A Two-tier Positioning Algorithm for Wireless Networks with Diverse Measurement Types

Zimu Yuan†, Shuhui Yang§, Wei Li†
†Institute of Computing Technology, Chinese Academy of Sciences, China
§Department of Mathematics, Computer Science, and Statistics, Purdue University Calumet
yuanzimu@ict.ac.cn, yang246@purdue.edu, liwei@ict.ac.cn

Abstract—A variety of measurement methods have been developed to obtain positions of nodes in wireless networks. However, most of existing positioning methods can only use limited types of measurement. These algorithms fail to exploit arbitrary measurement methods to improve the accuracy of positioning. In this paper, we propose a high-accuracy positioning algorithm which can utilize unlimited types of measurement. Our algorithm first selects a set of nodes equipped with more accurate measurement mechanisms to locate nodes whose positions are unknown. By this selection, a system of heterogeneous equations is built. Then, our algorithm uses the classical genetic method to solve equations which commonly are hard to solve. The comprehensive simulation shows that our algorithm significantly improves the accuracy of positioning compared with existing methods.

I. INTRODUCTION

Position information can be used to support many network functions, such as geographic routing [1], topology maintaining [2], coverage [3] and boundary detection [4]. These functions are fundamental for wireless network from the short-range bluetooth to the long-range telecommunication networks. Thus, various types of measurement such as time of arrival (ToA) [5] / time difference of arrival (TDoA) [10], angle of arrival (AoA) [11], radio signal strength (RSS, including bluetooth [17], telecommunication [18] and so on), coverage area (Area) [19], hop distance (Hop) [22] and neighbor proximity (Neighbor) [25] are developed to obtain the position information of nodes in wireless networks.

Though there are many types of measurement, most of existing positioning algorithms [31] can only utilize limited types of measurement. For example, the multilateration algorithm [6] depends on distance-based measurements such as ToA/TDoA, and is not compatible with AoA, Area or Neighbor measurements. Other positioning algorithms can combine two or three types of measurement [27][28][29][30], but they cannot utilize all types of measurement. Some device-free positioning algorithms [22][23][24] can work with unlimited types of measurement, but their accuracy for positioning is low because they can not fully utilize the merit of different types of measurement. In fact, there is still a lack of high-accuracy positioning algorithms that can exploit all types of measurement in wireless networks.

In this paper, we propose a new algorithm called Two-tier Selection Algorithm, which combines different types of measurements to calculate positions of nodes in wireless networks. This algorithm includes two tiers of process. In the first tier, after nodes with known positions apply all available measurements to calculate the location for nodes with unknown positions, a measurement selection strategy is designed to select a group of measurements among those calculated positions with higher accuracy. These group of measurements form a heterogenous equation system. Therefore, the second tier of our algorithm is to solve the system and calculate location. The challenge here is how to solve the heterogenous system effectively and efficiently. For our algorithm, we use the genetic method, a well-known optimization method, to solve the above challenging problem. By using the two tier of process, our algorithm can iteratively execute from a small set of nodes with known locations until all nodes are located.

The contribution of this work includes three aspects: 1) To the best of our knowledge, this is the first work to explore the possibility of including unlimited types of measurements and fully exploit the merit of each to improve the accuracy of node positioning in wireless networks; 2) a two-tier selection positioning algorithm that takes advantage of unlimited types of measurement is developed, which can locate all nodes in the network with a high accuracy; and 3) comprehensive experiments are conducted and the proposed algorithm is verified to have better performance in terms of positioning accuracy compared with other algorithms.

The rest of the paper is organized as follows: Section II introduces the related work; Section III presents the proposed two-tier selection algorithm; Section IV evaluates the two-tier selection algorithm through comprehensive simulation; and Section V concludes the work.

II. RELATED WORK

According to the ability to work with diverse measurement, existing positioning algorithms can be classified into two types: one that can work with limited types of measurement and one that can work with unlimited types.

Most existing positioning algorithms (Survey [31]) belong to the first type. That is, many algorithms only work with one type of measurement and others work with two or three types such as ToA/AoA [27], ToA/RSS [28], TDoA/AoA [29] and ToA/TDoA [30]. Commonly, the above algorithms can utilize the merit of accurate measurement types and provide accurate positioning, but they cannot work on the network with measurement types out of the above specified types.

