Efficient Navigation for Unmanned Agents in Sparse Wireless Sensor Networks

Donghoon Kim

Department of Aerospace Engineering and Engineering Mechanics, University of Cincinnati, Cincinnati, OH 45221, USA

Key Words: Sparse Wireless Sensor Networks, Extended Graph, Localizable Area, Navigation

1. Introduction

Many works have reported on various sensor network position estimation methods based on the relative distance measurement that can be used when the global navigation satellite system is environmentally denied or degraded.1-3 Among others, trilateration algorithms are widely adopted because of their simple principle.4-6 However, the algorithms possibly fail if the sensors have a low range of communication or the environment includes obstacles.7 Typically, such distance-based localization algorithms are used to construct a globally rigid network.8,9 In other words, albeit each sensor, called node herein, has a limited transmission range, unmanned agents, like unmanned aerial vehicles (UAVs), should be inside the coverage space to receive the sensors’ information.10 Therefore, the typical algorithm requires a network that can adequately cover a certain area and must be capable of communicating with at least three sensors at any point in the area. However, such a network is not always guaranteed.

This study proposes a strategy to maximize UAV’s navigation in a sparse wireless sensor network (SWSN) in the manner of the shortest distance travel. The overlapping (or localizable) area, which is calculated using the positions of three disks constructed by the sensor’s transmission range, is used to characterize the possibility of localizing UAVs through trilateration. To ensure that a UAV travels from a starting point to a destination point via the localizable area, it must pass the points that are defined by a sensor set, called vertices. The keys are to find such vertices to define a graph that is flexible to various network complexities that are determined by the combination of sensors and reduce the number of search nodes or the total distance. To determine the shortest path, the Dijkstra algorithm,11,12 one of the most widely used algorithms, is applied with proper modifications. The feasibility of the proposed method is verified through two-dimensional (2D) and 3D examples.

2. Method

The trilateration method estimates positions of stationary and/or mobile nodes using distance measurements from at least three sensors.4 Figure 1 represents a 3D WSN constructed with three omnidirectional sensors, and the shaded overlapping area that varies depending on both the configuration of the sensor and the operation altitude denotes a localizable area. Given sensor i’s position, $x_i = [x_i, y_i, z_i]^T \in \mathbb{R}^3$, where $i = 1, 2, \ldots, N$, the target’s position, $x_t = [x_t, y_t, z_t]^T \in \mathbb{R}^3$, in the localizable area is calculated using the method of least squares:13

$$x_t = [A^TA]^{-1}A^Tb,$$  \hspace{1cm} (1)

where $A \in \mathbb{R}^{N \times 3}$ and $b \in \mathbb{R}^N$. For example, if $N = 3$, then,

\[
A = 2 \begin{bmatrix}
(x_2 - x_1) & (y_2 - y_1) & (z_2 - z_1) \\
(x_3 - x_2) & (y_3 - y_2) & (z_3 - z_2) \\
(x_1 - x_3) & (y_1 - y_3) & (z_1 - z_3)
\end{bmatrix} \in \mathbb{R}^{3 \times 3},
\]

\[
b = \begin{bmatrix}
x_2^2 - x_1^2 + y_2^2 - y_1^2 + z_2^2 - z_1^2 + d_1^2 - d_2^2 \\
x_3^2 - x_2^2 + y_3^2 - y_2^2 + z_3^2 - z_2^2 + d_2^2 - d_3^2 \\
x_1^2 - x_3^2 + y_1^2 - y_3^2 + z_1^2 - z_3^2 + d_3^2 - d_1^2
\end{bmatrix} \in \mathbb{R}^3,
\]

where $d_i$ indicates the distance between $x_i$ and $x_t$ that are given as measurements. Note that it must satisfy either $z_1 \neq z_2$ or $z_2 \neq z_3$ or $z_3 \neq z_1$, and this constraint is usually satisfied in real-world environments due to uneven ground surfaces.

Figure 2 represents an example of vertices and the corresponding adjacency matrix that generates possible paths for a UAV’s navigation. It starts with marking every vertex as unvisited and inserting all vertices in the unvisited set. The temporary distances of all unvisited neighbors are calculated with the distance between the vertices of the graph based on the adjacency matrix. The newly calculated temporary distance less than the current value is assigned as a new value of the next vertex, $u$, which has the shortest path.

Fig. 1. A localizable area described by three omnidirectional sensors.
minimum value. After moving to the next vertex, the previous vertex is changed to the visited vertex, and it is removed from the vertex set, \( Q \), where it would not be checked again. This process is expressed as pseudo-code described in Algorithm 1, where \( \text{dist}[p] \) and \( S \) obtained through the algorithm represent the minimum distance and the path to go to the destination, respectively.

