IPv6 Lightweight Stateless Address Autoconfiguration for 6LoWPAN Using Color Coordinators

Hyojeong Shin, Elmurod Talipov and Hojung Cha
Department of Computer Science
Yonsei University
Seoul, Korea
{hjshin, elmurod and hjcha}@cs.yonsei.ac.kr

Abstract— **As resource-constrained network technology develops, such as wireless sensor networks, connectivity to an IP-based network has become an important requirement. Assigning the global unique address to network nodes is a prerequisite to the connectivity to the IP-based networks. Since conventional address autoconfiguration protocols require high network bandwidth and management cost, they are not suitable for wireless sensor networks. In this paper, we propose a lightweight address autoconfiguration mechanism for resource-constrained networks. The proposed algorithm uses three coordinators that assign geometric information to the network to remove the assumption that each node has location information. Each node gathers the hop distance from the coordinators and generates a unique address based on the location information. The proposed algorithm is implemented with real hardware, and the performance is evaluated. The result shows that the mechanism efficiently assigns unique addresses to sensor nodes.**

Keywords-component; IPv6; 6LoWPAN; Autoconfiguration; Wireless Sensor Networks

I. INTRODUCTION

As sensor network technology develops, connectivity to an IP-based network has become an important requirement. Various applications show the usefulness of a sensor network increases when the wireless sensor network (WSN) is connected to an IP-based network [1][2]. Application-based tools and web services provide interfaces to users showing detailed information from the sensor network and adjusting the network configuration. The 6LoWPAN standard [3] is a good example of recent efforts to connect WSNs to IP networks.

A node in an IP-based network is configured with an IP address, a netmask and a default gateway. The dynamic host configuration protocol (DHCP) [4] may help a node to be configured with a unique address in the local network. Stateful autoconfiguration approaches such as MANETconf[5], Boleng's protocol[6], Prophet Allocation protocol[7] and Buddy protocol[8] use the common or disjoint distributed address allocation table to manage address assignment. Centralized control and managing an allocation table are not suitable for the sensor network since most functionality requires large memory space and a reliable link-layer for broadcasts and multicasts.

Protocols using stateless approaches allow nodes to select an address by themselves and verify its uniqueness in a distributed manner. The Zeroconf working group [9] of the Internet Engineering Task Force (IETF) uses the duplicate address detection (DAD) procedure to configure hosts in the absence of a server dedicated to managing the allocation table. However, link-level broadcasts in the ad-hoc network are not guaranteed to reach all nodes. So duplicate address detection of Zeroconf is not feasible, and the protocol is heavy. Aware of the benefits of the stateless approach, Perkins et al. [10] proposed query-based DAD (QDAD). While an address request (AREQ) message is flooded, the intermediate nodes establish a reverse path to allow other nodes to unicast an AREP back to the originator. Although this approach efficiently reduces the DAD cost, the total DAD procedure is still heavy. The weak DAD (WDAD) [11] and the passive DAD (PDAD) [12] diagnose the address conflict while the network stack operates, instead of guaranteeing the uniqueness of the address before the stack runs. They reduce the overhead of the DAD procedure. However, latency and liability should be considered as tradeoff.

Autoconfiguration of the resource-constrained network is intended to assign link-local unique IP (UIP) addresses to nodes within a single autonomous network. Also, the autonomous network can be connected to an IP-based global network. Following the concept of the IPv6 address scheme, the network ID (NID) for a single autonomous network is assumed to be globally unique. Thus, the composition of the local IP and the network ID is globally unique.

In order to decrease packet overhead, the network stack may use a 16-bit address scheme to communicate with each other within the autonomous network. Adapting the 16-bit address scheme, the in-network networking protocol uses lightweight routing protocols. Not only are many routing protocols developed and optimized for a low-bandwidth network, but they also concern the energy cost of the network. For out-bound networking, the network stack uses the full version of the address scheme. This two-layer address scheme provides both global uniqueness for outgoing communication and a brief address format for in-bound communication.

978-1-4244-3304-9/09/$25.00 ©2009 IEEE

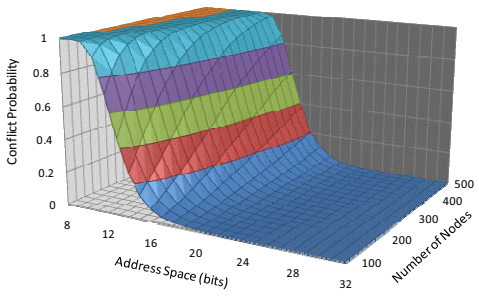

Figure 1. Address conflict rate

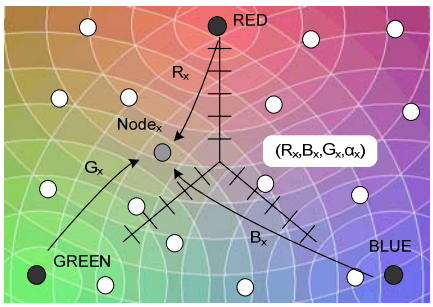

Figure 2. Algorithm of the three-axis method

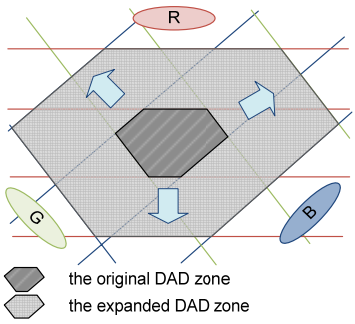

Figure 3. Expanding the DAD zone after address conflict

In this paper, we propose an address autoconfiguration protocol that quickly assigns network identification and reduces the network cost. The proposed algorithm limits the flooding range of the DAD procedure. The algorithm performs pseudo-localization and assigns unique labels to special regions. Using geometric information, the sensor nodes perform the DAD procedure only within the region, and the cost of the DAD is limited to the bounds of the region. The main goals of the proposed algorithm are as follows: The procedure should be independent from any other components such as network protocols or applications. Unaware of routing protocols, autoconfiguration is performed before applications are executed. The procedure should be completed in constant or reasonable time. The protocol is totally distributed and performed concurrently in each node. Time-consuming procedures are limited, and the protocol should be performed at low cost. The protocol should reduce management cost. The uniqueness of the address should always be guaranteed during the lifetime of the network. Entering the network, a new node can generate a unique address after local analysis.

