Geographic Routing Algorithm with Location Errors

Yuanwei JING†, Nonmember and Yan WANG†(a), Student Member

SUMMARY  Geographic routing uses the geographical location information provided by nodes to make routing decisions. However, the nodes can not obtain accurate location information due to the effect of measurement error. A new routing strategy using maximum expected distance and angle (MEDA) algorithm is proposed to improve the performance and promote the successive transmission rate. We firstly introduce the expected distance and angle, and then we employ the principal component analysis to construct the object function for selecting the next hop node. We compare the proposed algorithm with maximum expectation within transmission range (MER) and greedy routing scheme (GRS) algorithms. Simulation results show that the proposed MEDA algorithm outperforms the MER and GRS algorithms with higher successive transmission rate.

1. Introduction

Wireless sensor networks have various applications in battle field and environmental monitoring. The geographic routing is of great importance for wireless sensor networks due to its efficiency and simplicity. All nodes in the network have the knowledge of their own location information through Global Positioning Systems (GPS) receivers or localization methods. The nodes also know their neighbors’ location through locally exchanged location information. The location information plays a crucial role in selecting the next hop node. Since the location is the important information required for geographic routing, the accuracy of the location is crucial. Most geographic routing algorithms assume that the location information available at each node is perfect. However, only a rough estimate of a node location is available in practice.

Several geographic routing algorithms that consider the effect of location errors have been proposed for wireless sensor networks [1]–[4]. A location error compensation scheme was proposed for geographic routing [1]. It used the topology difference between the two relative neighborhood graph approaches to detect the location error. Once the location error is detected, it is compensated by using the triangulation method. A maximum expectation within transmission range (MER) algorithm [2] which incorporated location errors into the objective function was proposed. Each node forwarded packets to the next hop node that maximize the objective function. A location estimation and dynamic link detection scheme [3] was proposed for the geographic routing in the NLOS environments. This algorithm proposed a location error correction and dynamic link detection for geographic routing. These strategies can enhance the performance of geographic routing. An energy-efficient geographic routing algorithm [4] with adaptive transmission strategy was investigated to cope with the transmission failure caused by location errors. It incorporated the energy consumption in the geographic routing to achieve minimum power consumption.

In this letter, we propose a maximum expected distance and angle (MEDA) algorithm to improve the performance of routing in the presence of location errors. The aim of the proposed algorithm is to maximize the object function which consists of expected distance and angle. We employ principal component analysis to reconstruct the object function. The source node will select the node according the object function values of nodes.

The paper is organized as follows. In Sect. 2, the impact of location errors on geographic routing is presented. In Sect. 3, the maximum expected distance and angle (MEDA) algorithm is described. Numerical and simulation results are presented in Sect. 4. Section 5 concludes this paper.

2. The Impact of Location Errors on Geographic Routing

Information about the locations of nodes is of particular importance for the applications of wireless sensor networks. The location of node can be determined by GPS or location algorithm. But the inaccurate measurement can result in the location errors. Various node localization techniques have been proposed to improve the accuracy of location. However the location errors can not be eliminated absolutely [5]. The geographic routing is based on locally available position information, so the impact of location errors on the geographic routing is serious.

In this section, we introduce the impact of location errors on the performance of geographic routing. As shown in Fig. 1, S is the source node and D is the destination node. The source node S will select the next hop node in the forwarding area (shaded area). The influence of location errors on nodes can be summarized into the four possible situations. When node A is actually located in the forwarding area but the source node S believes it is outside the region, so node A will not be selected as the next hop node.
Likewise, node C will also lose the chance to be the candidate node even though it is located in the transmission range \( r_c \). Node B is located out of the forwarding region, but the source node \( S \) believes it is inside, therefore the node \( B \) will select as the candidate node. It will cause the backward progress. If the source node \( S \) selected node \( E \) as the candidate node, it will cause transmission failure since node \( E \) located out of the transmission range \( r_c \). As discussed above, location errors not only affect the delivery ratio, but also consume extra transit energy.

3. Proposed Algorithm

In this section, the involved methodologies are described. In order to alleviate the effect of location errors on the geographic routing, we proposed a maximum expected distance and angle (MEDA) algorithm which is a modified version of MER algorithm. This algorithm employs the expected distance and angle to determine the next hop node.

