

# 基于改进灰狼算法优化 BP 神经网络的 RSS 指纹定位<sup>\*</sup>

刘伟<sup>1,3</sup> 李艾龙<sup>2,3</sup> 李卓<sup>1,3</sup> 王智豪<sup>2,3</sup>

(1.桂林理工大学物理与电子信息工程学院 桂林 541006;2.桂林理工大学计算机科学与工程学院 桂林 541006;  
3.桂林理工大学广西嵌入式技术与智能系统重点实验室 桂林 541006)

**摘要:** 室内定位技术,特别是基于接收信号强度(RSSI)的指纹定位方法,因其成本低廉、设备支持广泛、易于部署、计算开销小等特点,受到了广泛关注。为了增强 RSSI 与实际物理距离之间的映射关系并提高测距精度,本文提出了一种基于改进灰狼优化(IGWO)算法与反向传播神经网络(BPNN)结合的 RSSI 测距算法。与遗传算法(GA)、粒子群算法(PSO)和经典灰狼优化算法(GWO)相比,改进的 GWO 算法在定位精度和全局搜索能力方面具有显著优势。通过实验,本文提出的 IGWO 算法在均方根误差 RMSE 上相比 GWO 算法、GA 算法、PSO 算法分别减少了 21.3%、15.7%、14.6%,IGWO 算法表现出了较好的定位性能,在精度和性能上均优于传统方法。

**关键词:** 室内定位;RSSI 测距;BP 神经网络;灰狼算法;粒子群算法

中图分类号: TN92 文献标识码: A 国家标准学科分类代码: 510.1050

## The RSS fingerprint positioning of BP neural network was optimized based on the improved gray wolf algorithm

Liu Wei<sup>1,3</sup> Li Ailong<sup>2,3</sup> Li Zhuo<sup>1,3</sup> Wang Zhihao<sup>2,3</sup>

(1. School of Physics and Electronic Information Engineering, Guilin University of Technology, Guilin 541006, China;  
2. School of Computer Science and Engineering, Guilin University of Technology, Guilin 541006, China;  
3. Guangxi Key Laboratory of Embedded Technology and Intelligent Systems, Guilin University of Technology, Guilin 541006, China)

**Abstract:** Indoor positioning technology, especially the Received Signal Strength Index(RSSI) -based fingerprinting positioning method, has received extensive attention due to its low cost, wide device support, easy deployment, and low computational overhead. In order to enhance the mapping relationship between RSSI and the actual physical distance and improve the ranging accuracy, this paper proposes an RSSI ranging algorithm based on Improved Grey Wolf Optimization(IGWO) algorithm and Back Propagation Neural Network(BPNN). Compared with Genetic Algorithm(GA), Particle Swarm Optimization(PSO) and classical Grey Wolf Optimization algorithm(GWO), the improved GWO algorithm has significant advantages in positioning accuracy and global search ability. Through experiments, the root mean square error(RMSE) of IGWO algorithm is reduced by 21.3%, 15.7% and 14.6% respectively compared with GWO algorithm, GA algorithm and PSO algorithm. IGWO algorithm shows better positioning performance, and is superior to the traditional methods in accuracy and performance.

**Keywords:** indoor positioning;RSSI ranging;BP neural network;Grey Wolf algorithm;Particle swarm optimization;

## 0 引言

随着移动互联网技术的快速发展,人们对基于位置的服务(location based service, LBS)的需求日益增长<sup>[1]</sup>。在室外环境中,全球导航卫星系统(global navigation satellite

system,GNSS)<sup>[2]</sup>,例如我国的北斗卫星导航系统能够为用户提供较高精度的定位服务,基本满足了用户在室外场景中对基于位置服务的需求,为人们的日常生活、工作和其他场景(如导航汽车和路线规划)创造了极大的便利。然而,在室内环境中,卫星信号可能会受多径效应、室内环境的干

扰(如路由器、微波炉等)等从而引起定位误差,使GNSS的定位精度大大降低,可能无法满足定位要求。因此,研究能够在室内环境中实现高精度定位的方法具有重要意义。

针对室内环境的复杂特性,人们提出了多种室内定位技术,例如基于蓝牙<sup>[3]</sup>、超宽带(ultra wide band, UWB)<sup>[4]</sup>、射频识别(radio frequency identification, RFID)<sup>[5]</sup>、无线局域网、磁场<sup>[6]</sup>、超声波和计算机视觉等方法。这些技术各具优缺点,适用于不同的应用场景。基于信道状态信息(channel state information, CSI)<sup>[7-8]</sup>和接收信号强度(received signal strength, RSS)<sup>[9-11]</sup>的室内定位方法得到了广泛的研究和应用。其中,基于CSI的定位方法能够提供更为精确的定位信息,但其实施难度较大,对硬件的要求较高。因此,基于RSS的室内定位技术,凭借其简便性和较低的设备要求,成为了研究者们更为关注的方向。RSS作为一种简单易用的信号强度指标,能够有效地反映接收设备与信号源之间的距离信息。通过测量不同位置的RSS值,结合合适的定位算法,可以实现对目标位置的估算。

然而,基于RSS的定位技术也面临诸多挑战,如多径效应、信号衰减、环境干扰等,这些因素可能会导致信号强度的不稳定,从而影响定位精度。为了解决这些问题,研究者们提出了各种优化方法,倪云峰等<sup>[12]</sup>设计了一种基于接收信号强度的室内人员卡尔曼滤波定位算法,该算法在提高定位精度和降低误差方面取得了显著的效果,但是文中的实验数据较为有限,没有提供更多的实际场景测试数据和对比分析(比如与其他的定位算法对比),不能更好的评估算法的性能。姚军等<sup>[13]</sup>提出的基于K-means聚类算法预处理样本数据的BP神经网络测距算法,能够有效降低RSS测距误差,并优化了目标RSS值与真实距离的映射关系,但是并未对BP神经网络进行进一步的优化处理,可能导致在复杂的环境下性能提升受到限制。朱清山等<sup>[14]</sup>结合了模拟退火算法(simulated annealing algorithm, SA)和遗传算法(genetic algorithm, GA)提出SAGA来优化神经网络的初始权重和阈值,结果表明,对比传统反向传播神经网络(back propagation neural networks, BPNN)和GA-BP有着更高的定位精度,但是文中的实验场景较为有限,无法更加全面的反应实际复杂环境中的多径效应、信号衰减等问题。余振宝等<sup>[15]</sup>提出了一种利用遗传算法(GA)优化的反向传播(BP)神经网络,构建了无线信号路径损耗模型(GA-BP),利用遗传算法对BP神经网络中的初始值和阈值进行优化,克服了BP神经网络局部极小解的缺陷,但是文中使用的数据集主要来自一个空旷的阳台环境,缺乏多样化的室内环境数据,不同的室内环境对信号传播的影响不同,模型在这些环境中的表现也会有所不同。林贻若等<sup>[16]</sup>提出的基于灰狼算法(grey wolf optimizer, GWO)结合BP神经网络的RSSI测距算法,通过优化BP神经网络的初始权重和偏置,避免了陷入局部极小解,相对于遗传算法具有更快的收敛速度和更强的稳定性,提高了网络的全

