on the link error probability), to measure message complexity of AAPs in reference to the link error probability, P_e. The control procedures for the retransmission limit have been included for each of the AAPs. Retransmission limit control is critical for efficient energy consumption of MANET nodes operating on limited portable energy. O-notation has been applied to analyze the upper bound of the number of messages generated by a MANET group of nodes. The AAPs investigated in this paper are strong duplicate address detection (DAD), weak DAD with proactive routing protocol (WDP), weak DAD with on-demand routing protocol (WDO), and MANETConf. Each AAP reacts different to link errors, as each AAP has different operational procedures. The required number of broadcasting, unicasting, relaying, and received messages of the nodes participating in a single-node joining procedure is investigated to asymptotically calculate the message complexity of each AAP. Computer simulation was conducted and the results have been analyzed to verify the theoretical message complexity bounds derived. The message complexity of WDP was lowest, closely followed by WDO, based on the simulation results and analysis of the message complexity under nominal situations. The message complexity of MANETConf was higher than WDO, and strong DAD resulted to be most complex among the four AAPs.**

Index Terms: **Address autoconfiguration protocol (AAP), ad hoc network, link error, message complexity, node mobility.**

I. INTRODUCTION

Mobile ad hoc network (MANET) is a wireless network composed of mobile nodes that have self-organizing and routing capabilities to communicate with one another over multi-hop wireless links without any fixed communication infrastructure (e.g., base station). MANET applications are increasing, based on new services that apply wireless sensor networks and mobile communication multihop relay connectivity. The majority of wireless network protocols are based on Internet protocol (IP) networks, especially for interoperability with heterogeneous networks. In order to receive IP packets, the IP address of a mobile node must be properly set to belong to the subnet in which it is logically located. The IP address must not overlap with other nodes of the network. IP networks with mobile nodes face two major challenges. One challenge is to maintain an IP address and routing path suitable for communication with the network, even when the node moves to another network. This process needs to be executed quickly and with as low complexity as possible. Another challenge is to overcome the instability of the wireless channel the mobile node experiences. The routing protocol needs to quickly adapt to the changing network topology, due to node mobility. Since network topology is based on the addresses and positions of the mobile nodes, the longer the address reconfiguration takes, the routing path setup will be that much more delayed.

MANET nodes directly send data packets to a destination node through routes, therefore, nodes need to have up-to-date routing tables proactively set for all the nodes, or need to find routes on-demand [1]. In MANETs, IP addresses of nodes are checked to determine if the connection and identification of the mobile nodes are properly configured [2]. Therefore, it is essential for all nodes to be able to perform the operations required for configuration of unique addresses to execute proper routing of data packets in a MANET. Address autoconfiguration is an important issue in MANETs, since address pre-configuration is not always possible.

This paper is organized as follows. In the following subsections of Section I, an overview of address autoconfiguration protocol (AAP) is presented followed by a brief description of the performance analysis approach and issues regarding node mobility. Section II first presents a system model definition, then presents several lemmas and proofs that derive the message complexity of strong duplicate address detection (DAD), weak DAD, and MANETConf. Section III provides computer simulation results and performance analysis, and Section IV states the conclusion.

A. AAPs Overview

AAPs can be classified as either a stateless or stateful [3]. Dynamic host configuration protocol (DHCP) is the most representative example of a stateful protocol. When DHCP is applied, a DHCP server assigns IP addresses to unconfigured nodes and keeps the state of address information in an address allocation table. In stateless protocols, a node can select an address and verify its uniqueness in a distributed manner using an algorithm for DAD [4]. Using a DAD algorithm, an addressless node can determine if the selected address can be used. Nodes with assigned addresses can prevent other nodes from accidentally using their

Manuscript received May 21, 2007; approved for publication by Hussein Mouftah, Division III Editor, June 11, 2008.

This work was supported by the Korea Research Foundation Grant funded by the Korean Government (MOEHRD, Basic Research Promotion Fund, KRF-2008-331-D00502), the Information Technology Research Center (ITRC) program supervised by the Institute of Information Technology Assessment (IITA-2008-C1090-0801-0038) of the Ministry of Knowledge Economy, Korea, and the Seoul R&BD program.

S.-C. Kim is with the School of Computer Science, Kookmin University, 861-1, Chongnung-dong, Songbuk-gu, Seoul, 136-702, Korea, email: sckim7@kookmin.ac.kr.

J.-M. Chung is with the School of Electrical & Electronic Engineering, Yonsei University, Seoul, 120-749, Korea, email: jmc@yonsei.ac.kr.

Table 1. Acronym table [*: Variable].

Acronym	Message	Acronym	Message
AB	Abort	LS	Link state
AC	Address cleanup	NR	Neighbor reply
AD	Advertised	NQ	Neighbor query
AE	Address error	RP	Route reply
AL	Allocated	RQ	Route request
AO	Allocation	RT	Requester request
AP	Address reply	m*	DAD retry count limit
AQ	Address request	n*	Retry count limit
IR	Initiator reply	S*	Retransmission count limit
IQ	Initiator request	P_e*	Probability of link error

addresses [5].

Algorithms for DAD can be classified as strong DAD, weak DAD, and MANETConf [6]–[8]. In strong DAD, by sending out an address request (AQ) message, a node joining the MANET randomly selects an address and checks if the address is currently being used in the MANET. Based on address reply (AP) messages in response to the broadcasted AQ message, the node can detect if address duplication exists within the MANET [2]. Weak DAD is proposed by [7], where ad hoc routing protocols are used to detect address duplication, by modification of the routing protocol packet fields. As a stateful protocol, MANETConf [8] uses a mutual exclusion algorithm for a node to acquire a new IP address. If a requester wishes to acquire an IP address, the IP address should be approved by all nodes in a MANET. The acronyms of messages and nomenclatures of the key variables used in this paper for strong DAD, weak DAD, and MANETConf are summarized in Table 1.

In related research, Weniger and Zitterbart summarized the current approach and future directions of address autoconfiguration schemes in MANETs [5]. Jeong *et al.* [9] studied hybrid ad hoc IP address autoconfiguration. The authors of [10] proposed an IP address configuration for Zeroconf. Mohsin and Prakah [11] introduced an IP address assignment method for MANETs. Zhou and Mutka [12] investigated prophet address allocation for large scale MANETs. Weniger [3] proposed a passive autoconfiguration scheme for MANETs. In [6], the message complexity of AAPs are analyzed for errorless communication environments.