We make the following assumptions in the study: the transmission range \( r_c \) is fixed, the location errors at different nodes are independent. \( X_i \) is the measured location of \( i \)-th node, \( X_i \) is the real location of \( i \)-th node. It can be expressed as \( X_i = X_i' + W_i \), where \( W_i \) is the location error modeled as zero mean white Gaussian with variance \( \sigma_j^2 \), namely, \( W_i \sim N(0, \sigma_j^2) \). Each sensor node estimates the location of itself through a localization algorithm, the standard deviation \( \sigma_j \) could obtain in an environment through simulation or experiment. We attach an error information field in a message for geographic routing and announce the statistical characteristics of the location error to neighbors with location information.

As shown in Fig. 2, we establish the polar coordinate. The coordinate of source node \( S \) is the pole and the polar axis is from the pole and pointing to the destination node \( D \). For example, the real position of node \( j \) can be expressed as \( (\| X_j \|, \theta_j) \). We set \( NE \) is the neighbor set of source node \( S \). For node \( j \), if \( \| X_j \| \leq r_c \) and \( \| X_D - X_j \| \leq \| X_D \| \), then node \( j \) is an element of neighbor set. When we select the next hop node from neighbor set, we consider the line from the source node \( S \) to the destination, pick the next hop node which is far away from the source node \( S \) and close to the line. It could reduce the energy consumption of sensor nodes and shorten the transmission distance. The combined effect of distance and angle could achieve selecting the next hop node from the neighbor set. The object function is established as follows:

\[
J_j = w_1 \frac{E_{jd}}{r_c} + w_2 E_{j\theta}, \quad j \in NE
\]

where \( E_{jd} \) and \( E_{j\theta} \) are the expected distance and angle respectively, \( w_1 \) and \( w_2 \) are the weight of expected distance and angle respectively. \( E_{jd}, E_{j\theta}, w_1 \) and \( w_2 \) will be described in the following subsections.

We select the next hop node which has the maximum values in Eq. (1).

3.1 Calculating the Expected Distance

The distance between measured and real position of node \( j \) can be expressed as \( Z_j = \| X_j - X_j' \| \). The distance follows a Rayleigh distribution. So we can obtain the probability density function of location error \( Z_j \) as follows:

\[
f_{jd}(Z_j) = \frac{Z_j}{\sigma_j^2} \exp \left( -\frac{Z_j^2}{2\sigma_j^2} \right)
\]

Let node \( j \) be a neighbor of source node \( S \). \( h_j = \| X_D \| - \| X_D - X_j \| \) is the measured progress to node \( j \). \( l_j = r_c - \| X_j \| \) is the measured margin form the boundary. \( \tau_j = \| X_D \| - \| X_D - X_j \| \) is the true progress to node \( j \). \( f_j(\| X_j \|) \) is the probability density function of \( \| X_j \| \). So the expected distance of node \( j \) can be approximated as:

\[
E_{jd}(\tau_j) = \int_{A_j} \tau_j f_j(\| X_j \|) d \| X_j \|
\]

\[
\approx h_j \int_{A_j} f_j(\| X_j \|) d \| X_j \|
\]

where, \( A_j = \{ X \in R^2 \mid \| X - X_j \| \leq v_j \} \). If \( v_j > l_j \) node \( j \) may be out of the transmission range of source node \( S \). If \( v_j > h_j \) it may cause backward progress. So we define \( v_j = \min(h_j, l_j) \).
The probability that the real location $X_j$ of node $j$ is located within $v_j$ from $X_j'$ can be calculated utilizing following formulation:

$$F(v_j) = P[\|X_j - X_j'\| \leq v_j] = \int_0^{v_j} f_{jd}(Z_j) dZ_j$$

$$= \int_0^{v_j} \frac{Z_j}{\sigma_j^2} \exp(-Z_j^2/2\sigma_j^2) dZ_j = 1 - \exp\left(-\frac{v_j^2}{2\sigma_j^2}\right)$$

(4)

So Eq. (3) can be rewritten as:

$$E_{jd}(\tau_j) = h_j \left(1 - \exp\left(-\frac{v_j^2}{2\sigma_j^2}\right)\right)$$

(5)

3.2 Calculating the Expected Angle

As shown in figure 2, $\theta$ illustrated the deviate degree between node $j$ and pole. If $\theta$ is relatively large in each hop, the total distance that deliver the packet from source node to destination node is large. In order to shorten the total distance, we select the node which has the smaller relative angle as the next hop node. Due to the effect of location errors, we estimate the angle as follows.