The rest of the paper is organized as follows. In Sections II and III, an autoconfiguration method and an alternative approach are proposed. Section IV discusses the implementation of our solutions. Section V evaluates the feasibility of the proposed protocols. Section VI discusses related work and previous approaches. Finally, we conclude the paper in Section VII.

II. LIGHTWEIGHT STATELESS ADDRESS AUTOCONFIGURATION

A. Background

A random address selection in a 128-bit address scheme may successfully assign a unique address to sensor nodes. However, the size of the standard message header with a 128-bit address in 6LoWPAN is assumed to be big, and many header compression techniques [3] cannot be fully applied. A 16-bit address scheme helps to reduce network overhead and allows the use of many robust network stacks for a traditional 16-bit address scheme. Although a 16-bit address scheme is compact, the duplication rate of the addresses increases.

$$P(E_c) = 1 - e^{-n}\left(1 - \frac{n}{r}\right)^{n-r-\frac{1}{2}} . \quad (1)$$

The probability of an address conflict is evaluated as (1) where an address is randomly selected; the number of nodes and the size of the address field are denoted as n and r, respectively. This probability is analyzed by PACMAN [12]. As the size of the address decreases, the duplication rate increases. Fig. 1 shows that the conflict probability is related to the size of the address's space and the network. The conflict probability in a 16-bit address scheme is much higher than that of a 128-bit address scheme. In the case of a 16-bit address space, which is used as a link-layer address, the conflict probability is as high as about 50% in a network of 300 nodes. Even a single duplication needs the DAD procedure, and the DAD procedure overwhelms network communication with a high autoconfiguration cost.

B. Color Tagging Method

The procedure assigns a unique color coordinate to a certain area and assigns a unique address to a node according to the color coordinate. The algorithm deploys three color coordinators: RED, GREEN, and BLUE. Each node in the network measures the distance from the coordinators and generates a random value, alpha. The final unique address of the sensor node is a color value (R, G, B, alpha). Fig. 2 shows the outline of the proposal.

Since the hop distance is not continuous, a zone that is represented by the same color is found. The zone is denoted by a monochrome zone in this paper. Monochrome zones are shown in Fig. 3. As a monochrome zone may have multiple nodes with the same color, the alpha channel distinguishes each node in the zone. To detect duplication of addresses, each node checks the sensor nodes in the same monochrome zone. The zone is generally smaller than the one-hop range. So, DAD messages only need to be shared between one-hop ranges. This feature dramatically reduces the DAD cost.

C. Address Conflict Rate and Treatment

The sensor network is divided into a set of monochrome zones in which address assignment and conflict detection are performed. An address conflict with the alpha channel is related to the number of nodes in the zone, which is the physical density of the sensor nodes, not the total number of sensor nodes in the network.

Each node broadcasts DAD messages containing its address in the monochrome zone. All nodes in the zone receive the

DAD messages from neighbor nodes and can detect address conflict. After a node detects address conflict, the node generates its address and broadcasts again. To prevent recursive conflict, the node expands its monochrome zone, which is the address range. As the address range increases, the conflict rate decreases, and the DAD cost increases. Fig. 3 shows the expansion of the monochrome zone. Initially, the monochrome zone is assigned a single (R, G, B) tag. After the conflict, each coordinate is expanded, and the zone is composed of multiple colors. Since the expanded zone embeds monochrome zones with a few nodes, the conflict rate decreases. If a node detects a conflict again, the node repeats the same procedure. The cost of the repeated DAD procedure is limited, because only the address conflict detected nodes repeat the DAD procedure and the conflict rate decreases due to zone expansion.

New nodes may join the network after the auto-configuration process is completed. Although the coordinators do not generate color tags for new nodes, the node can gather color tags from neighbor nodes and estimate the tag for the node. Gathered tags in one hop neighbor nodes have small variation since tag is built by hop distance from the coordinators. With a simple estimation averaging RGB factors of gathered tags, the new nodes also reduce the size of DAD zone. DAD process with the estimated tag is same with the regular DAD process during the autoconfiguration.

D. Deployment Policy of the Coordinators

Coordinators are static nodes that are the originators of the RGB coordinates. Coordinators broadcast the coordinates of a packet once in the initialization phase of the network. These coordinators form small monochrome zones and help the DAD zone to be small. Hence, the deployment policy of the coordinators mainly affects the performance of the proposed algorithm. The proposed algorithm settles three coordinators in a triangle formation, which is one of the ideal deployments. In the geometric scope, the monochrome zones form ideally on the near position of the coordinators and inside the triangle of the coordinators. Guidelines for coordinator formation are as follows: deploying three coordinators as far as possible from each other and deploying coordinators on the boundary of the network.

A simple deployment policy is proposed to make up for the weak points of manual deployment that require human input. The sink node of the network selects three nodes, R, G and B, among the neighbor nodes and broadcasts the initialization message to the network. With the message flooding, the selected nodes delegate the selection to one of their child nodes. After the flooding procedure is completed, the final selected nodes are coordinators. Although the algorithm is simple, it is an advantage in terms of power consumption and completion time.

E. Mobility Support

The size of the DAD zone needs to be adjusted where nodes are mobile. Mobility increases the size of the DAD zone. Each node increases the size of the node's DAD zone to avoid applying the same value to all nodes in the network. A node can detect mobility based on the information of the neighbor nodes: a change in the information of a neighbor node and the hop-distance gap of a neighbor node. A gap in the hop distance of a neighbor node gives a hint how much the DAD zone should be expanded. A node detecting mobility decides how far to broadcast the DAD messages according to the degree of mobility.