In the above mentioned papers, strong DAD, weak DAD, and MANETConf have been introduced as AAPs. To the best of the authors' knowledge, the message complexity of these protocols have not been analyzed against error events that might occur within the AAP execution process. This paper first presents some additional novel procedures that enable these AAPs (i.e., strong DAD, weak DAD, and MANETConf) to stably deal with error states. It also provides mathematical derivations and message complexity analysis when the MANET nodes experience link error events.

B. Performance Analysis Approach

Many factors influence MANET performance. Reduction of routing overhead is always a major concern, as it relates to power consumption of the mobile nodes and also takes up a significant portion of the very limited wireless channel resources. One essential measure of the quality of a MANET routing protocol is scalability to an increase in MANET nodes.

Message complexity is defined as the overhead of an algorithm measured in terms of the number of messages needed to satisfy the algorithm's request. Cao and Singhai use message complexity and synchronization delay to measure the performance of a mutual exclusion algorithm used to effectively share resources in distributed systems [13]. The authors of [14] use message complexity to statistically measure the performance of the cluster-based topology control (CLTC) protocol. In [15], the authors calculate storage complexity and communication complexity to analyze the scalability of various MANET routing protocols and introduce the routing overhead of periodically updated LS messages. A detailed investigation to derive the upper bound of the message complexity, considering erroneous link conditions, for MANETs has not yet been conducted. Therefore, in this paper, the upper bounds of the message complexity of the AAPs for MANETs are derived, based on the link error probability of the wireless mobile nodes.

To derive the upper bound of message complexity, the general methodology of [16] is applied, which uses a flowchart to analyze the time complexity of an image segmentation algorithm based on the recursive shortest spanning tree (RSST). In [17], it is pointed out that time complexity is one of the most important factors to use in comparing different algorithms.

Message complexity of MANET address autoconfiguration algorithms influenced by link errors is investigated in this paper based on the complexity analysis method of [16], where the method of adding the upper bounds of the time complexity measured at each step can be adapted in the proposed algorithm, since MANET address autoconfiguration algorithms are composed of a sequence of discrete distinctive procedures, where each step has its own message complexity. Therefore, by summing the message complexity measured at each step, the message complexity of a complex procedure can be calculated. Correspondingly, the method of adding the time complexity measured at each node to derive the time complexity, can be adapted in the proposed algorithm since MANET address autoconfiguration algorithms are composed of recursive procedures.

C. Node Mobility Issues

In regards to the mobility factor in MANETs, it is indicated in [18] that the rate of link failure, due to node mobility, is the main concern of routing in ad hoc networks. MANET nodes move around according to their mobility scenarios, while they perform routing procedures simultaneously. Many papers deal with mobility patterns and mobility-based frameworks. Alparslan *et al.* [19] propose a generalized random mobility model, classify the existing mobility models for wireless ad hoc models, and summarize the assumptions of the movement profiles. The authors indicate that the random mobility model, where a mobility model indicating the movements of nodes due to a random process, is appropriate to evaluate performance. Random mobility models describe the movement pattern of mobile hosts by consecutive random length intervals, called movement epochs. The velocity and direction for each epoch may or may not be correlated with the previous epoch values. The random waypoint model (RWP) is a widely used random mobility model used to measure analytically the performance of MANET routing algorithms [20]. The function of remoteness is used to define the relative mobility of nodes [21]. In [21], the authors propose a structure algorithm based on cluster topology, to reduce the far-reaching effects due to topological changes. Amid *et al.* investigate current mobility models used in simulation to evaluate the proposed routing algorithms. They found that the performance of ad hoc routing algorithms, such as the packet delivery ratio, control overhead, and data packet delay are significantly affected by the mobility models. The authors argue that the current mobility models are not fully compatible to simulate real-world movements [22].

In this paper, a generalized approach of link error probability is considered, in reference to these mobility modeling papers. The link error probability (P_e) is the same for all inter-node links within each MANET group. We use this type of approach based on the fact that different mobility models result in different error rates under specific conditions. Therefore, the performance evaluation would be dependent on the mobility model and mobile environment used in the computer simulation. The generalized approach of using link error probability provides a level of independence to any mobility model. Averaging the link error events to obtain the link error rate, and applying this value to the link error probability, this paper's results are directly applicable to complexity analysis of MANETs that use specific mobility models.

II. COMPLEXITY ANALYSIS

A MANET is represented as a graph $G(\boldsymbol{V}, \boldsymbol{E})$ where $\boldsymbol{V}$ is a finite nonempty set of nodes, which can be represented as $\boldsymbol{V} = \{V_1^G, V_2^G, \cdots, V_W^G\}$ where $|\boldsymbol{V}| = W$ and $\boldsymbol{E}$ is a collection of pairs of distinct nodes from $\boldsymbol{V}$ that form a link, that can be represented as $\boldsymbol{E} = \{E_1^G, E_2^G, \cdots, E_W^G\}$ [23].

Definition 1: In a MANET $P(\boldsymbol{V}, \boldsymbol{E})$, broadcasting an $address\ query$ (e.g., AQ message in strong DAD, LS, and RQ messages in weak DAD, or IQ message in MANETConf) message by a node is defined as a *trial*.

1. A *success trial* is defined as an event in which, after a node broadcasts an $address\ query$ message, it does not receive any AP message (e.g., AP message in strong DAD, AE in weak DAD, or negative IR message in MANETConf) within a specific period.
2. A *failure trial* is defined as an event in which, after a node broadcasts an $address\ query$ message, it receives at least one AP message, within a specific period.
3. A *successful address verification procedure* is defined from m consecutive $success\ trials$.
   (a) For a node to get a verified address, the node has to perform a sequence of m independent *trials*, where each *trial* has to become a $success\ trial$.
   (b) In strong DAD, m is defined as a positive number, greater than one ($m > 1$).
   (c) In weak DAD and MANETConf, since m is set to one ($m = 1$), the $successful\ address\ verification\ procedure$ is the same as a $success\ trial$.
4. An $address\ verification\ procedure$ including any $failure\ trial$ results in a *failure address verification procedure*.
   (a) In strong DAD, a $failure\ address\ verification\ procedure$ is composed of consecutive $x - 1$ times of $success\ trials$ and a $failure\ trial$ at the xth *trial* where $x = 1, 2, \cdots, m$.
   (b) In weak DAD and MANETConf, since m is set to one ($m = 1$), the $failure\ address\ verification\ procedure$ is the same as a $failure\ trial$.
5. A *session* is defined as a sequence of $successful$ and/or $failed\ procedures$. The maximum number of $procedures$ executed in the $session$ is limited by n in strong DAD, weak DAD, and MANETConf.
   (a) When computing the upper bound in strong DAD, the worst case of a $successful\ session$ is composed of $n - 1$ consecutive $failure\ address\ verification\ procedures$ and a $successful\ address\ verification\ procedure$ at the nth $address\ verification\ procedure$. A $failure\ session$ is composed of n $failure\ address\ verification\ procedures$.