The shortest-path problem can be defined for a graph (e.g., undirected, directed, and mixed). In particular, an undirected graph refers to a case where each of the edges does not contain a direction. The undirected graph is an ordered pair \( \langle V, E \rangle \) comprising, where \( V \) is a set of vertices and \( E \) is a set of edges connecting two arbitrary vertices \( v_i \) and \( v_j \). \( V \) and \( E \) are expressed as

\[
V = \{v_1, v_2, \cdots, v_n\} \subseteq \mathbb{R}^{3 \times n},
\]

\[
E \subseteq \{(v_i, v_j)| (v_i, v_j) \in V^2 \wedge i \neq j\} \subseteq \mathbb{R}^{n \times n},
\]

where \( v \in \mathbb{R}^3 \) is the position vector of vertex, \( n \) is the number of vertices, and the subscripts \( i \) and \( j \) are the positive integers up to \( n \). Note that \( V' \) represents the set of vertices that including all the new vertices. In this study, each vertex is formed by three sensors for trilateration, unlike the simplification of the sensor forming the overlapping area. Since the coverage shape is not a circle but a circular triangle, \( r \) cannot be defined as the constant value. Therefore, the criterion for \( r \) that determines the edge is different for each vertex, and it is found as follows:

\[
r = y_{i,j} + y_{j,i}
\]

where \( y_{i,j} \) represents the maximum distance from the 3th localizable area to \( v_j \).

As explained with Fig. 2, it is very possible that overlapping areas are not fully connected from the starting point to the destination point in SWSN environments. In other words, the path to the target point is not created as the vertices of the defined graph, so it requires extending the graph by finding feasible vertices. For example, if the sensors are being distributed on the ground, then lowering the altitude of UAVs increase the coverage area made by the omnidirectional sensors. That is, the unconnected area can be connected by lowering the altitude of the UAVs.
ple of a side-view for the extended graph according to altitude changes. A set of new vertices, \( \{v_{n+1}, \ldots, v_{n+m}\} \), consists of the maximum altitude for connecting the unconnected consecutive area (i.e., \( v_4 \) and \( v_5 \)), where \( n \) and \( m \) are the number of current and new vertices, respectively, and \( n = 3 \) and \( m = 2 \) are used in this example. Let a UAV travels from \( v_1 \) to \( v_3 \). Then, a set of vertices, \( \{v_1, v_2, v_3\} \), is formed at the same altitude where the UAV locates initially. Given such a current set of vertices, the only possible path is drawn with a black-colored line. However, this is not a feasible path because the edges, \( e_{1,2} \) and \( e_{2,3} \), do not fully lie within the overlapping area. By changing the altitude of the UAV, new vertices, \( v_4 \) and \( v_5 \), can be found, and these newly defined vertices provide more degree-of-freedoms to generate navigation path candidates for the UAV, such as blue and green lines. Obviously, the green-colored path is the shortest feasible path here, and the proposed strategy provides such a result.

Figure 4 illustrates the architecture of the proposed strategy that is composed of the following: i) determining the connection of adjacent areas and ii) finding new vertices. First, the object to which the algorithm is applied must be defined. The parameters for clarifying the object are: i) the designated region, ii) the type and number of sensors that are deployed, and iii) the locations of the sensors. In this area, the starting and destination points, which can be changed according to the mission to be executed and the reference altitude corresponding to the positions, should be set. In the graph, an adjacency matrix is constructed based on the Euclidean distance of the sensor, but adjacent vertices beyond a certain distance (twice of sensor range) are included as unconnected. Therefore, the overlapping areas need to be analyzed for finding localizable areas considering the coverage of three sensors. The center of each localizable area is a vertex, and the localizable area is shaped like a circular triangle. Note that the size, shape, and angle of the area are determined by the positions of the three sensors that form the area. Therefore, the edge connecting each vertex is based on the Euclidean distance, but the criterion for determining whether connected or not is changed. Through this process, one can obtain a graph that considers the localizable area of the designated area. However, in the case of SWSN, if the graph obtained according to that degree is incompletely connected to the destination point, the point cannot be reached.

In summary, it presents an extended-graph approach with vertices added to make the incompletely connected graph complete. The appropriate altitude, at which the contact points of two adjacent areas, is calculated using the property whose coverage range varies according to the altitude and set this contact point as a vertex. Obviously, the coverage of the localizable area is the largest at the altitude at which the sensor is located, but it limits the minimum altitude depending on the characteristics of terrains. The navigation path to the destination point is planned to use a set of newly defined vertices and edges.

3. Results

In an SWSN, the shortest-path problem that moves from
the starting point to the destination point set in a 3D space is solved through the localizable space. This method works by changing the altitude of each UAV optimally. Furthermore, in order to travel to the destination while maintaining the required altitude, the changeable localizable space according to the altitude is analyzed. To analyze the feasibility of the proposed concept, it assumes that 25 range sensors with omnidirectional coverage (300 m) are randomly distributed in the designated area (1000 m × 1000 m), the altitude of each sensor is constant, and the location is known. The adjacency matrix, $E$, of the graph is set as the distance between each vertex. However, assuming that each vertex has a uniformed coverage range, $E$ is considered that the connection is not established if it exceeds the applicable range.

