Regular Paper

Performance Analysis of Optimized Link State Routing-based Localization

Tomoya Takenaka,[†] Hiroshi Mineno,[††] Yuichi Tokunaga,[†††]
Naoto Miyauchi[†††] and Tadanori Mizuno[††]

Node localization obtained by estimating node positions is an essential technique for wireless multi-hop networks. In this paper, we present an optimized link state routing (OLSR)-based localization (ROULA) that satisfies the following key design requirements: (i) independency from anchor nodes, (ii) robustness for non-convex network topology, and (iii) compatibility with network protocol. ROULA is independent from anchor nodes and can obtain the correct node positions in non-convex network topology. In addition, ROULA is compatible with the OLSR network protocol, and it uses the inherent distance characteristic of multipoint relay (MPR) nodes. We reveal the characteristics of MPR selection and the farthest 2-hop node selection used in ROULA, and describe how these node selections contribute to reducing the distance error for a localization scheme without using ranging devices. We used a simulation to specify appropriate MPR_COVERAGE, which is defined to control the number of MPR nodes in OLSR, and give a comparative performance evaluation of ROULA for various scenarios including non-convex network topology and various deployment radii of anchor nodes. Our evaluation proves that ROULA achieves desirable performance in various network scenarios.

1. Introduction

Location information is an essential parameter for wireless multi-hop networks that include emerging adhoc and sensor networks. Several researchers have proposed location-based routing for adhoc networks to achieve efficient routing control [3)~6)]. In their algorithms, they assume that nodes are positioned in advance through some means such as using the Global Positioning System (GPS). In sensor networks, node positions determines the origin of an event, so localization techniques are of great interest. Moreover, location information is necessary for mobile computing and context-aware applications [1),2)]. One of the simplest solutions for localization is equipping each node with a GPS receiver. However, localization using GPS is infeasible for the two reasons. First, adding GPS devices to nodes increases their cost. Second, GPS cannot be used in areas where nodes cannot communicate with GPS satellites because of obstructions, for example, inside buildings.

In recent years, much research has been conducted on how to obtain node positions. Several localization schemes [10)~13),20)] assume that nodes are equipped with ranging devices, such as directional antennas or ultra-sound ranging devices. However, installing ranging devices is not suitable for large-scale networks due to the cost of the hardware.

In this paper, we study a localization technique with the following key design requirements:

(i) independency from anchor nodes[☆],
(ii) robustness for non-convex network topology,
(iii) compatibility with network protocol.

(i) Independency from anchor nodes is necessary due to the quantitative and geometric aspects of anchor nodes. In terms of the quantitative aspect, as mentioned earlier, increasing the number of anchor nodes increases the network cost. Second, in terms of the geometric aspect, localization algorithms using trilateration based on anchor nodes are limited to the geometric conditions of the anchor nodes. In other words, localization using trilateration based anchor nodes does not work when the deployment radius of the anchor nodes is restricted. This problem is related to the geometric dilution of precision (GDOP) [22)] which represents the error factor of positioning accuracy of trilateration obtained by calculating the geometric con-

[†] Graduate School of Science and Technology, Shizuoka University
[††] Faculty of Informatics, Shizuoka University
[†††] Mitsubishi Electric Corporation

[☆] An anchor node is a node that is known its actual position in advance through some system such as GPS.

ditions of the anchor nodes. We should not expect that anchor nodes can be deployed uniformly in a field, as anchor nodes equipped with GPS receivers are susceptible to obstructions.

(ii) Robustness for non-convex network topology is essential for localization. Before we give the formal definition of non-convex network topology, we introduce the definition of the convex set. Let $\overleftrightarrow{xy}$ denote line segment xy. For subset $A \in \mathbf{R^n}$, the convex set is satisfied with the following condition:

$$\forall x, y \in A \Longrightarrow \overleftrightarrow{xy} \subset A. \tag{1}$$

We define a set as non-convex if it does not satisfy the condition above Eq. (1). Consider a wireless network topology as a graph, $G = (V, E)$, where V is a set of nodes, and E is a set of links between (i, j), and where $i, j \in V$. Each node connects to another by limited communications ranges. If the nodes are deployed in a convex set, we call the network topology convex. If the nodes are deployed in a non-convex set, we call the network topology non-convex. A non-convex network topology can occur when nodes cannot be deployed in some areas because of obstructions, for example, buildings or natural features such as trees or mountains. We can easily imagine that the non-convex network topology occurs in many practical cases. Thus, robust performance for non-convex network topology is an essential requirement.

(iii) Compatibility between network and localization protocol is an essential scheme for the following reasons. A localization protocol is fundamentally composed of two steps: 1) measuring the distance and 2) positioning the node. The first step necessarily includes communicating with other nodes. If nodes can measure distances when they communicate with other nodes by using a message or process, they can integrate localization messages and network messages generated in the network layer. A localization scheme that is compatible with the network protocol enables the nodes to have a more efficient localization protocol. Therefore, a localization technique that uses the characteristics of the underlying network layer process is needed.

Here, we present an optimized link state routing (OLSR)-based localization (ROULA) that satisfies the key design requirements mentioned above. ROULA is independent of the anchor nodes and can obtain node positions in non-convex network topology. In addition, ROULA is compatible with the OLSR network protocol[23]. Each node in ROULA localizes nodes which are selected by the multipoint relay (MPR) selection used in OLSR and by the farthest 2-hop node selection developed for ROULA. One of our reasons for using OLSR in the network layer is the inherent distance characteristic of MPR nodes. We analyzed MPR selection and farthest 2-hop node selection, and exposed the relationship between connectivity, which shows how many nodes connect to other nodes in 1-hop on average, and node distances. Our analysis revealed that the characteristics of MPR selection and farthest 2-hop node selection contribute to reducing the distance error for a localization scheme without using ranging devices. In addition, we showed the effectiveness of MPR_COVERAGE, which controls the number of MPR nodes, and specified the appropriate MPR_COVERAGE for ROULA. Finally, we presented a comparable performance evaluation of ROULA in various network scenarios, and demonstrated that ROULA is independent of the anchor nodes and is robust in non-convex network topology.

This paper is organized as follows. Related work is reviewed in Section 2. The ROULA technique is described in Section 3. Section 4 presents the characteristics analysis of MPR selection and farthest 2-hop node selection. The performance evaluation of ROULA is described in Section 5. Section 6 concludes the paper with a brief summary and a mention of future work.

2. Related Work

Bulusu, et al.[7] proposed a simple localization method, called Centroid. In Centroid, anchor nodes flood beacon packets containing their own location information. Other unknown nodes then estimate their positions by using a centroid formula with overlapping received beacons. Since anchor nodes are needed to estimate node positions, Centroid can estimate only a few nodes when sufficient anchor nodes are not deployed in the network.

