Performance Comparison of *MISP*-based *MANET* Strong *DAD* Protocol

Sang-Chul Kim[1]
[1] School of Computer Science, Kookmin University,
861-1, Chongnung-dong, Songbuk-gu, Seoul, 136-702 Korea
[e-mail: sckim7@kookmin.ac.kr]
*Corresponding author: Sang-Chul Kim

Received March 5, 2015; revised June 25, 2015; accepted July 29, 2015;
published September 30, 2015

Abstract

A broadcast operation is the fundamental transmission technique in *mobile ad-hoc networks* (*MANETs*). Because a broadcast operation can cause a broadcast storm, only selected forwarding nodes have the right to rebroadcast a broadcast message among the one-hop and two-hop neighboring nodes of a sender. This paper proposes the *maximum intersection self-pruning* (*MISP*) algorithm to minimize broadcasting redundancy. Herein, an example is given to help describe the main concept of *MISP* and upper bounds of forward node have been derived based on induction. A simulation conducted demonstrated that when conventional *blind flooding* (*BF*), *self-pruning* (*SP*), an *optimized link state routing* (*OLSR*) *multipoint relay* (*MPR*) *set*, and *dominant pruning* (*DP*), are replaced with the *MISP* in executing *Strong duplicate address detection* (*DAD*), the performances in terms of the *energy consumption*, *upper bounds of the number of forward nodes*, and *message complexity* have been improved. In addition, to evaluate the performance in reference to the *link error probability*, P_e, an enhancement was achieved by computing a proposed *retransmission limit*, *S*, for error recovery based on this probability. Retransmission limit control is critical for efficient energy consumption of *MANET* nodes operating with limited portable energy where *Strong DAD* reacts differently to link errors based on the operational procedures.

Keywords: Ad-hoc Networks, Energy Efficiency, Address auto-configuration protocol, Selective-broadcasting technology, Link Error

---

A preliminary version of this paper appeared in IEEE ICUFN 2014, July 8-11, Shanghai, China. This version includes a concrete analysis and supporting implementation results on MISP algorithm.

http://dx.doi.org/10.3837/tiis.2015.09.010 **ISSN : 1976-7277**

1. Introduction

Owing to the quick growth in the number of *mobile ad-hoc network* (*MANET*) applications, short-range wireless applications, and robotic networks applications, more powerful and efficient *MANET* technology is strongly needed. A *MANET* consists of a set of sensor nodes with routing capability to forward packets. Each sensor host becomes a member of a self-organizing wireless network, where the hosts communicate with one another over multi-hop wireless links without relying on a fixed communication infrastructure, such as a base station or access point. Because a broadcast request can be issued at any time by any host with a packet to be delivered throughout the entire network, a broadcast operation has an important role in a *MANET*. A single transmission sent by each node will be received by all nodes within the node's transmission range because the transmission is broadcasted on a wireless channel. All other nodes need to cooperate in propagating the packet by rebroadcasting it because when a sender transmits a packet, all nodes within the sender's transmission range will be affected by this radio transmission [1].

Address auto-configuration is an important issue because an address pre-configuration is not always possible in a *MANET*. It is essential that all nodes be able to perform the operations required for the configuration of unique addresses to execute the proper routing of data packets in a *MANET*. In a conventional network, the *address auto-configuration protocol* (*AAP*) is categorized as either a stateless or stateful protocol [2]. The *dynamic host configuration protocol* (*DHCP*) is an example of a stateful protocol, where a *DHCP* server assigns unique addresses to non-configured nodes and maintains the state 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 *DAD* algorithms. Using a *DAD* algorithm, a node lacking an *IP* address in an *MANET* can determine whether the candidate address it selects is available. A node already equipped with an *IP* address also depends on *DAD* to protect its *IP* address from being accidentally used by another node in the *MANET*. Based on a conventional method [2], *DAD* can be classified as *Strong DAD* or *Weak DAD*. *Strong DAD* uses an address discovery mechanism where a node randomly selects and requests an address within the *MANET*, and verifies whether the address is already being used in the network. Based on a reply to the claimed request, which must arrive at the node within a finite-bounded time interval, the node can detect address duplications in the *MANET*.

To assign a new *IP* address to a newly joining node, a *MANET* conducts *AAP*s based on stateless or stateful approaches, where control messages are generated and transmitted from the source to destination through unicasting, multicasting, or broadcasting. Because a broadcast operation can trigger a broadcast storm when mobile nodes implement *AAP*s, this paper proposes a novel selective broadcasting technology called *maximum intersection self-pruning* (*MISP*), where only selected forwarding nodes have the right to rebroadcast a broadcast message. One of the application areas of this proposed *MISP* algorithm is in *wireless sensor networks* (*WSNs*), where sensor nodes operated by limited portable batteries need to save network costs for network maintenance, and where broadcast storms are a significant issue when delivering control messages. Many factors influence the performance of a *MANET*. A reduction of the control overhead is always a major concern because it relates to the power consumption of the mobile nodes and takes up a significant portion of the very limited wireless channel resources. One essential measure of the quality of a *MANET* routing or control protocol is scalability to the increase in *MANET* nodes.

The remainder of this paper is organized as follows. Section 2 summarizes the characteristics of related works, particularly concerning broadcast technologies and broadcast redundancy. Section 3 proposed several *claims* and *corollaries* leading to the mathematical expressions of the proposed algorithm, allowing the readers to easily understand the difference in the operation mechanisms between conventional *SP*, *OLSR MPR Set*, and *DP* algorithms and the proposed *MISP* algorithm. The *big O-notation* is used in order to derive the upper bound of the forward node of each protocol. Section 4 describes the numerical experiments conducted along with their results. Finally, Section 5 summarizes this work and provides some concluding remarks. The acronyms used in this paper are summarized in **Table 1**.

Table 1. Acronym Table

Acronym	***Meaning***	***Acronym***	***Meaning***
AAP	*Address auto-configuration protocol*	*ADB*	*Adaptive dynamic backbone*
AP	*Address Reply*	*AQ*	*Address Query*
BF	*Blind flooding*	*DAD*	*Duplicate address detection*
DHCP	*Dynamic host configuration protocol*	*DP*	*Dominant pruning*
DS	*Dominating set*	*F*	*Forward node list*
f_n/F_n	*Forward nodes*	*LCP*	*Link control protocol*
MAC	*Media Access Control*	*MANET*	*Mobile ad-hoc network*
MISP	*Maximum intersection self-pruning*	*MPR*	*Multipoint relay*
NP	*Nondeterministic polynomial*	*OLSR*	*Optimized link state routing*
PPP	*Point-to-Point*	*SP*	*Self-pruning*
TCP	*Transmission Control Protocol*	*TDP*	*Total dominant pruning*

2. Related Work

2.1 Selective-Broadcasting Technology

Several selective broadcasting techniques have been proposed to overcome the redundancy of flooding. In [1], the authors addressed the issue of broadcast operations causing a broadcast storm if forwarding nodes are not carefully designated. A *dominating set* (*DS*) is a subset of nodes in a network such that every node in the network is either a member node of the *DS* or a neighboring node of the member node of the *DS*. A member node of the *DS* can become a forwarding node when the wireless network does not have an isolated node, and has been fully connected through the wireless connection of the member node.

The concept of an *MPR set* was proposed to minimize the flooding of broadcast packets by reducing duplicate retransmissions [3]. The *MPR set* is a subset of one-hop neighbors of node that must cover all of the node's two-hop neighbors. The *OLSR* protocol uses *Hello* messages at each node to discover two-hop neighbor information, and conducts a distributed election of the *MPR* set.

In [4], the authors suggested that flooding should not be performed blindly, and proposed several schemes, including probabilistic, counter-based, distance-based, location-based, and cluster-based schemes, to reduce redundant rebroadcasts and differentiate their timing. The *enhanced partial dominant pruning* (*EPDP*) algorithm was proposed in [5]. Because the *total dominant pruning* (*TDP*), *partial dominant pruning* (*PDP*), and *EPDP* cause a defer time and additional *IP* packet fragmentation before a message is transmitted, additional wireless network resources are consumed. This paper focuses on and analyzes the *SP*, *OLSR MPR*, and *DP* algorithms, which do not experience message segmentation or additional wireless network resource consumption. Considering this concept, the proposed algorithm was developed in a simplistic fashion because there is no defer time or further steps required to broadcast a message. However, it is sufficiently robust to operate within *MANET* area where the contention-based *medium address control* (*MAC*) has been adopted.

2.2 Broadcasting Issue of Address Auto-Configuration Protocols

The *Lucent WaveLAN IEEE* 802.11 wireless network interface respectively consumes 1,327 and 967 *mW* of power when transmitting and receiving at a transmission rate of 2 *Mbps* [6]. In [7], a reduction in the number of broadcast messages is considered. The authors have focused on the concept of efficiency, which is represented as the number of forward nodes, rather than on reliability, which is described as the percentage of nodes receiving a broadcast packet.