局优化性,但是该方法对于数据没有进行滤波处理,在存在噪声干扰(如电磁干扰、多径效应、墙壁反射等)的环境下,其定位性能可能会受到显著影响。

由于RSSI信号受多种因素的影响,导致信号强度与距离之间的关系不再是简单的线性关系,BP神经网络有着较强的非线性映射能力,在对于室内定位尤其是基于RSSI信号的定位,BP神经网络能够有效学习并处理这些复杂的非线性关系。其次,不仅是RSSI数据,还可以处理其他的比如温度、湿度<sup>[17]</sup>等非线性数据。灰狼优化算法因其简单高效、自适应强等特点,已成功应用于各种优化问题,尤其是在复杂的非线性优化任务中。传统的灰狼优化算法在许多领域取得了显著的成绩,但其仍然面临着局部最优解和搜索效率低的问题,为了克服这些局限性,本文在BP神经网络的基础上提出了改进的灰狼算法优化BP神经网络,该方法利用改进后的灰狼优化算法寻找BPNN最优的权值和阈值,将其赋值于BP神经网络,从而提高测距精度。

## 1 基本理论

### 1.1 灰狼优化算法

灰狼优化算法(GWO)是一种模拟灰狼捕猎行为的群体智能优化算法。它由Mirjalili<sup>[18]</sup>等在2014年提出,GWO通过模拟灰狼的社会行为和捕猎策略来进行搜索和优化。灰狼有非常强的群体合作性,通常在捕猎时按照等级划分, $\alpha$ 狼、 $\beta$ 狼和 $\delta$ 狼分别起着主导作用,而其他狼则是 $\omega$ 狼,负责辅助捕猎。

图1为灰狼社会等级划分,展示了各个灰狼在种群中的社会等级。在算法中,灰狼优化算法在迭代过程中遵循严格的层级结构。

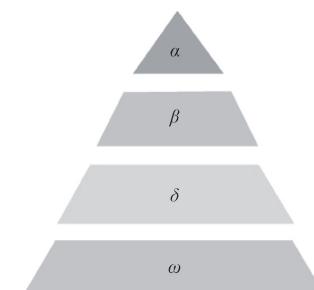


图1 灰狼社会等级划分

Fig. 1 Social hierarchy of gray wolves

其中, $\alpha$ 狼是灰狼群体中最高级的领导者,在灰狼群体中起着决策的作用。 $\beta$ 狼是群体中的副领导者,辅助 $\alpha$ 狼进行决策和执行。 $\delta$ 狼是 $\beta$ 狼的下属,听从 $\alpha$ 狼、 $\beta$ 狼的命令,负责内部的群体事务。 $\omega$ 狼是狼群体中的普通成员,负责执行 $\alpha$ 狼、 $\beta$ 狼和 $\delta$ 狼的命令和决策。在算法中,灰狼的行为可以分为3个主要阶段:围猎、狩猎、攻击猎物。

1)围猎,GWO算法围猎行为数学模型计算如式(1)和(2)所示。

$$D(t) = |\mathbf{C} \cdot \mathbf{X}_p(t) - \mathbf{X}(t)| \quad (1)$$

$$\mathbf{X}(t+1) = \mathbf{X}_p(t) - \mathbf{A} \cdot D(t) \quad (2)$$

式中:  $t$  表示当前迭代次数, 系数向量为  $\mathbf{A}$  和  $\mathbf{C}$ , 猎物和狼之间的距离为  $D(t)$ , 猎物的位置向量为  $\mathbf{X}_p(t), \mathbf{X}(t+1)$  表示灰狼在下一时刻  $t+1$  的位置。系数向量  $\mathbf{A}$  和  $\mathbf{C}$  计算公式如式(3)和(4)所示。

$$\mathbf{A} = 2\mathbf{a} \cdot \mathbf{r}_1 - \mathbf{a} \quad (3)$$

$$\mathbf{C} = 2\mathbf{r}_2 \quad (4)$$

式中:  $\mathbf{r}_1$  和  $\mathbf{r}_2$  是取值为  $[0, 1]$  均匀分布的随机向量,  $a$  是收敛因子, 根据迭代次数  $t$  从 2 线性减小到 0。

2) 狩猎, 狼群根据  $\alpha$  狼、 $\beta$  狼、 $\delta$  狼的位置来寻找猎物, 捕猎过程是通过搜索空间中的各个狼的位置更新来实现。这里, 猎物代表最优解, 而狼群需要通过与  $\alpha$  狼、 $\beta$  狼、 $\delta$  狼的相对位置调整来“捕猎”最优解。 $\omega$  狼更新他们的位置如下:

$$X_{\omega}(t+1) = \frac{X_{\alpha}(t) + X_{\beta}(t) + X_{\delta}(t)}{3} \quad (5)$$

其中,  $X_{\omega}(t+1)$  是第  $t+1$  次迭代时第  $i$  只  $\omega$  狼的位置, 通过当前迭代的  $\alpha$  狼、 $\beta$  狼、 $\delta$  狼位置来计算。 $X_{\alpha}(t), X_{\beta}(t), X_{\delta}(t)$ , 分别是当前迭代  $t$  中  $\alpha$  狼、 $\beta$  狼、 $\delta$  狼的位置, 考虑与  $\alpha$  狼、 $\beta$  狼、 $\delta$  狼的距离, 将三者取均值在全局搜索和局部开发之间取得平衡。 $X_{\alpha}(t), X_{\beta}(t), X_{\delta}(t)$  的更新如式(6)所示。

$$\begin{cases} X_{\alpha}(t+1) = X_{\alpha}(t) - A_1 \cdot D_{\alpha}(t) \\ X_{\beta}(t+1) = X_{\beta}(t) - A_2 \cdot D_{\beta}(t) \\ X_{\delta}(t+1) = X_{\delta}(t) - A_3 \cdot D_{\delta}(t) \end{cases} \quad (6)$$

