

DOI: 10.13382/j.jemi.B2306988

# 基于麻雀搜索的协同定位算法研究\*

夏斌<sup>1</sup> 张立晔<sup>2</sup>

(1. 山东理工大学电气与电子工程学院 淄博 255000; 2. 山东理工大学计算机科学与技术学院 淄博 255000)

**摘要:**无线传感器网络的定位问题可以转化为适应度函数最优问题,并通过经典的麻雀搜索算法进行求解。然而该算法所用的适应度函数并未使用未知节点之间的测量距离数据,从而导致定位精度的提升有限。为了解决这一问题,提出了一种基于麻雀搜索的协同定位算法。该算法主要包括两个搜索阶段:粗略搜索和精细搜索。在粗略搜索阶段,未知节点到锚节点之间的测量距离数据被用于确定未知节点的初始位置。在精细搜索阶段,未知节点之间的测量距离数据被用来确定未知节点的精确位置。首先,采用Cat混沌映射方法来保证初始种群的均匀分布,从而有助于确定最佳位置。其次,构建了两种不同的适应度函数,一种用于粗略搜索,另一种用于精细搜索。其中,用于精细搜索的适应度函数利用未知节点之间的测量距离数据来提高定位精度。最后,提出了一种新的精细搜索方法,以避免协同定位结果收敛到局部最优位置。通过对仿真和实测数据进行分析,验证了所提方法的有效性。

**关键词:**麻雀搜索;协同定位;适应度函数;定位精度

**中图分类号:** TP393 **文献标识码:** A **国家标准学科分类代码:** 510.50

## Research on cooperative localization algorithm based on sparrow search

Xia Bin<sup>1</sup> Zhang Liye<sup>2</sup>

(1. School of Electrical and Electronic Engineering, Shandong University of Technology, Zibo 255000, China;

2. School of Computer Science and Technology, Shandong University of Technology, Zibo 255000, China)

**Abstract:** The localization problem of wireless sensor network can be transformed into a fitness function optimization problem, which is solved by the classical sparrow search algorithm. However, the fitness function used in this algorithm does not use measured distance data between unknown nodes, resulting in limited improvement in positioning accuracy. To address this issue, a cooperative localization algorithm based on sparrow search is proposed. This algorithm mainly includes two search stages: rough search and fine search. In the rough search stage, the measured distance data between the unknown node and the anchor node is used to determine the initial position of the unknown node. In the fine search stage, the measured distance data between unknown nodes is used to determine the precise position of the unknown node. Firstly, the Cat chaotic mapping method is used to ensure the uniform distribution of the initial population, which helps to determine the optimal location. Secondly, two different fitness functions are constructed, one for rough search and the other for fine search. Among them, the fitness function used for fine search utilizes the measured distance data between unknown nodes to improve positioning accuracy. Finally, a new fine search method is proposed to avoid the convergence of cooperative localization results to the local optimal position. The effectiveness of the proposed method is verified through analysis of simulation and measured data.

**Keywords:** sparrow search; cooperative localization; fitness function; positioning accuracy

## 0 引言

无线传感器网络(wireless sensor network, WSN)在智能交通<sup>[1]</sup>、目标跟踪<sup>[2]</sup>、医疗保健<sup>[3]</sup>、智慧农业等场景中得到广泛应用。在这些应用场景中,如果节点的位置信息缺失,收集到的相关数据将无法发挥其应有的作用。例如在智慧农业中,无线传感器网络通过节点实时收集农作物所处环境数据,自动调整生产设备的工作状态,实现农业生产和管理的精准化。如果没有节点的准确位置信息,也就无法应用这些收集到的环境数据。由此可见,精确定位是无线传感器网络应用的前提。因此,对无线传感器网络的定位进行研究<sup>[4-5]</sup>具有重要的应用价值。

通过获取测量距离数据<sup>[6-9]</sup>可构建定位方程组。为了求解该方程组,常采用麻雀搜索算法。麻雀搜索算法<sup>[10-11]</sup>具有易于实现和快速收敛的特点,因此在定位领域中得到了广泛的应用。例如文献[12]成功地将麻雀搜索算法应用于无线传感器网络的定位,实现了较高的定位精度。此外,文献[13]还提出了一种基于麻雀搜索算法和反向传播神经网络的定位方法,用于估计未知节点的位置。而文献[14]针对无线传感器网络中的定位问题,提出了一种基于麻雀搜索的定位算法。与其他方法相比,这种方法展现了良好的定位性能。麻雀搜索算法所用的适应度函数包含的测量距离数据越多,那么该算法获得的结果就越接近最优解<sup>[15]</sup>。然而,以上提到的这种算法在构建适应度函数时,只考虑未知节点到锚节点的测量距离数据,而没有考虑未知节点之间的测量距离数据,导致定位精度的提升有限。