In many previous works, trilateration is used to estimate node positions. In DV-Hop[9], anchor nodes first estimate the average 1-hop node distance by propagating their location information to all other anchor nodes in the network. They can then use the average 1-hop distance to trilaterate unknown node positions. Similarly, in Amorphous[8], anchor nodes trilaterate node positions using the 1-hop node distance by calculating node density in the network. In Hop-TERRAIN[11], nodes first estimate initial node positions by trilateration using the hop count distances like DV-Hop. After that, the nodes improve their positions by trilaterating with ranging devices.

Priyantha, et al.[14] proposed an anchor-free localization scheme that does not require anchor nodes. Anchor-free localization (AFL) has two phases: in the first phase, node positions are roughly estimated without using ranging devices, and in the second phase, more accurate node positions are estimated using a ranging technique. In the first phase, the AFL selects five nodes that represent the x and y axes and the center of the network. Using these five nodes as reference nodes, AFL estimates all other node positions and assigns them relative coordinates using the specified formula[15]. Several authors[16)~19)] have proposed localization using multidimensional scaling (MDS). The MDS technique is used for converting the high-dimensional data into low-dimensional data. Since MDS can handle euclidean distances as input data, it can be applied to localization. MDS can also obtain relative node positions without using anchor nodes. Although we acknowledge at least two MDS-based algorithms[18),19)] can cope with non-convex network topology, ROULA has a better compatibility with the network protocol.

Lim and Hou[21] proposed a localization algorithm for non-convex network topology, called the proximity-distance map (PDM). In this algorithm, anchor nodes construct an optimal linear transformation that minimizes the mapping errors from a proximity matrix to a geographic distance matrix based on the anchor nodes. The nodes then estimate their positions by applying PDM to estimate the hop count distance to anchor nodes. The PDM needs anchor nodes to construct an optimal linear transformation.

3. Optimized Link State Routing-based Localization

3.1 Overview

The basic idea of ROULA is that each node matches regular triangles that form exactly convex curves, and grows them into global coordinates by merging overlapping regular triangles iteratively. **Figure 1** is a conceptual representation of ROULA in a non-convex network topology. As described earlier, one of the prior requirements for localization is robustness for non-convex network topology **(ii)**. A non-convex network topology appears to be a non-convex curve if the network is seen from a global point of view. However, if the network is viewed locally, each small set of the network, $P_i \subset V(i = 1, \ldots, n)$, appears to be a convex curve. In other words, a non-convex network topology is composed of partially convex curves. To find these convex curves, nodes in ROULA search for nodes that are arranged into regular triangles:

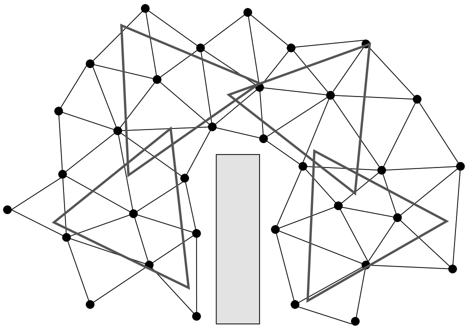


Fig. 1 Conceptual representation of ROULA in non-convex network topology.

$$\triangle\mathbf{T^r_{ABC}} \overset{\text{def}}{\Longleftrightarrow} \{\triangle ABC | AB = BC = CA, A, B, C \in V\}. \quad (2)$$

A node can estimate its positions relative to nodes A, B, and C on the vertices of regular triangles. This localization method satisfies the requirement of independency from anchor nodes **(i)**, because a node can determine the other nodes' positions relatively without using anchor nodes. Of course, other nodes could be found by using the pattern of v convex polygons ($v > 3$). However, finding more complex convex polygons would force the nodes to use a more complex procedure. We adopted a method that selects regular triangles for simplicity. In practical node placement, nodes may not find nodes that are arranged into ideal regular triangles. However, our approximation method using matching regular triangles was proven effective by simulation.

Lastly, we state requirement **(iii)**. Nodes in ROULA are assumed to use the OLSR protocol in the network layer. They localize MPR nodes as their 1-hop nodes without any modification of the MPR selection. Using the OLSR protocol has two advantages. First, as described in Section 3.3, the MPR selection used in OLSR has the inherent characteristic of reducing distance errors in localization without using ranging devices. Second, nodes in OLSR always hold and update the latest 2-hop node information and MPR nodes in a proactive action that periodically floods Hello packets. Therefore, flooding Hello packets and the computational task of MPR selection can be integrated by using the underlying network layer processes. These characteristics ensure that ROULA is compatible with the network protocol, and achieves an efficient localization scheme for wireless multihop networks.

In this work, we make the assumption that a sufficient distance of 2-hop paths are in the network to make 2-hop sized regular triangles. Nodes are deployed in a two-dimensional plane.

3.2 Algorithm

ROULA is operated as described below.

(1) **MPR selection:** Nodes flood Hello packets containing their own 1-hop nodes list to their 1-hop nodes. Once a node has a 2-hop nodes list, it selects MPR nodes.

(2) **Farthest 2-hop node selection:** Each node selects the farthest 2-hop node for each MPR node.

(3) **Matching regular triangle:** Nodes flood TRI_NOTICE packets to their farthest 2-hop nodes with their farthest 2-hop nodes list. Then, nodes that received TRI_NOTICE packets match regular triangles by using the received farthest 2-hop nodes lists. Next, nodes obtain local coordinates by merging their overlapping regular triangles.

(4) **Merging local coordinates:** Local coordinates are merged into one set of global coordinates. We assume a sink node merges all the maps of local coordinates in the network. A sink node floods MAP_REQ packets to all nodes in the network. Receiving nodes send back MAP_REP packets containing their local coordinates.

(5) **Converting to absolute coordinates:** Although the global coordinates are relative coordinates, if at least three anchor nodes are in the network, the relative coordinates can be converted into absolute coordinates that have the correct network orientation. This phase is optional.

Next, we give a detailed motivation for using MPR nodes for 1-hop node localization and explain in detail how ROULA achieves localization.

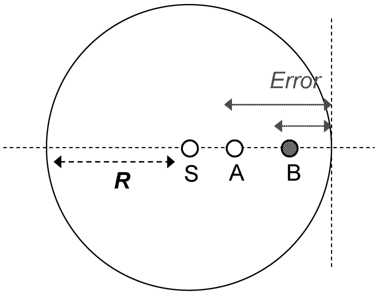


Fig. 2 Distances of 1-hop nodes.

3.3 MPR Selection

All nodes in ROULA must choose candidates out of all 2-hop nodes to make 2-hop regular triangles. However, which nodes should be chosen as 2-hop nodes? Here, to make this problem simple, we consider the distance between source and 1-hop nodes instead of the 2-hop node distance. **Figure 2** shows that node S with its communication range R has 1-hop nodes, A and B. Assume that when a node knows the length of its communication range, it estimates 1-hop node distances without using any ranging devices. The node regards the distance between the source and 1-hop nodes as R since it cannot measure the node distances. Therefore, if node S in Fig. 2 selects node B, which is closer to the radio boundary, rather than node A as a 1-hop node, the distance error is smaller than that of selecting node A. To find the node that is close to its radio boundary, we introduce the MPR selection used in OLSR[23].