式中:  $A_1, A_2, A_3$  分别表示  $\alpha$  狼、 $\beta$  狼、 $\delta$  狼的系数,  $D_{\alpha}(t), D_{\beta}(t), D_{\delta}(t)$  分别表示第  $t$  次迭代时  $\omega$  狼与  $\alpha$  狼、 $\beta$  狼、 $\delta$  狼的距离, 如式(7)所示。

$$\begin{cases} D_{\alpha}(t) = |\mathbf{C}_1 \cdot \mathbf{X}_{\alpha}(t) - \mathbf{X}(t)| \\ D_{\beta}(t) = |\mathbf{C}_2 \cdot \mathbf{X}_{\beta}(t) - \mathbf{X}(t)| \\ D_{\delta}(t) = |\mathbf{C}_3 \cdot \mathbf{X}_{\delta}(t) - \mathbf{X}(t)| \end{cases} \quad (7)$$

其中,  $C_1, C_2, C_3$  分别表示  $\alpha$  狼、 $\beta$  狼、 $\delta$  狼的系数, 控制  $\omega$  狼与  $\alpha$  狼、 $\beta$  狼、 $\delta$  狼之间的距离。 $X_{\alpha}(t), X_{\beta}(t), X_{\delta}(t)$  分别表示  $\alpha$  狼、 $\beta$  狼、 $\delta$  狼的当前位置,  $X(t)$  表示  $\omega$  狼在当前迭代的位置。 $C_1, C_2, C_3$  的计算公式为:

$$C_i = 2 \cdot r, i = 1, 2, 3 \quad (8)$$

式中:  $r$  为  $[0, 1]$  范围内随机生成均匀分布的数值, 使得狼的移动方向和幅度具有不确定性, 增强搜索的多样性。

3) 攻击猎物, 攻击猎物是捕猎的最后阶段, 此时狼群已经非常接近猎物, 优化过程逐渐从全局搜索过渡到局部搜索。此阶段通过动态调整系数来精确控制搜索范围。动态调整系数  $a$  由式(9)所示。

$$a = 2 - \frac{2t}{T} \quad (9)$$

式中:  $t$  为当前迭代次数,  $T$  为最大迭代次数。

## 1.2 BP 神经网络

BP 神经网络(BPNN)是一种广泛应用的人工神经网

络, 是最早被广泛使用的多层前馈神经网络之一。它由输入层、隐藏层和输出层组成, 其结构如图 2 所示, 每一层由多个神经元节点构成, 隐藏层可以是单层或多层结构。相邻层之间的节点通过连接形成连接权重和阈值, 这些连接权重通过训练过程进行优化。BPNN 的学习过程分为两个阶段: 在正向传播阶段, 输入数据通过输入层传入, 经过隐藏层的计算后传递至输出层, 最终生成网络的输出结果; 在反向传播阶段, 计算输出结果与真实值之间的误差, 并不断调整网络的连接权重和阈值, 使误差最小化, 进而提高网络的准确性。

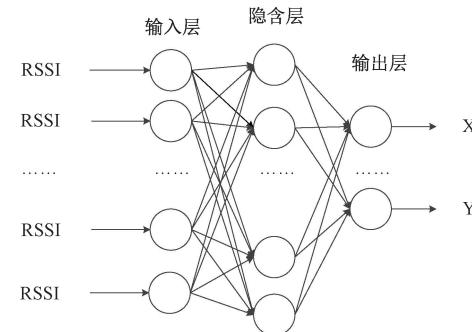


图 2 BP 神经网络结构

Fig. 2 BP neural network structure

BP 神经网络在处理非线性和复杂数据问题时表现出色, 尤其在室内定位中, 能够有效处理由不同设备获取的射频信号 RSSI 数据, 并进行差异校正。由于其优异的性能, BP 神经网络已被广泛应用于实际生活中的各种任务。然而, 网络的初始权重和阈值选择对网络的性能和训练速度具有重要影响。如果初始权重和阈值设置不合理, 可能会导致网络陷入局部最优解, 无法收敛, 或者训练过程异常缓慢。

## 1.3 粒子群算法

粒子群算法(particle swarm optimization, PSO)是一种自然启发的元启发式算法, 在 1995 年中首次提出, 它主要是通过模拟鸟群的觅食行为来建模的。PSO 算法基于群体智能, 管理多个协作粒子来探索整个搜索空间。在 PSO 算法中, 每个粒子在  $S$  维搜索空间中的位置用一个向量表示。如果粒子  $i$  的位置是  $x_i$ , 则它可以表示为:

$$x_i = [x_{i1}, x_{i2}, x_{i3}, \dots, x_{iS}] \quad (10)$$

式中:  $x_i$  是粒子  $i$  的当前位置向量。 $x_{ij}$  是粒子  $i$  在第  $j$  维的位置分量,  $j = 1, 2, \dots, S$ 。每个粒子都有一个唯一的位置  $x_i$  和速度  $v_i$ , 它们可以代表一个候选解。在搜索过程中, 第  $i$  个粒子的位置受自身历史最优位置  $P_{i,best}$  和整个群体中所有粒子的最佳位置  $G_{best}$  的影响。因此, 使用以下等式更新粒子位置  $x_i$ :

$$x_i(t+1) = x_i(t) + v_i(t+1) \quad (11)$$

其中,  $v_i$  通过以下公式计算粒子速度:

$$\begin{aligned} v_i(t+1) &= \omega v_i(t) + c_1 r_1 (P_{i,best} - x_i(t)) + c_2 r_2 \\ (G_{best} - x_i(t)) \end{aligned} \quad (12)$$

其中,  $t$  是迭代次数,  $c_1$  和  $c_2$  是加速度系数,  $r_1$  和  $r_2$  表示区间  $[0, 1]$  内均匀分布的随机变量,  $\omega$  表示惯性权重。

#### 1.4 Logistic 映射与 Tent 映射

Logistic 映射是一种简单的非线性动力学系统,通常用于描述种群增长或其他生物过程。其数学表达如式(13)所示。

$$x(t+1) = r \cdot x(t) \cdot (1 - x(t)) \quad (13)$$

其中,  $x(t)$  是第  $t$  次迭代的状态,  $x(t+1)$  是第  $t+1$  次迭代的状态,  $r$  是控制参数, 通常叫做增长率, 取值范围为  $(3.569, 4]$ <sup>[19]</sup>, 随着  $r$  的增加, Logistic 映射从稳定的固定点到周期性行为再到混沌现象发生转变。它是研究混沌行为和分岔现象的经典模型, 广泛应用于生态学、混沌理论和数学研究中。

Tent 映射是一种具有“帐篷形状”的非线性映射, 其公式为:

$$x(t+1) = \begin{cases} r \cdot x(t), & 0 \leq x(t) \leq 0.5 \\ r \cdot (1 - x(t)), & 0.5 < x(t) \leq 1 \end{cases} \quad (14)$$