In [8], the authors addressed two research approaches, probabilistic and deterministic, to obtain an efficient broadcast. A probabilistic approach uses no or a limited amount of neighbor information and requires high broadcasts to maintain an acceptable packet delivery ratio. A deterministic approach determines the list of forward nodes to guarantee full network coverage. In [9], the authors indicate that, unlike in a wired network, a packet transmitted by a node in an ad hoc wireless network can reach all neighbors. Therefore, the total number of transmissions is used as the performance metric for broadcasting. A node does not forward a broadcast packet if the *SP* algorithm is satisfied based on the neighborhood information. Although only a set of nodes forward a broadcast packet, this process guarantees complete network delivery. *SP*-based broadcast protocols collect neighborhood topology information based on a *Hello* message and form a connected *DS* through the forward nodes. *DP* also offers a promising approach to reducing redundant transmissions caused by *BF* [9], which is considered an approximation to the minimum flood tree problem. In [10], various *AAPs* for *MANET* are analyzed. One essential measure of the quality of a *MANET* protocol is its scalability with regard to an increase in the number of *MANET* nodes. Message complexity is defined as the overhead of an algorithm measured in terms of the number of messages required to satisfy the algorithm request [10].

2.3 *MISP* algorithm in *Strong DAD*

This paper proposes a new *SP* algorithm called *MISP* and verifies whether *AAP* algorithms are able to reduce the broadcast redundancy using this new algorithm. Furthermore, because the improvement achieved when the *SP*, *DP*, and *MISP* replace the conventional *BF* in the *AAP*, particularly for *Strong DAD*, is unknown, the performance is investigated in reference to the message complexity and energy consumption. Because *Strong DAD* uses additional recursive broadcast mechanisms to resolve duplicated *IP* addresses compared with other *AAP*s, the reduction rate achieved by *MISP* is expected to be significant compared with the reduction rate achieved by the *BF*, *SP*, *OLSR*, and *DP* algorithms. Therefore, the first objective of this paper is to obtain a quantitative ratio of the percentage of reduction for a broadcasting operation when the selected broadcasting algorithms are used in *Strong DAD*. The detailed operation of *Strong DAD* is shown in **Fig. 1**. This research adopts an analysis of a worst-case scenario [10] to conduct a quantitative analysis of the message complexity, upper bounds of the number of forward nodes, and energy consumption.

2.4 Wireless Channel Condition and Link Error Probability

The wireless communication environment and the mobility of the nodes make a link unstable, thereby resulting in link errors. In this paper, a generalized approach to the *link error probability* (P_e) is considered. P_e is the same for all inter-node links within each *MANET* group. We use this type of approach because different mobility and channel models result in different error rates under specific conditions. Therefore, the performance evaluation is dependent on the mobility model and wireless communication environment such as *Rayleigh* and *Rician* fading channels used in the computer simulation.

```
Start of Strong DAD
Step 01:  A node selects a temporary address
          and configures it as its network interface address
Step 02:  n=0; (Set retry count (n) =0)
Step 03:  m=0; (Set DAD retry count (m) = 0)
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 IP address
          and makes an Address Query (AQ) message for the IP address
Step 07:  The node broadcasts the AQ
Step 08:  if (all nodes receive the AQ == TRUE)
Step 09:      if (an Address Reply (AP) message arrives to the node
                  before timer expires == TRUE)
Step 10:          if ((retry count <= n ) == TRUE)
Step 11:              goto Step 4;
Step 12:          else
Step 13:              goto Step 21;
Step 14:      else (an AP message arrives to the node before timer
                  expires == FALSE)
Step 15:          if ((DAD retry count <= m ) == TRUE)
Step 16:              The node replaces the source IP address with its IP
                          address (Successful assignment of IP address)
                  break;
Step 17:          else
Step 18:              goto Step 5;
Step 19:  else
Step 20:      goto Step 7;
Step 21:  The node fails to get a source IP address
End of Strong DAD
```

Fig. 1. Pseudo code of Strong DAD

A generalized approach to using P_e provides a level of independence to any mobility model and fading channel. By averaging the link error events to obtain the link error rate, and applying this value to the P_e, the results of this paper are directly applicable to a performance analysis of a *MANET* that uses a specific mobility model and fading channel [11]. For a given P_e, the *retransmission count limit value* (S) can be defined based on the network manager's desired settings, some optimal criteria, and/or the priority of the mobile node. For a given P_e, the *average number of transmissions* (N_{TM}) required for a successful reception is provided in (1), which can be used as a reference value for the S [11].

$$N_{TM} = (1 - P_e)^{-1} \text{ for } 0 \le P_e < 1 \tag{1}$$

Because a link error can stop the propagation of *AQ* messages, a node that experiences link errors needs to retry broadcasting the *AQ* message to its neighboring nodes. *Point-to-Point* (*PPP*) protocol uses *link control protocol* (*LCP*) to negotiate, set up links, and detect link error occurred on the *Wide Area Network* (*WAN*) data link where *PPP* provides a standard method for transporting multiprotocol datagrams over point-to-point links. When the *LCP* closes the link, it notifies network layer protocols so that they may take appropriate action [12]. Therefore, this paper adopts this kind of notification mechanism from the lower layers such that an *IP* node can be able to learn of the link failure of a transmission using *LCP*.

3. Proposed Algorithm

3.1 Introduction of Selective-Broadcasting Technology

The broadcast storm problem is a serious issue in a *MANET*. Hence, several algorithms have been introduced to reduce the number of broadcast messages. In [8], the authors concluded that finding a minimum flood tree that provides the minimum number of forward nodes is *nondeterministic polynomial* (*NP*) time-complete [8]. They argued that although a minimum flood tree is constructed, the maintenance cost of the tree in a mobile environment is too high to be useful in practice. The *DP* algorithm [9] can reduce redundant transmissions using two-hop neighborhood information [8]. Because a source node knows the list of forward nodes, based on its neighboring nodes selected using the *SP* or *DP* algorithm, it is not necessary for all neighboring nodes to rebroadcast a packet issued by the source node. Conversely, all neighboring nodes rebroadcast a packet issued by the source node in *BF*. The *SP* and *DP* algorithms can reduce the total number of rebroadcasted packets and rebroadcast nodes compared to *BF*. By adopting the *SP* and *DP* algorithms, the performance of the *Strong DAD* algorithm when using the decision by the nodes to rebroadcast packets can be evaluated.

There seem to be few studies that have described in a systematic and easy to understand way the exact differences among broadcasting algorithms, such as *BF*, *MPR*, *SP*, and *DP*, for a sample *MANET*. Therefore, the following section describes the basic differences between the *BF*, *MPR*, *SP*, and *DP* algorithms.

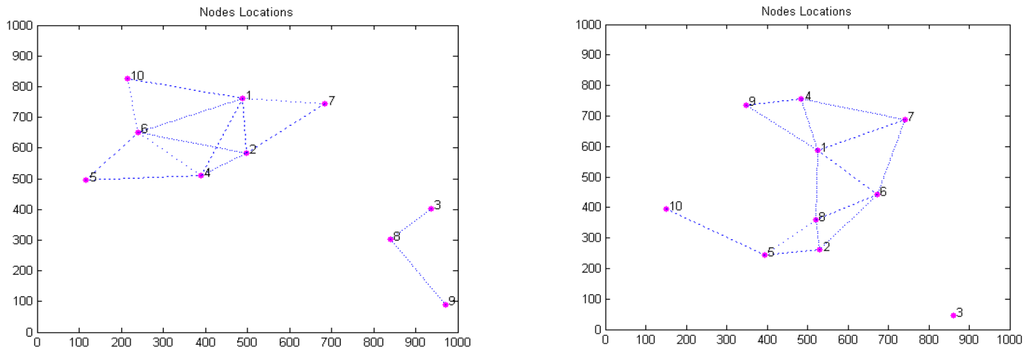
3.2 Analysis of *SP* Algorithm

Fig. 2 (*a*) and (*b*) illustrate sample *MANETs* at an instantaneous time, where the differences among the *SP*, *DP*, *MPR*, and *MISP* algorithms can be seen. In the *SP*, when a receiver node (*r*) receives a packet that piggybacks a neighboring list of a sender node (*s*), it calculates if the set of $N(r)$ - $N(s)$ is empty. If the set is empty, *r* does not rebroadcast the packet because $N(r)$ is covered by *s*. Otherwise, *r* rebroadcasts the packet.

***Claim* 1.** The upper bound of the number of forward nodes of *SP* can be denoted as $O(\cup\,[n(i), n(j)$ only if when $\exists_n\{n{:}n\in N(j)$ and $n\notin N(i)\}]$ for $j=1,\ldots, N_T$ and $i=1,\ldots, N_T)$ where i is the index number of node (n), j is the index number of neighboring nodes of the node i, and $n(i)$, $n(j)$ indicates the i^{th} and j^{th} node. In addition, when i is used as the index number of outer loop, j is used as the index number of inner loop where i and j are substituted from 1 to N_T, respectively and N_T indicates the total number of nodes in a network.