MPR selection was developed for optimizing the relaying 1-hop node in the OLSR protocol. MPR selection finds the optimized 1-hop nodes, referred to as the MPR nodes, which are more accessible 2-hop nodes for relaying, and this can reduce the number of redundant retransmission nodes. Consequently, MPR selection can find nodes that are close to the radio boundary.

Here, we introduce some notations. Let $N(u)$ be the 1-hop nodes of node u. Let $N_2(u)$ define the 2-hop nodes of u. The set of MPR selected by node u is $MPR(u)$. For a node $v \in N(u)$, let $d_u^+(v)$ be the number of nodes of $N_2(u)$ that are in $N(v)$: $d_u^+(v) = |N_2(u) \cap N(v)|$. For a node $w \in N_2(u)$, let $d_u^-(w)$ be the number of nodes of $N(u)$ that are in $N(w)$: $d_u^-(w) = |N(u) \cap N(w)|$. The numbers in **Fig. 4** show the corresponding $d_S^-(w)$.

The algorithm for MPR selection is presented in **Fig. 3**. Here, we briefly explain the operation of MPR selection. A detailed descrip-

```
Algorithm MPR selection ($u \in V$)
1: for all nodes $v \in N(u)$ do
2:   if ($\exists w \in N(v) \cap N_2(u) | d_u^-(w) <$ MPR_COVERAGE) then
3:     Select $v$ as $MPR(u)$
4:     $N(u) \leftarrow N(u) - \{v\}$
5:     $N_2(u) \leftarrow N_2(u) - \{w\}$
6: while ($N_2(u) \neq \emptyset$) do
7:   for all nodes $v \in N(u)$ do
8:     if ($d_u^+(v) = \max_{w \in N(u)} d_u^+(w)$) then
9:       Select $v$ as $MPR(u)$
10:      $N(u) \leftarrow N(u) - \{v\}$
11:      if ($\exists s \in N(v) \cap N_2(u) || MPR(u) \cap N(s)| =$ MPR_COVERAGE) then
12:        $N_2(u) \leftarrow N_2(u) - \{s\}$
```

Fig. 3 Pseudo code for MPR selection.

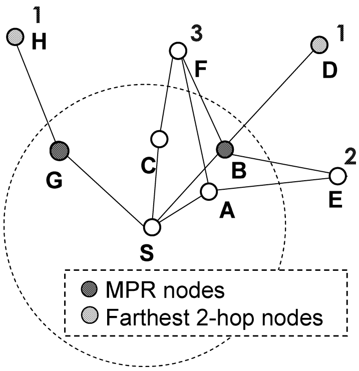


Fig. 4 Example node selection for MPR nodes and farthest 2-hop nodes (MPR_COVERAGE = 1).

tion of MPR selection can be found in Ref. 23). MPR selection is composed of two steps. The first step in lines 1–5 selects as MPR nodes such nodes $v \in N(u)$ that cover $N_2(u)$ connected by less than MPR_COVERAGE nodes in $N(u)$. MPR_COVERAGE is defined to control the number of MPR nodes so as to increase the redundancy of flooding in OLSR. MPR_COVERAGE can assume any integer value > 0. The second step in lines 6–12 selects the node that covers most nodes in $N_2(u)$, and it also ensures that 2-hop nodes are connected by at least MPR_COVERAGE nodes in $MPR(u)$. The second step finds the minimum number of MPR nodes. In MPR selection, finding 1-hop nodes that cover more 2-hop nodes causes the selection of the 1-hop nodes that are far away from the source node. Consequently, MPR nodes are closer to the radio boundary.

An example of MPR selection for node S when MPR_COVERAGE is set to 1 is presented in Fig. 4. Node B is selected as an MPR node in the second step since node B covers most of $N_2(S)$. Then, nodes D, E, and F, which are connected by at least MPR_COVERAGE (i.e., MPR_COVERAGE = 1) node in $MPR(S)$ are removed from $N_2(S)$. Loop in the second step, node G is selected as an MPR node since it covers node H, which is the remainder of $N_2(S)$. Node H is then removed from $N_2(S)$, and completes the MPR selection. In case of MPR_COVERAGE = 2, nodes G and B are selected as MPR nodes in the first step since nodes D and H are connected by only one (i.e., less than MPR_COVERAGE = 2) node, B and G, respectively. Nodes D and H are then removed from $N_2(S)$. Node A is then selected as an MPR node since it covers most of the remainder of $N_2(S)$ in the second step. Nodes E and F are then removed from $N_2(S)$ since they are connected by at least MPR_COVERAGE (i.e., MPR_COVERAGE = 2) nodes in $MPR(S)$, and complete MPR selection. Since MPR_COVERAGE can increase the number of MPR nodes, it affects how many nodes can make regular triangles in ROULA. A detailed description of MPR_COVERAGE's effectiveness is given in Section 5.

According to the analysis in Ref. 24), the computational complexity of MPR selection is $O(N_{2n}^2)$, where N_{2n} is the number of 2-hop and 1-hop nodes.

3.4 Farthest 2-hop Node Selection

In the farthest 2-hop node selection, each node finds the farthest 2-hop nodes of all their 2-hop nodes. Let $F2(u)$ be the set of the farthest 2-hop nodes of node u. $F2(u)$ are the farthest in $N_2(u)$ from u for each $MPR(u)$. For the farthest 2-hop node selection, nodes use the $d_u^-(w), w \in N_2(u)$ that was determined in MPR selection. Note that nodes in the farthest 2-hop node selection do not require any connectivity information other than the MPR selection. In Fig. 4, the number above each node shows the corresponding $d_S^-(w)$, or how many 2-hop nodes are covered by $N(S)$. As shown in Fig. 4, if the node distance from node S is farther, such as nodes S and D, $d_S^-(w)$ is small. This is because, with a uniform node density, the connectivity between 2-hop nodes and the source node is smaller when the node distance is farther. Based on this assumption, nodes select the farthest 2-hop nodes, as described in **Fig. 5**. For example, in Fig. 4, $F2(S)$ is node D and H

```
Algorithm Farthest 2-hop node selection (u ∈ V)
for all nodes $v \in MPR(u))$ do
  if $(d_u^-(z)_{z\in N(v)-N(u)} = \min d_u^-(z))$ then
    Select $z$ as $F2(u)$
```

Fig. 5 Pseudo code for farthest 2-hop node selection.

when MPR_COVERAGE is set to 1.

The computational complexity of the farthest 2-hop node selection can be computed by $O(M_1^2)$ since $d_u^-(w), w \in N_2(u)$ can be provided in MPR selection, where M_n denotes the number of shortest n-hop nodes.

3.5 Matching Regular Triangles

We illustrate how nodes are arranged into regular triangles in **Fig. 6**. Here, we focus on matching $\triangle T^r_{ABC}$ for node B. Each arrow in Fig. 6 shows the direction of a farthest 2-hop node; for instance, node A has three farthest 2-hop nodes B, C, and D. Nodes A and C flood TRI_NOTICE packets to the farthest 2-hop nodes containing their farthest 2-hop nodes list. The list in Fig. 6 shows the received farthest 2-hop nodes list for node B after the TRI_NOTICE packets are exchanged. Node B knows that nodes A and C selected node B as the farthest 2-hop node. Next, node B finds two combinations of $\triangle T^r_{ABC}$ by matching AC and CA in the received farthest 2-hop nodes list.