式中:  $x(t)$  是第  $t$  次迭代的状态,  $x(t+1)$  是第  $t+1$  次迭代的状态,  $r$  是控制参数, 通常叫做增长率, 它展示了类似于 Logistic 映射的混沌行为, 但数学形式更简单, 且具有分段线性结构。Tent 映射常用于混沌加密、伪随机数生成以及动力学系统的研究中。

## 2 改进灰狼优化算法策略

### 2.1 引入粒子群算法策略

GWO 在优化过程中未充分利用个体的历史经验, 同时缺乏个体与群体之间的有效信息交流, 容易导致早熟收敛, 陷入局部最优。为了解决这一问题, 本文综合考虑了算法的收敛速度和精度, 融合了 PSO 算法中的速度更新和位置调整机制。通过引入 PSO 的速度更新公式, 综合改进  $\alpha$  狼、 $\beta$  狼、 $\delta$  狼的位置, 使得 GWO 在寻优过程中具备记忆能力。同时, 通过动态调节惯性权重, 有效平衡了混合算法的全局搜索与局部开发能力, 进一步提升了优化性能。距离公式更新如下:

$$\begin{cases} D_\alpha = |C_1 \cdot X_\alpha(t) - \omega(t)X(t)| \\ D_\beta = |C_2 \cdot X_\beta(t) - \omega(t)X(t)| \\ D_\delta = |C_3 \cdot X_\delta(t) - \omega(t)X(t)| \end{cases} \quad (15)$$

式中: 灰狼在搜索空间中的探索和开发由一个自适应惯性常数  $\omega(t)$  控制,  $\omega(t)$  由式(16)给出,  $\omega_{\max}$  和  $\omega_{\min}$  分别为最大和最小惯性权重,  $t$  为当前迭代次数,  $T$  为最大迭代次数。

$$\omega(t) = \omega_{\max} - \frac{\omega_{\max} - \omega_{\min}}{T} \cdot t \quad (16)$$

基于以上所有内容, 通过更新速度和更新位置执行 GWO 和 PSO 变体的组合, 如式(17)所示。

$$\begin{cases} v_i(t+1) = \omega(t)v_i(t) + c_1 r_1 (X_\alpha(t) - x_i(t)) + \\ c_2 r_2 (X_\beta(t) - x_i(t)) + c_3 r_3 (X_\delta(t) - x_i(t)) \\ X_i(t+1) = X_i(t) + v_i(t+1) \end{cases} \quad (17)$$

其中,  $v_i(t+1)$  是第  $t+1$  次迭代时第  $i$  只  $\omega$  狼的速度,  $v_i(t)$  是第  $t$  次迭代时第  $i$  只  $\omega$  狼的当前速度,  $c_1, c_2, c_3$  为加速因数,  $r_1, r_2, r_3$  为均匀分布在  $[0, 1]$  区间内的随机数,  $x_i(t)$  表示当前灰狼  $i$  的位置,  $X_\alpha(t), X_\beta(t), X_\delta(t)$  为  $\alpha$  狼、 $\beta$  狼、 $\delta$  狼的位置, 这些位置分别代表灰狼优化算法中的三种领导者个体, 引导普通灰狼(粒子)向更优解靠近。 $X_i(t+1)$  表示第  $i$  只狼在第  $t+1$  轮迭代后的新位置,  $X_i(t)$  表示第  $i$  只狼在第  $t$  轮迭代时的当前位置。位置更新公式将当前的位置和速度结合, 决定灰狼在搜索空间中的下一步。

### 2.2 混沌映射初始化

启发式算法的优化过程受到初始种群质量的显著影响。传统的灰狼优化算法通常通过随机方式生成初始种群, 但这种方式往往导致搜索效率低, 且全局搜索能力较弱。因此, 本文采用了 Logistic 映射与 Tent 映射相结合的方式来生成初始种群, 从而提高了种群的分布均匀性, 并增强了全局搜索能力。引入动态权重  $\alpha$  来调整这两种映射的相对影响力, 加入非线性调整参数  $\beta$  来放大混沌值的变化, 并在混合映射时进一步增强随机性和不可预测性。可以使得种群更加随机地、均匀地分布在搜索空间内。

将 Logistic 映射和 Tent 映射进行加权组合, 并使用动态调整的权重  $\alpha$  来根据当前迭代阶段确定两者的影响。改进后如式(18)所示。

$$\begin{cases} x(t+1) = \\ \left[ \alpha \cdot r \cdot x(t) \cdot (1 - x(t)) + (1 - \alpha) \cdot \frac{4 - r}{2} \cdot x(t) \right]^\beta \\ \text{mod } 1, \quad x(t) < 0.5 \\ \left[ \alpha \cdot r \cdot (1 - x(t)) \cdot x(t) + (1 - \alpha) \cdot \frac{4 - r}{2} \cdot (1 - x(t)) \right]^\beta \\ \text{mod } 1, \quad x(t) \geq 0.5 \end{cases} \quad (18)$$

式中:  $x(t)$  是在第  $t$  次迭代时的值,  $x(t+1)$  是在第  $t+1$  次迭代时的值,  $\beta$  为非线性调整参数, 通常选择大于 1 的值, 目的是增加混沌值的随机性,  $r$  为一个影响映射效果的控制参数。动态权重  $\alpha$  根据当前个体迭代步数进行动态调整 Logistic 映射和 Tent 映射的贡献, 其计算过程如式(19)所示。

$$\alpha = \frac{1}{1 + e^{-r \cdot (i+j-\frac{S}{2})}} \quad (19)$$

$\gamma$  是控制权重变化速率的参数, 通常选择小的正数值。 $i$  和  $j$  是当前种群的索引, 表示种群个体的行和列的位置。 $S$  为种群中个体的数量。随着迭代的推进, 动态调整 Logistic 映射和 Tent 映射的贡献比例, 初期偏向全局探索

(Logistic 映射), 后期偏向局部开发(Tent 映射)。

### 2.3 精英反向学习

精英反向学习机制, 由群体中的精英个体组成逆向群体, 在灰狼算法中精英个体代表  $\alpha$  狼、 $\beta$  狼、 $\delta$  狼, 再从逆向群体中与现有群体中选择优良个体组成新的群体, 从而提高群体的多样性。在 GWO 算法优化过程中, 将狼种群的极值点个体设为种群中的精英个体:

$$X_{i,j}^E = (X_{i,1}^E, X_{i,2}^E, \dots, X_{i,D}^E) \quad (20)$$

反向求解灰狼精英个体为:

$$\bar{X}_{i,j}^E = (\bar{X}_{i,1}^E, \bar{X}_{i,2}^E, \dots, \bar{X}_{i,D}^E) \quad (21)$$