When computing the upper bound in weak DAD and MANETConf, the worst case of a *successful session* is composed of $n - 1$ consecutive $failure\ trials$ and a $success\ trial$ at the nth $address\ verification\ procedure$. A *failure session* is composed of n $failure\ trials$.

In this paper, the most common flooding method is used to broadcast an $address\ query$ message, where every node retransmits an AQ message to all one-hop neighbors, whenever it receives the first copy of the $address\ query$ message [24]. Since each member node in a MANET will relay the $address\ query$ message initiated at node V_i, assuming that the duplicated packet discard scheme is applied, the maximum number of nodes relaying an $address\ query$ message is $N - 1$. Therefore, the maximum number of $address\ query$ messages broadcasted or relayed in the free tree is N. This can be represented as $O(N)$ in the case of an ideal errorless channel.

For a given link error probability of P_e, the retransmission count limit value S can be defined based on the network manager's desired setting, some optimal criteria, and/or the mobile node's priority. For a given link error probability, the average number of transmissions (N_T) required for successful reception is provided in (1). This can be used as a reference value for the retransmission count limit value S.

$$N_T = (1 - P_e)^{-1}, \text{ for } 0 \leq P_e < 1. \tag{1}$$

Since a link error can stop propagation of AQ messages, a node that experiences link errors needs to retry broadcasting the AQ message to its neighboring nodes. It is assumed that a node is able to learn of transmission failure using acknowledgments from the lower layers. Based on the detected link error probability, a network controller can set the retransmission count limit S to a desired value, then the maximum number of $address\ query$ messages broadcasted or relayed in the MANET can be represented as $SO(N)$. The AAPs can now be generalized in the following definition.

Definition 2: For a MANET with N nodes, $SO(N)$ is the upper bound of the maximum number of broadcast or relayed $address\ query$ messages to assist a node joining the MANET, based on a retransmission count limit of S selected in reference to the link error rate P_e.

Lemma 1: For a MANET routing tree with t nodes in the maximum length path, $SO(t)$ is the upper bound of the maximum number of unicast or relayed AP messages to assist a node joining the MANET, based on a retransmission count limit of S selected in reference to the link error rate P_e.

Proof: Since each member node in a path of $d(j, i)$ relays an AP message initiated by the AP source node, the maximum number of nodes relaying an AP message is $t - 2$, where the rule of discarding duplicated messages at a node is adapted, and node V_i does not relay an AP message. Therefore, the maximum number of AP messages unicasted or relayed in the free tree is $t-1$, where the message complexity bound would be $O(t)$, if the wireless link was errorless. When the link error probability is considered, the retransmission count of S is multiplied to $O(t)$. □

A. Strong DAD

The pseudocode of strong DAD (Algorithm 1) is used to derive the upper bound of the message complexity of the strong DAD protocol. To compute the upper bound of the message complexity, a scenario where a node experiences a $failure\ address\ verification\ procedure$ is considered. Since the procedure is composed of a total of $(m - 1)$ $success\ trials$ and a $failure\ trial$ at the mth $trial$, the message complexity of a $failure\ address\ verification\ procedure$ can be represented as $S(mO(N) + O(t))$. The following lemma is based on this.

Lemma 2: $S(mO(N) + O(t))$ is the upper bound of the maximum number of broadcast/relayed AQ messages and unicast/relayed AP messages when a node needs to verify its address in a MANET with the strong DAD protocol, based on a retransmission count limit of S selected in reference to the link error rate P_e, in an $address\ verification\ procedure$.

Proof: The $address\ verification\ procedure$ including a $failure\ trial$ at the mth $trial$ is composed of $(m - 1)$ $success\ trials$, which gives $S(m - 1)O(N)$ number of broadcasted or relayed AQ message based on Definition 2, and a $failure\ trial$ at the mth $trial$, which gives $SO(N)$ number of broadcasted or relayed AQ message based on Definition 2, and $SO(t)$ unicast or relayed AP messages, based on Lemma 1. Therefore, the message complexity of the $failure\ verification\ procedure$ can be represented as $S(m - 1)O(N) + SO(N) + SO(t)$, which sums the upper bound of the maximum number of broadcast, unicast, and relayed AQ and AP messages in $m - 1$ $success\ trials$ and a $failure\ trial$ at the mth $trial$. Rearranging $S(m-1)O(N)+SO(N)+SO(t)$ yields $S(mO(N)+O(t))$. □

Lemma 3: In a $session$, $nS(mO(N) + O(t))$ is the upper bound of the maximum number of broadcast/relayed AQ messages and unicast/relayed AP messages using the strong DAD protocol, based on a retransmission count limit of S selected in reference to the link error rate P_e.

Proof: Strong DAD has a $session$ and the maximum number of retries of the $address\ verification\ procedure$ is limited by n in the $session$. Since the $session$ consists of a maximum number n $address\ verification\ procedures$ and the upper bound of the maximum number of $address\ verification\ procedures$ is $S(mO(N) + O(t))$, based on Lemma 2, the message complexity of the $session$ can be represented $nS(mO(N) + O(t))$. □

Algorithm 1: Pseudocode of strong DAD AAP operations.

```
while strong DAD AAP do
    step 01: A node selects a temporary address and
             configures it as its network interface address;
    step 02: $n = 0$, initialization;
    step 03: $m = 0$, initialization;
    step 04: $n++$, (Increase the retry count ($n$) by 1);
    step 05: $m++$, (Increase the DAD retry count ($m$) by 1);
    step 06: The node randomly selects a source address and
             forms an AQ message for the address;
    step 07: The node broadcasts the AQ;
    step 08: if (all MANET nodes receive the AQ in the
             situation where there might be link errors in a
             MANET == TRUE)[$SO(N)$] then
        step 09: if (an AP arrives at the node before the timer
                 expires in the situation where there might be
                 link errors in a MANET == TRUE)[$SO(t)$]
                 then
            step 10: if (retry count $\leq n$) then
                step 11: goto step 4; [$nS\{mO(N)+O(t)\}$:
                         Strong DAD with Session] ;
            else
                step 12: goto step 17;
            end
        else
            step 13: if (DAD retry count $\leq m$ ) then
                step 14: The node replaces the source address
                         with its address, goto end of while ;
            else
                step 15: goto step 5;
                         [$S\{mO(N)+O(t)\}$: Strong DAD with
                         address verification]
            end
        end
    else
        step 16: goto step 7;
    end
    step 17: The node fails to get a source address
end
```