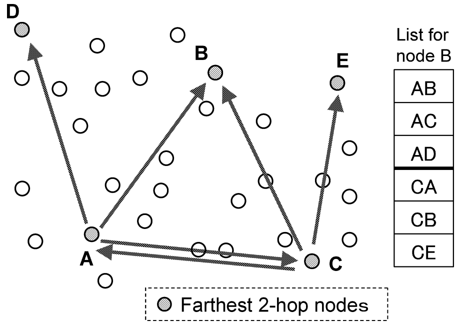


Fig. 6 Illustration of matching regular triangles for node B. The list shows the received farthest 2-hop nodes list for node B after receiving TRI_NOTICE packets from nodes A and C.

The regular triangles consist of the farthest 2-hop nodes and MPR nodes on the vertexes and edges. These nodes on the vertexes and edges of the regular triangles are given local coordinates on the basis of the coordination of relative regular triangles; which is to say, the farthest 2-hop nodes are positioned at $R \times 2$, and the MPR nodes are positioned at R ideally, where R is the length of communication range. However, as revealed in Section 4, this simple relative coordination is not suitable for practical node density. According to the analysis given later in Section 4, we introduced an approximate distance function for MPR nodes and the farthest 2-hop node, and applied it to the relative coordination of regular triangles.

The computational complexity of matching regular triangles is $O(l \log_2 l)$, where l is the size of the received farthest 2-hop node lists.

3.6 Merging Local Coordinates

```
Algorithm Merging local coordinates (u ∈ V)
 1: while $|LM(u)| = 1$ do
 2:   repeat
 3:     repeat
 4:       GET $CM_{LM(u)}(v|v \in LM(u))$
 5:     until (maximum no. of com. nodes $< th_{com}$)
 6:     repeat
 7:       $z, v \in CM_{LM(u)}(v)|\max|LC(z) \cap LC(v)|$
 8:       GET $\{s,t\} = \min_{s,t\in CN(z,v)} \mathbf{MERGE_{(s,t)}}(z,v)$
 9:     until (distance of $\{s,t\} < th_e$)
10:     until (distance of $\{s,t\} < th_e$)
11:     MERGE$_{(s,t)}$ $(z,v)$
12:     $LM(u) \leftarrow LM(u) - \{v\}$
```

Fig. 7 Pseudo code for merging local coordinates.

The protocol for merging local coordinates is as follows. Let $LM(u)$ be the maps of local coordinates held by node u. $LC(m)$ is the set of nodes in local coordinates of map m. $CN(z,v)$ is the set of common nodes of two maps $\{z,v\}$, or formally $CN(z,v) = \{LC(z) \cap LC(v)\}$. $CM_{LM(u)}(z)$ is the set of map combinations with the number of common nodes between maps z and all $v \in LM(u)$ sorted in descending order. Let $\mathrm{MERGE_{(p,q)}(z,v)}$ define merging two maps $\{z,v\}$ that are centered around nodes $\{p,q\}$. $\mathrm{MERGE_{(p,q)}(z,v)}$ returns the average distance errors among the common nodes.

The algorithm for merging local coordinates is presented as **Fig. 7**. The loop defined in lines 3–5 enables the node to find a map that has more common nodes with map v. If *maximum number of common nodes* of $CM_{LM(u)}(v)$ is less than a threshold value th_{com}, we set $th_{com} = 5$, the node continues to look for the common nodes with another map excluding map v. Next, in lines 6–9, the node searches for nodes that have minimum errors in common nodes between maps $\{z,v\}$. Once the node finds the nodes such that *distance of* $\{s,t\}$ is greater than th_e, we set $th_e = R \times 0.5$, and it merges the two maps. If *distance of* $\{s,t\}$ is less than th_e, the node tries to find the nodes that minimize the error in common nodes by using another map. If the node cannot find the nodes such that *distance of* $\{s,t\}$ is less than th_e in $CM_{LM(u)}(v)$, it tries to find nodes with another map in lines 2–10. When the node cannot find the nodes such that *distance of* $\{s,t\}$ is less than th_e in all $v \in LM(u)$, the merging halts on the way. Otherwise, the node lets $|LM(u)| = 1$ by iterating merging the maps, and the merging completes successfully.

Figure 8 illustrates the merging of local coordinates. We assume that the node A has several maps of local coordinates that consist of overlapping regular triangles as shown in top of Fig. 8. First, in lines 3–5, node A has $CM_{LM(A)}(v)$, and is ready to merge the local coordinates of $\triangle T^r_{ABC}$ and $\triangle T^r_{ABG}$ which are drawn with dashed lines in Fig. 8 because they have the maximum number of common nodes of all local coordinates. Next, in 6–11, the local coordinates are merged. In other words, the local coordinates of the common nodes A, B, D, E, and F at the bottom of Fig. 8 are averaged and those of the non-common nodes C and G are added. Merging of local coordinates is run iteratively until a node obtains one set of global coordinates.

The computational complexity of merging local coordinates is as follows. Finding the map that has the maximum number of common nodes is $O(\overline{LN}\ \overline{BM})$, where $\overline{LN}$ denotes the average number of nodes in local coordinates, and $\overline{BM}$ denotes the average number of maps to which each node belongs. The computational complexity of searching for the nodes that have the minimum distance errors in line 8 is $O(c^3)$, where c is the average number of common nodes. The value of c is usually six or more because our local coordinates has six nodes on the regular triangle initially. The total computational complexity of merging local coordinates is $O(|LM(u)|\alpha\ (\beta\overline{LN}\ \overline{BM} + \gamma c^3))$, where α, β, and γ are the average numbers of loops repeated in lines 2, 3, and 6, respectively. Although α, β, and γ add computational complexity to finding more common nodes, we have found α, β, and γ take small values. Since merging local coordinates can be implemented in a distributed manner, $|LM(u)|$ can take also a small value. The major computational complexity of ROULA is merging local coordinates. However, since each parameter determining the computational complexity of merging local coordinates is slow to increase as the number of nodes increases, merging local coordinates can be computed with small computational complexity while the computational complexity of $O(N^3_{2n})$, which is applying MDS to each node[18] rapidly increases in proportion to the number of nodes.

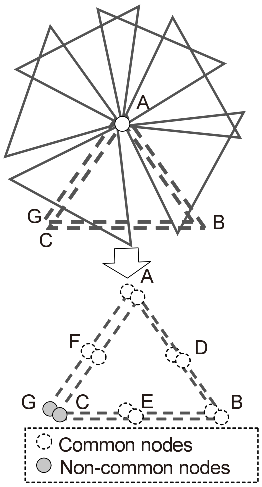


Fig. 8 Illustration of merging local coordinates.