Hop-based algorithms [22][23][24][26] and Neighbor Proximity algorithm [25] belong to the second type, which can work with unlimited types of measurement due to their device-free feature. As a matter of fact, these algorithms use hop-
distance or neighbors’ positions instead of more accurate measurement types such as AoA. That is, these algorithms can not utilize the merit of different measurement and hence have positioning results with low accuracy.

Compared to the above methods, the algorithm we propose can not only work on unlimited types of measurement but also fully exploit the merit of each measurement to achieve high accuracy of positioning. In fact, this is the fundamental difference between our algorithm and the hop-based algorithm or Neighbor Proximity algorithm.

III. TWO-TIER SELECTION ALGORITHM

Basically, our algorithm is motivated by the simple consideration that, in an environment with diverse measurement types, we should be able to use more accurate measurement to obtain more accurate positioning results. In addition, by iteratively selecting known nodes to calculate unknown nodes, we can finally obtain the positions of all nodes. By selecting various measurements, we actually set up a system of heterogenous equations. Therefore, a key challenge of our algorithm is to solve the heterogenous equation system efficiently. For this, our idea is to use the genetic method to solve the system, which needs carefully design of selecting operators [33]. Before presenting our algorithm, we first give the network model used for wireless network.

A. Network Model

We use a graph \( G = \langle V, E \rangle \) to model a wireless network with \( N \) nodes, where \( V \) is the set of wireless nodes and \( E \) is the set of edges, i.e., the connection between nodes. It is assumed that the graph is connected. Before the execution of the positioning algorithm, there are two types of nodes. One is called beacon node whose locations are already known. All the other nodes are target nodes with unknown locations. All possible measurements can then be applied for positioning. Then measurement results are collected and sent to a node or a server, which will execute our algorithm based on the above measurement results.

B. Tier 1: Process of Measurement Selection

In this process, there are two key parts: one is to assign weight values for different types of measurement; the other one is to select a sufficient number of measurements to construct a system of equations that calculate positions.

1) Assigning Weight to Different Measurements: Suppose that there are \( i \) beacon nodes, \( n_1, n_2, ..., n_i \), collaboratively measuring the location of a target node, \( n_t \). Let \( m \) and \( W_m \) denote this measurement and its weight. \( W_{n_1}, W_{n_2}, ..., W_{n_i} \) are the weighted values of these beacon nodes. We first define a transitional weighted value \( W_{t_{n,m}} \) as

\[
W_{t_{n,m}} = \min \{W_{n_1}, W_{n_2}, ..., W_{n_i}\} \times W_m \tag{1}
\]

in which, the minimal value is selected from the weighted value of Beacon nodes \( n_1, n_2, ..., n_i \). Then, it is multiplied by \( W_m \) to calculate the transitional weighted value \( W_{t_{n,m}} \).

At the beginning of our algorithm, beacon nodes and target nodes are assigned with weighted value \( 1 \) and \( 0 \), respectively. During node positioning, the target nodes are gradually become beacon nodes with updated weighted value. This procedure iterates and the algorithm terminates when all nodes in the network become beacon nodes.

The weighted value of a measurement \( m \) is assigned based on its accuracy. Table I is the list of common measurement types based on existing research. The weighted values of these measurement methods have a relationship of \( 1 \geq A_{ToA}, A_{TDoA}, A_{AoA} > A_{RSS}, A_{Area} > A_{Hop} > A_{Nbr} > 0 \) according to the accuracy listed in Table I. Therefore, a measurement of higher accuracy is assigned with an higher weighted value. Additionally, any newly-proposed measurement type can be added to table I later.

For example, in Fig. 1, beacon node \( n_1 \) (with known location \( X_{n_1} = [x_{11}, x_{12}] \)) applies the ToA measurement on target node \( n_t \) with location \( X_{n_t} = [x_{t1}, x_{t2}] \). Let \( dist_{n_1,n_t} \) denote the distance between these two
nodes, and the relationship between these two nodes can be written as an equation
\[ \sqrt{(x_{11} - x_{11})^2 + (x_{12} - x_{11})^2} = \text{dist}_{n_1 n_1}. \] (2)

The transitional weighted value of this measurement is
\[ W_{t_1m_{0T}} = W_{n_1} * A_{T_{ToA}}. \]

Beacon node \( n_2 \) (with known location \( X_{n_2} = [x_{21}, x_{22}] \)) applies AoA measurement on target node \( n_1 \). The relationship between these two nodes can be written as an equation
\[ \tan \theta_2 = (x_{21} - x_{11})/(x_{22} - x_{12}) \] (3)
which has \( W_{n_1n_2_{0A}} = W_{n_2} * A_{AoA} \).