B. Weak DAD

The pseudocode of weak DAD (Algorithm 2) is used to derive the upper bound of the message complexity of the weak DAD protocol. In WDP, nodes periodically broadcast LS messages to inform other nodes of the network topology. In WDO, only when a source node needs to send data to a destination node where the source node does not have a route to the destination, the source node broadcasts a RQ message to find a route to a destination node and a node, which is the destination node or a node having a fresh enough route, unicasts a RR messages in response to the RQ message. When a node finds an address that is duplicated with an entry in its routing table, after investigating an address in a LS, RQ, or RR message, the node takes additional steps to inform other nodes of the duplicated address [7]. In such a case, the node that was already using the address will unicast an AE message to the node that has the duplicated address [9]. If a node does not find a duplicated address after investigating an address in a LS, RQ, or RR message, the node normally relays the LS, RQ, or RR message. The following lemmas can be derived based on the above specifications.

Lemma 4: In an address verification procedure, $S(O(N) + O(t))$ is the upper bound of the maximum number of broadcast/relayed LS messages and unicast/relayed AE messages when a node needs to verify its address in a MANET using WDP, based on a retransmission count limit of S selected in reference to the link error rate P_e.

Proof: The maximum number of messages occurs when the $address\ verification\ procedure$ results in a $failure\ trial$. Since, the $failure\ trial$ gives $SO(N)$ number of broadcast or relayed LS messages based on Definition 2, and $SO(t)$ unicasted or relayed AP message based on Lemma 1, the message complexity of the $failure\ trial$ can be represented as $S(O(N) + O(t))$. This sums the upper bound of the maximum number of broadcast and relayed LS messages and unicast and relayed AE messages, where S is introduced to consider the link errors in a MANET. □

Lemma 5: In a $session$, $nS(O(N) + O(t))$ is the upper bound of the maximum number of broadcast/relayed LS messages and unicast/relayed AE messages using WDP, based on a retransmission count limit of S selected in reference to the link error rate P_e.

Proof: WDP has a $session$ and the maximum number of retries of the $address\ verification\ procedure$ is limited by n in the $session$. Since the $session$ consists of n maximum number of $address\ verification\ procedure$ and the upper bound of the maximum number of an $address\ verification\ procedure$ is $S(O(N)+O(t))$, based on Lemma 4, the message complexity of the $session$ can be represented as $nS(O(N) + O(t))$, where n is the number of retry count of the $verification\ procedures$. □

In WDO, a node broadcasts or relays a RQ message and it can unicast a RP message, if it is a destination node based on the normal routing procedure. It unicasts an AE message when a node finds a duplicated address. Based on the above results, the following Corollaries that are similar to the WDP case are given.

Corollary 1: In an $address\ verification\ procedure$, $S(O(N) + 2O(t))$ is the upper bound of the maximum number of broadcasted/relayed RQ messages and unicasted/relayed RP messages and AE messages, when a node needs to verify its address in a MANET using WDO, based on a retransmission count limit of S selected in reference to the link error rate P_e.

Proof: The maximum number of messages occurs when the $address\ verification\ procedure$ results in a $failure\ trial$. Since the $failure\ trial$ gives $SO(N)$ number of broadcasted or relayed RQ messages based on Definition 2, and $2SO(t)$ unicasted or relayed RP messages and AE messages based on Lemma 1, the message complexity of the $failure\ trial$ can be represented as $S(O(N) + 2O(t))$, which sums the upper bound of the maximum number of broadcasted and relayed RQ and unicasted and relayed RP and AE messages. □

Algorithm 2: Pseudocode of weak DAD AAP operations.

```
while weak DAD AAP do
    step 01: A node selects a temporary address and
             configures it as its network interface address;
    step 02: $n = 0$, initialization;
    step 03: $n++$, (Increase the retry count ($n$) by 1);
    step 04: The node randomly selects a source address and
             picks a unique key value (e.g., MAC address)
             as the identification of the node;
    step 05: if (Proactive routing protocol is used $==$ TRUE)
             then
        step 06: The node broadcasts a LS periodically ;
        step 07: if (all MANET nodes receive the LS in the
                 situation where there might be link errors in a
                 MANET $==$ TRUE)[$SO(N)$] then
            step 08: if (the node receives an AE for the
                     selected address in the situation where
                     there might be link errors in a MANET
                     $==$ TRUE) [$SO(t)$] then
                step 09: if (retry count $\leq n$) then
                    step 10: goto step 3;
                             [$nS\{O(N)+O(t)\}$: WDP,
                             $nS\{O(N)+2O(t)\}$: WDO]
                else
                    step 11: The node fails to get a source
                             address, goto end of while;
                end
            else
                step 12: The node replaces the source address
                         with its address, goto end of while;
            end
        else
            step 13: goto step 6;
        end
    else
        step 14: The node broadcasts RQ when needed;
        step 15: if (all MANET nodes receive the RQ in the
                 situation where there might be link error in a
                 MANET) [$SO(N)$] then

        else
            step 16: if (the node is the destination of a RQ)
                     then
                The node unicasts a $RP$ in the situation where
                there might be link errors in a MANET. [$SO(t)$];
            else
                step 17: goto step 8;
            end
            step 18: goto step 14;
        end
    end
end
```

Corollary 2: In a $session$, $nS(O(N) + 2O(t))$ is the upper bound of the maximum number of broadcasted/relayed RQ messages and unicasted/relayed RP messages and AE messages in WDO, based on a retransmission count limit of S selected in reference to the link error rate P_e.