可定义精英个体反向解为:

$$\bar{X}_{i,j}^E = \xi \cdot (u_{bj} + l_{bj}) - X_{i,j}^E \quad (22)$$

式(22)中:  $\xi$  为  $[0,1]$  的动态系数;  $u_{bj}$  和  $l_{bj}$  分别为动态边界的上界和下界。若反向解  $\bar{X}_{i,j}^E$  超出边界, 则利用式(23)进行重置, 以确保搜索过程的有效性和稳定性。式中  $rand$  为  $[0,1]$  中的随机数。

$$\bar{X}_{i,j}^E = l_{bj} + rand \cdot (u_{bj} - l_{bj}) \quad (23)$$

传统的精英反向学习机制虽然能够增加种群的多样性, 避免早熟收敛, 但其反向点的影响是固定的, 无法适应算法搜索过程的不同阶段。因此, 改进精英反向学习机制引入基于当前迭代次数的 sigmoid 函数的动态因子, 随着迭代次数的增加, 因子的值会逐渐从 0 上升到 1。这样, 反向点在早期的影响较小, 而在后期逐渐增强对更新的影响, 从而帮助算法跳出局部最优并加速全局搜索。动态因子如式(24)所示。

$$\alpha_f = \frac{1}{1 + e^{(-\gamma \cdot (t - \frac{T}{2}))}} \quad (24)$$

式中:  $\gamma$  是调整因子, 控制因子的变化速率;  $t$  是当前迭代次数,  $T$  是最大迭代次数。

在引入动态因子后, 反向点的计算不再是固定的, 而是随着迭代次数的增加进行加权调整。新的精英个体反向解为如式(25)所示。

$$\bar{X}_{i,j}^E = \alpha_f \cdot \bar{X}_{i,j}^E + (1 - \alpha_f) X_{i,j}^E \quad (25)$$

式中:  $\bar{X}_{i,j}^E$  是狼群精英个体的反向位置,  $X_{i,j}^E$  为狼群精英个体的当前位置。通过式(23)和式(25)计算出反向位置, 计算反向位置的自适应值, 若反向位置的自适应值比当前精英位置更优, 则更新精英位置, 如式(26)所示。

$$X_{i,j}^E = \begin{cases} \bar{X}_{i,j}^E & f(\bar{X}_{i,j}^E) < f(X_{i,j}^E) \\ X_{i,j}^E & \text{otherwise} \end{cases} \quad (26)$$

式中:  $f(\bar{X}_{i,j}^E)$  表示反向位置  $\bar{X}_{i,j}^E$  的自适应值,  $f(X_{i,j}^E)$  表示当前位置  $X_{i,j}^E$  的自适应值, 具体自适应函数将在下文说明。

### 2.4 非线性收敛因子

本文中将式(9)中的线性收敛因子改进为非线性收敛因子, 非线性收敛因子通过动态调整优化过程中的搜索策略, 在不同的优化阶段实现不同的收敛速度, 从而平衡探索

和开发的需求。与线性收敛因子相比, 非线性收敛因子能够更有效地避免过早收敛到局部最优, 并在接近全局最优时加速收敛。在本文中改进的收敛因子  $a(t)$  的动态变化由式(27)控制。

$$a(t) = \begin{cases} a_s - \alpha \left( \frac{t}{T/2} \right)^2, & t \leq \frac{T}{2} \\ a_e + \beta \left( 1 - \frac{t - T/2}{T/2} \right)^2, & t > \frac{T}{2} \end{cases} \quad (27)$$

式中:  $a_s$  为初始收敛因子的值,  $a_e$  是最终的收敛因子值,  $\alpha$  和  $\beta$  是控制收敛速度参数,  $t$  是算法的当前迭代次数,  $T$  是算法的最大迭代次数。在本文中,  $a_s, a_e, \alpha, \beta$  分别取 1.0、0.5、0.5。

图 3 是本文改进的收敛因子与灰狼算法中的收敛因子对比图, 描述了在迭代过程中收敛因子的一个变化过程, 从图 3 可以看出, GWO 算法采用的是线性递减收敛因子, 而本文提出的改进控制参数方法则采用非线性递减策略。与原始灰狼算法线性收敛方法相比, 本文的方法在初期允许狼群以较大的步长进行全局搜索, 收敛速率相对较慢, 这样不仅能保证种群的多样性, 还能增强全局搜索能力。到了中期, 收敛速率加快, 从而提升了算法的收敛速度; 在后期, 收敛因子的步长逐渐减小, 收敛速率变缓, 有助于算法在局部区域进行精细搜索, 从而提高了解的精度和效率。

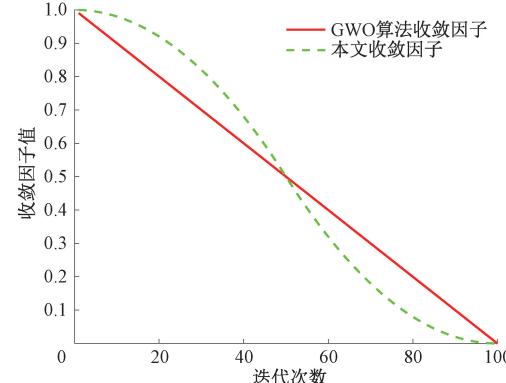


图 3 收敛因子变化曲线

Fig. 3 Convergence factor change curve

### 2.5 系统流程

为了克服 BP 神经网络对初始权重敏感的问题, 本文采用改进的灰狼优化算法(improved grey wolf optimizer, IGWO)来优化 BPNN(IGWO-BP)的权值和阈值。通过 IGWO 算法, 将优化后的权重和阈值将作为初始值传递给 BP 神经网络进行训练, 从而提高训练速度和精度, 有效降低定位误差。

通过改进 GWO 优化 BP 神经网络的 RSSI 测距算法主要包含以下步骤。

步骤 1) 数据预处理。采集数据, 进行滤波处理, 标准化和归一化处理。

步骤 2) 判断: 如果已有优化后的 BPNN 预测模型, 直

接进入步骤9,使用该模型进行预测。如果没有优化后的预测模型,进入步骤3),开始构建并优化BPNN。

步骤3)确定模型输入层节点数*i*、输出层节点数*o*,这里设置输出层节点数*o*为2,指横纵坐标X,Y。根据式(28)确定隐含层节点数*h*,常数*k*为[1,10]的取值。

$$h = k + \sqrt{i + o} \quad (28)$$

步骤4)设置GWO优化的参数。种群规模,搜索空间大小,最大迭代次数,训练终止条件,设定训练误差达到期望值时结束训练。

步骤5)生成初始灰狼个体。每只灰狼的个体包含BP神经网网络所有权重和阈值。个体位置为:

$$X = [W_1, B_1, W_2, B_2] \quad (29)$$

式中:*W*<sub>1</sub>、*B*<sub>1</sub>是输入层到隐含层的权重和偏置。*W*<sub>2</sub>、*B*<sub>2</sub>是输入层到隐含层的权重和偏置。