In Fig. 2, beacon node \( n_1(X_{n_1}) \), \( n_2(X_{n_2}) \) and \( n_3(X_{n_3}) \) jointly apply the Area measurement on target node \( n_t(X_{n_t}) \) by using the centroid of overlap coverage area of node \( n_1 \), \( n_2 \) and \( n_3 \). The relationship between these four nodes can be written as an equation
\[ X_{n_t} = \text{Centroid}(\text{Overlap}(\text{Area}(X_{n_1}), \text{Area}(X_{n_2}), \text{Area}(X_{n_3}))) \] (4)
which has the transitional weighted value, \( W_{t_{n_1m_{Area}}} = \min\{W_{n_1}, W_{n_2}, W_{n_3}\} * A_{Area} \).

2) Selecting A Sufficient Number of Measurement: For a target node \( n_t \) with an unknown location, it can get a group of \( j \) transitional weighted values of measurements. These values then are sorted into descending order, \( W_{t_{n_1m_1}} \geq W_{t_{n_1m_2}} \geq ... \geq W_{t_{n_1m_j}} \). We choose \( C \) maximum values from them, \( C < j \). These \( C \) measurement results can then form a system of equations for solving the position of \( n_t \). The numerical value of \( C \) is determined by the computability of the measurements on node \( n_t \). Here, the meaning of computability is how many measurements are sufficient to calculate the position of unknown nodes. For example, as shown in Table I, at least \( C = 3 \) equations of distance types (ToA/TDoA, RSS, Hop) are required to calculate a target node in a 2D network. To calculate, the value chosen for \( C \) depends on the surrounding environment of node \( n_t \). For example, in Fig. 1, there are \( C = 4 \) equations of ToA, AoA and RSS measurements selected. Since just two equations of TOA and AOA measurements are not sufficient to calculate \( n_t \), two equations of RSS measurement of the same weighted value need to be selected. If node \( n_3 \) changes to use the AoA measurement instead of RSS, then only \( C = 3 \) equations of node \( n_1 \), \( n_2 \) and \( n_3 \)’s measurement are needed.

After the \( C \) transitional weighted values are selected, an elected weighted value of node \( n_t \), \( W_{t_{n_t}} \), is defined as
\[ W_{t_{n_t}} = \min\{W_{t_{n_1m_1}}, W_{t_{n_1m_2}}, ..., W_{t_{n_1m_C}}\} \] (5)
in which \( W_{t_{n_t}} \) is the minimum value of \( C \) selected transitional weighted values. It can be interpreted as the weighted value of the most inaccurate measurement in these \( C \) values. The calculation error of a target node’s position mainly depends on the error of the most inaccurate measurement. Therefore, \( W_{t_{n_t}} \) can be used to indicate the magnitude of the final calculation error of node \( n_t \). For example, in Fig. 1, we assume that the weighted values are \( W_{n_1} = W_{n_2} = W_{n_3} = W_{n_4} = 1, A_{T_{ToA}} = 1, A_{AoA} = 1 \) and \( A_{RSS} = 0.1 \). Then, we get \( W_{t_{n_t}} = W_{n_3}A_{RSS} = W_{n_4}A_{RSS} = 0.1 \).

After a period of time in the positioning process, assume that there are still \( k \) target nodes needed to be located. Let \( W_t \) denote the maximal elected weighted value of these target nodes, which can be written as
\[ W_t = \max\{W_{t_{n_1}}, W_{t_{n_2}}, ..., W_{t_{n_k}}\} \] (6)
Suppose the node \( n_{max} \) has the maximal weighted value among these \( k \) target nodes. This indicates \( n_{max} \) has the lowest calculation error among these nodes. Then, the \( C \) measurement results on \( n_{max} \) are selected to calculate its position in the second tier of the positioning algorithm. In fact, the \( C \) measurement will construct a system of measurement equations which is sufficient to find the solution for the position of unknown nodes. In the next subsection, we will discuss how to solve the system efficiently.