Proof: WDO has a $session$ and the maximum number of retries of the $address\ verification\ procedure$ is limited by n in the $session$. Since the $session$ consists of n maximum number of $address\ verification\ procedures$ and the upper bound of the maximum number of an $address\ verification\ procedure$ is $S(O(N) + 2O(t))$ based on Corollary 1, the message complexity of the $session$ can be represented as $nS(O(N) + 2O(t))$ where n is the number of retry count of the $address\ verification\ procedure$. □

C. MANETConf

In order to derive the upper bound of the message complexity in MANETConf, the pseudo code of Algorithm 3 is used. When a node (which is a $requestor$) tries to join a MANET and to obtain a verified address, it broadcasts a NQ message to its neighbors. When the $requestor$ does not receive any NR messages before the neighbor reply timer expires, it repeats broadcasting the NQ message by a threshold number. After finishing the repetition, the $requestor$ decides that there is only one node and configures itself with the address. The $initialization$ procedure of MANETConf described above is not considered into the message complexity since the message complexity is focused on the procedures of a single node joining into a MANET group.

If the $requestor$ receives NR messages, the $requestor$ selects an $initiator$ and unicasts a RR message to the $initiator$. The message complexity of unicasting the RR message can be represented as $SO(1)$ where S equals one when the P_e equals zero. After receiving a RR message, the $Initiator$ broadcasts an IQ message to all nodes of the MANET group in order to verify the address of the $requestor$. The message complexity of broadcasting the IQ message can be represented as $SO(N)$ based on Definition 2. Recipient nodes will reply with an affirmative or a negative response through the IR message, to the $initiator$. The message complexity of unicasting the IR message by all nodes in the MANET group can be represented as $SO(tN)$, since all N nodes unicast IR messages and each IR message has message complexity $SO(t)$ based on Lemma 1. If the Initiator receives positive IR messages from all the recipient nodes, it broadcasts an AO message to all the recipient nodes of the MANET group. The message complexity of broadcasting the AO message can be represented as $SO(N)$, based on Definition 2. If the $initiator$ receives negative IR messages from the recipient nodes, it selects another address and repeats the step of broadcasting IQ and receiving IR messages until the retry count reaches the $initiator\ request\ retry$, set to n in this paper. The following lemma can be derived based on these results.

Lemma 6: In an $address\ verification\ procedure$ of a single node joining case, $SO((t + 1)N)$ is the upper bound of the maximum number of broadcast/relayed IQ messages and unicast/relayed IR messages, when a node needs to verify its address in a MANET with MANETConf, based on a retransmission count limit of S selected in reference to the link error rate P_e.

Proof: The maximum number of messages occurs when the $address\ verification\ procedure$ results in a $failure\ trial$. Since, the $failure\ trial$ gives $SO(N)$ broadcast or relayed IQ messages based on Definition 2, and $SO(tN)$ unicast/relayed IR messages based on Lemma 1, the message complexity of the $failure\ trial$ can be represented as $S(O(N) + O(tN))$. This sums the upper bound of the maximum number of broadcast and relayed IQ and unicast and relayed IR messages.

Algorithm 3: Pseudocode of MANETConf AAP operations.

```
while MANETConf AAP do
    step 01: A requester (new joining node) selects an initiator
             and unicasts RR to the initiator [$SO(1)$] ;
    step 02: $n = 0$, initialization;
    step 03: $n++$, (Increase the retry count ($n$) by 1);
    step 04: The initiator broadcasts an IQ to all the nodes of
             the MANET group with the address of the
             requestor;
    step 05: if (all MANET nodes receive the IQ in the
             situation where there might be link errors in a
             MANET == TRUE) [$SO(N)$] then
        step 06: Recipient nodes reply with an affirmative or a
                 negative response (IR) to the
                 initiator in the situation where there might be
                 link errors in a MANET [$SO(tN)$] ;
    else
        step 07: goto step 4;
        step 08: if (the initiator receives affirmative IR
                 messages from all nodes == TRUE) then
            step 09: The initiator assigns the address to the
                     requestor;
            step 10: The initiator broadcasts an AO message
                     to all recipient nodes of the MANET
                     group;
                     goto end of while [$SO(N)$];
        else
            step 11: The initiator selects another address;
            step 12: if (retry count $\leq n$) [$nSO((t+1)N)$] then
                step 13:The initiator sends an AB message to
                        the requestor [$SO(1)$];
                        goto end of while ;
            else
                goto step 3;
            end
        end
    end
    [$nSO((t+1)N)+SO(N)+SO(2)$: MANETConf]
end
```

This can be rearranged as $SO((t + 1)N)$. □

Therefore, the message complexity of broadcasting an IQ message and receiving IR messages until the retry count reaches n can be represented as $nSO((t + 1)N)$. After n repetitions, if the $initiator$ receives negative IR messages, it sends AB messages to the $requestor$. The message complexity of unicasting the AB message can be represented as $SO(1)$. Therefore, the message complexity of a single node joining can be represented as $nSO((t+1)N)+SO(N)+SO(2)$ where $SO(2)$ indicates the message complexity of unicasting RR and AB messages. The following lemma can be derived based on these results.

Lemma 7: In a $session$ where a single node joins, $nSO((t+1)N) + SO(N) + SO(2)$ is the upper bound of the maximum number of broadcast or relayed IQ and AO messages and unicast or relayed IR, RR, and AB messages in MANETConf to assist a node joining the MANET, based on a retransmission count limit of S selected in reference to the link error rate P_e.

Proof: MANETConf has a $session$ and the maximum number of retries of the $address\ verification\ procedure$ is limited to n in the $session$. Since the $session$ consists of a maximum n $address\ verification\ procedures$ and the upper bound of the maximum number of an $address\ verification\ procedure$ is $SO((t+1)N)$, based on Lemma 6, the message complexity of the $session$ can be represented as $nSO((t+1)N) + SO(N) + SO(2)$ where n is the number of $address\ verification\ procedures$, $SO(N)$ indicates the message complexity of broadcasting the AO message and $SO(2)$ indicates the message complexity of unicasting RR and AB messages. □

III. PERFORMANCE ANALYSIS

Computer simulation is used to analyze message complexity of the AAPs. The nodes are randomly distributed with uniform density in a network area of 1 km^2 based on an independent MANET environment with mobile nodes having no connection to an external network (such as the Internet).

The random node generator and simulator performance was verified (for 100, 125, 150, and 175 nodes) so that the average number of nodes per cluster, as well as several specifications in the adaptive dynamic backbone (ADB) algorithm of [14], matched the results in [14], with less than a 1% difference in most cases, performed by QualNet.