The DAD zone expands in proportion to the velocity of the mobile nodes and the address configuration latency. The DAD zone should be expanded according to (2), since the DAD zone should cover distance mobile nodes moved. The address configuration latency will be discussed in the Evaluation section.

$$\text{Expanded DAD zone} = \text{DAD zone} + \frac{R_{radio}}{v_{max} \times T_{autoconf}}. \quad (2)$$

Conversely, the speed of mobile node is limited by (3).

$$\frac{R_{radio}}{\Delta\text{DAD zone} \times T_{autoconf}}. \quad (3)$$

F. Features of the Proposed Algorithm

Coordinators can simultaneously broadcast location messages. These messages are independent of each other. Without the influence of network bandwidth and message collision, the broadcasting location information of the coordinators is related only to the network size. After receiving the location information, a node is able to generate its address. This address assignment phase is performed in constant time.

The proposed algorithm uses distance information from three coordinators as the location identification. The algorithm assigns a unique label to the position where sensor nodes are located and performs the DAD procedure. Generally, a label on the position, denoted by the *monochrome zone*, is smaller than the one-hop radio range because the zone is divided by hop count. The network node is able to detect the address conflict within a small area, the *monochrome zone*. This feature dramatically reduces the DAD cost, while DAD messages in the IPv6 standard are flooded to the whole network.

The network is divided by radio range and labeled up to a value of 15 in each color. Distance over 15 hops is assumed to be 15 hops. Under a large network over 15 hops by 15 hops, nodes on the boundary have the maximum hop count. Address conflicts are easily detected, and DAD zone adjustment is required. Total DAD cost on the large network increases. Although the proposed method prefers a small network, this limitation is endurable, because the network size is assumed to be short in the IEEE 802.15.4 network. For example, a routing protocol, RIPng, which supports IPv6, is limited to networks whose longest path is 15 hops [13].

After labeling, tiny monochromes are found because the partitions are not evenly divided. Some monochromes do not exist, such as (2,2,2) in Fig. 2. Some monochromes are too small to include network nodes. These monochromes make the algorithm wasteful. Some address zones of these wasted monochromes cannot be assigned to network nodes. However, these addresses are available in the additional DAD phase after address conflict.

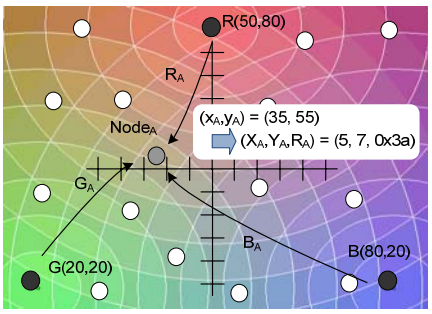

Figure 4. Algorithm of two-axis method

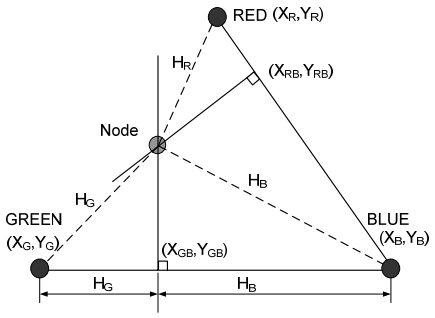

Figure 5. Calculating a node's position

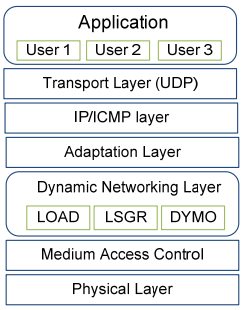

Figure 6. System architecture

III. ADVANCED TAGGING METHOD

The Color Tagging Address Assignment has a limit on flexibility. Encoding from a three-axis location tag to a flexible tag increases address utilization. The advanced tagging address assignment is proposed for assigning monochrome zones. The method performs a virtual-hop-based localization and encodes the address base on the two-axis geometric information.

A. Localization

Virtual-hop-based localizations are proposed to calculate the positions of sensor nodes in previous research [14][15]. Virtual-hop-based localization measures distances from some coordinator nodes in the hop count and calculates the position of the sensor nodes. The localization result is affected by the deployment of the sensor nodes because the hop count does not reflect a strict distance. Since the proposal adapts this position information to divide the network field into DAD zones, errors from virtual-hop-based localization can be ignored. The proposal performs a pseudo-virtual-hop-based localization, obtains the DAD zone information of the sensor nodes and generates a random value that is unique within the zone.

The advanced approach has also three coordinators that broadcast their position messages. After receiving the messages, the sensor nodes calculate their position by the trilateration method. The sensor network is divided by position data in a grid. The sensor nodes generate a random value to be distinguished from neighbor nodes in the partition. While the previous approach assigns a tag in the hop distance, the advanced approach assigns a tag on the x and y coordinates. The visualized process is shown in Fig. 4.

Assuming that the hop distance between sensor nodes reflects the physical distance, the trilateration method calculates the position of the node based on the hop-distance information. To avoid converting the hop count to a distance based on two-dimensional coordinates, the approach uses the rate of the hop count, H_r, H_g and H_b, shown in Fig. 5. Previous research [16] showed that at least four coordinators are required to track the location with rational distance. However, the proposal performs a localization procedure with only three coordinators because the proposal uses only rough location information and deploys fewer coordinators to reduce the managing cost. With lack of fourth coordinator, we use more simple and errant method to calculate a node's location which is shown in Fig. 5. It is proven that this simple method has error but it does not have a bad effect upon the autoconfiguration process, in the Evaluation Section.

The size of the address is 16 bit. The fields of x and y are 4 bits, and the field for the random field is 8 bit by default. So, the sensor field can be divided into 16 by 16 grids.