The probability density function of $\theta_j$ can be given by:

$$f_0(\theta_j) = \begin{cases} 
1/\pi & -\pi/2 < \theta_j < \pi/2 \\
0 & \text{others}
\end{cases}$$

(6)

The calculation process of Eq. (6) is given in Appendix A. We employ $\zeta_j(\theta_j) = \cos(\theta_j)$ to represent the influence of angle since we attempt to maximum the object function. So the expected angle of node $j$ can be calculated using the following formulation:

$$E_{\theta j}(\zeta_j(\theta_j)) = \int_{A_j} \zeta_j(\theta_j) f_0(\theta_j) d\theta_j$$

$$= \int_{A_j} \cos(\theta_j) f_0(\theta_j) d\theta_j$$

(7)

where, $A_j = \{X \in R^2 \mid \|X - X_j'\| \leq v_j\}$, $\theta_j \in [\theta_j' - \alpha, \theta_j' + \alpha]$, and as illustrated in figure 2, $\alpha = \arcsin\left(\frac{v_j}{\|X_j'\|}\right)$.

So the Eq. (7) can be rewritten as:

$$E_{\theta j}\zeta_j(\theta_j) \approx \cos \theta_j \int_{A_j} f_0(\theta_j) d\theta_j$$

$$= \cos \theta_j \int_{\theta_j' - \alpha}^{\theta_j' + \alpha} \frac{1}{\pi} d\theta_j$$

$$= \frac{2}{\pi} \cos \theta_j' \arcsin\left(\frac{v_j}{\|X_j'\|}\right)$$

(8)

where, $v_j = \min(h_j, l_j)$.

3.3 Principal Component Analysis Using Covariance Method

When we select the next hop node from neighbor set, the weights ($w_1$ and $w_2$) of $E_{jd}$ and $E_{j\theta}$ in Eq. (1) are difficult to determine quantitatively. In this subsection we employ principal component analysis [6] to reconstruct the Eq. (1). We firstly standardize $X = [X_1, X_2]^T$ to obtain a matrix $Z = [Z_1, Z_2]^T$, where $X_1 = [\xi_1, \xi_2, \xi_3, \xi_4]$ and $X_2 = [E_{i0}, \cdots, E_{iN}]$. Then we perform linear transformation of the vectors $Z_1$ and $Z_2$, this transformation is a geometric equivalent of coordination system rotation. It forms new comprehensive variables $Y_1$ and $Y_2$, namely principal components. The operation can make the comprehensive variables $Y_1$ and $Y_2$ independently and reserve all the information of primitive variables $X_1$ and $X_2$. Finally we compute the variance contribution rates of $Y_1$ and $Y_2$—namely the weights of $Y_1$ and $Y_2$.

The process steps are described as follows. Step 1-5 are the processes of standardized $X$. Step 6 is the linear transformation.

**Step 1** Calculate the mean of variables:

$$u_i = \frac{1}{N} \sum_{j=1}^{N} X_{i}(j), i = 1, 2$$

(9)

We define: $u = [u_1, u_2]^T$.

**Step 2** Calculate the deviations from the means: We subtract the mean vector $u$ from each column of the data matrix $X$. Then we can obtain a new data set whose mean is zero. The new data set as follows:

$$C = X - (uh)^T$$

(10)

where $h$ is a $1 \times N$ row vector of all 1s: $h_i = 1, i = 1, \cdots, N$.