4. Characteristics Analysis

4.1 Characteristics of MPR Selection

We investigated the distance characteristic and the number of selected nodes for the MPR selection and the farthest 2-hop node selection described in Sections 3.3 and 3.4. **Table 1** lists the simulation parameters assumed in the characteristics analysis. We placed the nodes randomly in a field with no obstructions and set the length of the communication range to 100 [m]. **Figure 9** presents the numerical results of calculating the average distance of all 1-hop nodes and MPR nodes with varying MPR_COVERAGE. We calculated the actual distances between the MPR nodes with varying MPR_COVERAGE and their source nodes, and averaged the results to analyze the distance characteristics for MPR nodes. The average distances are plotted with dots, and the standard deviations are plotted with lines. MPR_COVERAGE was varied from 1 to 4. Each result was plotted against connectivity.

As shown in Fig. 9, the average distances for the MPR nodes were closer to 100 than that for all 1-hop nodes. Furthermore, the standard deviations for the MPR nodes were smaller than

Table 1 Simulation parameters for characteristics analysis.

Field	500 × 500 [m]
Communication range	100 [m]
Number of nodes	60–230 (interval=10)
Node deployment	Random

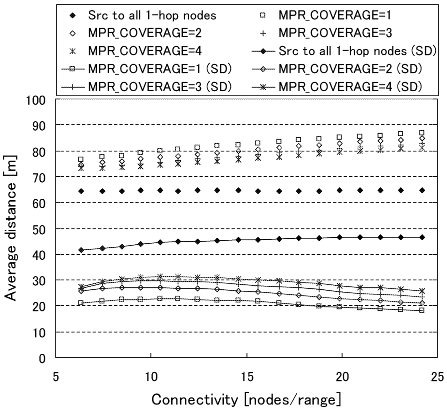


Fig. 9 Average distance and standard deviation of all 1-hop nodes and MPR nodes.

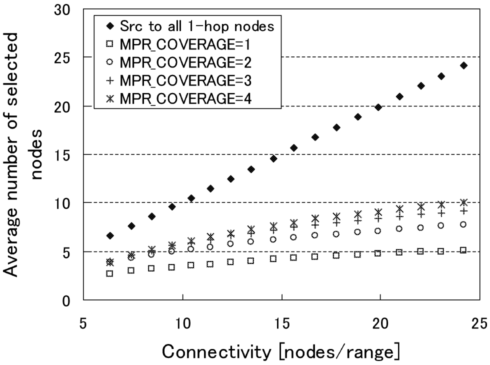


Fig. 10 Average number of all 1-hop nodes and MPR nodes.

that for all 1-hop nodes. These results demonstrate that MPR selection can select the nodes closest to the radio boundary with a small variance. For each MPR_COVERAGE, the average distances for the MPR nodes slightly decreased as MPR_COVERAGE was increased. The standard deviations for the MPR nodes for each MPR_COVERAGE increased slightly. This is because the nodes selected every node that had 2-hop nodes connected by less than MPR_COVERAGE nodes in the first step or at least MPR_COVERAGE nodes in the second step as MPR nodes regardless of the optimality.

Figure 10 plots the average number of selected MPR nodes with varying MPR_COVERAGE and of all 1-hop nodes per node. We calculated the number of nodes that each node selected as MPR nodes with varying MPR_COVERAGE, and averaged the results to analyze the characteristics of the selected number of MPR nodes. MPR selection can significantly reduce the number of 1-hop nodes used as relaying 1-hop nodes. This is a well known characteristic of OLSR, i.e., reducing the number of redundant retransmission nodes. Out of all MPR_COVERAGEs, MPR_COVERAGE = 1 produced the least number of MPR nodes. The other MPR_COVERAGEs (i.e., MPR_COVERAGE = 2, 3, and 4) slightly increased the number of MPR nodes as MPR_COVERAGE increased, respectively. This is because nodes select MPR nodes with redundancy in addition to optimized MPR nodes. In ROULA, the number of MPR nodes is related to the number of farthest 2-hop nodes and the possibility of matching regular triangles.

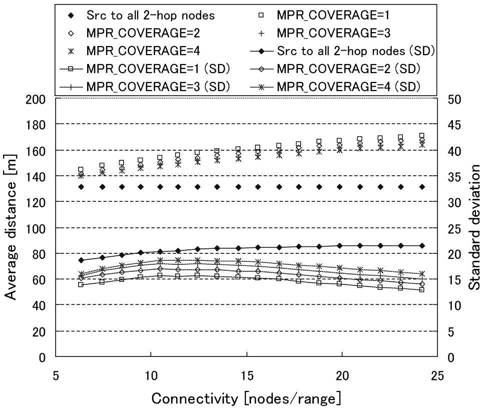


Fig. 11 Average distance and standard deviation of all 2-hop nodes and the farthest 2-hop nodes.

4.2 Characteristics of Farthest 2-hop Node Selection

Figure 11 plots the numerical results of the average distance of all 2-hop nodes and the farthest 2-hop nodes. We calculated the actual distances between the farthest 2-hop nodes and their source nodes, and averaged the results to analyze the distance characteristics for the farthest 2-hop nodes. The average distances of the farthest 2-hop nodes were closer to 200, which is the 2-hop ahead radio boundary of the source node, than those of all 2-hop nodes. Moreover, the standard deviations for the farthest 2-hop nodes were smaller than that for all 2-hop nodes. These results demonstrate that the farthest 2-hop node selection can select those nodes close to the 2-hop ahead radio boundary from a source node with small variance. As shown in Fig. 9 and Fig. 11, both MPR nodes and farthest 2-hop nodes are expected to reach their radio boundary as the number of nodes increases. However, the average distances of MPR nodes and farthest 2-hop nodes never reach 100 or 200 even if connectivity is high. This is because the possibility of selecting nodes exactly at the radio boundary is small in practical node density. Therefore, nodes cannot select nodes at the radio boundary. Here in ROULA, to minimize the distance error of selecting nodes simply, we previously conducted linear approximations of these distances from source nodes to MPR nodes and farthest 2-hop nodes, and nodes used these approximations when they assigned local coordinates in matching regular triangles. To model the rigorous distance characteristics of MPR nodes and farthest 2-hop nodes is beyond the scope of this work and is left for future work.

5. Performance Evaluation

5.1 Simulation Parameters

We evaluated the performance of ROULA in a simulator. The simulation environment we used was a discrete event simulation environment, OMNeT++[25] with Mobility Framework[26]. **Table 2** shows the simulation parameters. We distributed anchor nodes in a circle that was centered at $(100, 100)$ with a variable deployment radius, R_d, to evaluate how the geometric conditions of the anchor nodes affect localization algorithms. We used $\overline{\text{GDOP}}$, obtained by averaging all GDOP[22], as a parameter to indicate the geometric conditions of the anchor nodes defined as the following equation.

$$GDOP = \sqrt{\frac{N_A}{\sum_{i \in S_A} \sum_{j \in S_A, j>i} A_{ij}^2}}, \qquad (3)$$
$$A_{ij} = \sin \theta_{ij}.$$

In Formula (3), N_A indicates the number of anchor nodes, and S_A is the set of anchor nodes, and A_{ij} is the angle from an unknown node to anchor nodes $\{i, j\}$. We also defined the obstruction $[height, width]$ in a way that shows the obstruction size. We placed the obstruction in the middle of the field and set its *width* to 100. We varied the *height* of the obstruction to evaluate a non-convex network topology, and used curvature $c(t)$ as a parameter to indicate the degree of a non-convex network topology defined as the following equation.