B. Features of the Advanced Tagging Method

The advanced approach divides the network into 16 by 16 partitions. The size of the grid is flexible, and the scale of the coordinates is related to the size of the network. As the size of the network increases, the size of the grid increases. While the three-axis RGBa coding technique wastes lots of address labels, the advanced technique fully uses the address labels.

As the DAD procedure performs within the grid, the cost of DAD is related to the number of nodes in the grid, which is the DAD zone. Because the number of DAD zones is fixed, the DAD cost is related to the number of nodes in the network. Although the two-axis method reduces the total DAD cost, the cost in the big O notation is still heavy compared to the three-axis method.

The advanced technique is not concerned with obtaining the exact location but the rough position of the sensor node, which is an address zone. Although previous MCL-based localization was concerned with resolution and localization error, the proposal is concerned with the number of nodes in the zone and the cost of the DAD procedure.

IV. IMPLEMENTATION

The proposed algorithms have been implemented in the RETOS [17] operating system. RETOS is a multithreaded operating system and provides the commonly used thread model of programming interface to developers. Fig. 6 shows the IPv6 over low power wireless personal area network (6LoWPAN) protocols of RETOS divided into six layers: the transport layer, the IP/ICMP layer, the adaptation layer, the dynamic networking layer (DNL), the medium access control (MAC) layer and the physical layer. The RETOS network stack provides a flexible layered architecture to support diverse applications, routing protocols and various network functionalities including autoconfiguration techniques. The proposed algorithm is implemented in the ICMP layer.

The advanced approach assumes that the coordinators have their position information. Basing on these anchors, nodes in the network calculate their relative position. The accuracy of

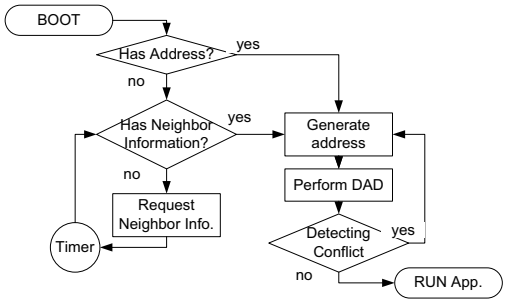

Figure 7. Boot sequence of autoconfiguration

the localization process is not critical to perform the auto-configuration process, since the position information is only used for dividing the network in grid. With lack of the position information, the coordinators estimate their position in wide-spread. Assuming the coordinators locate in ideal position that the coordinators locate at the border of the network and in triangle shape the estimation is adaptable. Error tolerance towards errant localization result is evaluated in Evaluation section.

A. Procedure

RETOS is a general-purpose operating system and manages multiple applications. The operating system initializes system devices and prepares system functionalities to satisfy the diverse requirements of applications. Since the physical address of a network node is essential for network communication, the system performs autoconfiguration before applications and routing protocols run. Fig. 7 shows the system boot sequence. A configured node runs applications after the DAD procedure. A newbie node tries to get the tag information from the neighbor nodes and performs the autoconfiguration procedure. If the network does not have tag information, the nodes in the network wait until the tag information is broadcast.

B. Message Format

TABLE I. MESSAGES FOR AUTOCONFIGURATION

MSG.	Table Column Head		
	Semantic	*Source*	*destination*
CS	Look for Network ID	N/C[a]	broadcast
CA	Inform Network ID including RGB information	N/C or IP address	broadcast
NS	DAD message	IP address	broadcast
NA	Inform Conflict of Address	IP address	unicast or broadcast

a. Not Configured

Message types of the proposed algorithm are categorized as standard 6LoWPAN ICMP message types. The messages are shown in Table I. The location information from coordinators, the Color Advertisement (CA) message, is carried out from a pre-configured coordinator node to non-configured nodes. The message is generated by coordinator nodes only, and non-configured nodes broadcast the message without a source address. The nodes forward the message a single time with the same sequence number of the message. The Color Solicitation (CS) message is used by non-configured nodes that enter the network after the configuration phase to query the coordinator information. The newbie nodes gather the coordinator information from neighbor nodes and generate their addresses. The NS and NA messages are informing messages for address assignment and duplication.

The protocol provides a manual configuration zone that allows a network administrator to deploy manually configured nodes. The total address zone is composed of the manual configuration zone, pre-allocated coordinators addresses and autoconfiguration zone. Since the address format is in the form of (RED, GREEN, BLUE, alpha) and the manual addresses can be configured by zero hop in distance vector.

C. Fault Tolerance

Packet loss of system messages causes a permanent wait of the protocols. A node that does not receive a message may wait for the message forever. This problem can be solved by periodic requests with the CS message. Non-configured nodes may request coordinator information periodically until every information datum is ready.

The hop distance is used only for dividing the network. So hop-distance error can be ignored when the DAD zone covers the error rate of the hop-distance data. In the implementation, the DAD zone starts with a two-hop range, and it shows adaptable overhead in the experiment.

D. Implementation Footprint

We measured the code size and memory footprint of the proposed protocol to evaluate code complexity. Table II shows that the proposed protocol produces reasonable overhead. Code size increased by 3.33% and memory size increased 0.92% for coordinator motes, and 3.34% and 0.93% for other motes.

TABLE II. CODE SIZE AND MEMORY FOOTPRINT

	Coordinator Motes		Other Motes	
	Code Size (byte)	*Memory Footprint (bytes)*	*Code Size (bytes)*	*Memory Footprint (bytes)*
RETOS Kernel (v1.4)	27964	2018	27964	2018
6LoWPAN Module	7324	602	6716	538
Proposed protocol	1174	24	1194	24
TOTAL	36462	2644	35874	2580

Furthermore, the code complexity of the proposed protocol with the 6LoWPAN module alone is small enough to be implemented on various sensor devices.

V. EVALUATION

The proposed algorithms are simulated and implemented to evaluate feasibility and performance. Although the proposed algorithms are implemented in a real environment and performed, the algorithms are evaluated in the simulator for fine-grained configuration. The parameters in the simulator reflect the hardware specification of the Tmote Sky mote[18].