根据以上研究,为了进一步提高定位精度,本文提出了一种基于麻雀搜索的协同定位算法。该算法主要从以下3个方面进行改进:

1)与随机生成的初始种群相比,Cat混沌映射方法生成的初始种群能够尽可能地实现均匀分布,有助于确定未知节点的最佳位置。

2)测量距离数据被用来构造两种不同的适应度函数:一种用于粗略搜索,另一种用于精细搜索。粗略搜索所用的适应度函数利用未知节点到锚节点的测量距离数据来确定未知节点的初始位置,从而加快算法的收敛速度;精细搜索所用的适应度函数则利用未知节点之间的测量距离数据来提高定位精度,改善定位性能。

3)针对构造的两种适应度函数,协同定位算法采用

不同的搜索方法。在粗略搜索阶段,协同定位算法沿用原有的麻雀搜索方法,而在精细搜索阶段,协同定位算法采用一种新的搜索方法,避免最优解陷入局部最优,同时还提高了搜索效率。

实验结果表明,本文提出的算法在随机部署的 WSN 中具有更快的收敛速度和更好的定位性能。

## 1 经典的麻雀搜索定位算法

假设 WSN 定位区域有  $K$  个锚节点,锚节点  $k$  的位置表示为  $(u_k, v_k)$ ,  $k \in \{1, \dots, K\}$ ,锚节点  $k$  到未知节点  $(x, y)$  的测量距离数据为  $d_k$ ,则定位方程组可以表示为:

$$\begin{cases} d_1 = \sqrt{(x - u_1)^2 + (y - v_1)^2} + e_1 \\ \vdots \\ d_k = \sqrt{(x - u_k)^2 + (y - v_k)^2} + e_k \\ \vdots \\ d_K = \sqrt{(x - u_K)^2 + (y - v_K)^2} + e_K \end{cases} \quad (1)$$

其中,  $\sqrt{(x - u_k)^2 + (y - v_k)^2}$  表示实际值,  $e_k$  表示测量距离数据误差。

假设有  $N$  只麻雀,  $\bar{\mathbf{X}}_n^l = (x_n^l, y_n^l)^T$  表示第  $n$  只麻雀,  $l$  为搜索次数。根据公式(1),适应度函数定义为:

$$f(\bar{\mathbf{X}}_n^l) = \frac{1}{K} \sum_{k=1}^K (\sqrt{(x_n^l - u_k)^2 + (y_n^l - v_k)^2} - d_k)^2 \quad (2)$$

根据式(2)计算适应度函数值,并相应地更新麻雀的位置。如果搜索次数达到最大值,搜索过程结束,此时将得到未知节点的位置。值得注意的是,经典的麻雀搜索定位算法无法同时确定多个未知节点的位置,只能获得一个未知节点的位置。如果想要获得其他未知节点的位置,就需要重新调用麻雀搜索定位算法。

## 2 基于麻雀搜索的协同定位算法

协同定位过程主要包括两个搜索阶段:粗略搜索和精细搜索。在粗略搜索阶段,未知节点到锚节点之间的测量距离数据被用于确定未知节点的初始位置。在精细搜索阶段,未知节点之间的测量距离数据被用来确定未知节点的精确位置。假设 WSN 定位区域有  $I$  个未知节点和  $K$  个锚节点,锚节点与未知节点间的测量距离数据为  $d_{i,k}$ ,未知节点之间的测量距离数据为  $d'_{i,j}$ ,则协同定位方程组可以表示为:

$$\left\{ \begin{array}{l} d_{1,1} = \sqrt{(x_1 - u_1)^2 + (y_1 - v_1)^2} + e_{1,1} \\ \vdots \\ d_{i,k} = \sqrt{(x_i - u_k)^2 + (y_i - v_k)^2} + e_{i,k} \\ \vdots \\ d_{l,K} = \sqrt{(x_l - u_K)^2 + (y_l - v_K)^2} + e_{l,K} \\ d'_{1,2} = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} + e'_{1,2} \\ \vdots \\ d'_{i,j} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} + e'_{i,j} \quad i < j \\ \vdots \\ d'_{(l-1),l} = \sqrt{(x_{(l-1)} - x_l)^2 + (y_{(l-1)} - y_l)^2} + e'_{(l-1),l} \end{array} \right. \quad (3)$$

其中,  $k \in \{1, \dots, K\}$ ,  $i \in \{1, \dots, l\}$ ,  $j \in \{1, \dots, l\}$ 。