Proof. **Fig. 2**(*a*) is provided to explain the systematic procedure of the *SP*. Because node 7 (or $n(i)$ which is i^{th} node when i=1) is a source node, $N(s)$ (or $N(i)$ where i=1, the neighboring nodes of i^{th} node) becomes {1, 2, 7}. Node 1 (or $n(j)$ which is j^{th} node (j=1) when i=1) and node 2 (or $n(j)$ which is j^{th} node (j=2) when i=1) are receiver nodes. Because node 1 (or $n(j)$ which is j^{th} node (j=1) when i=1) is considered to be a receiver node, $N(r)$ (or the neighboring nodes $N(j)$ where j=1, i=1) becomes {1, 2, 4, 6, 7, 10} and $N(r)$ - $N(s)$ equals {4, 6, 10}, which is not a null set. This process can be expressed as $\exists_n\{n{:}n\in N(j)$ and $n\notin N(i)\}$ because $\exists_n$ means there exist nodes such as {4, 6, 10} satisfying the rule of $\{n{:}n\in N(j)$ and $n\notin N(i)\}$. Therefore, node 1 should be a member of the *forwarding node* (F_n), which can be written as $F_n = \{1\}$. This process is represented as $\cup(n(i), n(j))$ which results in $n(j) = \{1\}$ because the source node 7 ($n(i)$) is not counted as forward node. Furthermore, this process can be generalized as $\cup[n(i)$, $n(j)$ only if when $\exists_n\{n{:}n\in N(j)$ and $n\notin N(i)\}]$ for j=1, i=1. Because node 2 (or $n(j)$ which is j^{th}

node ($j=2$) when $i=1$) is considered a receiver node, $N(r)$ (or the neighboring nodes $N(j)$ where $j=2$, $i=1$) becomes {1, 2, 4, 6, 7, 10}. Here, $N(r)$ - $N(s)$ equals {4, 6}, which is not a null set. This process can be expressed as $\exists_n\{n:n\in N(j) \text{ and } n\notin N(i)\}$ because $\exists_n$ means there exist nodes such as {4, 6} satisfying the rule of $\{n:n\in N(j) \text{ and } n\notin N(i)\}$. Therefore, node 2 should be a member of F_n, which can be written as $F_n = \{1, 2\}$. This process is represented as $\cup(n(i), n(j))$ which results in $n(j) = \{2\}$ because the source node 7 ($n(i)$) is not counted as forward node. Furthermore, this process can be generalized as $\cup[n(i), n(j) \text{ only if when } \exists_n\{n:n\in N(j) \text{ and } n\notin N(i)\}]$ for $j=2$, $i=1$.Therefore, the upper bound of the number of forward nodes of the first step can be denoted as $O(\cup[n(i), n(j) \text{ only if when } \exists_n\{n:n\in N(j) \text{ and } n\notin N(i)\}])$ for $j=1,\ldots,N_T$, $i=1$.



(a) *Sample network for SP, OLSR, DP*

(b) *Sample network for MISP*

Fig. 2. Sample networks

In the next step, for $F_n = \{1, 2\}$, because node 1 (or $n(i)$ which is i^{th} node when $i=2$) becomes a source node, $N(s)$ (or $N(i)$ where $i=2$, the neighboring nodes of i^{th} node) becomes {1, 2, 4, 6, 7, 10}. Because node 10 (or $n(j)$ which is j^{th} node ($j=1$) when $i=2$) is considered a receiver node, $N(r)$ (or the neighboring nodes $N(j)$ where $j=1$, $i=2$) becomes {1, 6, 10}. Here, $N(r)$ - $N(s)$ equals a null set, which violates the rule of $\exists_n\{n:n\in N(j) \text{ and } n\notin N(i)\}$. Therefore, node 10 should not be a member of F_n, which can be written as $F_n = \{1, 2\}$. This process can be represented as $\cup(n(i), n(j))$ which results in $F_n = \{1, 2\}$ because the previous forward nodes found in $i=1$ have been accumulated. Because node 6 (or $n(j)$ which is j^{th} node ($j=2$) when $i=2$) is then considered as a receiver node, $N(r)$ (or the neighboring nodes $N(j)$ where $j=2$, $i=2$) becomes {1, 2, 4, 5, 6, 10}. Here, $N(r)$ - $N(s)$ equals {5}, which is not a null set, which complies with the rule of $\exists_n\{n:n\in N(j) \text{ and } n\notin N(i)\}$. Therefore, node 6 should be a member of F_n, which can be written as $F_n = \{1, 2, 6\}$. This process can be represented as $\cup(n(i), n(j))$ which results in $n(j) = \{6\}$ because the source node 1 ($n(i)$) is already counted as forward node and the forward nodes previously found have been accumulated. Because node 4 (or $n(j)$ which is j^{th} node ($j=3$) when $i=2$) is then considered a receiver node, $N(r)$ (or the neighboring nodes $N(j)$ where $j=3$, $i=2$) becomes {1, 2, 4, 5, 6}. In this case, $N(r)$ - $N(s)$ equals {5}, which is not null set, which complies with the rule of $\exists_n\{n:n\in N(j) \text{ and } n\notin N(i)\}$. Therefore, node 4 should be a member of F_n, which can be written as $F_n = \{1, 2, 4, 6\}$. This process can be represented as $\cup(n(i), n(j))$ which results in $n(j) = \{4\}$ because the source node 1 ($n(i)$) is already counted as forward node and the forward nodes previously found have been accumulated. Because node 2 (or $n(j)$ which is j^{th} node ($j=4$) when $i=2$) is thus considered to be a receiver node, $N(r)$ (or the neighboring nodes $N(j)$ where $j=4$, $i=2$) becomes {1, 2, 4, 6, 7}, and $N(r)$ - $N(s)$ equals a null set, which violates the rule of $\exists_n\{n{:}n\in N(j)$ and $n\notin N(i)\}$. Therefore, node 2 should not be a member of F_n, which can be written as $F_n = \{1, 2, 4, 6\}$. The upper bound of the number of forward nodes of the second step can be denoted as $O(\cup[n(i), n(j)$ only if when $\exists_n\{n{:}n\in N(j)$ and $n\notin N(i)\}])$ for j=1,…, N_T, i=2. Therefore, the upper bound of the number of forward nodes of *SP* can be denoted as $O(\cup[n(i), n(j)$ only if when $\exists_n\{n{:}n\in N(j)$ and $n\notin N(i)\}]$ for j=1,…, N_T and i=1,…, $N_T)$ which summarizes the upper bounds of the number of forward nodes generated from the first step (i=1) through the last step (i= N_T). ■

***Corollary* 1.** A summary of the systematic operation of the *SP* can be written as below. Nodes 4 and 6 should rebroadcast the message because nodes 4 and 6 have node 5 as a neighboring node. Node 5 receives duplicated messages transmitted from nodes 4 and 6. This operation can be considered a disadvantage of the *SP*. The following operation yields a termination point of the *SP*. Although node 2 has already broadcast the message in *Step* I, in *Step* II, based on s = 1 and r = 2, yielding the result of a null set, node 2 is not required to rebroadcast the message. This means that node 2 terminates the broadcast at *Step* II. The following illustrates a comparison between the conventional *BF* and *SP* algorithms. In *BF*, even if the set of $N(r)$ - $N(s)$ is empty, r always rebroadcasts the packet, thereby increasing broadcast redundancy. In the above example, although node 10 achieves a null set result based on s = 1 and r = 10, it broadcasts the message, which results in an increase in network cost.

3.3 Analysis of *OLSR* Algorithm

***Claim* 2.** The upper bound of the number of forward nodes of *OLSR* can be denoted as $O(\cup[n(i), n(j)$ only if when $\exists_n\{n{:}n\in N(j)$ and $n\notin\{N(N(i))\text{-}N(i)\}\}]$ for j=1,…, N_T and i=1,…, $N_T)$ where i is the index number of node (n) and j is the index number of neighboring nodes of the node i and $n(i)$, $n(j)$ indicates the i^{th} and j^{th} node.

Proof*.** Fig. 2(a) is provided to explain the systematic procedure of the *MPR set* algorithm. In the *OLSR* algorithm, the *MPR set* of ***node 7 (or $n(i)$ which is i^{th} node when i=1, j=1) should cover all ***two-hop neighbors of node 7***, which is {4, 6, 10} (or $\{N(N(i))\text{-}N(i)\}$ where i=1, j=1). To cover ***node 10*** (or $n(i)$ of $\{N(N(i))\text{-}N(i)\}$ where i=1, j=1, ***one of nodes among n(i)'s two-hop neighboring nodes***), ***node 1*** (or $N(j)$, where i=1, ***j***=1, ***one of nodes among n(i)'s one-hop neighboring nodes***) must be the subset of the ***one-hop neighbors of node 7*** (or $N(i)$ where i=1, the neighboring nodes of i^{th} node). Therefore, the *MPR set* becomes {1}. F_n can be written as $F_n = \{1\}$. This process can be expressed as $\exists_n\{n{:}n\in N(j)$ and $n\notin\{N(N(i))\text{-}N(i)\}\}$ because $\exists_n$ means there exist nodes such as {1} satisfying the rule of $\{n{:}n\in N(j)$ and $n\notin\{N(N(i))\text{-}N(i)\}\}$. To cover the ***node 6*** (or $n(i)$ of $\{N(N(i))\text{-}N(i)\}$ where i=1, j=2, ***one of nodes among n(i)'s two-hop neighboring nodes***), node 1 and ***node 2*** (or $N(j)$, where i=1, ***j***=2, ***one of nodes among n(i)'s one-hop neighboring nodes***) must be a subset of the ***one-hop neighbors of node 7*** (or $N(i)$ where i=1, the neighboring nodes of i^{th} node). Therefore, the *MPR set* becomes {1, 2}. F_n can then be written as $F_n = \{1, 2\}$. This process can be expressed as $\exists_n\{n{:}n\in N(j)$ and $n\notin\{N(N(i))\text{-}N(i)\}\}$ because $\exists_n$ means there exist nodes such as {2} satisfying the rule of $\{n{:}n\in N(j)$ and $n\notin\{N(N(i))\text{-}N(i)\}\}$. To cover ***node 4*** (or $n(i)$ of $\{N(N(i))\text{-}N(i)\}$ where i=1, j=3, ***one of nodes among n(i)'s two-hop neighboring nodes***), the one-hop neighboring nodes 1 and 2 (or $N(j)$, where i=1, ***j***=3) should be the subset of the ***one-hop neighbors of node 7*** (or $N(i)$ where i=1, the neighboring nodes of i^{th} node). Therefore, the *MPR set* becomes {1, 2}. F_n can thus be written as $F_n = \{1, 2\}$. This process can be expressed as $\exists_n\{n:n\in N(j)$ and $n\notin\{N(N(i))-N(i)\}\}$ because $\exists_n$ means there exist nodes such as {2} satisfying the rule of $\{n:n\in N(j)$ and $n\notin\{N(N(i))-N(i)\}\}$. Because F_n already includes 2 as its element, for uniqueness, F_n can be written as $F_n = \{1, 2\}$. Therefore, the upper bound of the number of forward nodes of the first step can be denoted as $O(\cup\,[n(i), n(j)$ only if when $\exists_n\{n:n\in N(j)$ and $n\notin\{N(N(i))-N(i)\}\}])$ for j=1,…, N_T and i=1.