C. Tier 2: Process of Solving Equations

After the selection process of the first tier, \( C \) equations of measurements are selected to construct a system of heterogeneous equations in (7). Assume there are \( p \) unknown variables in this equation system, i.e., \( X = [x_1, x_2, ..., x_p] \). Here, \( p \) could be 2 in a 2D case and 3 in a 3D case. We formalize the equation system as
\[
\begin{align*}
  f_1(x_1, x_2, ..., x_p) &= 0 \\
  f_2(x_1, x_2, ..., x_p) &= 0 \\
  &... \\
  f_C(x_1, x_2, ..., x_p) &= 0
\end{align*}
\] (7)
Since errors exist in these \( C \) equations, it cannot always find a \( X \) that makes all equation established. So solving this system is equivalent to an optimization problem of the following
\[
\begin{align*}
  \text{find: } X &= [x_1, x_2, ..., x_p], X \in \Phi \\
  \min: f(X) &= \sum_{q=1}^{C} f_q^2(X)
\end{align*}
\] (8)
in which, \( \Phi \) denotes the solution interval of this system. If \( f(X) \) gets minimum value at point \( X' \), then \( X' \) is a solution of the equation system in (7).

Usually, there are two ways to solve the problem in (8). One way is to apply iterative methods. Iterative methods such as the most commonly used Newton-Raphson method. However, if the initial guessed point does not belong to the convergence domain, the method will fail to get the result. Since the \( C \) equations are derived from various measurement methods such as ToA, Angle and Area, these \( C \) equations have heterogeneous characteristics for iterative methods. This will make iterative methods difficult to converge. Other approaches include direct numerical optimization methods such as the steepest descent method, simplex method and Powell method [32]. These methods also have the problem of high sensitivity of the initial guess and poor convergence.

From the knowledge of optimization theory, we know that the genetic method [33] has global convergence and strong robustness, which helps solve the problem in (8). In fact, in order to use the genetic method to calculate the location of the target node, we need to design three operators: crossover, mutation and tournament selecting operators. Next, we will
introduce our design for the genetic method. Before presenting any operators, we need to do some initial work. That is, at the initial part, a polygon boundary is consisted by the coordinates of nearby beacon nodes. The initial population, \(X_{0,1}, X_{0,2}, \ldots, X_{0,g}\), for the genetic method is randomly generated in this boundary limit. The equation in (8) is taken as an evaluation function to evaluate the fitness of individuals such as \(X_{0,1}\), which indicates the 1st individual of the initial population. After the initialization, we are ready to present our design of the operators for the genetic method.

The first operator we need to determine is the crossover operator. This operator is used for generating the next generation population. Assume that the crossover process is applied between individual \(X_{i,j}\) and \(X_{i,k}\). The crossover selecting operator is designed as

\[
\begin{align*}
X'_{i+1,j} &= \alpha X_{i,j} + (1 - \alpha)X_{i,k} \\
X'_{i+1,k} &= (1 - \alpha)X_{i,j} + \alpha X_{i,k}
\end{align*}
\]  

in which, \(\alpha \in [0, 1]\) is a randomly selected value.

The second operation is the mutation selecting operator, which is used for maintaining the genetic diversity of the next generation population. The traditional genetic algorithm has the problem of slow convergence and premature. This is mainly due to the mutation operation having no obvious effect to overcome the local minimum point. Thus, we have designed a new random direction selecting operator of mutation. Let \(X'_{i+1,j}\) be the selected mutation individual of the next generation and \(d_{i+1,j}\) be the randomly generated mutation direction. The mutation process is a one-dimensional search operation that takes \(X'_{i+1,j}\) as the starting point and seeks the most fitting individual along the direction \(d_{i+1,j}\). The process can be formalized as

\[
\min \Phi(\lambda) = f(X'_{i+1,j} + \lambda d_{i+1,j})
\]

in which, \(\lambda\) denotes the step length in direction \(d_{i+1,j}\) and \(\Phi\) is the solution interval as was previously mentioned. In fact, here we use the fibonacci search \([32]\) to find the optimal step length \(\lambda_0\). Then, the new individual \(X_{i+1,j}\) is written as

\[
X_{i+1,j} = X'_{i+1,j} + \lambda_0 d_{i+1,j}
\]