A. Simulation

1) Three-axis Method

The basic algorithm generates the IP address based on three-axis coordinate data and generates an IP address. After receiving all coordinator information, each node quickly

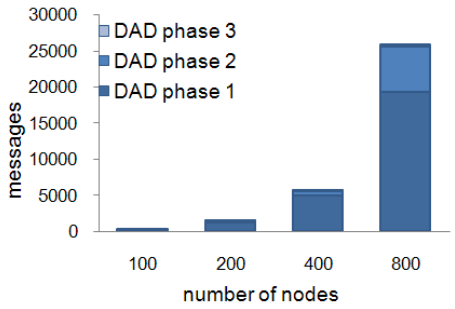

Figure 8. DAD cost of three-axis Method

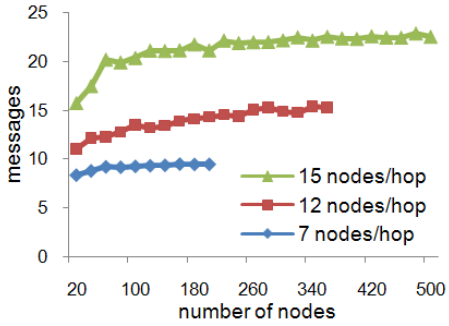

Figure 9. DAD cost of three-axis Method

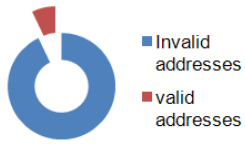

Figure 10. Valid addresses on the visualized monochrome plane

generates its address and then performs duplicate address detection. Fig. 9 shows the network cost during the DAD phases. In the DAD phase, the node broadcasts the NS message to its neighbor nodes within the bounds of the minimum DAD zone, denoted as DAD phase 1. If a node detects the address conflict by receiving the NA message, the node increases its DAD zone and allocates a new address, which is denoted as a DAD phase *n* in Fig. 8. With a fixed size of the network, we increased the number of network nodes and measured the DAD cost. As shown in Fig. 9, as the number of nodes increases, the address conflict also increases. Although the message overhead of the extended DAD phase is big, the number of nodes in the extra DAD phase decreases. In the view of a single node, this cost is endurable. To evaluate the relationship of DAD cost and network size, we fixed the average distance between nodes, which is network density, and increased the size of the network. Fig. 9 shows that the network cost per node is stable as the network size increases to 500 nodes. Meanwhile, the cost increases as the number of nodes in the neighborhood increases.

One of the limitations of the three-axis method is the waste of the address pool. Fig. 10 shows the size of the addresses shown on the visualized plane. About 93% of the address pool is hidden during the first address allocation phase due to the location of the coordinators. Since the coordinators are on the boundary of the network, the network nodes are located on one side of the coordinators' address space. Also, most of the monochromes are too small to have nodes or invalid to be configured. These unused addresses are utilized during additional address assignment after address conflict. They reduce the conflict probability of the additional DAD procedure. Also, small monochromes reduce the conflict probability of each network node.

2) Two-axis Method

In the two-axis method, the nodes measure the hop distance from three coordinators but do not evaluate the actual distance. The distance rate from three coordinators is not sufficient to calculate the actual position. Due to the lack of information, the result of localization has errors, as shown in Fig. 11. Nodes on the boundary of the network are estimated as being located inside the network. As mentioned earlier, the accuracy of the localization does not ruin the address assigning procedure. Since the result reflects the relational position, each node successfully generates an address based on the location information. For high accuracy of the procedure, additional coordinators can be used. HCRL[16] performs a fine-grain localization by deploying four coordinator nodes and adjusting the radio power. The final result of the localization distributes network nodes in the proper position, reduces the conflict rate and removes the additional DAD procedure.

One of the advantages of the two-axis method is that the DAD cost is almost independent from the size of the network. We fixed the average density of the nodes and increased the size of the network. Fig. 12 shows that the cost per address is almost constant, as the network size increases. In this case, addresses are well distributed, and address conflicts rarely occur.

3) Comparing with Conventional Approaches

In order to compare our approaches against conventional approaches, we conducted the simulation with various approaches on MANET: strong DAD [9], MANETConf [5] and Perkins's work [10]. We fixed the average density of the nodes and increased the number of nodes from 100 to 1000. Fig. 13 shows the network overhead of various autoconfiguration approaches. Communication overhead is the average number of messages required per address allocation. As the network size increases, the overhead of strong DAD increases exponentially. The proposed algorithm reduces the DAD cost. It shows that the DAD cost of autoconfiguration has a huge overhead, and reducing network cost is the main goal of revising the autoconfiguration protocol. MANETConf uses broadcast messages such as address request and address cleanup, one-hop broadcast messages like neighbor query, unicast messages like neighbor reply and etc. As MANETConf is a stateful address configuration based on mutual exclusion, each node should reply to the DAD initiating the node's request. The communication overhead increases in proportion to the number of nodes. Although the three-axis method reduces the size of the DAD zone, the method increases the DAD zone after the address conflict. Hence, the network cost is influenced by the number of nodes in the network. The two-axis method well distributes the location information of the network nodes and assigns nodes' addresses at a low conflict rate. The method also reduces the size of the DAD area, so the cost is stable. The strong DAD and Perkins's proposal show much higher networking overhead because the network is flooded with AREQ messages for the DAD procedure. Although Perkins's proposal reduces message overhead for AREP messages, the conflict rate causing AREP message is rarely found. Ninety-nine percent of nodes assign a unique link local address in the first random configuration of strong DAD and Perkins's