In the second step, the above procedure is executed recursively for each member node of the *MPR set*, which is {1, 2} (or $n(i)$ which is i^{th} node when i=2, j=1, 2). The *MPR set* of node 1 should cover all two-hop neighbors of node 1. To cover ***node 5*** (or $n(i)$ of $\{N(N(i))-N(i)\}$ where i=2, j=1, ***one of nodes among n(i)'s two-hop neighboring nodes***), ***node 6*** (or $N(j)$, where i=2, j=1, ***one of nodes among n(i)'s one-hop neighboring nodes***) must be a subset of the ***one-hop neighbors of node 1*** (or $N(i)$ where i=2, the neighboring nodes of i^{th} node). Therefore, the *MPR set* becomes {1, 2, 6}. F_n can be written as $F_n = \{1, 2, 6\}$. This process can be expressed as $\exists_n\{n:n\in N(j)$ and $n\notin\{N(N(i))-N(i)\}\}$ because $\exists_n$ means there exist nodes such as {6} satisfying the rule of $\{n:n\in N(j)$ and $n\notin\{N(N(i))-N(i)\}\}$. To cover ***node 5*** (or $n(i)$ of $\{N(N(i))-N(i)\}$ where i=2, j=2, ***one of nodes among n(i)'s two-hop neighboring nodes***), ***node 4*** (or $N(j)$, where i=2, j=2, ***one of nodes among n(i)'s one-hop neighboring nodes***) must be a subset of the ***one-hop neighbors of node 1***(or $N(i)$ where i=2, the neighboring nodes of i^{th} node). Therefore, the *MPR set* becomes {1, 2, 4, 6}. F_n can then be written as $F_n = \{1, 2, 4, 6\}$. This process can be expressed as $\exists_n\{n:n\in N(j)$ and $n\notin\{N(N(i))-N(i)\}\}$ because $\exists_n$ means there exist nodes such as {4} satisfying the rule of $\{n:n\in N(j)$ and $n\notin\{N(N(i))-N(i)\}\}$. The upper bound of the number of forward nodes of the second step can be denoted as $O(\cup\,[n(i), n(j)$ only if when $\exists_n\{n:n\in N(j)$ and $n\notin\{N(N(i))-N(i)\}\}$ for j=1,…, N_T, i=2. Therefore, the upper bound of the number of forward nodes of *SP* can be denoted as $O(\cup\,[n(i), n(j)$ only if when $\exists_n\{n:n\in N(j)$ and $n\notin\{N(N(i))-N(i)\}\}]$ for j=1,…, N_T and i=1,…, $N_T)$ which summarizes the upper bounds of the number of forward nodes generated from the first step (i=1) through the last step (i= N_T). ■

***Corollary* 2.** As with the *SP* algorithm, nodes 6 and 4 are counted recursively in the *MPR set* to cover node 5. Because node 5 receives duplicate messages transferred from nodes 4 and 6, this can be considered a disadvantage of the *OLSR MPR set* algorithm.

3.4 Analysis of *DP* Algorithm

Based on the sample network shown in **Fig. 2**(*a*), the detailed procedure of *DP* can be explained through the following two steps. The first step determines the *forward nodes* (f_n) *of the source node* based on the information regarding the one-hop and two-hop neighboring nodes of the source node. The second step is identifying the next forward node based on the one-hop and two-hop neighboring node information of the previously found f_n. To describe the *DP* and *MISP* algorithms, **Fig. 3** is provided to illustrate the operational steps of the *SP*, *OLSR*, *DP*, and *MISP* algorithms. Based on a literature review, the *SP*, *OLSR*, *DP*, and *MISP* algorithms were shown to identify recursively the forward nodes based on the previously found forward nodes. When the union set of the forward nodes covers the entire network, the *SP*, *OLSR*, *DP*, and *MISP* algorithms terminate.

***Claim* 3.** The upper bound of the number of forward nodes of *DP* can be denoted as $O(\cup\,[n(i), n(j)$ only if when $\exists_n\{n:n\in N(j)$ and $n\notin\{N(N(i))-N(i)\}\}]$ for j=1,…, N_T and i=1,…, $N_T)$ where i is the index number of node (n) and j is the index number of neighboring nodes of the node i and $n(i)$, $n(j)$ indicates the i^{th} and j^{th} node.

Proof. The following describes the method used to determine f_n based on the information regarding the source node's one-hop and two-hop neighboring nodes. Here, $N(j)$ is {1, 2, 7}, and $N(N(i))$ becomes {1, 2, 4, 6, 7, 10} based on **Fig. 2**(*a*). The set of the difference between $N(N(i))$ and $N(i)$, represented as $N(N(i))$ - $N(i)$, (where i=1, j=1) becomes {4, 6, 10}. For the one-hop neighboring node set of ***node 7*** (or $n(i)$ which is i^{th} node when i=1, j=1), which is {1, 2, 7}, the following procedure indicates what node should be selected as the forward node to efficiently cover the set of $N(N(i))$ - $N(i)$, that is, {4, 6, 10}. For *DP*, $N(1) \cap \{N(N(i)) - N(i)\}$ is calculated as {1, 2, 4, 6, 7, 10} ∩ {4, 6, 10}, which yields {4, 6, 10}. Because $N(1) \cap \{N(N(i))\text{-}N(i)\}$ is not a null set, ***node 1*** (or $N(j)$ where i=1, j=1, ***one of nodes among n(i)'s one-hop neighboring nodes***) should be selected as one of the forward nodes, f_n={1}. This means that node 1 should be used for broadcasting because this operation has significance in covering the remaining nodes that have not yet been covered.

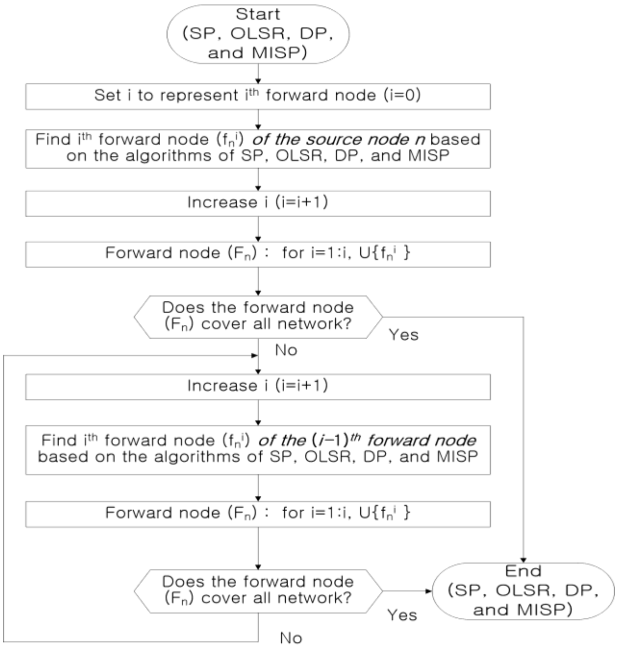


Fig. 3. Operational flowchart of SP, OLSR, DP, and MISP algorithms

The *forwarding node* (F_n) performed by the *DP* algorithm can be written as F_n = {1}. This process can be expressed as $\exists_n\{n{:}n \in N(j) \text{ and } n \notin \{N(N(i))\text{-}N(i)\}\}$ because $\exists_n$ means there exist nodes such as {1} satisfying the rule of $\{n{:}n \in N(j) \text{ and } n \notin \{N(N(i))\text{-}N(i)\}\}$.

***Corollary* 3.** The following summary can be provided based on the observation of the above *DP* operation. Because $N(N(v))$ indicates the potential coverage area by node v, $N(v)$ indicates the nodes already being covered by node v, and $N(N(v))$ - $N(v)$ indicates the nodes to be covered. Here, $N(1) \cap \{N(N(v)) - N(v)\}$ indicates the number of uncovered nodes (or remaining nodes) that can be covered by the broadcasting of node 1. If the result is a null set, the broadcasting of node 1 is without value because it does not cover any of the remaining nodes. If the result is not a null set, the broadcasting of node 1 is significant because it does cover the remaining nodes.