**Step 3** Calculate the covariance matrix $\Sigma$ of matrix $C$.

**Step 4** Calculate the eigenvectors $\gamma = [\gamma_1, \gamma_2]$ and eigenvalues $\lambda = [\lambda_1, \lambda_2]$ of the covariance matrix $\Sigma$. By ordering the eigenvectors in the order of descending eigenvalues, we can create an ordered orthogonal basis with the first eigenvector having the direction of the largest variance of the data. We define $\lambda_1 \geq \lambda_2$, and $\gamma_1, \gamma_2$ is the corresponding eigenvectors.

**Step 5** Convert the data to $z$-scores:

$$Z = \frac{C}{s \cdot h}$$

(11)

where $s = \sqrt{\Sigma[i,i]}$ is a $2 \times 1$ standard deviation vector from the square root of each element along the main diagonal of the covariance matrix $\Sigma$.

**Step 6** Obtain a new data set $(2 \times N)$ matrix through linear transformation as follows:

$$Y = \begin{bmatrix} Y_1 \\ Y_2 \end{bmatrix} = \gamma^T Z = \begin{bmatrix} \gamma_1^T \\ \gamma_2^T \end{bmatrix} \begin{bmatrix} Z_1 \\ Z_2 \end{bmatrix}$$

(12)

The variance contribution rate:

$$a_k = \frac{\lambda_k}{\lambda_1 + \lambda_2} (k = 1, 2)$$

(13)

Since the linear transformation makes the comprehensive variables $Y_1$ and $Y_2$ reserve all the information of primitive variables $X_1$ and $X_2$. So maximizing the objective function (1) is equivalent to maximizing the reconstructed objective function.
Table 1 Default Parameter Values.

| Parameter            | Symbol | Default Value |
|----------------------|--------|---------------|
| Transmission range   | rc     | 200           |
| Number of deployed nodes | N     | 400           |
| Variance of measurement noise | σ     | 30            |

The source node S will select the node which has the maximum value in Eq. (14) as the next hop node.

\[ J'_j = a_1 Y_{1j} + a_2 Y_{2j}, \quad j \in NE \] (14)

4. Performance Evaluation

In this section, we present simulation results of the proposed algorithm. We consider a 1000m × 1000m square area with hundreds sensor nodes, and the nodes are uniformly deployed in the field. Table 1 presents the default parameter values in the experiments. In each simulation case, 2000 Monte Carlo runs are performed with the same parameters. We compare the performance of the algorithms with MER and GRS algorithm.

The performance of MER, GRS and MEDA algorithm with different communication radius is illustrated in Fig. 3. It can be observed that with the increase of communication radius, the success rate of algorithms will increase. This is because with the larger communication radius, the number of neighbor nodes will increase. The proposed MEDA algorithm has the highest success rate. The MEDA algorithm does not consider the influence of location errors, so it owns the lowest success rate.

Figure 4 shows the relation between success rate and σ values. The results imply that the higher σ value, the better effect of proposed algorithm is. When the σ value is relatively large, the MEDA algorithm outperforms MER and GRS algorithms with higher success rate.

Figure 5 depicts the performance of the algorithms with two location error models: uniformly distributed error and exponentially distributed error. The minimum and maximum values of parameters of uniform distribution are −30 and 30, namely, \( U(−30, 30) \). The rate parameter of exponential distribution is defined as \( \epsilon = 30 \). The MEDA algorithm has the highest success rate, so the algorithm is robust to different location error models. When the location errors are uniform distribution, the MEDA algorithm has much higher success rate than MER and GRS algorithms, about 22.8% and 49.1% respectively. The MEDA algorithm improved 25% and 10.2% in comparison with GRS and MER algorithms when the location errors are exponential distribution.

5. Conclusion

A maximum expected distance and angle (MEDA) algorithm was proposed in the presence of location errors. The expected distance and angle were introduced firstly to construct the object function. Then the principle component analysis was employed to reconstruct the object function. The source selected the next hop node which had the maximum object function value. The performance evaluation of the proposed geographic routing algorithm surpassed the MER and GRS algorithms. It was shown that the proposed algorithm improves the successive transmission rate.