$\sqrt{(x_i - u_k)^2 + (y_i - v_k)^2}$  表示未知节点  $(x_i, y_i)$  到锚节点  $(u_k, v_k)$  的真实距离,  $\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$  表示未知节点  $(x_i, y_i)$  到未知节点  $(x_j, y_j)$  的真实距离,  $e_{i,k}$  和  $e'_{i,j}$  表示测量距离数据误差。

基于麻雀搜索的协同定位过程如下:

步骤 1) 麻雀种群初始化。假设有  $N$  只麻雀,  $\mathbf{X}^n = (\mathbf{X}_1^n, \mathbf{X}_2^n, \dots, \mathbf{X}_i^n, \dots, \mathbf{X}_N^n)^T$  表示第  $n$  只麻雀, 其中  $\mathbf{X}_i^n = (x_i^n, y_i^n)^T$ ,  $n \in \{1, 2, \dots, N\}$ 。为了保持麻雀种群的多样性, 并确保初始种群尽可能均匀分布, 采用了 Cat 混沌映射。Cat 混沌映射是一个二维的可逆混沌映射, 具有较好的混沌特性, 能生成更好的初始序列<sup>[16-18]</sup>。该映射方程如下:

$$\begin{bmatrix} x_{i+1} \\ y_{i+1} \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} x_i \\ y_i \end{bmatrix} \bmod(1) \quad (4)$$

其中,  $\bmod(1)$  表示只取实数的小数部分。

步骤 2) 基于未知节点  $i$  到每个锚节点的测量距离数据, 用于粗略搜索的适应度函数定义为:

$$f_1(\mathbf{X}_i^n) = \frac{1}{K} \sum_{k=1}^K (\sqrt{(x_i^n - u_k)^2 + (y_i^n - v_k)^2} - d_{i,k})^2 \quad (5)$$

步骤 3) 更新发现者的位置。发现者寻找食物, 为跟随者提供搜索范围, 其位置更新公式如下:

$$\mathbf{X}_i^n(l_1 + 1) = \begin{cases} \mathbf{X}_i^n(l_1) \exp\left(\frac{-n}{\alpha \times L_1}\right) & R_2 < ST \\ \mathbf{X}_i^n(l_1) + QL & R_2 > ST \end{cases} \quad (6)$$

其中, 随机数  $\alpha \in [0, 1]$ ; 随机数  $Q$  服从标准正态分布;  $l_1$  表示粗略搜索次数;  $L_1$  表示粗略搜索最大次数; 矩阵  $\mathbf{L}$  中的元素都为 1;  $ST$  为安全阈值; 报警值  $R_2 \in [0, 1]$ 。

步骤 4) 更新跟随者的位置。跟随者跟随发现者寻找食物, 其位置更新公式如下:

$$\mathbf{X}_i^n(l_1 + 1) = \begin{cases} \mathbf{X}_i^n(l_1) \\ Q \exp\left(\frac{\mathbf{X}_{worst}^n(l_1) - \mathbf{X}_i^n(l_1)}{n^2}\right), n > N/2 \\ \mathbf{X}_i^p(l_1 + 1) + |\mathbf{X}_i^n(l_1) - \mathbf{X}_i^p(l_1)| \mathbf{A}^+ \mathbf{L}, \text{其他} \end{cases} \quad (7)$$

其中,  $\mathbf{X}_{worst}$  表示最差位置,  $\mathbf{X}_i^n$  是发现者搜索到的最优位置, 矩阵  $\mathbf{A}$  的元素为 -1 或 1 的随机值, 矩阵  $\mathbf{A}^+ = \mathbf{A}^T(\mathbf{A}\mathbf{A}^T)^{-1}$ 。

步骤 5) 更新警戒者的位置。警戒者的位置更新公式如下:

$$\mathbf{X}_i^n(l_1 + 1) = \begin{cases} \mathbf{X}_{best}^n(l_1) + \beta |\mathbf{X}_i^n(l_1) - \mathbf{X}_{best}^n(l_1)|, f_n \neq f_g \\ \mathbf{X}_i^n(l_1) + K \left[ \frac{|\mathbf{X}_i^n(l_1) - \mathbf{X}_{worst}^n(l_1)|}{(f_n - f_w) + \varepsilon} \right], f_n = f_g \end{cases} \quad (8)$$

其中, 随机数  $K \in [-1, 1]$ , 控制步长的参数  $\beta \in N(0, 1)$ ,  $f_w$  表示  $\mathbf{X}_{worst}$  对应的适应度函数值,  $f_g$  表示最佳位置  $\mathbf{X}_{best}$  对应的适应度函数值,  $f_n$  表示第  $n$  只麻雀对应的适应度函数值, 最小常数  $\varepsilon$  用来避免分母出现 0。