The same procedure can be adopted if ***node 2*** (or $N(j)$ where i=1, j=2, ***one of nodes among n(i)'s one-hop neighboring nodes***) is selected as the forward node to efficiently cover the set of $N(N(i)) - N(i)$, which is {4, 6, 10}. Therefore, in *DP*, $N(2) \cap \{N(N(i)) - N(i)\}$ is calculated as $\{1, 2, 4, 6, 7, 10\} \cap \{4, 6, 10\}$, which yields {4, 6, 10}. Because $N(2) \cap \{N(N(v)) - N(v)\}$ is not a null set, node 2 (or $N(j)$ where i=1, j=2, ***one of nodes among n(i)'s one-hop neighboring nodes***) is selected as a forward node. This means that node 2 should be broadcasted because it covers the remaining nodes {4, 6, 10}. Because node 2 is newly counted as a forward node, the forward node set becomes f_n = {1, 2}. F_n can be written as F_n = {1, 2}. This process can be expressed as $\exists_n\{n{:}n \in N(j)$ and $n \notin \{N(N(i))\text{-}N(i)\}\}$ because $\exists_n$ means there exist nodes such as {2} satisfying the rule of $\{n{:}n \in N(j)$ and $n \notin \{N(N(i))\text{-}N(i)\}\}$.

The same procedure can be adopted to determine whether ***node 7*** (or $N(j)$ where i=1, j=3, ***one of nodes among n(i)'s one-hop neighboring nodes***) should be selected as a forward node to efficiently cover the set of $N(N(i)) - N(i)$, which is {4, 6, 10}. Therefore, in *DP*, $N(7) \cap \{N(N(v)) - N(v)\}$ is calculated as $\{1, 2, 7\} \cap \{4, 6, 10\}$, which yields a null set. Because $N(7) \cap \{N(N(v)) - N(v)\}$ is a null set, node 7 cannot be selected as a forward node, which means that a broadcasting by node 7 is without value because this action does not cover any of the remaining nodes. Because there is no change in the set of the forward nodes, the forward node set remains as f_n = {1, 2}, and is called the first forward nodes. The first step of the *DP* algorithm is complete at this point. F_n can be written as F_n = {1, 2}. Because there does not exist nodes satisfying the rule of $\{n{:}n \in N(j)$ and $n \notin \{N(N(i))\text{-}N(i)\}\}$, this process does not satisfy the requirement of $\exists_n\{n{:}n \in N(j)$ and $n \notin \{N(N(i))\text{-}N(i)\}\}$. The upper bound of the number of forward nodes of the firsr step can be denoted as $O(\cup[n(i), n(j)$ only if when $\exists_n\{n{:}n \in N(j)$ and $n \notin \{N(N(i))\text{-}N(i)\}\}$ for $j=1,\ldots, N_T$, i=1.

The second step is determining the next forward node based on the information of the previously found f_n where the information regarding the one-hop and two-hop neighboring nodes of the previously found f_n is used. This begins with the calculation of $N(1) \cap \{N(N(i)) - N(i)\}$ and $N(2) \cap \{N(N(i)) - N(i)\}$, where $N(1)$ and $N(2)$ indicate the one-hop neighboring nodes of f_n, $N(i)$ indicates the one-hop neighboring nodes of the source node, and $N(N(i))$ indicates the two-hop neighboring nodes of the source node i, which yield {4, 6, 10} and {4, 6, 10}, respectively. To determine the second forward nodes based on the first forward nodes f_n = {1, 2}, $N(N(f_n))$ and $N(f_n)$ for node 1 are calculated as {1, 2, 4, 5, 6, 7, 10} and {1, 2, 4, 6, 7, 10}, respectively. The difference in the set of $N(N(f_n)) - N(f_n)$ is the presence of {5}. The difference in sets $N(f_n) - N(i)$, which is {1, 2, 4, 6, 7, 10} - {1, 2, 7}, equals {4, 6, 10}.

***Corollary* 4.** Based on the above selection procedure for forward nodes in the *DP* algorithm, the observations can be summarized as below. Here, $N(N(i))$ and $N(N(f_n))$ indicate the potential coverage area by node i in the first forward nodes and by node f_n in the second forward nodes, respectively. In addition, $N(i)$ and $N(f_n)$ indicate the nodes already covered by node i in the first forward nodes and by node f_n in the second forward nodes, $N(i) - i$ and $N(f_n) - N(i)$ indicate the first and second forward candidate nodes, and $N(N(i)) - N(i)$ and $N(N(f_n)) - N(f_n)$ indicate the nodes to be covered for the first and second forward nodes, respectively.

Moreover, $N(N(f_n))$ of node 1 indicates the two-hop neighboring nodes of forward node 1, which are {1, 2, 4, 5, 6, 7, 10}, and $N(f_n)$ of node 1 indicates the one-hop neighboring nodes of forward ***node 1*** (or $n(i)$ which is i^{th} node when i=2, j=1), which are {1, 2, 4, 5, 6, 7, 10}. The difference between sets $N(N(f_n) - N(f_n)$ is {5}. This means that one of the one-hop neighboring nodes of forward node 1, i.e., {4, 6, 10}, could be a candidate node for the second forward node; however, the candidate nodes of the second forward node must include {5} as its one-hop neighboring node.

First, ***node 4*** (or $N(j)$ where i=2, j=1, ***one of nodes among n(i)'s one-hop neighboring nodes***) are determined as {1, 2, 4, 5, 6}. Because node 4 incudes node 5 as its one-hop neighboring node, which should be covered by a forward node, node 4 can be a candidate node. F_n can be written as F_n = {1, 2, 4}. This process can be expressed as $\exists_n\{n{:}n\in N(j)$ and $n\notin\{N(N(i))\text{-}N(i)\}\}$ because $\exists_n$ means there exist nodes such as {1} satisfying the rule of $\{n{:}n\in N(j)$ and $n\notin\{N(N(i))\text{-}N(i)\}\}$. The ***node 6*** (or $N(j)$ where i=2, j=2, ***one of nodes among n(i)'s one-hop neighboring nodes***) are then determined as {1, 2, 4, 5, 6, 10}. Because node 6 includes node 5 as its one-hop neighboring node, which should be covered by a forward node, node 6 can be a candidate node of the second forward node. F_n can be written as F_n = {1, 2, 4, 6}. This process can be expressed as $\exists_n\{n{:}n\in N(j)$ and $n\notin\{N(N(i))\text{-}N(i)\}\}$ because $\exists_n$ means there exist nodes such as {1} satisfying the rule of $\{n{:}n\in N(j)$ and $n\notin\{N(N(i))\text{-}N(i)\}\}$. In the last step, ***node 10*** (or $N(j)$ where i=2, j=3, ***one of nodes among n(i)'s one-hop neighboring nodes***) are determined as {1, 6, 10}. Because node 10 does not include node 5 to be covered by a forward node, node 10 cannot become a candidate node of the second forward node. F_n can then be written as F_n = {1, 2, 4, 6}. Because there does not exist nodes satisfying the rule of $\{n{:}n\in N(j)$ and $n\notin\{N(N(i))\text{-}N(i)\}\}$, this process does not satisfy the requirement of $\exists_n\{n{:}n\in N(j)$ and $n\notin\{N(N(i))\text{-}N(i)\}\}$.

The following section illustrates the last step used for selecting the forward nodes. Because the candidate nodes of the second forward node such as nodes 4 and 6 cover the same node 5, there is no need for both nodes 4 and 6 to become a forward node to cover the same node 5. Therefore, only node 4 is selected as a forward node of the second forward node in the *DP* because node 1 has the information regarding the two-hop neighboring nodes of nodes 4 and 6. The forwarding node (F_n) determined by the *DP* can thus be written as F_n = {1, 2, 4}. The upper bound of the number of forward nodes of the second step can be denoted as $O(\cup[n(i)$, $n(j)$ only if when $\exists_n\{n{:}n\in N(j)$ and $n\notin\{N(N(i))\text{-}N(i)\}\}$ for j=1,…, N_T, i=2. Therefore, the upper bound of the number of forward nodes of *SP* can be denoted as $O(\cup$ $[n(i)$, $n(j)$ only if when $\exists_n\{n{:}n\in N(j)$ and $n\notin\{N(N(i))\text{-}N(i)\}\}]$ for j=1,…, N_T and i=1,…, N_T) which summarizes the upper bounds of the number of forward nodes generated from the first step (i=1) through the last step (i= N_T). ■

3.5 Analysis of *MISP* Algorithm

Fig. 2(*b*) is used to describe the proposed *MISP*, where node 1 is a source node. Here, node 6 locally broadcasts its *Hello* message to node 1, reporting that its neighbors are $N(6)$ = {1, 2, 6, 7, 8}. Node 8 also broadcasts its *Hello* message to node 1, reporting that its neighbors are $N(8)$ = {1, 2, 5, 6, 8}. When r = 6, source node 1 calculates $N(s)$ = {1, 4, 6, 7, 8, 9} and $N(r)$ = {1, 2, 6, 7, 8}, and determines $N(r)$ - $N(s)$ = {2}. When r = 8, source node 1 calculates $N(s)$ = {1, 4, 6, 7, 8, 9} and $N(r)$ = {1, 2, 5, 6, 8}, and determines $N(r)$ - $N(s)$ = {2, 5}. Therefore, source node 1 can select node 8 as its forward node because node 8 can cover nodes 2 and 5, whereas node 6 can only cover node 2. In a conventional *SP*, nodes 6 and 8 are selected as forward nodes. However, in the proposed *MISP*, only node 8 is selected as the forward node, and node 6 is not selected. This results in a reduction of the number of unnecessary flooding nodes by eliminating the unnecessary forward nodes in each step of the algorithm.