初始化优化参数的搜索空间。确定优化的参数为BP神经网络中的所有权重和偏置。优化参数的搜索空间维度为:

$$D = I \times H + H \times O + H + O \quad (30)$$

式中:*I*为输入层节点数,*H*为隐含层节点数,*O*为输出层数。具体包括:输出层到隐含层的所有权重(*I*×*H*),隐含层的所有偏置(*H*),隐含层到输出层的所有权重(*H*×*O*),输出层的所有偏置(*O*)。通过式(18)初始化来生成初始种群的位置。

步骤6)计算每只灰狼的适应度函数。GWO算法需要通过自适应函数来计算自适应值,其自适应函数定义为GWO算法通过自适应函数来计算每只狼的适应度值。自适应函数的形式为:

$$F_i = \frac{1}{S} \sum_{i=1}^S \sqrt{(y_i - y'_i)^2} \quad (31)$$

式中:*F*<sub>i</sub>为第*i*只灰狼的自适应值;*y*<sub>i</sub>为样本的实际值;*y'*<sub>i</sub>为样本的预测值;*S*为样本总量。

步骤7)计算反向点的自适应值,满足更新条件即根据式(26)更新灰狼位置,否则通过步骤6)计算的适应度值,根据式(6)、(15)、(16)、(17)更新灰狼的位置,检查灰狼位置是否超越搜索空间的边界,如果超出则将灰狼位置重新映射到边界内。

步骤8)检查是否满足终止条件。在每次迭代后,检查当前误差是否达到预期最小值或者已达到最大迭代次数*T*,满足则停止,否则返回步骤6)和步骤7)进行迭代优化。

步骤9)输出最优解。在优化过程中,选择适应度值最小的灰狼位置作为最终的解,得到优化后的权重和偏置。使用优化后的权重和偏置作为初始值,训练BPNN直到达到迭代次数。

图4展示了基于改进GWO算法优化BP神经网络的测距算法的流程。当完成IGWO-BP测距算法的训练后,输入新的RSSI数据,即可估计其与无线接入点(access point,AP)之间的距离。

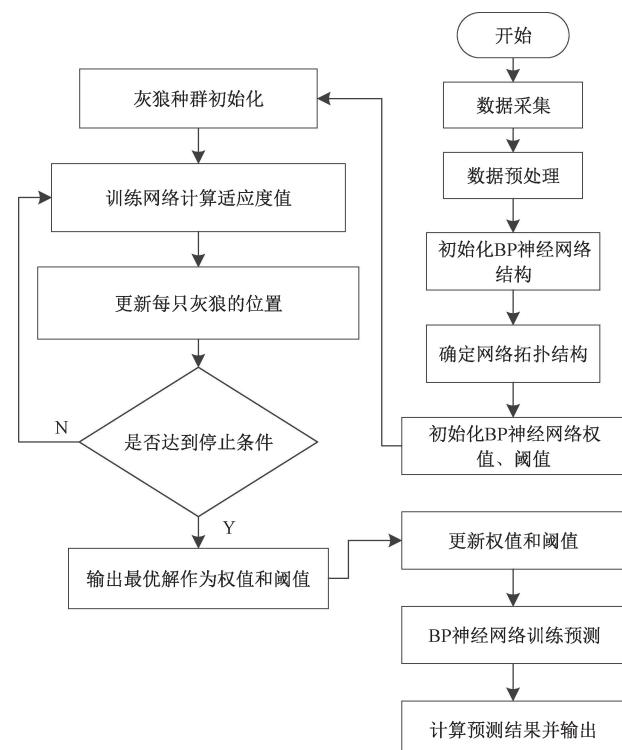


图4 基于改进灰狼算法优化BP神经网络的预测流程图

Fig. 4 Prediction flow chart of BP neural network optimized by improved grey Wolf algorithm

### 3 试验及分析

#### 3.1 实验数据及误差指标

实验数据1选取UCI的BLE RSSI室内定位数据集<sup>[20]</sup>,实验区域在西密歇根大学的图书馆一楼,面积大约为15 m×18 m,室内信标节点平面图如图5所示,实验选取13个蓝牙信标,每个信标间隔2~3 m,分别布设在不同位置,参考点一共130个,每个参考点采集5~10次指纹,一共1136组参考点指纹数据。测试点一共50个,每个测试点采集5~10次指纹,一共284组测试指纹数据,并将数据进行均值滤波处理。

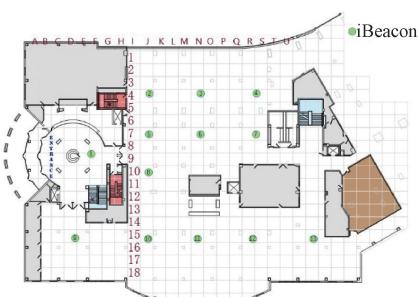


图5 蓝牙信标分布实验平面图

Fig. 5 Experimental floor plan of bluetooth beacon distribution

本文选取3个误差指标评估模型预测精度:

$$MAE = \frac{1}{N} \sum_{i=1}^N |y_i - y'_i| \quad (32)$$

$$MAPE = \frac{1}{N} \sum_{i=1}^N \left| \frac{y_i - y'_i}{y_i} \right| \times 100\% \quad (33)$$

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^N (y_i - y'_i)^2} \quad (34)$$

式中: MAE 是平均绝对误差, 衡量的是预测值与真实值之间的平均绝对差异。MAPE 平均绝对百分比误差, 是通过计算每个样本的误差占真实值的百分比, 衡量预测值的相对误差。RMSE 是均方根误差, 衡量的是预测值与真实值之间误差均方根。 $y_i$  为真实值,  $y'_i$  为预测值,  $N$  为样本个数。

### 3.2 参数选择

在构建和训练 BPNN 模型时, 超参数的选择对模型的性能具有重要影响, 为了确定最优的超参数组合, 本文采用网格搜索, 通过遍历不同的参数组合, 评估模型在测试集上的表现, 最终选择误差指标最小的参数组合。本文主要调整 3 个超参数: 迭代次数、学习率、期望误差。候选参数如表 1 所示。

表 1 候选参数

Table 1 Candidate parameters

| 参数名    | 范围                                     |
|--------|--|
| 最大迭代次数 | [80, 90, 100, 110, 120, 130, 140, 150] |
| 学习率    | [0.000 1, 0.001, 0.01, 0.1]            |
| 期望误差   | [0.000 01, 0.000 1, 0.001, 0.01]       |