Acknowledgements

This work is supported by the National High-Tech Research and Development Plan of China under Grant No. 2010AA101702; the National Natural Science Foun-
In 2-dimensional Cartesian coordinates, we define the truth location of the $j$-th node as $(x_j, y_j)$, and the estimated location of the $j$-th node is $(\hat{x}_j, \hat{y}_j)$. We set $\varphi_j = \arctan((y_j - \hat{y}_j)/(x_j - \hat{x}_j))$. The probability density function of $\varphi_j$ is expressed as $f_\varphi(\varphi_j)$. Due to the $\theta_j \approx \varphi_j$, we can use $f_\varphi(\varphi_j)$ to approximate $f_\theta(\theta_j)$. $f_\varphi(\varphi_j)$ is calculated as follows:

Due to the location errors are distributed normally, i.e. $\Delta x_j = (x_j - \hat{x}_j) \sim N(0, \sigma_1^2)$, $\Delta y_j = (y_j - \hat{y}_j) \sim N(0, \sigma_2^2)$. The joint normal distribution of $\Delta x_j, \Delta y_j$ can be expressed as:

$$f_{\Delta x_j, \Delta y_j}(\Delta x_j, \Delta y_j) = \frac{1}{2\pi\sigma_1\sigma_2 \sqrt{1 - r^2}} \exp\left\{-\frac{1}{2(1 - r^2)} \left(\frac{\Delta x_j^2}{\sigma_1^2} - \frac{2r \Delta x_j \Delta y_j}{\sigma_1 \sigma_2} + \frac{\Delta y_j^2}{\sigma_2^2}\right)\right\}.$$  

(A-1)

According to Eq. (A-1), we can obtain the probability density function of $\varphi = \Delta y_j / \Delta x_j$ after some calculations:

$$f_\varphi(\varphi) = \frac{1}{2\pi\sigma_1\sigma_2 \sqrt{1 - r^2}} \int_0^\infty \Delta x_j \exp\left[-\frac{\Delta x_j^2}{2(1 - r^2)} \left(\frac{\varphi_j^2}{\sigma_1^2} - \frac{2r \Delta x_j \Delta y_j}{\sigma_1 \sigma_2} + \frac{1}{\sigma_2^2}\right)\right] d\Delta x_j$$  

(A-2)

Since $\int_0^\infty x \exp(-x^2/(2a^2))dx = a^2 \int_0^\infty \omega \exp(-\omega) d\omega = a^2$. The Eq. (A-2) can be rewritten as:

$$f_\varphi(\varphi) = \frac{\sqrt{1 - r^2}\sigma_1\sigma_2/\pi}{\varphi^2(\varphi - r\sigma_1/\sigma_2)^2 + \sigma_2^2(1 - r^2)}$$  

(A-3)

So $\varphi = \Delta y_j / \Delta x_j$ is the Cauchy distribution with center in $\varphi = r\sigma_1/\sigma_2$.

When $\Delta x_j$ and $\Delta y_j$ are independent, the $r = 0$ and Eq. (A-3) can be simplified as:

$$f_\varphi(\varphi) = \frac{\sigma_1/\pi\sigma_2}{\varphi^2 + \sigma_1^2/\sigma_2^2}$$  

(A-4)

Since the variances $\Delta x_j$ and $\Delta y_j$ are independently normally distributed with mean zero and we assume $\sigma_1 = \sigma_2$, the probability density of $\varphi_j$ will be determined in the following content.

For $\varphi = \Delta y_j / \Delta x_j$, according to Eq. (A-4), we can obtain $f_\varphi(\varphi) = 1/(\pi(\varphi^2 + 1))$. Since $-\pi/2 < \varphi_j < \pi/2$, the equation $\varphi_j = \arctan z$ has the single solution: $z = \tan \varphi_j$. Because of $d\varphi_j/dz = \cos^2 \varphi_j$, So,

$$f_\varphi(\varphi_j) = \frac{f_\varphi(z)}{\cos^2 \varphi_j} = \frac{1/\pi}{\cos^2 \varphi_j(1 + \tan^2 \varphi_j)} = \frac{1}{\pi}.$$  

$$|\varphi_j| < \frac{\pi}{2}.$$  

(A-5)

We get $f_\theta(\theta_j) = \left\{\begin{array}{ll} 1/\pi & , -\pi/2 < \theta_j < \pi/2 \\ 0 & , others \end{array}\right.$