The conflict probability (P_c) is defined as the probability in which the address that a node requests to use is already being used in the MANET group. As the conflict probability approaches 1.0, the message complexity approaches the derived theoretical message complexity upper bound. The conflict probability depends on the size of the address and the number of nodes in a MANET group [3]. The authors of [3] calculate the conflict probability which is shown to be as high as 50% when an address space size of 16 bits is used as MANET local addresses in a network of 300 nodes. When the conflict probability approaches one, the selected (or reselected) joining node's addresses will almost always conflict with one of the addresses in the MANET group, resulting in maximum message complexity, obtained through the derivations in Section II.

It can be expected that in the simulation of WDO, having a different occurrence probability of unicasting a RP message at a certain conflict probability will result in a different message complexity value. Therefore, for simplicity, in the simulation experiments to follow, it is assumed that the occurrence probability of unicasting a RP message is the same as the conflict probability of the requested address.

In the computer simulation, P_e used are 0, 0.2, 0.25, 0.5, 0.75, and 0.8; P_c is fixed at 0.1. Corresponding to each of the P_e values of 0.2, 0.25, 0.5, 0.75, and 0.8, the retransmission count S has been set to 1.25, 1.33, 2, 4, and 5, respectively, based on (1).

The most common flooding method used in the simulation is to have every node retransmit an AQ message to all of its one-hop neighbors whenever it receives the first copy of the AQ message [24]. Dijkstra's shortest path algorithm at each node is used to calculate the number of hops in unicasting or relaying a unicast AP message from a destination node to a source node. The transmission range of the nodes changes the number of hops. The upper bound of the message complexity derived in Section II is compared to the simulation results, where the maximum number of nodes in a reverse path at each unicast case is used to calculate $O(t)$ in each upper bound equation.

Table 2. Simulation parameters [1].

Parameter	Value
Network area	1000×1000 m^2
Actual transmission range	100 m
S (retransmission count limit)	$(1-P_e)^{-1}$
m (DAD retry count limit)	1 (WDP, WDO, and MANETConf)
m (DAD retry count limit)	3 (strong DAD)
n (retry count limit)	5
P_c (conflict probability)	0.1
N (number of mobile nodes)	50 to 350
Simulation time	2 hours
Number of trials	20
Confidence level	95%

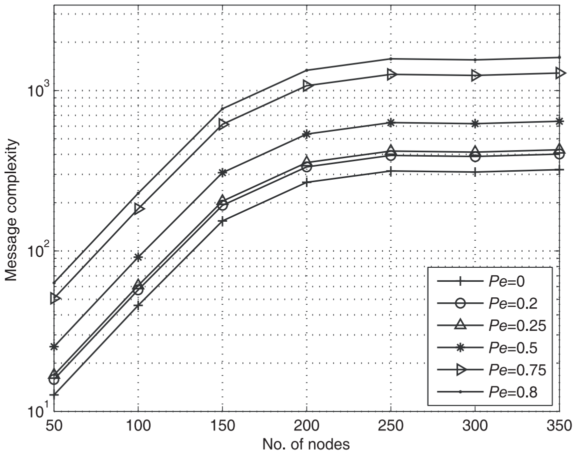

Fig. 1. Message complexity of strong DAD.

In the strong DAD protocol, the retry count limit (n) is set as 5, and the DAD retry count limit (m) is set to 3. In the weak DAD and MANETConf protocols, they are 5 and 1, respectively. The node transmission range is set to 100 m, where the number of mobile nodes varies from 50 to 350. Table 2 summarizes the parameters used in the strong DAD, WDP, WDO, and MANETConf computer simulations.

Figs. 1–4 illustrate the message complexities of strong DAD, WDP, WDO, and MANETConf protocols based on various numbers of mobile nodes communicating over an erroneous wireless links. The horizontal axis represents the number of nodes in the network area and the vertical axis indicates the number of messages for each case. As the conflict probability increases, it is shown that the number of messages to resolve the duplicated address also increases.

In Figs. 1–4, $P_e = 0$ indicates an errorless wireless communication environment. Based on Figs. 1–4, for a given wireless link error probability, the mechanism of re-broadcasting (e.g., $address\ query$) messages and re-unicasting (e.g., AP) messages result in different control signaling overhead for each

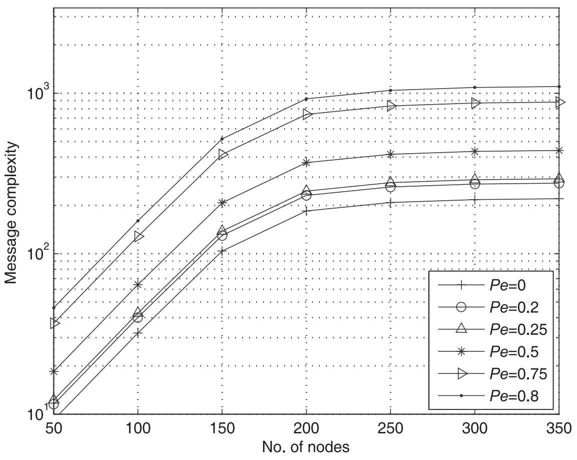

Fig. 2. Message complexity of WDP.

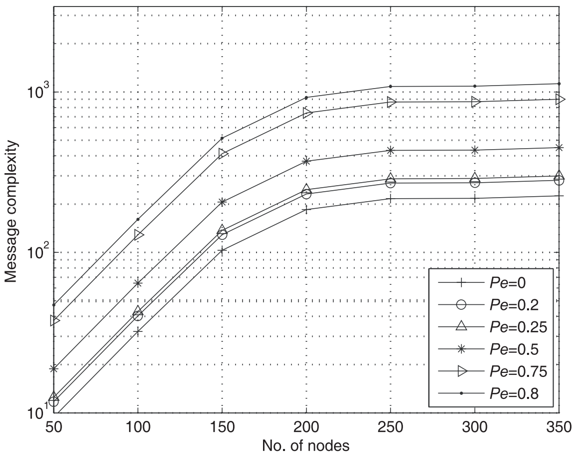

Fig. 3. Message complexity of WDO.

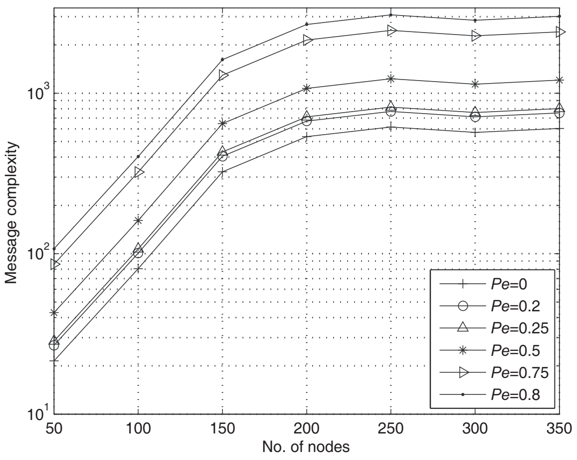

Fig. 4. Message complexity of MANETConf.