步骤 6) 如果粗略搜索次数达到最大值, 则完成未知节点  $i$  位置的粗略搜索; 否则,  $l_1 = l_1 + 1$  进入步骤 3)。

步骤 7) 如果所有未知节点的初始位置估计没有完成, 那么进入步骤 2), 开始对下一个节点位置进行粗略搜索; 否则, 直接进入步骤 8)。

步骤 8) 基于节点之间的所有测量距离数据, 用于精确搜索的适应度函数定义为:

$$f_2(\mathbf{X}^n) = \frac{1}{K_1} \sum_{i=1}^l \sum_{k=1}^K (\sqrt{(x_i^n - u_k)^2 + (y_i^n - v_k)^2} - d_{i,k})^2 + \frac{1}{K_1} \sum_{i=1}^{l-1} \sum_{j=1}^l (\sqrt{(x_i^n - x_j^n)^2 + (y_i^n - y_j^n)^2} - d'_{i,j})^2 \quad (9)$$

其中,  $K_1$  为测量距离的总数。

步骤 9) 通过式 (10) 进行精确搜索, 更新最优位置。

$$\mathbf{X}^n(l_2 + 1) = \mathbf{X}_{best}^n(l_2) + \varepsilon \times (L_2 - l_2) / L_2 \quad (10)$$

其中,  $\mathbf{X}_{best}^n(l_2)$  为第  $l_2$  次精确搜索时的最佳位置,  $L_2$  为精确搜索最大次数。采用新的精细搜索方法能简化操作并提升效率, 主要通过统一的搜索策略实现。与原有方法不同的是, 新方法不再需要各类麻雀 (如发现者、跟随者和警戒者) 分别执行不同的搜索策略, 这样不仅使整个搜索过程更加高效, 而且还减少了对复杂参数调整的需求。

步骤 10) 如果精确搜索次数达到最大值, 则完成协同定位; 否则,  $l_2 = l_2 + 1$  进入步骤 9)。

基于麻雀搜索的协同定位算法可以同时确定多个未知节点的位置, 其流程如图 1 所示。

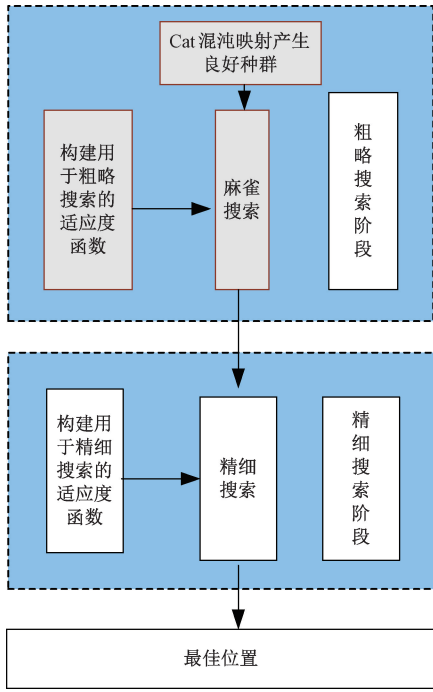


图1 基于麻雀搜索的协同定位流程

Fig. 1 Cooperative localization process based on sparrow search

### 3 实验结果与分析

在本节中,对提出的麻雀协同定位算法的性能进行评估,通过实验来证该方法的有效性。将所提算法与相同网络条件下的鸡群优化定位算法<sup>[19]</sup>、蜣螂优化定位算法<sup>[20]</sup>和经典的麻雀搜索定位算法进行了比较,采用Octave 软件进行仿真模拟实验。

假定未知节点和锚节点随机分布在  $D \times D$  的无线传感器网络定位区域,边长  $D = 12$ 。测量距离数据误差服从均值为0,方差为  $\sigma^2$  的高斯分布。为了验证算法的准确性,本文引入平均定位误差评价指标。平均定位误差定义为:

$$error = \frac{\sum_{i=1}^I \sqrt{(\hat{x}_i - x_i)^2 + (\hat{y}_i - y_i)^2}}{I} \quad (11)$$