Figure 5 shows the path comparison results according to whether the proper $r$ is applied or not in a 2D SWSN. As explained, finding the criterion for $r$ is important to obtain the shortest feasible navigation path. The localizable area (shaded with the grey color) generated by the randomly distributed sensors marked with ‘$*$’ differs depending on the combination of the sensors, and the vertices found are marked with ‘$+$’ at the center of the circular triangle (shaded with the purple color). The black path indicates the shortest but not the feasible path (path length: 1448 m) found from the starting point (1) to the target point (109) without considering such a criterion. The blue path (path length: 1758 m) that uses the constant $r$ value is included to highlight the importance of determining the proper $r$. This path looks like created almost within a localizable area, but it does include a non-feasible sub-route that is marked with the green circle. On the other hand, the adjacency matrix constructed by the proposed method guarantees the feasible shortest paths (path length: 1821 m) because it determines the possibility of inter-area movement.

Depending on the specifications of the range sensor with omnidirectional coverage, the size of the localizable space varies with the operating altitude of UAVs. Therefore, subject to the combination of the sensors positioned randomly, area(s) not connected to the movement from the predetermined operating altitude to the destination may be included. However, if an intersection between unconnected areas is created by changing altitude, a UAV can navigate within the localizable area by including potential vertices into path planning. Figure 6 shows that no feasible path can be planned at the reference altitude of 150 m because of the incompletely connected area from the starting point (1) to the destination (118). By including the optimally found vertices at the different altitudes, it is shown that the feasible shortest navigation path (path length: 1951 m) is achieved. Here, a user-defined discrete altitude change (50 m) is considered to prove the feasibility of the proposed approach. It is important to note that different shortest paths can be found according to altitude change variations.

4. Conclusion

This study presents an extended-graph approach by defined with an adjacency matrix that takes into account the shape and size of the various localizable areas formed by trilateration. By using the proposed matrix, the method avoids the wrong path planning that occurs when using the existing matrix applying the distance-based criterion. In addition, if no path exists to the destination in a sparse wireless sensor network, a new feasible shortest path is planned considering the potential vertices generated according to the altitude changes. This will be useful to assist unmanned agents’ navigation for missions in GPS denied/degraded environments through either locally or globally available wireless sensor networks composed of a group of spatially dispersed and dedicated sensors. The future study includes accuracy and uncertainty analyses to understand the influence on the navigation path in cluttered environments.
References

1) Liu, Y., Yang, Z., Wang, X., and Jian, L.: Location, Localization, and Localizability, *J. Comput. Sci. Technol.*, **25**, 2 (2010), pp. 274–297.
2) Moore, D., Leonard, J., Rus, D., and Teller, S.: Robust Distributed Network Localization with Noisy Range Measurements, Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems, 2004, pp. 50–61.
3) Park, S. and Yi, S.: Least-square Matching for Mobile Robot Slam Based on Line-segment Model, *Int. J. Control Automation Systems*, **17**, 11 (2019), pp. 2961–2968.
4) Eren, T., Goldenberg, O. K., Whiteley, W., Yang, Y. R., Morse, A. S., Anderson, B. D. O., and Belhumeur, P. N.: Rigidity, Computation, and Randomization in Network Localization, IEEE INFOCOM 2004, Vol. 4, 2004, pp. 2673–2674.
5) Hendrickson, B.: Conditions for Unique Graph Realizations, *SIAM J. Computing*, **21**, 1 (1992), pp. 65–84.
6) Yang, B., Guo, L., Guo, R., Zhao, M., and Zhao, T.: A Novel Trilateration Algorithm for RSSI-based Indoor Localization, *IEEE Sensors J.*, **20**, 14 (2020), pp. 8164–8172.
7) Oliva, G., Pantziar, S., Pascucci, F., and Setola, R.: Sensor Networks Localization: Extending Trilateration via Shadow Edges, *IEEE Trans. Automatic Control*, **60**, 10 (2015), pp. 2752–2755.
8) Jackson, B. and Jordán, T.: Connected Rigidity Matroids and Unique Realizations of Graphs, *J. Combinatorial Theory, Series B*, **94**, 1 (2005), pp. 1–29.
9) Laman, G.: On Graphs and Rigidity of Plane Skeletal Structures, *J. Engineering Mathematics*, **4**, 4 (1970), pp. 331–340.
10) Chang, W.-L., Zeng, D., Chen, R.-C., and Guo, S.: An Artificial Bee Colony Algorithm for Data Collection Path Planning in Sparse Wireless Sensor Networks, *Int. J. Machine Learning Cybernetics*, **6**, 3 (2015), pp. 375–383.
11) Dijkstra, E. W.: A Note on Two Problems in Connexion with Graphs, *Numerische Mathematik*, **1**, 1 (1959), pp. 269–271.
12) Paige, P. and Kruskal, C.: Parallel Algorithms for Shortest Path Problems, Proc. the 1985 International Conference on Parallel Processing, 1985, pp. 14–20.
13) He, S., Shin, H.-S., Xu, S., and Tsourdos, A.: Distributed Estimation over a Low-cost Sensor Network: A Review of State-of-the-art, *Information Fusion*, **54** (2020), pp. 21–43.

Masayuki Sato
Associate Editor