Finally, we need to design the tournament selecting operators which is used for choosing better fitting individuals out of the next generation population. Let \(t\) denote the tournament scale. The tournament selecting operator chooses the most fitting individual out of \(t\) randomly selected candidates as the individual of the next generation. We adopt the elitist strategy here to avoid the bad influence on many random factors. The best fitting individual of current generation does not participate in crossover and mutation process. It is set as the winner in the choosing process of tournament selecting operator and used to replace the worst individual of the next generation.

Based on the above three operators, the genetic method will repeatedly generate a new generation until two termination conditions are met. The first condition is the repeat times that has reached the upper limit of generation \(U\). The other condition is the best fitting individual \(X\) of the current generation substitutes its value into equations (7) and get the error \(\epsilon = \max\{f_1(X), f_2(X), \ldots, f_C(X)\} < \epsilon_0\). In Section IV, \(U\) is set to 100 and \(\epsilon_0\) is set to \(10^{-3}\). The two-tier selection algorithm is formally presented in Algorithm 1.

### Algorithm 1: Two-tier Selection Algorithm

**Input:** The position \(X\) of all beacon nodes, All measurement results, The weight \(W_m\) of each measurement type, The upper limit of generation \(U\), The error \(\epsilon_0\)

**Output:** The position \(X\) of all nodes

```plaintext
for each node \(n\) do
  if \(n\) is beacon node then
    The weight \(W_n = 1\);
  else
    The weight \(W_n = 0\);
end

while still have target nodes do
  for each target node \(n_t\) do
    for all measurement \(m\) on \(n_t\) do
      \(W_{tn,m} = \min\{W_{n_1}, W_{n_2}, \ldots, W_{n_t}\} \times W_m\);  
    end
    Select \(C\) maximal transitional weight measurement;
    \(W_{tn} = \min\{W_{tn,m_1}, W_{tn,m_2}, \ldots, W_{tn,m_C}\}\);
    \(\epsilon_{max} = \arg\max\{W_{tn}\}\);
    Establish the system of \(C\) equations to locate \(n_{max}\);
    Randomly generate the initial population \(X_0 = \{X_{0,1}, X_{0,2}, \ldots, X_{0,g}\}\);
    Set the generation index \(i = 0\);
    while \(i < U\) do
      Crossover operator randomly select \(X_{i,j}, X_{i,k}\);
      \(X'_{i+1,j} = \alpha X_{i,j} + (1 - \alpha)X_{i,k}\);
      \(X'_{i+1,k} = (1 - \alpha)X_{i,j} + \alpha X_{i,k}\);
      Mutation operator randomly select \(X_{i+1,j}\);
      Mutation direction \(d_{i+1,j}\) randomly generate;
      \(\lambda_0 = \arg\min f(X'_{i+1,j} + \lambda d_{i+1,j})\);
      \(X_{i+1,j} = X'_{i+1,j} + \lambda_0 d_{i+1,j}\);
      Tournament operator does selection;
      Get the best individual \(X\) in \(i\)th generation;
      \(\epsilon = \max\{f_1(X), f_2(X), \ldots, f_C(X)\}\);
      if \(\epsilon < \epsilon_0\) then
        break;
      end
      \(i = i + 1\);
    end
    The position \(X\) of node \(n_{max}\) is calculated out;
    Set \(n_{max}\) as beacon node;
    The Weight \(W_{n_{max}} = W_t\);
  end
return The position \(X\) of all nodes;
```

Since most of the existing algorithms can not work on unlimited types of measurement, in our experiments, we
select the classical hop-distance-based multilateration system [22] (refer to Section II) as a baseline to compare with our algorithm. In order to verify the effectiveness of this selection for our algorithm, we also have designed a random algorithm that can randomly decide the next node to be calculated instead of the weighted value selection process in the first tier and randomly determines the next generation instead of selecting the operators’ calculation.