其中,  $(\hat{x}_i, \hat{y}_i)$  表示未知节点  $i$  的估计位置。定位的相关参数设置如表1所示。

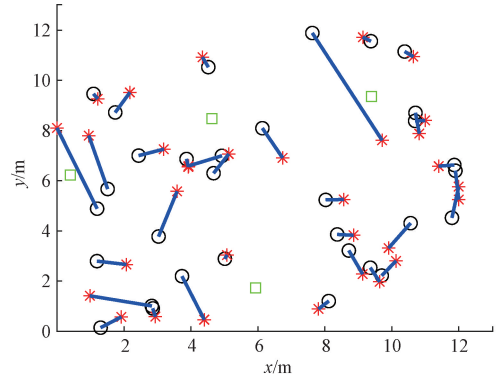
表1 参数设置

Table 1 Parameter setting

算法	参数设置
经典的麻雀搜索定位算法	麻雀数目 $N = 50$ , 最大搜索次数 $L = 200$ , 安全阈值 $ST = 0.6$
鸡群优化定位算法	鸡群数目为 $N = 50$ , 最大搜索次数 $L = 200$
蜣螂优化定位算法	蜣螂数目 $N = 50$ , 最大搜索次数 $L = 200$
麻雀协同定位算法	麻雀数目 $N = 50$ , 粗略搜索次数 $L_1 = 100$ , 精细搜索次数 $L_2 = 100$ , 安全阈值 $ST = 0.6$

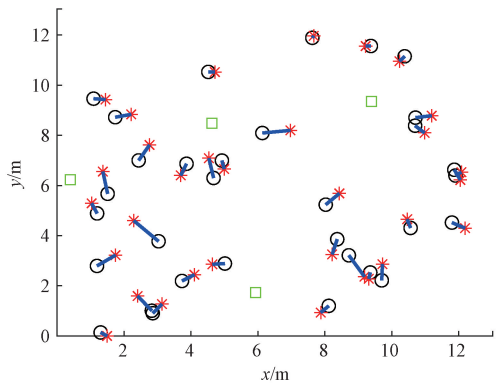
#### 1) 定位效果

图2展示了经典的麻雀搜索定位算法和麻雀协同定位算法的定位效果对比。正方形表示锚节点的位置,圆圈表示未知节点的实际位置,星号表示未知节点的估计位置,并且实线表示实际位置和估计位置之间的差值。经典的麻雀搜索定位对应的平均定位误差为0.95 m,而麻雀协同定位对应的平均定位误差为0.35 m。这表明麻雀协同定位算法的定位精度要高于经典的麻雀搜索定位算法的定位精度。



(a) 经典麻雀定位算法 ( $K=4, I=32, \sigma^2=1.2$ )

(a) Classic sparrow localization algorithm ( $K=4, I=32, \sigma^2=1.2$ )



(b) 麻雀协同定位算法 ( $K=4, I=32, \sigma^2=1.2$ )

(b) Sparrow collaborative localization algorithm ( $K=4, I=32, \sigma^2=1.2$ )

图2 定位效果

Fig. 2 Localization effect

#### 2) 测量距离数据误差对协同定位性能的影响

测量距离数据误差对协同定位性能的影响如图3所示。随着测量距离数据误差的不断减小,经典的麻雀搜索算法、蜣螂优化定位算法、鸡群优化定位算法和麻雀协同定位算法的定位误差都在减小。此外,对于相同的测量距离数据误差,麻雀协同定位算法的定位误差明显低于其他定位算法,这是因为它利用未知节点之间的测量距离数据进行协同定位,减少了测量距离数据误差对定位性能的影响,进一步提高了定位精度。

由表2和3可知,麻雀协同算法在定位精度上相较



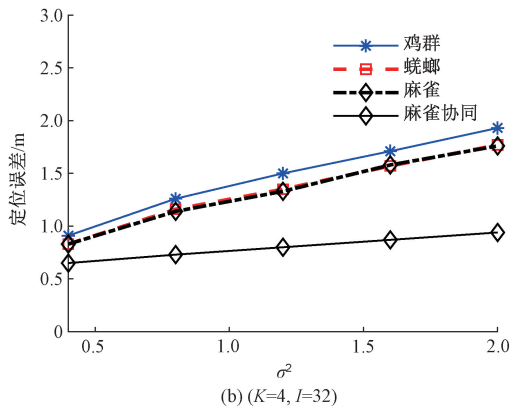
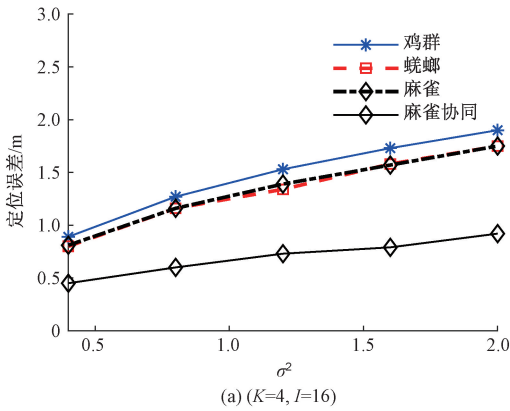


图 3 测量距离数据误差对定位误差的影响  
Fig. 3 Influence of measured distance data error on localization error