通过评估每种参数组合在测试集上的均方根误差(RMSE), 最终选择使 RMSE 最小的参数组合作为最优参数。寻优结果如表 2 所示。

根据寻优结果, 本文确定最优的参数组合为最大迭代次数 100, 学习率 0.01, 期望误差 0.000 1。

对于隐含层节点数的选择, 根据式(28)得到取值范围  $h$  为 3~12, 并结合网络误差指标 MAPE 来确定隐含层节点数。表 3 为不同隐含层节点数对应的误差指标值。通过表中的对比可知隐含层节点数为 12 时输出的 MAPE 值最小为 0.045 387。为了确保不同模型在相同样本下进行公平的对比, 本文对各个算法的种群数量、迭代次数和优化目标进行了相同的设置, 尽量减少因实验设置差异带来的影响。

本文在输入层到隐含层传递函数选择 tansig 函数, 其具有非线性映射能力。在隐含层到输出层的传递函数选择 purelin 函数, 该函数的线性特性使其能够输出任意范围的数值, 符合本文的需求, 隐含层通过 tansig 函数引入了足够的非线性, 输出层使用 purelin 函数在保留非线性特征的同时, 简化模型的计算过程, 避免对输出值的范围造成不必要的限制, 更适合本文的需求。

表 2 参数寻优结果

Table 2 Results of parameter optimization

| 迭代次数       | 学习率         | 期望误差           | RMSE         |
|------------|-------------|----------------|--------------|
| 80         | 0.01        | 0.000 1        | 1.356        |
| 80         | 0.001       | 0.000 1        | 1.263        |
| 80         | 0.000 1     | 0.001          | 1.348        |
| 90         | 0.01        | 0.000 1        | 1.333        |
| 90         | 0.001       | 0.000 1        | 1.275        |
| <b>100</b> | <b>0.01</b> | <b>0.000 1</b> | <b>1.232</b> |
| 100        | 0.01        | 0.001          | 1.241        |
| 100        | 0.001       | 0.000 01       | 1.249        |
| 100        | 0.001       | 0.000 1        | 1.236        |
| 110        | 0.01        | 0.000 1        | 1.294        |
| 110        | 0.001       | 0.001          | 1.334        |
| 120        | 0.01        | 0.001          | 1.304        |
| 120        | 0.001       | 0.001          | 1.346        |
| 130        | 0.01        | 0.001          | 1.335        |
| 130        | 0.001       | 0.001          | 1.315        |
| 140        | 0.01        | 0.001          | 1.353        |
| 140        | 0.001       | 0.000 1        | 1.424        |
| 150        | 0.01        | 0.000 1        | 1.410        |
| 150        | 0.001       | 0.001          | 1.371        |

表 3 不同隐含层节点数对应 MAPE 值

Table 3 Different numbers of hidden layer nodes correspond to MAPE values

| 隐含层节点数 | MAPE 值    | 隐含层节点数 | MAPE 值           |
|--------|-----------|--------|------------------|
| 3      | 0.119 56  | 8      | 0.052 096        |
| 4      | 0.092 583 | 9      | 0.072 837        |
| 5      | 0.091 848 | 10     | 0.068 574        |
| 6      | 0.072 453 | 11     | 0.052 448        |
| 7      | 0.085 005 | 12     | <b>0.045 387</b> |

综上, 本章实验使用单隐含层的三层神经网络结构, 将 RSSI 数据作为输入, 接收点坐标作为输出。本文 BP 神经网络的参数设置如表 4 所示。

表 4 参数设置

Table 4 Parameter settings

| 参数名         | 值       |
|-------------|---------|
| 最大迭代次数      | 100     |
| 学习率         | 0.01    |
| 期望误差        | 0.000 1 |
| 输入层到隐含层传递函数 | tansig  |
| 隐含层到输出层传递函数 | purelin |
| 隐含层节点数      | 12      |

### 3.3 结果对比与分析

本文用IGWO优化BP神经网络对灰狼算法(GWO)、遗传算法(GA)、粒子群算法(PSO)优化BP神经网络进行对比预测分析。为了验证本文的IGWO对于预测稳定性的能力,由式(35)可知,RMSE对误差进行平方运算,对较大的误差更敏感,因此,本文根据计算RMSE指标的标准差来主要衡量预测稳定能力。而对于实验数据,本文对原始数据进行直接预测和滤波后预测两种进行对比,取50次预测结果(RMSE)。

图6为实验数据1滤波前后实验结果,通过改进的混沌映射初始化,增加初始种群的均匀性,降低RMSE波动,从图6中可以直观看出IGWO算法对在实验数据1上滤波前后相较于其他算法波动幅度更小,整体误差曲线偏向下方,展现出更小的误差。其他3种算法的曲线从相对贴近到经过滤波处理后相对分离可以得出,BPNN对于处理后的数据有着更好的预测表现,并且经过处理的数据对于不同算法会体现出更大的误差差异。

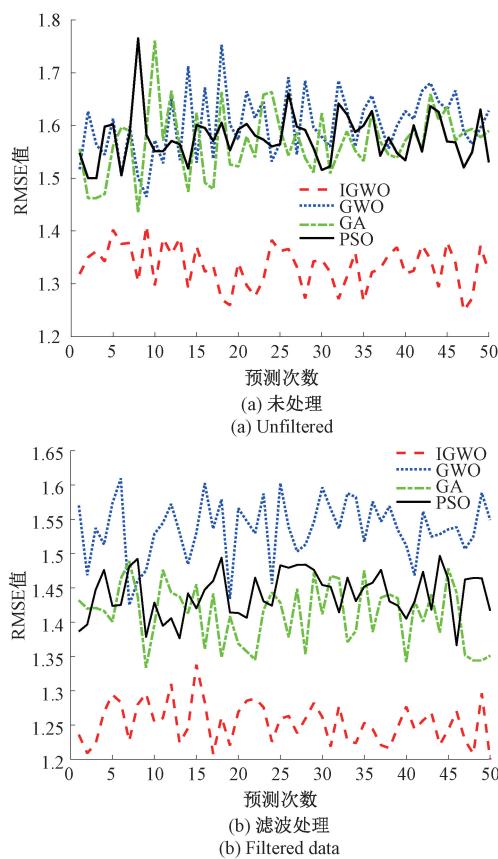


图6 实验数据1预测结果

Fig. 6 Experimental data 1 predicted results

对实验数据1的结果计算标准差,计算结果如表5所示,本文的IGWO算法在对数据进行滤波前和滤波后相比其他3种算法,在RMSE指标上的标准差更小,体现了其预测的稳定性。

表5 实验数据1标准差对比

Table 5 Comparative analysis of experimental data 1 standard deviation

| 算法   | 未处理          | 均值滤波处理       |
|------|--------------|--------------|
| IGWO | <b>0.040</b> | <b>0.030</b> |
| GWO  | 0.060        | 0.043        |
| GA   | 0.062        | 0.044        |
| PSO  | 0.046        | 0.033        |