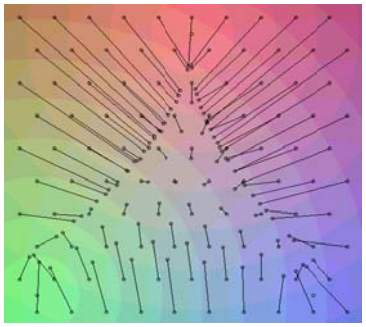

Figure 11. Error of localization with lack of the fourth coordinator

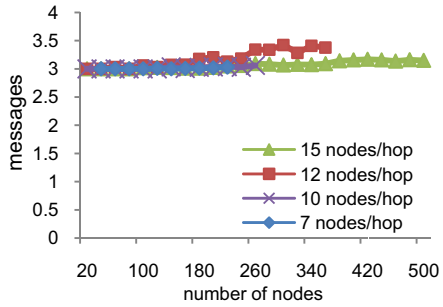

Figure 12. DAD cost of 2-axis Method

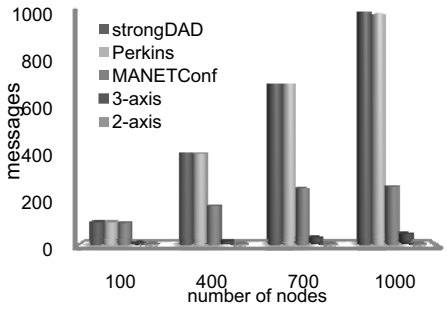

Figure 13. Comparison of network overhead

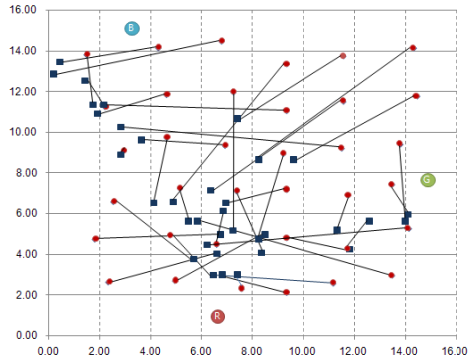

Figure 14. Localization error

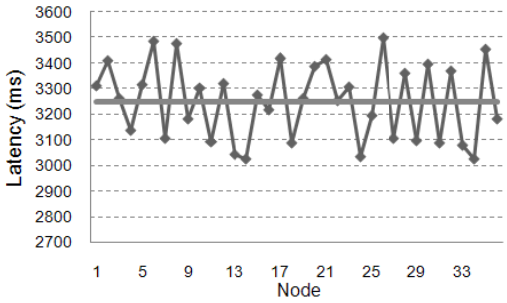

Figure 15. Autoconfiguration Latency

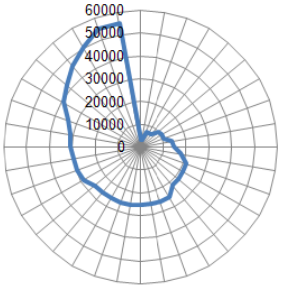

Figure 16. Autoconfiguration address space

proposal in the network with thousand of nodes. So the network overhead of Perkins's proposal is similar to the original method in this simulation.

The result can be explained with the message complexity of the protocols. The IP verification method of the proposed algorithm within a DAD zone is similar with that of strong DAD in single-node joining case. The IP verification cost of strong DAD is $m\mathrm{O}(N) + \mathrm{O}(t)$ where AQ (Address Request) messages are broadcasted into the network with N nodes and AP(Address Reply) messages are unicasted or relayed via a MANET routing tree with t nodes[19]. The IP verification cost of the proposed algorithm is lighter than strong DAD since the proposed algorithm reduces size of DAD zone. AR-like messages are broadcasted in DAD zone and AP-like messages are relayed at most size of DAD zone. Although the proposed method does not use multiple successful trials for IP verification, the cost can be evaluated in same way. In an IP verification procedure, $m\mathrm{O}(N_{\mathrm{DAD}})+\mathrm{O}(t_{\mathrm{DAD}})$ is the upper bound of the maximum cost of the proposed algorithm, where N_{DAD} and t_{DAD} are the number of node and the number of AP forwarding nodes in the DAD zone. N_{DAD} and t_{DAD} are function of physical density of nodes, d, and trial counts, since they increase with DAD failure and they are bounded by the physical density of nodes in the network. However, they are independent from total number of nodes, N, and they are quite static and limited. So, they can be assumed as constants. With the maximum number of tries, n, $n(m\mathrm{O}(N_{\mathrm{DAD}})+\mathrm{O}(t_{\mathrm{DAD}}))$ is the total cost of the proposed algorithm. Additionally in the localization phase, the original and advanced method broadcast R, G and B tags into the network, so message complexity of the localization is $\mathrm{O}(N)$. Perkins's proposal has same message complexity with that of strong DAD, and in MANETconf $n\mathrm{O}((t+1)N) + \mathrm{O}(N) + \mathrm{O}(2)$ is the upper bound of the maximum number of messages in single node-joining case [19].

B. Testbed Experiment

We deployed 36 Tmote Sky [18] motes in a 250 m^2 area. All nodes were randomly positioned at a distance of 2 to 2.5 meters, and three coordinator nodes were positioned on the edges of the network. We set the radio communication range to about 3 meters, which may also vary up to 2 meters. Each node started autoconfiguration when the node received color advertisement from the coordinator motes.

Fig. 14 shows the deployment and localization error; the circles are the actual positions of the nodes, and the squares are the localized positions. The localization error is the average of the experiments. Results show that the minimum localization error is 0.2 meters, the maximum 8.8 meters, and the average 4.2 meters. The localization error largely varies due to the radio communication range and the link quality. However, localization error does not have a large impact on the proposed protocol. Hence, in all experiments address conflicts do not occur.

Fig. 15 shows the autoconfiguration latency of each node, Autoconfiguration latency is the average time needed to

complete autoconfiguration and is measured as the difference between the emission time of the very first neighbor solicitation and the expiration time of the neighbor advertisement wait. In our experiment, each node waits 3 seconds after the emission of the neighbor solicitation. We assume the period is long enough to perform DAD by waiting neighbor advertisement. In our experiment, the autoconfiguration latency of each node does not vary much due to the absence of collision. Hence, the mean autoconfiguration latency is 3.25 seconds, and the standard deviation is 0.147.