于经典的麻雀搜索算法有了显著的提升,具体来说,其定位精度提高了至少 21%。

表 2 麻雀协同定位精度的提升百分比 ( $K=4, I=16$ )

Table 2 The percentage improvement of sparrow cooperative positioning accuracy ( $K=4, I=16$ )

测距误差	平均定位误差/m		提高百分比/%
	麻雀	麻雀协同	
0.4	0.81	0.45	25
0.8	1.16	0.60	48
1.2	1.39	0.73	57
1.6	1.57	0.79	50
2.0	1.75	0.92	47

表 3 麻雀协同定位精度的提升百分比 ( $K=4, I=32$ )

Table 3 The percentage improvement of sparrow cooperative positioning accuracy ( $K=4, I=32$ )

测距误差	平均定位误差/m		提高百分比/%
	麻雀	麻雀协同	
0.4	0.83	0.65	21
0.8	1.14	0.73	35
1.2	1.33	0.80	40
1.6	1.58	0.87	45
2.0	1.76	0.94	46

### 3) 精细搜索对协同定位性能的影响

如图 4 所示,精细搜索对协同定位性能有明显的影 响。从图中可以看出,当使用原有的麻雀算法进行精细搜索时,协同定位算法容易陷入局部最优解。这可能导致定位结果的准确性降低,从而影响其定位精度。

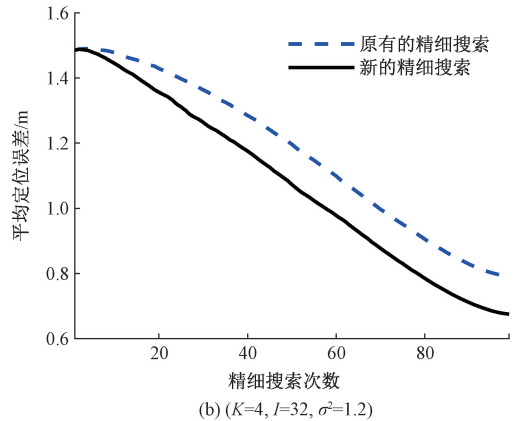
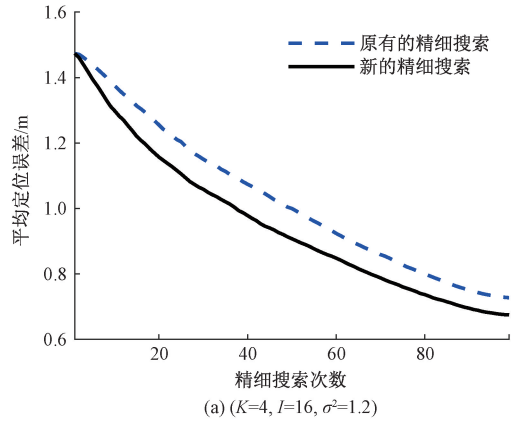


图 4 精细搜索对协同定位性能的影响  
Fig. 4 Influence of fine search on cooperative localization performance

### 4) 粗略搜索次数对协同定位性能的影响

图 5 展示了不同的粗略搜索次数对协同定位性能的影响。从图中可以看出,随着粗略搜索次数的增加,未知节点的初始位置逐渐接近最佳位置,这有助于协同定位算法更快地找到最佳位置。

### 5) 复杂度分析

通过麻雀搜索算法获得未知节点位置,其时间复杂度约为  $O(L \times K \times I \times N)$ ,其中  $L$  表示最大搜索次数, $N$  表示麻雀数, $K$  表示锚节点数, $I$  表示未知节点数。在协同定位的过程中,首先通过粗略搜索获得未知节点的初始位置,这个过程的时间复杂度约为  $O(L_1 \times K \times I \times N)$ ,其中  $L_1$  表示粗略搜索次数。然后,通过精细搜索来进一步确定未知节点的最佳位置,这个过程的时间复杂度约为  $O(L_2 \times N \times I^2)$ ,其中  $L_2$  表示精细搜索次数。因此,麻雀协同定位算法的总时间复杂度约为  $O(L_1 \times K \times$