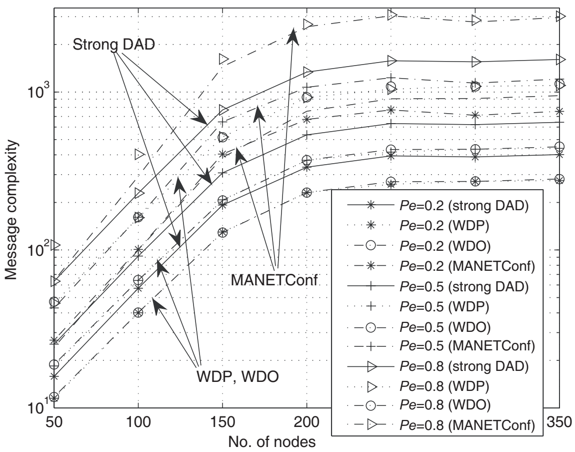

Fig. 5. Message complexity comparison of AAPs at link error probability $P_e = 0.2$, 0.5, and 0.8.

protocol. For various P_e values, strong DAD has approximately 45% more overhead compared to WDP. WDO has approximately 1.1% more overhead compared to WDP. MANETConf has 174% more overhead compared to WDP.

Fig. 5 compares strong DAD, WDP, WDO, and MANETConf when the link error probabilities vary from 0.2, 0.5 to 0.8. At each link error probability, WDP and WDO have the lowest message complexity and MANETConf has the highest message complexity. The message complexity of WDO is almost the same as WDP (without considering the proactive signaling messages).

IV. CONCLUSION

The wireless communication environment and the mobility of the nodes make the link unstable, which results in link errors. Node mobility results in nodes moving from one MANET group to another group, requiring address reconfiguration. The main objective of this paper is to perform a mathematical and computer simulation analysis of message complexity for MANET AAPs to quantize the effects of node mobility that results in address changes and link errors. The original publications on MANET AAPs are not equipped with procedures to deal with error events that occur during AAP operation.

Novel procedures to deal with error events had to be added to each AAP protocol. In each AAP protocol, the retransmission count (S) has been added to consider the link errors that occur in end-to-end wireless connections. This is similar to the reason why a retry count limit (n) is needed in the $session$ control of strong DAD. It helps eliminate the possibility of an infinite loop under certain conditions. The retransmission count (S) is effective in preventing too many transmission attempts from being executed when channel conditions are poor.

In effect, this supports two essential roles. First, if the channel condition is too poor, then energy of the mobile nodes should be saved by limiting the retransmission attempts. The AAP procedures can be reattempted later, when channel conditions improve. This is critical for MANET nodes that operate on limited battery energy. Second, incumbent mobile nodes of the MANET are conducting other tasks (e.g., various data transfers and route updates) in addition to the AAP procedures to

Table 3. Comparison of message complexity.

AAP	Message complexity
Strong DAD	$nS(mO(N) + O(t))$
WDP	$nS(O(N) + O(t))$
WDO	$nS(O(N) + 2O(t))$
MANETconf	$nSO((t + 1)N) + SO(N) + SO(2)$

assist a new node attempting to enter the network. Therefore, the processing power could be more effectively used to support ongoing MANET operations. If there is no limit to the consecutive attempts, then the AAP procedures alone could consume all the network resources. Therefore, this could also be used as a method of denial of service (DoS) attack against the MANET.

Table 3 summarizes the message complexity of a single node joining in strong DAD, WDP, WDO, and MANETConf. Since node mobility introduces significant challenges to network operations, such as, routing, resource management, and quality of service provisioning, this paper uses P_e to present disconnections of wireless links and dynamic network topology changes. The most appropriate retransmission count limit S can be decided from P_e and the priority of the newly joining mobile node. As the link error probability approaches one, the retransmission count approaches infinity.

Based on the simulation results and analysis of the message complexity, for nominal n, m, t, N, P_e, P_c, S values and transmission range, the message complexity compares as follows: WDP $<$ WDO $<$ strong DAD $<$ MANETConf. In regards to message complexity, if a MANET area has a high address conflict probability, weak DAD with MANET routing protocols becomes the most suitable protocol, compared to MANETConf and strong DAD. Weak DAD with MANET routing protocols execute routing path signaling and address autoconfiguration together, resulting in lower message complexity compared to MANETConf and strong DAD.

The authors feel that there is still more to be done before this area can be claimed sufficiently complete. The limitations of the proposed effort are summarized as follows. One limitation of the proposed research presented in this paper is that the message complexity was analyzed based on the upper bound performance. Instead, the message complexity could be analyzed using the average and other higher-order statistics to express the differences in performance more accurately among the AAPs. However, in the authors' point of view, this attempt would most likely require significantly more complex mathematics. If accomplished, the average and higher-order statistics and computer simulation results could serve as a more accurate guideline in complexity estimation and protocol design.

In addition, in this paper, the link error probability was assumed the same for all links within a given wireless area. This approach was applied to simplify the mathematical derivations, protocol design, and computer simulations. However, in reality the error rate of the links would be all different and time-varying. By averaging the link error events to obtain the link error rate, and by using the average error rate as the momentary link error probability, the results of this paper can be directly applied to complexity analysis. The authors intend to apply the analysis to various MANET mobility models and various channel environments. In future research, these two major factors (among many others) are seen as worth attempting.

REFERENCES

[1] Z. J. Haas and B. Liang, "Ad hoc mobility management with uniform quorum systems," *IEEE/ACM Trans. Netw.*, vol. 7, no. 2, pp. 228–240, Apr. 1999.

[2] C. E. Perkins, J. T. Malinen, R. Wakikawa, E. M. Royer, and Y. Sun, "IP address autoconfiguration for ad hoc networks," *IETF draft*, 2001.

[3] K. Weniger, "PACMAN: Passive autoconfiguration for mobile ad hoc networks," *IEEE J. Sel. Areas Commun.*, vol. 23, no. 3, pp. 507–519, Mar. 2005.

[4] S. Thomson and T. Narten, "IPv6 stateless address autoconfiguration," IETF, Tech. Rep. RFC 2462, Dec. 1998.