Table 2 Simulation parameters.

Field	500 × 500 [m]
Communication range	100 [m]
Number of nodes	90–270 (interval=30)
Node deployment	Random
Center of anchor nodes deployment	(100, 100)
Deployment radius of anchor nodes	$100 \leq R_d \leq 400$ (interval=50)
Obstruction [height, width]	$0 \leq$ height ≤ 300 (interval=50) width = 100 [m]

$$c(t) = \frac{ab}{(a^2 \sin^2 t + b^2 \cos^2 t)^{3/2}}. \qquad (4)$$

In Formula (4), we set $a = \frac{width}{2}$, $b = height$, and $t = \frac{\pi}{2}$. A circle formed by the obstruction bends sharply at a point, so it has high curvature. We defined the positioning error as the distance between the actual and estimated node positions, and normalized it by the communication range. We defined the coverage as the percentage of nodes that could localize themselves out of all nodes. We assumed symmetrical link communication with a fixed range. We compared ROULA's performance with that of other localization techniques, Centroid[7], DV-Hop[9], and AFL[14]. We ran the simulations 50 times with random seeds, and plotted the averaged results.

5.2 Results for various MPR_COVERAGEs

First, we investigated how MPR_COVERAGE affects the localization performance of ROULA in a field with no obstructions, and specified appropriate MPR_COVERAGE. **Figure 12** presents the coverage for various MPR_COVERAGEs. The coverage increased as MPR_COVERAGE increased. As revealed in Section 4, the larger the MPR_COVERAGE, the more MPR nodes per node, so the nodes can match more regular triangles. A node can thus more easily find a map combination for merging local coordinates. Consequently, coverage increases as MPR_COVERAGE increases. The coverages of ROULA for MPR_COVERAGEs were lower when connectivity was lower because ROULA cannot find a sufficient number of regular triangles when the node density is low. Increasing the number of nodes increases the number of common nodes, which are used when

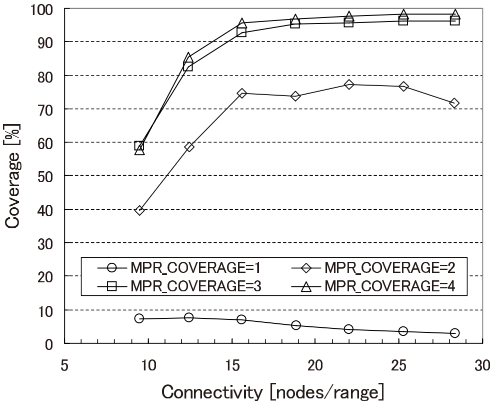


Fig. 12 Coverage for various MPR_COVERAGEs.

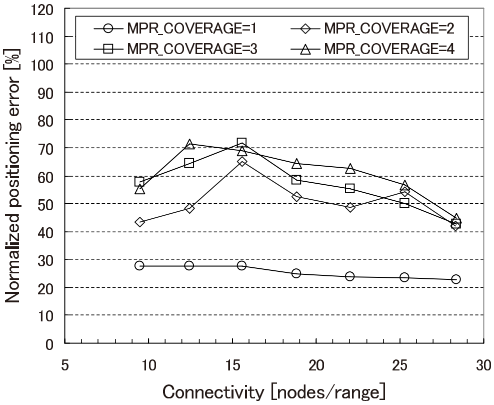


Fig. 13 Normalized positioning error for various MPR_COVERAGEs.

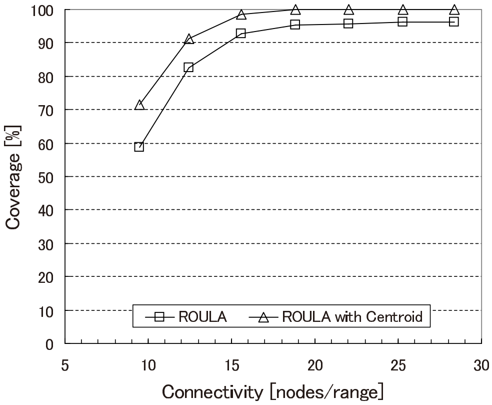


Fig. 14 Coverage after executing Centroid (MPR_COVERAGE = 3).

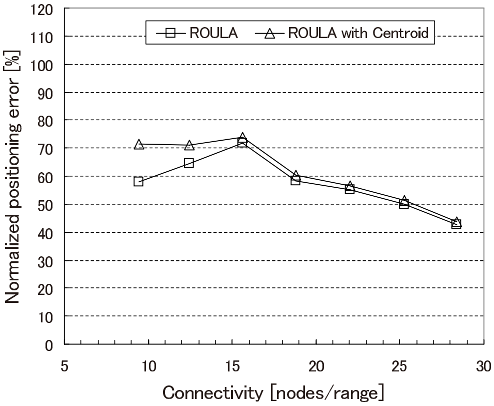


Fig. 15 Normalized positioning error after executing Centroid (MPR_COVERAGE = 3).

merging local coordinates. Therefore, coverages improve with an increase in the number of nodes. The coverages were almost 100% except for MPR_COVERAGE = 1 and 2, when connectivity was over 15. ROULA never achieved 100% even if connectivity was high enough. This is because ROULA does not localize all nodes in local coordinates. However, 100% coverage can be achieved if the nodes execute Centroid [7] after finishing a sink node merging phase. Considering each MPR_COVERAGE, MPR_COVERAGE = 1 does not work for localization because the coverage is too poor. While coverage was much better for MPR_COVERAGE = 2, MPR_COVERAGE = 4 improved the coverage slightly compared with MPR_COVERAGE = 3. On the other hand, **Fig. 13** gives the positioning errors for each MPR_COVERAGE. Except for MPR_COVERAGE = 1, the positioning errors increased slightly as MPR_COVERAGE increased. This is because, as presented in Fig. 9 and Fig. 11, the nodes increase the distance errors when estimating 1-hop and 2-hop nodes since the standard deviation of MPR nodes and farthest 2-hop nodes increases. Therefore, MPR_COVERAGE should be kept as small as possible to obtain better positioning accuracy. In terms of network protocol, MPR_COVERAGE is a tradeoff between the broadcast efficiency and the robustness. Reducing MPR_COVERAGE reduces network traffic. For the above reasons, we specified an appropriate value for MPR_COVERAGE as 3 because it results in better positioning accuracy and coverage with smaller MPR_COVERAGE.