***Claim* 4.** The upper bound of the number of forward nodes of *MISP* can be denoted as $O(\cup[\boldsymbol{max}(|n(i), n(j)$ only if when $\exists_n\{n{:}n\in N(j)$ and $n\notin N(i)\}|$ for $j=1,\ldots, N_T$ and $i=1, \ldots, N_T)])$ where i is the index number of node (n), j is the index number of neighboring nodes of the node i, $n(i)$, $n(j)$ indicates the i^{th} and j^{th} node, and $|\cdot|$ represents the number of nodes from the difference in set $\{N(j)\text{-}N(i)\}$.

Proof. The procedure of the *MISP* can be written as the seven steps generalized below.

-. *Step* 1: Calculate $\{N(j)\text{-}N(i)\}$ (or $\{n{:}n\in N(j)$ and $n\notin N(i)\}$) for a neighboring node j of source node i.

-. *Step* 2: Determine whether $\{N(j)\text{-}N(i)\}$ for j of i is a null set.(or $n(i)$, $n(j)$ only if when $\exists_n\{n{:}n\in N(j)$ and $n\notin N(i)\}$).

-. *Step* 3: If $\{N(j)\text{-}N(i)\}$ is a null set, stop considering j as a forward node.

-. *Step* 4: If $\{N(j)\text{-}N(i)\}$ is not a null set, calculate the number of nodes from the difference in set $\{N(j)\text{-}N(i)\}$ (or $|\, n(i), n(j)$ only if when $\exists_n\{n{:}n\in N(j)$ and $n\notin N(i)\}|$) .

-. *Step* 5: Repeat the above *Steps* 1 through 4 for the other neighboring nodes (or $|\, n(i), n(j)$ only if when $\exists_n\{n{:}n\in N(j)$ and $n\notin N(i)\}|$ for $j=1,\ldots, N_T$ and $i=1$).

-. *Step* 6: Choose node j as the forward node of source node i when it gives the maximum number of nodes from the difference in set $\{N(j)\text{-}N(i)\}$ obtained from *Steps* 4 and 5 or $\boldsymbol{max}(|\, n(i), n(j)$ only if when $\exists_n\{n{:}n\in N(j)$ and $n\notin N(i)\}|$ for $j=1,\ldots, N_T$ and $i=1$)).

- *Step* 7: The forward node j chosen becomes the new source node i, and repeat *Steps* 1 through 6 until the network has been fully covered $O(\cup[\boldsymbol{max}(|n(i), n(j)$ only if when $\exists_n\{n{:}n\in N(j)$ and $n\notin N(i)\}|$ for $j=1,\ldots, N_T$ and $i=1, \ldots, N_T)])$. ■

To begin, the proposed *MISP* calculates the intersection set between $N(r)$ - $N(s)$ of node 6 and $N(r)$ - $N(s)$ of node 8, which can be represented as $\{2\}\cap\{2,5\}$, to determine whether the candidate forward nodes such as nodes 6 and 8 cover the same remaining network area. If the intersection set is not a null set, it means that the candidate forward nodes ($\{2\}\cap\{2, 5\} = \{2\}$, which is not a null set) cover the same remaining network area. Therefore, the *MISP* chooses only one node yielding the maximum number of set elements among the candidate forward nodes. Because $N(r)$ - $N(s)$ of node 6 equals $\{2\}$, the number of set elements of $N(r)$ - $N(s)$ of node 6 is 1 and because $N(r)$ - $N(s)$ of node 8 equals $\{2, 5\}$, the number of set elements of $N(r)$ - $N(s)$ of node 8 is 2. That is, node 8 provides the maximum number of set elements among the candidate forward nodes 6 and 8. If the intersection set is a null set, it means that the candidate forward nodes (for example, $\{2\}\cap\{5\} = \{\}$, which is a null set), cover a different remaining network area. Therefore, we call the proposed algorithm the *maximum intersection SP* algorithm, or *MISP*. The term *intersection* indicates the verification that the candidate forward nodes cover the same remaining network area, whereas the term *maximum* illustrates the process of removing the redundancy of rebroadcasting when the candidate forward nodes cover the same remaining network. In the above case, for the three candidate forward nodes, after investigating every intersection set of $N(r)$ - $N(s)$ of the candidate forward nodes of source node u (or forward node u), source node u (or forward node u) selects the forward node that provides the maximum number of element nodes in the set of $N(r)$ - $N(s)$. If the element nodes in the set of $N(r)$ - $N(s)$ of the candidate forward nodes are the same, only one forward node is selected in *MISP* in order to consider the network coverage. Based on systematic procedure of *SP*, in the conventional *SP*, source node 1 selects nodes 4 and 6 as its forward nodes; however, in *MISP*, node 4 is selected as its forward node in order to consider the network coverage. In here, the selection of node 6 yields same network coverage compared to the network coverage of node 4. **Table 2** summarizes the results of the upper bounds of the number of *forward node* (F_n) and the its real number of forward nodes based on the sample network shown in **Fig. 2.** It is found that the upper bounds of *MPR* and *DP* have the same mathematical expression.

Table 2. Comparison of upper bound of the number of Forward node (F_n) and their number

Algorithms	***BF***	***MPR***	***SP***	***DP***	***MISP***
Upper bound of the number of Forward node (F_n)	$O(N_T)$	$O(\cup[n(i), n(j)$ only if when $\exists_n\{n{:}n\in N(j)$ and $n\notin\{N(N(i))-N(i)\}\}]$ for $j=1,\ldots, N_T$ and $i=1,\ldots, N_T)$	$O(\cup[n(i), n(j)$ only if when $\exists_n\{n{:}n\in N(j)$ and $n\notin N(i)\}]$ for $j=1,\ldots,$ N_T and $i=1,\ldots, N_T)$	$O(\cup[n(i), n(j)$ only if when $\exists_n\{n{:}n\in N(j)$ and $n\notin\{N(N(i))-N(i)\}\}]$ for $j=1,\ldots, N_T$ and $i=1,\ldots, N_T)$	$O(\cup[\mathbf{max}(\vert n(i), n(j)$ only if when $\exists_n\{n{:}n\in N(j)$ and $n\notin N(i)\}\vert$ for $j=1,\ldots,$ N_T and $i=1, \ldots, N_T)])$
Real Forward node (*The number of* F_n)	1,2,4,5,6,10 (6)	1,2,4,6 (4)	1,2,4,6 (4)	1,2,4 (3)	1,2,4 (3)

4. Numerical Results and Observations

4.1 Energy Model

For sending a k-bit data packet in a *WSN*, a model of the energy consumption of the wireless transmitter and receiver was studied in [13]. For a sensor node, to transmit a k-bit packet, the node that operates the radio electronics consumes an energy amount of $k\cdot E_{elec}$, and the transmit power amplifier consumes an energy amount of $k\cdot\ \varepsilon_{amp}\ \cdot d^4$ or $k\cdot\varepsilon_{fs}\cdot d^2$, where d indicates the transmission distance between the transmitter and receiver. To receive a k-bit packet, the node operates the radio electronics, where an energy amount of $k\cdot E_{elec}$ is consumed [13]. In [13], it was determined that to achieve an acceptable E_b/N_0, the radio should dissipate $E_{elec} = 50$ nJ/bit to operate the transmitter or receiver circuitry, and $\varepsilon_{amp}=100$ $pJ/bit/m^2$ for the transmit amplifier. The equation of the energy consumption was derived in [13], where it was assumed that the squared value of the distance is used for the energy loss owing to the channel transmission in the simulation model. Therefore, to transmit a k-bit message a distance of $d_{nm}^{\ 2}$ using the radio model, where d_{nm} indicates the distance between a node (n) to its neighboring node (m), the transmitter expends the following:

$$E_{Tx}=k\cdot E_{elec} + k\cdot\varepsilon_{fs}\cdot d_{nm}^{\ 2}. \tag{2}$$

To receive a k-bit message, the receiver expends

$$E_{Rx}=k\cdot E_{elec}. \tag{3}$$

In this research, we used this energy equation to transmit a k-bit message. In the simulation, 10 $pJ/bit/m^2$ was used for ε_{fs}, 50 nJ/bit was used for E_{elec}, and 525 and 1,485 *bytes* were used as the lengths of a single packet, where 500 and 1,460 *bytes* are assigned to the data message [14] and 25 *bytes* is assigned to the packet header. In the simulation, the case of a single node joining the largest clustered network among several clustered sub-networks was evaluated. Clustering is randomly generated based on the network size and transmission range. A computer-based simulator was written to implement the proposed algorithm. During the simulation, the *forward node list* (*F*) implemented by the *MPR*, *SP*, *DP*, and *MISP* algorithms was selected to rebroadcast the messages. Because only the forward nodes in the neighboring list can broadcast a *Strong DAD* message, it was shown that the message complexities of *Strong DAD* are significantly reduced compared with the *BF* method.

The energy consumption of *BF* is determined based on Equation (4), where each node in the cluster member dissipates energy to locally broadcast packets. Therefore, the total energy consumption of *BF* is the summation of the energy consumed in each member node, which is represented as $E_{tx\_local\_broad}$, where *n* and *m* represent member nodes in a cluster.

$$E_{\mathrm{tx\_local\_broad}} = \sum\nolimits_{n,m} (k\cdot E_{\mathrm{elec}} + k\cdot \varepsilon_{amp}\cdot d_{nm}^{2}) \tag{4}$$

Because only forward nodes selected by the *SP*, *OLSR*, *DP*, and *MISP* algorithms have the right to broadcast messages, the energy consumption model of the forward nodes is the same as that of the cluster heads in the energy consumption model in [13], where the forward nodes have the same role as the cluster heads.