In our experiments, we use Positioning Error ($PE_{n_t}$) to indicate the accuracy of node $n_t$’s position. That is, we have

$$PE_{n_t} = \sqrt{(x_1' - x_1)^2 + (x_2' - x_2)^2 + \ldots + (x_p' - x_p)^2}$$

where $X' = [x_1', x_2', \ldots, x_p']$ is the calculated position and $X = [x_1, x_2, \ldots, x_p]$ is the true position.

### A. Simulation Settings

In order to investigate the adaptation of our algorithm, we provide three network topologies for our experiments: the square, the ring and the E shape network of 200 nodes (shown in Fig. 3, 4 and 5), which are randomly generated. The square network is $250m \times 250m$. The ring network is with the radius of $160m$ in the outer cycle and $65m$ in the inner cycle. The E network is with the height of $340m$ and the maximal width of $240m$. The signal coverage of each node in these networks is randomly generated. The measurement types of high, median and low accuracy in Table I are set with the error range of 5%, 25% and 40%, respectively. For the parameters of the second tier, the selecting operator calculation is set with the values that the population scale is 50, the crossover probability is 0.6, the mutation probability is 0.05 and the termination condition is the upper limit of generation $U = 100$ or the maximal error of equations $\epsilon_0 < 10^{-3}$. 

In our experiments, we use Positioning Error ($PE_{n_t}$) to indicate the accuracy of node $n_t$’s position. That is, we have

$$PE_{n_t} = \sqrt{(x_1' - x_1)^2 + (x_2' - x_2)^2 + \ldots + (x_p' - x_p)^2}$$

where $X' = [x_1', x_2', \ldots, x_p']$ is the calculated position and $X = [x_1, x_2, \ldots, x_p]$ is the true position.
B. Simulation Results

Given the above configuration, in our experiments we mainly explore two factors that affect the accuracy of positioning: the measurement error and the number of beacon nodes.

1) Effects of Measurement Error: Fig. 6, 7 and 8 show the positioning error of the case of square shape network with 16 beacon nodes. In these figures, the x and y axis indicates the changing of percentage of low and median accuracy measurements, respectively. (The percentage of high accuracy of measurements is \( \text{pct}_{\text{high}} = 1 - \text{pct}_{\text{low}} - \text{pct}_{\text{median}} \).) It can be observed in Fig. 6 and 7 that the maximum positioning error for both the two-tier selection and the random algorithm is where \( \text{pct}_{\text{low}} \) approaches to 100%, and the minimum positioning error is where \( \text{pct}_{\text{high}} \) approaches to 100%. The two-tier selection algorithm is better than the two-tier random algorithm, especially where \( \text{pct}_{\text{high}} \) is low. The hop-distance algorithm is device-free and uses hop-based distance instead of measurements of high accuracy, so its error (Fig. 8) remains the same no matter how the percentage of accuracy of measurements changes.

2) Effects of Various Numbers of Beacon Nodes: The comparison of different algorithms are shown in Fig. 9, 10 and 11. The positioning error is averaged from all the cases of percentage of high, median and low accuracy measurements changing; i.e., the value of all of the points on the surface in Fig. 6 is averaged and the average value is just the point of 16 beacon nodes on the error curve of the two-tier selection algorithm in Fig. 9. For all positioning errors in Fig. 9, 10 and 11, the two-tier selection algorithm is better than the hop-distance algorithm with an average 29.8m less error, and it is better than the two-tier selection algorithm with an average 7.7m less error. We can see that the two-tier selection algorithm can achieve high accuracy in these different shapes of square, ring and E networks, which confirms the algorithm's stability.

V. CONCLUSION AND FUTURE WORK

A two-tier selection algorithm is proposed for positioning in wireless networks. It can combine all different possible measurement methods to jointly calculate the position of nodes with high accuracy. Its effectiveness is verified by measuring the error of node positioning in experiments. In the future, we will focus on further improving the performance of the two-tier selection algorithm in terms of accuracy and time complexity.

ACKNOWLEDGMENT

This work is supported in part by the Hi-Tech Research and Development (863) Program of China (Grant No. 2013AA01A212, 2013AA01A209), the National Basic Research (973) Program of China (Grant No.2011CB302502,2011CB302803), and the major national science and technology special projects (Grant No. 2010ZX03004-003-03).

REFERENCES