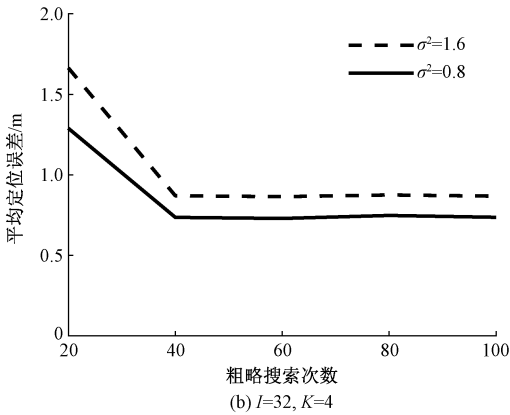
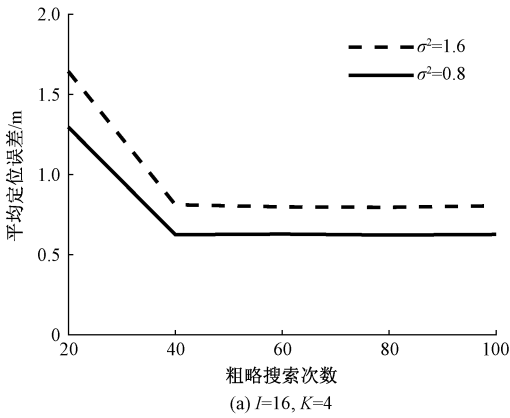


图5 粗略搜索次数对协同定位性能的影响  
Fig. 5 Influence of rough search number on cooperative localization performance

$I \times N) + O(L_2 \times N \times I^2)$ 。由表4可知,与经典的麻雀搜索定位算法相比,麻雀协同定位算法的平均运行时间复杂度有所增加。

表4 定位算法的平均运行时间( $\sigma^2 = 1.2$ )  
Table 4 The average running time of the localization algorithm ( $\sigma^2 = 1.2$ )

节点数目	平均运行时间/s	
	麻雀	麻雀协同
$I = 16, K = 4$	0.15	0.20
$I = 32, K = 4$	0.30	0.48

在实测实验中,使用了基于到达时间的测距设备来获取实际的距离数据。基于这些数据,计算得出的定位结果如表5所示。结果显示,麻雀定位算法的定位误差为0.70 m,而麻雀协同定位算法的定位误差降低到了0.44 m。由此可见麻雀协同定位效果优于麻雀定位效果。

表5 实测环境下定位结果( $I=8, K=4$ )

Table 5 The localization results in actual measurement environment ( $I=8, K=4$ )

未知节点 实际坐标/m	麻雀算法 估计坐标/m	麻雀协同算法 估计坐标/m
(8.4, 8.4)	(8.75, 8.5)	(8.77, 8.77)
(9.6, 4.8)	(10.13, 4.73)	(10.20, 5.14)
(9.6, 1.2)	(10.74, 2.44)	(10.4, 1.63)
(4.8, 9.6)	(5.1, 9.79)	(4.82, 9.8)
(8.4, 2.4)	(9.51, 3.61)	(8.79, 2.55)
(2.4, 7.2)	(2.69, 7.36)	(2.44, 7.3)
(8.4, 7.2)	(8.6, 7.49)	(8.72, 7.5)
(1.2, 3.6)	(0.89, 3.39)	(1.28, 3.4)

### 4 结论

本文提出了一种全新的麻雀搜索协同定位算法,该算法充分利用未知节点之间测量距离数据。仿真结果表明,这种算法能够显著提高定位精度,从而满足无线传感器网络的定位需求。然而,该算法并未考虑节点的移动性,因此无法支持移动定位应用。为了解决这一问题,在接下来的研究中将探讨如何将麻雀搜索协同定位算法应用于移动定位场景。

### 参考文献

[1] VERMA S, ZEDADALLY S, KAUR S, SHARMA AK. Intelligent and secure clustering in wireless sensor network (WSN)-based intelligent transportation systems[J]. IEEE Transactions on Intelligent Transportation Systems, 2022, 23(8):13473-13481.

[2] DA K, LI T, ZHU Y, et al. Recent advances in multisensor multitarget tracking using random finite set [J]. Frontiers of Information Technology and Electronic Engineering, 2021, 22:5-24.

[3] DASH P K, PANDA M. An efficient collision avoidance scheme for healthcare application [C]. International Conference on Intelligent Controller and Computing for Smart Power, 2022:1-4.

[4] MA M, XU S, JIANG J. A distributed gradient descent method for node localization on large-scale wireless sensor network [J]. IEEE Journal on Miniaturization for Air and Space Systems, 2023, 4(2):114-121.

[5] CACERES N L A, SONG I, KIM K. Fundamental limitations and state-of-the-art solutions for target node localization in WSNs: A review [J]. IEEE Sensors Journal, 2022, 22(24):23661-23682.