Fig. 16 shows the address space usage based on the two-axis method. The solid line represents assigned addresses, which largely varies, and it indicates that the address conflict rate is low.

VI. Related Work

Although there are many IPv6 address autoconfiguration protocols for mobile ad-hoc networks (MANET), they rarely consider the autoconfiguration protocols for resource-constrained networks such as wireless sensor networks. WSNs may use already standardized mechanisms for any type of IPv6 network, namely the IPv6 neighbor discovery (NDP) [20] and the IPv6 stateless address autoconfiguration (SAA) [21]. Both mechanisms expect layer 2 multicast capable links, i.e., multicast packets sent to the link are received by all on-link nodes. Some link technologies such as IEEE 802.15.4 do not support link layer multicast by default. One solution is to use layer 2 broadcasts to distribute multicast packets to the network instead of multicast functionality. However, an intensive use of broadcast would lead to a significant consumption of bandwidth, processing power and battery power in sensor networks, something that has to be limited as much as possible.

[22] investigated how standard NDP and SAA can be optimized for the IEEE802.15.4 sensor networks. The authors suggested nodes and the PAN coordinator can exchange router solicitation and router advertisement messages using existing mesh (layer 2) routing protocols. The duplicate address detection (DAD) protocol is very similar to the protocol proposed in [21]: a node sends a neighbor solicitation message with solicited node multicast, and a node that detects conflicted address replies using a neighbor advertisement message. However, this also causes large overhead for a low-power sensor network, and to exploit mesh routing, all nodes should be configured with valid short 16-bit or long 64-bit addresses.

The pioneer work of Perkins et al. on ad-hoc address autoconfiguration [10] is an adaptation of the stateless IETF Zeroconf protocol for MANETs. A node randomly chooses an address and performs a DAD by flooding the network with an Address Request (AREQ) message, which contains the chosen address. A node having the same address defends it by replying with an Address Reply (AREP) message, which is sent over the reverse path established by the AREQ message. If there is no other node in the network with this address, a timer at the originator node expires and the address is considered unique. A serious drawback of this approach is that the overhead caused to fulfill DAD is extremely large.

Another DAD mechanism specific for MANET called Weak DAD (WDAD) was proposed in [11]. Other than the query associated DAD methods, WDAD is integrated with the routing protocol and can continuously detect duplicate addresses with information added to the routing protocol packets. Thus, the routing protocol packet format has to be modified. The main idea is to add a key to each address that is distributed by the routing protocol. The key can be of arbitrary length and is chosen once by each node either randomly or based on a Universal Unique ID (UUID). A node detects a conflict if the node receives two address-key-pairs with the same address but different keys. Thus, a conflict cannot be detected if two nodes choose the same address and the same key. In the case of random keys, the probability of an undetectable conflict decreases as the key length increases. Since the key length determines the additional overhead, there is a trade-off between routing protocol overhead and the probability that some conflicts cannot be detected. A similar approach is used in [6]. Here, the author proposed a new network layer addressing scheme with variable-length addresses. An advantage of variable-length addresses is that the overhead of the protocols sending a lot of addressing information can be reduced significantly. Yet, the overhead caused by the protocol is large enough to deplete network bandwidth and energy.

An example of a stateful approach is MANETconf [5]. Using MANETconf, each configured node is able to assign addresses to new nodes and therefore maintains an allocation table of already assigned addresses in the network. A new node called a requester searches for an already configured node called an initiator by sending a special broadcast message. The initiator replies by choosing an unassigned address and ensures the uniqueness of this address by a mutual exclusion algorithm. It floods a special message and asks all nodes in the network for permission to assign this address. The address is assigned only if all nodes send a positive reply. If a node does not reply at all, the initiator assumes that this node has left the network. Thus, the node's address is removed from the allocation table.

Allocating unique address for internal network is a good way to reduce the complexity of autoconfiguration process. Roofnet manages internal address scheme, which is an experimental 802.11b/g mesh network providing Internet connectivity to multiple users [23]. Roofnet consists of multiple wireless nodes and gateway nodes to the wired Internet. Roofnet does not use global unique IP address internally. Instead it uses its own routing protocol for internal communication. Internal addresses need to be unique within Roofnet nodes. Roofnet nodes use a class A address with the lower 24 bits of IPv4 Ehternet address as identification for the internal link. The identification field is assigned by the DHCP server wired to Ethernet. The internal address scheme reduces the complexity of the system and installation burden of Roof net node.

Although the previous approaches are robust and reliable, they do not satisfy the resource limitation of a resource-constrained network such as the IEEE 802.15.4 network. Performing autoconfiguration in a low-powered network is quite challenging, since most of the previous approaches are based on powerful functionality of the link layer and have large overhead in an ad-hoc network.

VII. CONCLUSION

The first step for operating a 6LoWPAN network is assigning unique addresses to network nodes. With globally unique addresses, the network can be attached to a global Internet or infra-structure network. To build the 6LoWPAN network with resource-constrained nodes, an autoconfiguration protocol should consider efficiency and robustness in terms of energy consumption. We proposed lightweight autoconfiguration protocols for a 6LoWPAN network that uses location information to reduce DAD cost. The proposed protocols are simple and operate at low cost.

A flexible DAD zone provides mobility of network nodes during autoconfiguration. However, the mobility of the nodes over a long-term period, which occurs over the network, may break the consistency of the DAD zone and network division or significantly increase the DAD cost. Supporting long-term mobility is a good challenge in the autoconfiguration protocol. And merge and division between individual networks are quite heavy, since the proposed algorithm is based on the location labeling technique. These challenges are considered the next step in research.

The proposed algorithm is designed for general ad-hoc network. We have experience on some applications for sensor network, such as a structural health monitoring system [24], a parking lot surveillance system [25], an acoustic source localization [26] and a sensor network monitoring tool [27]. Since these applications are built with static nodes and have an always-on monitoring terminal, the applications are nicely applicable to the proposed auto-configuration algorithm.