Figures 14 and **15** show the coverage and positioning error obtained by executing Centroid after the nodes estimated their positions by ROULA with MPR_COVERAGE = 3. To execute Centroid, a sink node floods the node positions in the network. As shown in Fig. 14, 100% coverage is achieved if the nodes execute Centroid after ROULA. Figure 15 shows that executing Centroid has little effect on the posi-

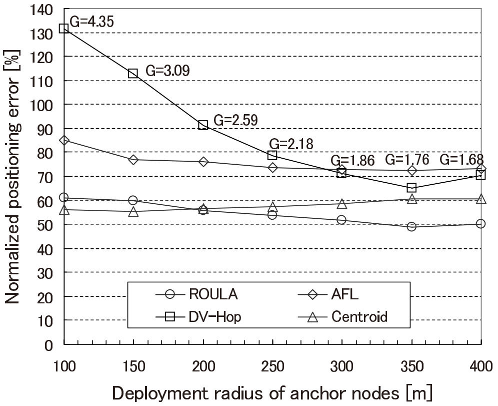


Fig. 16 Normalized positioning error for various deployment radii of anchor nodes (240 nodes and 5 anchor nodes).

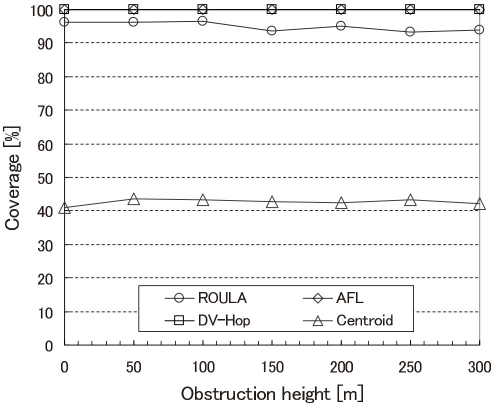


Fig. 17 Coverage for various deployment radii of anchor nodes (240 nodes and 5 anchor nodes).

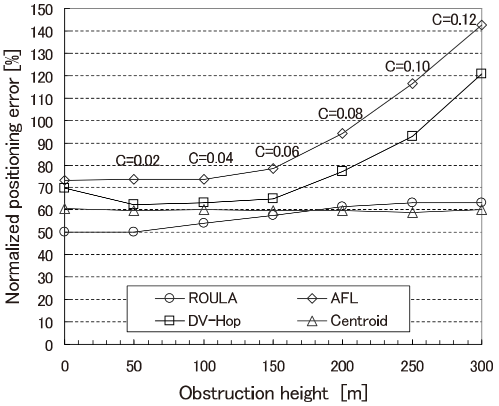


Fig. 18 Normalized positioning error for various obstruction heights (240 nodes and 5 anchor nodes).

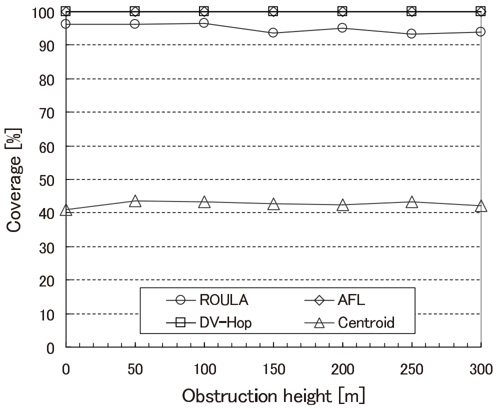


Fig. 19 Coverage for various obstruction heights (240 nodes and 5 anchor nodes).

tioning error.

5.3 Results for Various Deployment Radii of Anchor Nodes

Figures 16 and **17** give the normalized positioning error and coverage for various deployment radii of anchor nodes for 240 nodes and 5 anchor nodes. $\overline{\text{GDOP}}$, represented by G, is plotted against deployment radius of anchor nodes. As shown in Fig. 16, DV-Hop had the highest positioning error when the deployment radius of anchor nodes was 100. This is because DV-Hop trilaterates nodes based on anchor nodes, so the positioning accuracy is affected by GDOP, which is large when the anchor nodes are close together. Thus, DV-Hop is infeasible when the deployment of anchor nodes is limited. ROULA, AFL, and Centroid had better positioning accuracy regardless of various deployment radii of anchor nodes. Although Centroid had good positioning accuracy, it had the worst coverage, as shown in Fig. 17. The coverage of Centroid depends on the number of anchor nodes. The positioning error for ROULA and AFL slightly decreased as the deployment radius became larger. This is because ROULA and AFL become more robust against errors when a node converts to absolute coordinates, as the deployment radius of anchor nodes is larger.

5.4 Results for Various Obstruction Heights

Figures 18 and **19** present the results for normalized positioning error and coverage for various obstruction heights for 240 nodes and 5 anchor nodes. Curvature represented by C is plotted against obstruction height. DV-Hop and AFL had the higher positioning errors when the obstruction height was large. As the obstruction height increased, the curvature forming the non-convex network topology became steeper. Therefore, DV-Hop and AFL were directly affected by the curvature. On the contrary, ROULA achieved about 50–60% positioning errors regardless of the curvature of non-convex network topology.

As a whole, our performance evaluation demonstrated that ROULA can accurately estimate node positions in various scenarios including a non-convex network topology and a limited deployment radius of anchor nodes. Although ROULA had the positioning errors of about 40–50% at least, nodes such as in the sensor networks should be able to identify the origin of an event with better accuracy in the scenarios we assumed. Evaluating the performance of a network protocol based on the location information provided by ROULA remains as future work.

6. Conclusion

In this paper, we presented ROULA, a localization algorithm that satisfies the following key design requirements: **(i)** independency from anchor nodes, **(ii)** robustness for non-convex network topology, and **(iii)** compatibility with network protocol. ROULA is compatible with the OLSR network protocol, and it uses characteristics of the underlying network layer process. We revealed the characteristics of the MPR selection and the farthest 2-hop node selection that contribute to reducing the distance error for a localization scheme without using ranging devices. Using simulation, we first investigated the appropriate MPR_COVERAGE, which is defined to control the number of MPR nodes in OLSR, and found that MPR_COVERAGE = 3 is preferable for ROULA. We compared the performance of ROULA with that of other localization techniques and found that it achieved desirable performance in various network scenarios including limited deployment radius of anchor nodes and a non-convex network topology.

In this work, we assumed that nodes have a fixed communication range to enable us to determine the basic ROULA performance. A validation of the optimal value of MPR_COVERAGE considering an unstable communication range and a performance evaluation of ROULA in such a range are our future work. As a possible solution for unstable communication range on localization, removing the asymmetric link communication which incurred unstable communication range by using the mechanism identifying asymmetric link communication in OLSR is expected to mitigate the localization error. We plan to evaluate ROULA in a simulator with more realistic parameters.

References

1) Leonhardt, U. and Magee, J.: Towards a General Location Service for Mobile Environments, *Int. Workshop on Service in Distributed and Networked Environments*, pp.43–50 (1996).

2) Harter, A., Hopper, A., Steggles, P., Ward, A. and Webster, P.: The Anatomy of a Context-Aware Application, *ACM/IEEE MobiCom*, pp.59–68 (1999).