- [ 6 ] WATANABE F. Grid arrangement constraints-based localization for location-unknown anchors using AOA and TOA measurements [ C ]. IEEE International Conference on Indoor Positioning and Indoor Navigation, 2022; 1-8.
- [ 7 ] WANG Y, WAN J, LAI J. A wireless sensor networks positioning method in NLOS environment based on TOA and parallel Kalman filter [ C ]. IEEE International Conference on Communication Technology, 2019, 446-450.
- [ 8 ] VINAY P S, SINHA S. RSSI based improved weighted centroid localization method using indirect transmission and error estimation [ C ]. IEEE International Conference on Data Science and Information System, 2022, 1-6.
- [ 9 ] 高健, 陆阳, 李庆巧, 等. 采用三次通信的 TOF 与 TDOA 联合定位算法 [ J ]. 电子测量与仪器学报, 2020, 34(3):66-73.  
GAO J, LU Y, LI QQ, et al. TOF and TDOA joint positioning algorithm using three times communication [ J ]. Journal of Electronic Measurement and Instrumentation, 2020, 34 ( 3 ): 66-73.
- [ 10 ] LEI Y, DE G, FEI L. Improved sparrow search algorithm based DV-Hop localization in WSN [ C ]. 2020 Chinese Automation Congress ( CAC ), IEEE, 2020; 2240-2244.
- [ 11 ] 陈必帅, 王燕杰, 贾生尧, 等. 基于 Chan 与改进麻雀搜索算法的协同定位算法 [ J ]. 激光与光电子学进展, 2024, 61(4):376-384.  
CHENG B SH, WANG Y J, JIA SH Y, et al. Collocation algorithm based on Chan and improved sparrow search algorithm [ J ]. Laser & Optoelectronics Progress, 2024, 61(4):376-384.
- [ 12 ] ZHANG Q, ZHANG Y, ZHU X. A novel node localization algorithm based on sparrow search for WSNs [ C ]. IEEE International Conference on Electronics Information and Emergency Communication, 2021, 74-78.
- [ 13 ] ZHANG P, et al. RSSI-based indoor localization using sparrow search algorithm and backward propagation neural network [ C ]. International Conference on Big Data, Artificial Intelligence and Internet of Things Engineering, 2022, 151-155.
- [ 14 ] ZHU X, ZHANG Y, ZHANG Q. An improved DV-Hop node localization algorithm based on sparrow search for WSNs [ C ]. IEEE International Conference on Electronics Information and Emergency Communication, 2022, 154-157.
- [ 15 ] ZHAO Y, LI X, XIA M. Cooperative localization in wireless powered communication network [ C ]. IEEE International Conference on Communications, 2021, 1-6.
- [ 16 ] CHOI U S, CHO S J, KANG S W. Color medical image encryption using 3D chaotic cat map and NCA [ C ]. IEEE International Conference on New Technologies, Mobility and Security, 2019, 1-5.
- [ 17 ] Yanqiang T, Chenghai L, Yafei S, et al. Adaptive mutation sparrow search optimization algorithm [ J ]. Journal of Beijing University of Aeronautics and Astronautics, 2021, 49(3): 681-692.
- [ 18 ] CHOI U S, CHO S J, KIM J G, et al. Color image encryption based on PC-MLCA and 3-D chaotic cat map [ C ]. 2019 IEEE 4th International Conference on Computer and Communication Systems ( ICCCS ). IEEE, 2019; 272-277.
- [ 19 ] SANDELI M, BOUANAKA M A, KITOUNI I. An efficient localization approach in wireless sensor networks using chicken swarm optimization [ C ]. International Conference on Information Systems and Advanced Technologies, 2021, 1-6.
- [ 20 ] 潘志远, 卜凡亮. 基于蜣螂算法优化的 DV-Hop 定位算法 [ J ]. 电子测量与仪器学报, 2023, 37(7):33-41.  
PAN ZH Y, BU F L. DV Hop localization algorithm optimized based on dung beetle optimizer [ J ]. Journal of Electronic Measurement and Instrumentation, 2023, 37(7):33-34.

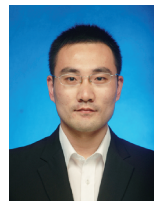
### 作者简介



夏斌 (通信作者), 2006 年于北京邮电大学获得博士学位, 现为山东理工大学电气与工程学院副教授, 主要研究方向为无线传感器网络。

E-mail: xiabin@sdut.edu.cn

**Xia Bin** (Corresponding author) received his Ph. D. degree from Beijing University of Posts and Telecommunications in 2006. He is currently an associate professor at School of Electrical and Electronic Engineering, Shandong University of Technology. His main research interest includes wireless sensor network.



张立晔, 2018 年于哈尔滨工业大学获得博士学位, 现为山东理工大学计算机科学与技术学院副教授, 主要研究方向为无线传感器网络。

E-mail: zhangliye@sdut.edu.cn

**Zhang Liye** received his Ph. D. degree from Harbin Institute of Technology in 2018. He is currently an associate professor at School of Computer Science and Technology, Shandong University of Technology. His research interest includes wireless sensor network.