Therefore, based on Equations (5) and (6), the energy consumption of the *SP*, *OLSR*, *DP*, and *MISP* algorithms is composed of two steps, as described below. First, the forward nodes transmit 525-*byte* and 1,485-*byte* packets among the forward nodes, which are represented as $E_{tx\_cluster}$ in (5). Then, each forward node locally conducts a broadcast represented as $E_{tx\_local\_dominant}$ in (6). Therefore, the total energy consumption of the *SP*, *OLSR*, *DP*, and *MISP* algorithms is the summation of the energy consumed in the forward nodes plus the energy consumed during the broadcasting, which can be represented as $E_{tx\_cluster} + E_{tx\_local\_dominant}$, where *i* and *j* represent the forward nodes, and d_{in} represents the distance between a forward node *i* (which is the same as cluster head *i*) and a cluster member node *n*. Because every link in a cluster is composed of one hop, the maximum distance between a cluster and the farthest node was selected to reach all member nodes.

$$E_{\mathrm{tx\_cluster}} = \sum\nolimits_{i,j} (k\cdot E_{\mathrm{elec}} + k\cdot \varepsilon_{amp}\cdot d_{ij}^{2}) \tag{5}$$

$$E_{\mathrm{tx\_local\_dominant}} = k\cdot E_{\mathrm{elec}} + k\cdot \varepsilon_{amp}\cdot max(d_{in}^{2}) \tag{6}$$

4.2 Simulation Environment

The system model used to analyze the proposed *MISP* algorithm follows the system model introduced in [10]. A standalone *MANET* environment is required to compare the number of algorithms used, as well as the energy consumption, message complexity, and number of forward nodes among the *MPR*, *SP*, *DP*, and *MISP* algorithms, where the *MANET* nodes have no connection to an external network such as the Internet. A computer-based simulator was developed with nodes randomly distributed with a uniform density within a network area of 1 km^2. A discrete-event simulator was developed in *Matlab* to verify the various network topologies and calculate the message complexity and energy consumption. The random node generator and simulator performance verified (number of nodes, 100, 125, 150, and 175) that the average number of nodes per cluster and several of the specifications in the *adaptive dynamic backbone* (*ADB*) algorithm [10] matched the results in [10] with less than a 1% difference in most cases, where *QualNet* was used to simulate *ADB*.

In our analysis, the *conflict probability* (P_c) is defined as the probability that the *IP* address requested by a node is in use in the *MANET* group. The *conflict probability* depends on the size of the address and the number of nodes in the *MANET* group. *BF* used in the simulation, which is compared to the *SP*, *OLSR*, *DP*, and *MISP* algorithms, has every node retransmit a message to all of its one-hop neighbors whenever it receives the first copy of a message. In the computer simulation, two values of 0.25 and 0.75 are used for P_e, and a value of 0.5 is used for P_c. In ad hoc networks, it is found that the *Transmission Control Protocol* (*TCP*) throughput degradation remains at less than 10 percent for packet error probabilities ranging from 0 to 0.1, especially for one-hop ad hoc networks. This is attributed to the link-level retransmissions of error packets in *IEEE* 802.11, which can largely reduce the impairment of the channel error. However, the throughput has been severely deteriorated from the range of 0.1 to 1 [14]. Therefore, in this paper, P_e of 0.25 (> 0.1) and P_e of 0.75 (< 1) including the packet loss due to channel errors, *MAC* layer medium contention, and mobility have been selected from the range of 0.1 to 1. **Table 3** summarizes the simulation parameters.

Table 3. Simulation parameters

Parameter	***Value***
Network area	1,000 × 1,000 m^2
Transmission range	300 *m*
S (*Retransmission control limit*)	$(1- P_e)^{-1}$
m (*Strong DAD retry control limit*)	3
n (*Retry count limit*)	5
P_c (*Conflict probability*)	0.5
P_e (*Link Error Probability*)	0.25 and 0.75
The number of mobile nodes	10 to 50
ε_{fs} (*Free space channel model*)	10 $pJ/bit/m^2$
ε_{amp} (*Multipath fading channel model*)	100 $pJ/bit/m^2$
E_{elec} (*Radio electronics energy consumption*)	50 *nJ/bit*
The lengths of one packet	525 & 1,485 *Bytes*
Simulation time	40 *hours*

4.3 Simulation Results

Fig. 4 compares the energy consumption of the *BF*, *SP*, *OLSR*, *DP*, and *MISP* when a *Strong DAD* algorithm is applied with a P_c of 0.5 and a P_e of 0.25 (**Fig. 4** (*a*)) and 0.75 (**Fig. 4** (*b*)) with the packet lengths of 525 and 1,485 *Bytes*. As it can be expected, in all cases of P_e,, the energy consumption of 1,485 *Bytes* consumed more energy of 184%, 183%, 182%, 183%, and 182% than the energy consumption of 525 *Bytes* with *BF*, *SP*, *OLSR*, *DP* and *MISP*, respectively. In addition, in all cases of P_e, the total energy consumption of *MISP* can be reduced by 280%, 163%, 123%, and 23% compared with the *BF*, *SP*, *OLSR*, and *DP*, respectively.

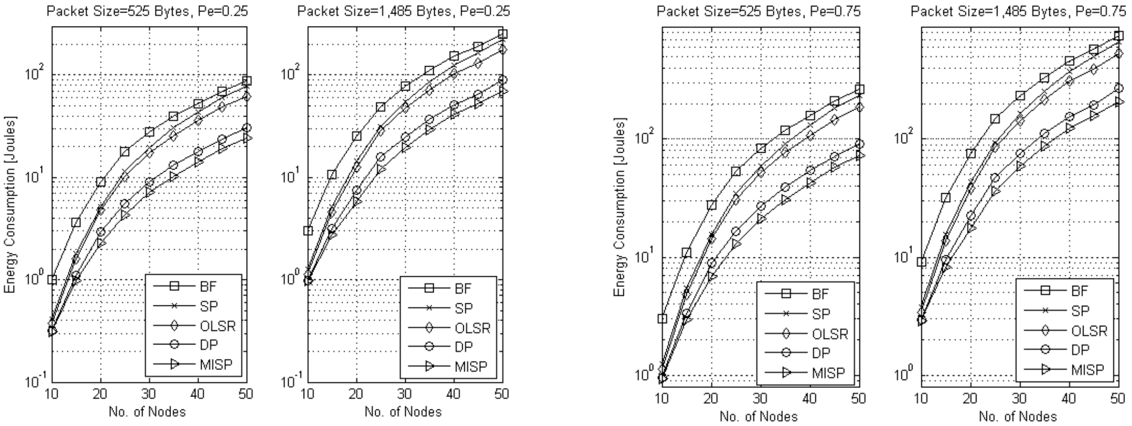
Fig. 5 illustrates the upper bounds of the number of forward nodes among *BF*, *SP*, *OLSR*, *DP*, and *MISP* when the *Strong DAD* algorithm is applied with a P_c of 0.5 and with a P_e of 0.25 and 0.75 with the packet lengths of 525 and 1,485 *Bytes*. In all cases of P_e, the upper bounds of the number of forward nodes of *MISP* can be reduced by 107%, 65%, 51%, and 11% compared with the *BF*, *SP*, *OLSR*, and *DP*, respectively.

Fig. 6 illustrates the message complexity among *BF*, *SP*, *OLSR*, *DP*, and *MISP* when the *Strong DAD* algorithm is applied with a P_c of 0.5 and with a P_e of 0.25 and 0.75 with the packet lengths of 525 and 1,485 *Bytes*. The message complexity of *MISP* can be reduced by 98%, 61%, 48%, and 10% compared with the *BF*, *SP*, *OLSR*, and *DP* algorithms, respectively. Because the optimal design of packet fragmentation is beyond the scope of this paper, 525 and

1,485 *Bytes* of packet lengths did not affect to the results of upper bounds of the number of forward nodes and message complexities, but affect to the results of energy consumption.

4.4 Observation

Figs. 4, 5, and **6** show that, in the case of P_e *=0.75*, the energy consumption, the number of forward nodes, and the message complexity of *BF*, *SP*, *OLSR*, *DP*, and *MISP* in *Strong DAD* have been increased nearly 200% compared to the case of P_e =0.25. It was also shown that in the case of P_e =0.75, the energy consumption, message complexity, and number of forward nodes of the *BF*, *SP*, *OLSR*, *DP*, and *MISP* in *Strong DAD* are increased nearly 50% compared to the case of P_e = 0.25. Therefore, it can be stated that with an increased degree of *link error probability* (P_e), the network cost used for recovering from the link failure is increased. For a given P_e, the mechanism of re-broadcasting (e.g., *AQ*) and re-unicasting (e.g., *AP*) messages result in different control signaling overhead in the *Strong DAD* protocol, and finally causes increased energy consumption and an increased number of forward nodes in the network. The main contribution of this paper is that when the *Strong DAD* algorithm is applied to configure the *IP* address in a *MANET* automatically, the proposed *MISP* algorithm significantly reduces the consumption of energy, the number of forward nodes, and the message complexity.