ACKNOWLEDGMENT

We would like to thank our shepherd, David Yates, and the reviewers for their useful feedback on the paper. This research was supported by the NRL (National Research Laboratory) program of the Korean Science and Engineering Foundation (No.M10500000059-6J0000-05910) and MKE (Ministry of Knowledge Economy), Korea, under the ITRC support program supervised by the IITA (IITA-2008-C1090-0801-0015).

REFERENCES

[1] D. E. Culler and J. Hui, 6LoWPAN Tutorial. IP on IEEE 802.15.4 Low-Power Wireless Networks. http://www.archrock.com/downloads/resources/6LoWPAN-tutorial.pdf

[2] Sensor Map. http://atom.research.microsoft.com/sensormap/

[3] G. Montenegro, N. Kushalnagar, J. Hui, and D. Culler, “Transmission of IPv6 Packets over IEEE 802.15.4 Networks”, RFC 4944

[4] R. Droms, “Dynamic Host Configuration Protocol”, Network Working Group - RFC 2131, March 1997.

[5] S. Nesargi and R. Prakash, “MANETconf: Configuration of Hosts in a Mobile Ad Hoc Network,” Proc. IEEE INFOCOM, June 2002.

[6] J. Boleng, “Efficient network layer addressing for mobile ad hoc networks”, in Proc. of ICWN'02, Las Vegas, USA, June 2002, pp. 271-277.

[7] H. Zhou, L. M. Ni, and M. W. Mutka, “Prophet Address Allocation for Large Scale Manets”, Proc. IEEE INFOCOM 2003, San Francisco, CA, Mar. 2003.

[8] M. Mohsin and R. Prakash, “IP Address Assignment in a Mobile Ad Hoc Network”, Proc. IEEE MILCOM 2002, Anaheim, CA, Oct. 2002.

[9] M. Günes and J. Reibel, “An IP Address Configuration Algorithm for Zeroconf Mobile Multihop Ad Hoc Networks”, Proc. Int’l. Wksp. Broadband Wireless Ad Hoc Networks and Services, Sophia Antipolis, France, Sept. 2002.

[10] C. Perkins, J. T. Malinen, R. Wakikawa, E. M. Belding-Royer, and Y. Sun, “IP address autoconfiguration for ad hoc networks,” IETF Draft, 2001.

[11] N. H. Vaidya, “Weak Duplicate Address Detection in Mobile Ad Hoc Networks”, in Proc. of ACM MobiHoc 2002, Lausanne, Switzerland, June 2002, pp. 206-216.

[12] K. Weniger, “PACMAN: Passive Autoconfiguration for Mobile Ad Hoc Networks”, Special issue, IEEE JSAC, Wireless Ad Hoc Networks, vol. 23, Mar. 2005, pp. 507-19.

[13] G. Malkin and R. Minnear, “RIPng for IPv6”, RFC 2080.

[14] A. Baggio and K. Langendoen, "Monte-Carlo Localization for Wireless Sensor Networks", MSN 2006, The Netherlands.

[15] D. Niculescu and B. Nath, "DV Based Positioning in Ad Hoc Networks", Telecommunication Systems, vol. 22, 2003, pp. 1-4, 267-28.

[16] S. Yang, J. Yi, and H. Cha, "HCRL: A Hop-Count-Ratio based Localization in Wireless Sensor Networks," Fourth Annual IEEE Communications Society Conference on Sensor, Mesh, and Ad Hoc Communications and Networks (SECON 2007), San Diego, USA, June 2007.

[17] H. Cha, S. Choi, I. Jung, H. Kim, H. Shin, J. Yoo, and C. Yoon, “RETOS: Resilient, Expandable, and Threaded Operating System for Wireless Sensor Networks”, The Sixth International Conference on Information Procedures in Sensor Networks (IPSN 2007)

[18] t-mote sky. http://www.sentilla.com/

[19] S. Kim, J. Chung, “Message Complexity Analysis of Mobile Ad Hoc Network Address Autoconfiguration Protocols,” IEEE Transactions on Mobile Computing, Vol. 7, No. 3, March 2008.

[20] N. Kushalnagar, G. Montenegro, and C. Schumacher “IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs): Overview, Assumptions, Problem Statement, and Goals”, RFC 4919

[21] S. Thomson and T. Narten, “IPv6 Stateless Address Autoconfiguration”, RFC2462

[22] S. Chakrabarti and E. Nordmark, "LowPan Neighbor Discovery Extensions", draft-chakrabarti-6lowpan-ipv6-nd-04.txt, Internet Draft (Work in Progress), November 18, 2007

[23] J. Bicket, D. Aguayo, S. Biswas, and R. Morris, “Architecture and Evaluation of an Unplanned 802.11b Mesh Network,” Proc. Mobicom 2005, Cologne, Germany, Aug 2005.

[24] H. Choi, S. Choi, H. Cha, "Structural Health Monitoring System based on Strain Gauge Enabled Wireless Sensor Nodes", INSS: International Conference on Networked Sensing Systems, Kanazawa, Japan, June 17-19, 2008

[25] K. Na, Y. Kim, H. Cha, "Acoustic Sensor Network-based Parking Lot Surveillance System", 6th European Conference on Wireless Sensor Networks(EWSN 2009), Cork, Ireland, February 2009

[26] Y. You, H. Cha, "Scalable and Low-Cost Acoustic Source Localization for Wireless Sensor Networks,"The 3rd International Conference on Ubiquitous Intelligence and Computing (UIC) 2006, Wuhan and Three Gorges, China, September 2006.

[27] I. Jung, H. Cha, "RMTool: Component-Based Network Management System for Wireless Sensor Networks," 2007 IEEE Consumer Communications and Networking Conference (CCNC), Las Vegas, January 2007.