[5] K. Weniger and M. Zitterbart, "Address autoconfiguration in mobile ad hoc networks: Current approaches and future directions," *IEEE Network*, vol. 18, no. 4, pp. 6–11, July/Aug. 2004.

[6] S.-C. Kim and J.-M. Chung, "Message complexity analysis of mobile ad hoc network address autoconfiguration protocols," *IEEE Trans. Mobile Comput.*, vol. 7, no. 3, pp. 358–371, Mar. 2008.

[7] N. H. Vaidya, "Weak duplicate address detection in mobile ad hoc networks," in *Proc. ACM MobiHoc 2002*, Lausanne, Switzerland, June 2002, pp. 206–216.

[8] S. Nesargi and R. Prakash, "MANETconf: Configuration of hosts in a mobile ad hoc network," in *Proc. IEEE INFOCOM 2002*, New York, USA, June 2002.

[9] J. Jeong, J. Park, and H. Kim, "Auto-networking technologies for IPv6 mobile ad hoc networks," in *Proc. ICOIN 2004*, Busan, Korea, Feb. 2004, pp. 257–267.

[10] S. Cheshire and B. Aboba, "Dynamic configuration of IPv4 link-local address," The Internet Society, Tech. Rep. RFC 3927, Mar. 2005.

[11] M. Moshin and R. Prakash, "IP address assignment in a mobile ad hoc network," in *Proc. IEEE MILCOM 2002*, Oct. 2002, pp. 856–861.

[12] H. Zhou, L. M. Ni, and M. W. Mutka, "Prophet address allocation for large scale MANETs," *Ad Hoc Networks*, vol. 1, no. 4, pp. 423–434, Nov. 2003.

[13] G. Cao and M. Singhai, "A delay-optimal quorum-based mutual execution algorithm for distributed systems," *IEEE Trans. Parallel Distrib. Syst.*, vol. 12, no. 12, pp. 1256–1268, Dec. 2001.

[14] C.-C. Shen, C. Srisathapornphat, R. L. Z. Huang, C. Jaikaeo, and E. L. Lloyd, "CLTC: A cluster-based topology control framework for ad hoc networks," *IEEE Trans. Mobile Comput.*, vol. 3, no. 1, pp. 18–32, Jan.–Mar. 2004.

[15] X. Hong, K. Xu, and M. Gerla, "Scalable routing protocol for mobile ad hoc networks," *IEEE Network*, vol. 16, no. 4, pp. 11–21, July/Aug. 2002.

[16] S. H. Kwok and A. G. Constantinides, "A fast recursive shortest spanning tree for image segmentation and edge detection," *IEEE Trans. Image Process.*, vol. 6, no. 2, pp. 328–332, Feb. 1997.

[17] A. Boukerche, S. Hong, and T. Jacob, "An efficient synchronization scheme of multimedia streams in wireless and mobile systems," *IEEE Trans. Parallel Distrib. Syst.*, vol. 13, no. 9, pp. 911–923, Sept. 2002.

[18] A. B. McDonald and T. F. Znati, "A mobility-based framework for adaptive clustering in wireless ad hoc networks," *IEEE J. Sel. Areas Commun.*, vol. 17, no. 8, pp. 1466–1487, Aug. 1999.

[19] D. N. Alparslan and K. Sohraby, "A generalized random mobility model for wireless ad hoc networks and its analysis one-dimensional case," *IEEE/ACM Trans. Netw.*, vol. 15, no. 3, pp. 602–615, June 2007.

[20] E. Hyytia, P. Lassila, and J. Virtamo, "Spatial node distribution of the random waypoint mobility model with applications," *IEEE Trans. Mobile Comput.*, vol. 5, no. 6, pp. 18–32, June 2006.

[21] B.-J. Kwak, N.-O Song, and L. E. Miller, "A mobility measure for mobile ad hoc networks," *IEEE Commun. Lett.*, vol. 7, no. 8, pp. 379–381, Aug. 2003.

[22] A. P. Jardosh, E. M. Belding-Royer, K. C. Almeroth, and S. Suri, "Real-world environment models for mobile network evaluation," *IEEE J. Sel. Areas Commun.*, vol. 23, no. 3, pp. 622–632, Mar. 2005.

[23] J. Gross and J. Yellen, *Graph Theory and Its Applications.* Boca Ranton, FL: CRC, 1998.

[24] M. Sheng, J. Li, and Y. Shi, "Relative degree adaptive flooding broadcast algorithm for ad hoc networks," *IEEE Trans. Broadcast.*, vol. 51, no. 2, pp. 216–222, June 2005.

Sang-Chul Kim received the B.S. and M.S. degrees in Electrical Engineering from Kyungpook National University and Computer Science from Changwon National University in 1994 and 1998, respectively. He received the Ph.D. degree in Electrical & Computer Engineering from Oklahoma State University, U.S.A. in 2005. During 1994–1999, he worked in Samsung SDS as a System Engineer. He is now an Asistant Professor in the School of Computer Science at Kookmin University, Seoul, Korea.

Jong-Moon Chung is an Associate Professor in the School of Electrical and Electronic Engineering at Yonsei University, Seoul, Republic of Korea, since September 2005. He received the Ph.D. degree in Electrical Engineering from the Pennsylvania State University in 1999, and the M.S. and B.S. degrees in Electronic Engineering from Yonsei University, Seoul, Republic of Korea, in 1994 and 1992, respectively. From 1997 to 1999, he was an Instructor and Assistant Professor in the Department of Electrical Engineering at the Pennsylvania State University. From 2000 to 2005, he served as Director of the Oklahoma Communication Laboratory for Networking and Bioengineering (OCLNB) and Associate Professor in the School of Electrical and Computer Engineering at the Oklahoma State University. His research is in the area of mobile and ad hoc communication and networking. In 2008 and 2007, respectively, he received the Outstanding Professor Award and the Outstanding Teaching Award both from Yonsei University. In October 2005, he received the Regents Distinguished Research Award (U.S.A.) and in the same year September, he received the Halliburton Outstanding Young Faculty Award (U.S.A.). In 2004 and 2003, respectively, he received the Technology Innovator Award and the Distinguished Faculty Award both from the Oklahoma State University (U.S.A.). In addition, in 2003 he received the Top Gun Award (U.S.A.) and in 2000, he received the 'The First Place' Outstanding Paper Award at the IEEE EIT conference (Chicago, IL, U.S.A.).