(*a*) P_e *of 0.25* (*b*) P_e *of 0.75*

Fig. 4. *Energy consumption of BF, SP, OLSR, DP, and MISP*

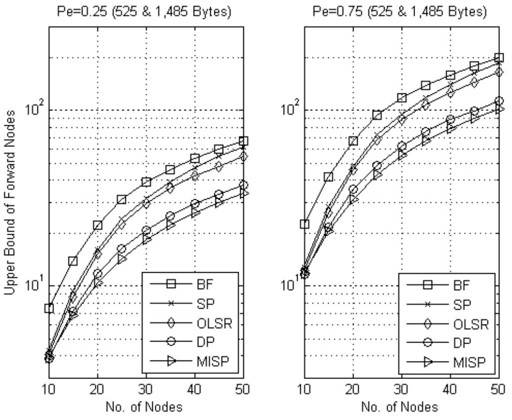


Fig. 5. *Upper bounds of the number of forward nodes* (*Packet lengths of 525 and 1,485 Bytes*)

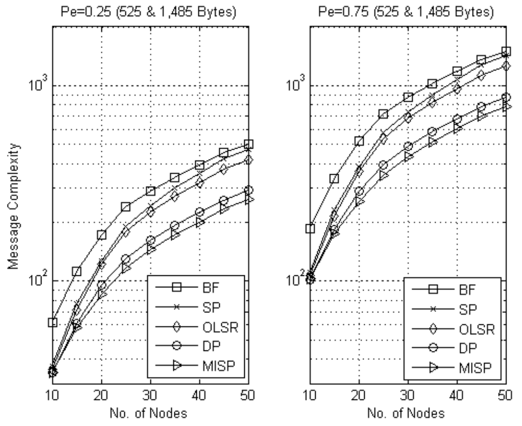


Fig. 6. *Message complexity* (*Packet lengths of 525 and 1,485 Bytes*)

Because the performance of the *MISP* is much better than that of the *DP*, the steps of which are much more complex that those of *MISP*, this paper focuses on this factor as an important contributor. The amount of network resources is saved by reducing the number of unnecessary flooding nodes when the *Strong DAD* is applied along with the *MISP*. The main idea of *MISP* comes from an earlier examination to determine whether nodes can become candidate forward nodes. The earlier elimination of nodes from the set of candidate forward nodes reduces of the number of unnecessary flooding nodes. In addition, the *MISP* adopts the concept of an intersection to verify that when the candidate forward nodes cover the same remaining network, the *MISP* finds the node that covers the maximum amount of network area by removing redundancy in the rebroadcasting.

5. Conclusion

The proposed *MISP* algorithm was modified and updated from a conventional *SP* algorithm to minimize the retransmissions of broadcasting operations, where only selected forwarding nodes among one-hop neighboring nodes of the sender have the right to rebroadcast a broadcast message when a generalized approach to the link error probability is considered. The proposed *MISP* algorithm is equipped with a novel property, i.e., an intersection operation is conducted to verify whether the areas covered by the candidate forward nodes are the same as the network area. By implementing an intersection operation into a conventional *SP* algorithm, unnecessary forward nodes are eliminated and the network resource requirements for broadcasting a message are reduced. When the number of intersected nodes (or covered nodes) incurred by the broadcasting of candidate forward nodes is the same, the *MISP* algorithm selects one candidate forward node in order to consider the network coverage. An example is given to help describe the main concept of *MISP*. Upper bounds of the number of forward node have been derived based on induction. It is found that the upper bounds of the number of *MPR* and *DP* forward nodes have the same mathematical expression. It was determined that the *MISP* algorithm yields a smaller number of forward nodes compared to the *SP*, *OLSR*, and *DP* algorithms because the *MISP* algorithm reduces the number of candidate forward nodes that need to be examined. Moreover, the total energy consumption of *MISP* can be reduced by 280%, 163%, 123%, and 23% compared with the *BF*, *SP*, *OLSR*, and *DP* algorithms, respectively. The proposed *MISP* algorithm also saves 107%, 65%, 51%, and 11% in the number of forward nodes compared to the *BF*, *SP*, *OLSR*, and *DP* algorithms, respectively. A simulation indicated that the *MISP*-based *Strong DAD* algorithm reduces the number of broadcasting nodes by 107% compared to a conventional *Strong DAD* with *BF* within a transmission range of 300 m when the link error probability is considered. Moreover, the proposed *MISP* algorithm saves 98%, 61%, 48%, and 10% in message complexity compared to the *BF*, *SP*, *OLSR*, and *DP* algorithms, respectively. As a future research, because adopting selective-broadcasting technologies may not guarantee giving the maximized network connectivity, the combined research between the selective-broadcasting and network connectivity algorithms could be conducted. In addition, as usual in *WSN*s, the research on the lifetime extension for the selected forward nodes can be another topic because the nodes tend to consume more power in order to service (transfer or relay) packets coming from their neighboring nodes. Because this research has mainly focused on designing the efficient energy consumption instead of providing quality of service (such as low error rates, high bandwidth, high throughput, low transmission delay, and low jitter, etc.), there might be several trade-off settings between the energy consumption and the quality of service in controlling protocols according to the purpose of network design.

References

[1] W. Lou and J. Wu, "Toward Broadcast Reliability in Mobile Ad Hoc Networks with Double Coverage," *IEEE Trans. Mobile Computing*, vol. 6, no. 2, pp. 148-163, Feb. 2007. Article (CrossRef Link).

[2] L. J. G. Villalba, J. G. Matesanz, A. L. S. Orozco, and J. D. M. Díaz, "Auto-Configuration Protocols in Mobile Ad Hoc Networks," *Sensors*, vol. 11, no.4, pp.3652-3666, 2011. Article (CrossRef Link).

[3] P. Jacquet, P. Muhlethaler, T. Clausen, A. Laouiti, A. Qayyum, and L. Viennot, "Optimized Link State Routing Protocol for Ad hoc Networks," in *Proc. of IEEE INMIC* 2001, pp.62-68, 2001. Article (CrossRef Link).

[4] Y.C. Tseng, S.Y. Ni, Y.S. Chen, and JP Sheu, "The Broadcast Storm Problem in a Mobile Ad Hoc Network," *Wireless Networks*, vol. 8, pp. 153-167, 2002. Article (CrossRef Link).

[5] A. Rahman, M. E. Hoque, F. Rahman, S. K. Kundu, and P. Gburzynski, "Enhanced Partial Dominant Pruning (EPDP) based Broadcasting in Ad hoc Wireless Networks," *Journal of Networks*, vol. 4, no 9, pp. 895-904, Nov. 2009. Article (CrossRef Link).

[6] C.-C. Shen, Z. Huang, and C. Jaikaeo, "Directional Broadcast for Mobile Ad hoc Networks with Percolation Theory," *IEEE Trans. Mobile Computing*, vol. 5, no. 4, pp. 317-332, Apr. 2006. Article (CrossRef Link).

[7] J. Wu and F. Dai, A "Generic Distributed Broadcast Scheme in Ad Hoc Wireless Networks," *IEEE Trans. on Computers*, vol. 53, no. 10, pp. 1343-1354, Oct. 2004. Article (CrossRef Link).

[8] H. Lim and C. Kim, "Flooding in wireless ad hoc networks," *Computer Comm. J.*, vol. 24, no.3-4, pp. 353-363, 2001. Article (CrossRef Link).

[9] W. Lou and J. Wu, "On Reducing Broadcast Redundancy in Ad hoc Wireless Networks," *IEEE Trans. on Mobile Computing*, vol. 1, no. 2, pp. 111-122, Apr.-June, 2002. Article (CrossRef Link).

[10] S.-C. Kim and J-.M. Chung, "Message Complexity Analysis of Mobile Ad Hoc Network Address Auto-configuration Protocols," *IEEE Trans. Mobile Computing*, vol. 7, no. 3, pp. 358-371, Mar. 2008. Article (CrossRef Link).

[11] S.-C. Kim and J. -M. Chung, "Analysis of Link Error Effects in MANET Address Autoconfiguration Protocols," *Journal of Communications and Networks*, vol. 11, no. 1, pp. 84-93, Feb. 2009. Article (CrossRef Link).

[12] Z. Shelby, C. Bormann, "6LoWPAN: The Wireless Embedded Internet," *John Wiley & Sons*, December 2009. Article (CrossRef Link).

[13] W. R. Heinzelman, A. Chandrakasan, and H.Balakrishnan, "An Application-Specific Protocol Architecture for Wireless Micro-sensor Networks," *IEEE Trans. Wireless Communications*, vol. 1, no. 4, pp. 660-670, Oct. 2002. Article (CrossRef Link).

[14] X. Li, P.-Y. Kong and K.-C. Chua, "TCP Performance in IEEE 802.11-Based Ad Hoc Networks with Multiple Wireless Lossy Links," *IEEE Trans. Mobile Computing*, vol. 6, no. 12, pp. 1329- 1342, Dec. 2007. Article (CrossRef Link).

Sang-Chul Kim is an Associate Professor in the School of Computer Science at Kookmin University, Seoul, Republic of Korea, since March 2006. He received the Ph.D. degree in the department of Electrical and Computer Engineering from the Oklahoma State University in 2005. He was a System Engineer in Samsung SDS and Aerospace from 1994 to 1999 and was an Instructor in the school of Computer Science at the Changwon National University from 1998 to 1999. His research is in the areas of mobile and ad hoc communication, Internet of things, and wireless sensor networks. He served as TPC Vice Co-Chairs of IEEE ICUFN 2015, Local Chair of IEEE VNC 2012, TPC member of IEEE ICUFN 2011, and Local Arrangement Co-Chairs of IEEE ICUFNs 2012, 2013, and 2014. He is a life member of the Korean Society for Internet Information (KSII) and Korean Institute of Communication and Information Sciences (KICS). In April 2015, he was nominated as one of the outstanding scholars in the release of the 2016 33rd edition of "Marquis Who's Who in the World 2016".