3) Navas, J. and Imielinski, T.: GeoCast - Geographic Addressing and Routing, *ACM/IEEE MobiCom*, pp.66–76 (1997).

4) Ko, Y. and Vaidya, N.H.: Location-Aided Routing (LAR) in Mobile Ad Hoc Networks, *ACM/IEEE MobiCom*, Vol.6, pp.307–321 (1998).

5) Bose, P., Morin, P., Stojmenovic, I. and Urrutia, J.: Routing with Guaranteed Delivery in Ad hoc Wireless Networks, *Int. Workshop Descrete Algorithms and Methods for Mobile Computing Communications*, Vol.7, pp.609–616 (1999).

6) Karp, B. and Kung, H.T.: GPSR: Greedy Perimeter Stateless Routing for Wireless Networks, *ACM/IEEE MobiCom*, pp.243–254 (2000).

7) Bulusu, N., Heidemann, J. and Estrin, D.: GPS-less Low Cost Outdoor Localization For Very Small Devices, *IEEE Personal Communications Magazine*, Vol.7, No.5, pp.28–34 (2000).

8) Nagpal, R.: Organizing a Global Coordinate System from Local Information on an Amorphous Computer, *A.I. Memo 1666*, MIT A.I. Laboratory (Aug. 1999).

9) Niculescu, D. and Nath, B.: Ad Hoc Positioning System (APS), *IEEE Globecom*, Vol.5, pp.2926–2931 (2001).

10) Savvides, A., Han, C. and Strivastava, M.B.: Dynamic Fine-grained Localization in Ad-hoc Networks of Sensors, *ACM/IEEE Mobicom*, pp.166–179 (2001).

11) Savarese, C., Rabaey, J. and Langendoen, K.: Robust Positioning Algorithms for Distributed Ad-Hoc Wireless Sensor Networks, *USENIX Technical Annual Conference*, pp.317–327 (2002).

12) Nasipuri, A. and Li, K.: A Directionality Based Location Discovery Scheme for Wireless Sensor Networks, *ACM Int. Workshop on Wireless Sensor Networks and Applications*, pp.105–111 (2002).

13) Niculescu, D. and Nath, B.: Ad Hoc Positioning System (APS) Using AoA, *IEEE Infocom*, Vol.3, pp.1734–1743 (2003).

14) Priyantha, N.B., Balakrishnan, H., Demaine, E. and Teller, S.: Anchor-free Distributed Localization in Sensor Networks, *Technical Report TR-892*, MIT LCS (2003).

15) Priyantha, N.B., Balakrishnan, H., Demaine, E. and Teller, S.: Mobile-assisted Localization in Wireless Sensor Networks, *IEEE Infocom*, Vol.1, pp.172–183 (2005).

16) Shang, Y., Ruml, W., Zhang, Y. and Fromherz, M.P.J.: Localization from Mere Connectivity, *ACM Mobihoc*, pp.201–212 (2003).

17) Raykar, V., Kozintsev, I. and Lienhart, R.: Position Calibration of Audio Sensors and Actuators in a Distributed Computing Platform, *ACM Multimedia*, pp.572–581 (2003).

18) Shang, Y. and Ruml, W.: Improved MDS-based Localization, *IEEE Infocom*, Vol.4, pp.2640–2651 (2004).

19) Ji, X. and Zha, H.: Sensor Positioning in Wireless Ad-hoc Sensor Networks using Multidimensional Scaling, *IEEE Infocom*, Vol.4, pp.2652–2661 (2004).

20) Moore, D., Leonard, J., Rus, D. and Teller, S.: Robust Distributed Network Localization with Noisy Range Measurements, *ACM Sensys*, pp.50–61 (2004).

21) Lim, H. and Hou, J.C.: Localization for Anisotropic Sensor Networks, *IEEE Infocom*, Vol.1, pp.138–149 (2005).

22) Spirito, M.A.: On the Accuracy of Cellular Mobile Station Location Estimation, *IEEE Trans. on Vehicular Technology*, Vol.50, pp.674–685 (2001).

23) Clausen, T. and Jacquet, P.: Optimized Link State Routing Protocol (OLSR), IETF RFC 3626 (2003).

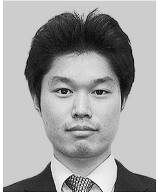
24) Adjih, C., Jacquet, P. and Viennot, L.: Computing Connected Dominated Sets with Multipoint Relays, *Int. Journal of Ad Hoc and Sensor Wireless Networks*, Vol.1 (2005).

25) OMNeT++ Descret Event Simulation System. http://www.omnetpp.org/

26) Mobility Framework for OMNeT++. http://mobility-fw.sourceforge.net/

(Received December 5, 2006)
(Accepted June 5, 2007)
(Online version of this article can be found in the IPSJ Digital Courier, Vol.3, pp.541–554.)

Tomoya Takenaka received the B.S. and M.S. degrees in computer science from Shizuoka University, Japan in 2005 and 2006, respectively. He is currently working towards the Ph.D. degree at Shizuoka University. His current research interests include localization and network protocol in wireless multi-hop networks. He has been with Mitsubishi Electric Research Laboratories (MERL), Massachusetts, USA, as an intern since April 2007. He is a member of IPSJ.



Hiroshi Mineno received his B.E. and M.E. degrees from Shizuoka University, Japan in 1997 and 1999, respectively. In 2006, he received a Ph.D. degree in Information Science and Electrical Engineering from Kyushu University, Japan. Between 1999 and 2002 he was a researcher in the NTT Service Integration Laboratories. In 2002, he joined the Department of Computer Science of Shizuoka University as an Assistant Professor. His research interests include heterogeneous network convergence as well as wireless sensor networks. He is a member of IEEE, ACM, IEICE, and IPSJ.

Yuichi Tokunaga graduated from Science University of Tokyo, Tokyo, Japan, in 1990. Since he joined Mitsubishi Electric Corporation in 1990, he has been engaged in research and development on embedded computers and systems in the Mitsubishi Electric Corporation Information Technology R&D Center. His current R&D activities focus on time-synchronization and localization of wireless sensor networks. He is a member of IPSJ.

Naoto Miyauchi is a senior engineer of Kobe Works, Mitsubishi Electric Co., Since joining the company in 1987, he had been engaged in the researches on implementation of OSI protocols, directory services, network management systems and ad-hoc networks. Currently he has been engaged in developments for the electric power control systems. He receives the B.E. from Chuo University in 1987.

Tadanori Mizuno received the B.E. degree in industrial engineering from the Nagoya Institute of Technology in 1968 and received the Ph.D. degree in computer science from Kyushu University, Japan, in 1987. In 1968, he joined Mitsubishi Electric Corp. Since 1993, he is a Professor of Shizuoka University, Japan. Now, he is a Dean of graduate school of Science and Technology of Shizuoka University. His research interests include mobile computing, distributed computing, computer networks, broadcast communication and computing, and protocol engineering. He is a member of Information Processing Society of Japan, the institute of electronics, information and Communication Engineers, the IEEE Computer Society and ACM.