图7展示了4种算法优化BP神经网络(分别是IGWO、GA、GWO、PSO)在实验数据1上进行的50次随机预测并基于3个误差指标取均值的结果对比,从图中可直观看出IGWO算法在整体的表现最好,通过对滤波前后结果,4种算法的误差下降8%左右,通过计算滤波后的误差,在RMSE指标上的误差分别比GWO、PSO、GA低18.1%、12.7%、11.1%,表明IGWO算法在引入速度更新公式和精英反向学习扩充解的多样性后,提高了模型预测精度,表现出了较明显的优势。

IGWO算法不仅在预测结果上的误差相较于其他3种算法有着更大的优势,在两个实验数据的标准差上也有着更好的表现。另一个方面,通过对比滤波前后的结果可以得出,选择更合适的数据处理方法显著提高了定位精度。IGWO算法不仅在精度上优于其他算法,还表现出更高的稳定性。IGWO引入精英反向学习机制,优化了学习过程,避免了过早收敛;同时,改进的线性收敛因子保持了全局和局部搜索的平衡,进一步提升了精度。综上所述,IGWO算法能够较好的应对RSSI波动带来的不稳定影响,能够有效的提升室内定位精度。

### 4 应用及分析

在实验中,为区分不同环境对定位精度的影响,将实验场景分为两类:噪声(电磁干扰、障碍物等)较少的简单环境(标记场景1,实验数据2)和噪声较多的复杂环境(标记为场景2,实验数据3),实验数据2对应相对较空旷的环境,仅存在少量设备。实验数据3模拟相对障碍物较多的场景,包括更多的电子设备产生的电磁干扰、工位间的遮挡等。实验利用上述的两个数据在两种不同环境下的数据进行定位分析,第一种环境是未处理(带噪声)的实验数据,以此反应室内环境下RSS信号在传播过程中存在多径和障碍物等干扰;第二种数据是经过均值滤波处理后的实验数据。将实验数据2和实验数据3分别去除噪声后录入指纹库,计算测试点位置。仿真平台,选用MATLAB 2022a仿真软件,在AMD R7 7840H处理器的Windows11系统笔记本本下进行仿真实验。

#### 4.1 场景1下实验

在桂林理工大学教五楼六层采集实验数据2进行干扰较少的实验验证,教五楼六层平面结构如图8所示,小教室

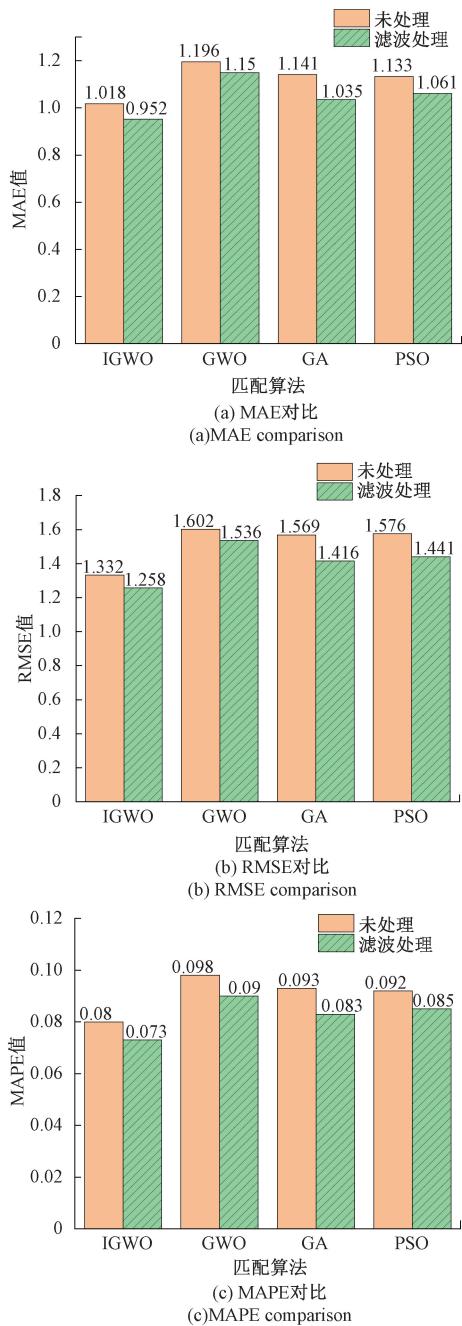


图 7 实验数据 1 随机 50 次预测误差均值

Fig. 7 Experimental data 1 mean prediction error of 50 random times

室内规格约为  $7.5 \text{ m} \times 10 \text{ m}$ , 布置 1 个 AP, 中教室约为  $7.5 \text{ m} \times 12.5 \text{ m}$ , 布置 2 个 AP, 大教室约为  $12.5 \text{ m} \times 17.5 \text{ m}$ , 布置 3 个 AP, 共计 4 个小教室, 2 个中教室, 2 个大教室, 14 个 AP, 每  $2.5 \text{ m}$  取一个点, 每个点采集十次数据, 取参考点(reference point, RP)共计 110 个, 共计约 1100 条训练数据。取测试点(test point, TP)共计 38 个, 共计 380 条测试数据。

场景 1 下对于 BPNN 的超参数调优, 候选参数如表 1 所示, 寻优结果如表 6 所示。

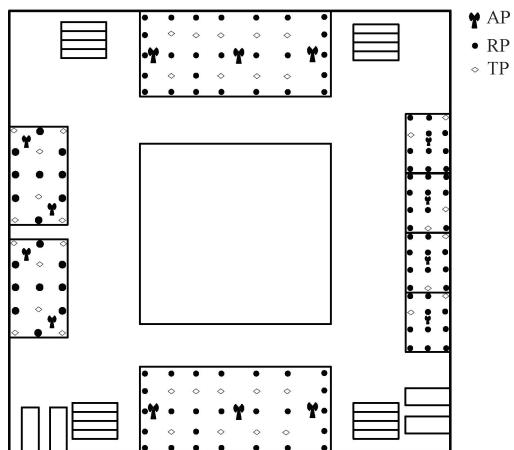


图 8 教五楼六层平面图

Fig. 8 Teach the floor plan of the fifth and sixth floors

表 6 参数寻优结果

Table 6 Results of parameter optimization

| 迭代次数       | 学习率         | 期望误差           | RMSE         |
|------------|-------------|----------------|--------------|
| 80         | 0.01        | 0.000 1        | 1.007        |
| 80         | 0.001       | 0.000 1        | 1.073        |
| 80         | 0.000 1     | 0.001          | 1.033        |
| 90         | 0.01        | 0.000 1        | 0.972        |
| 90         | 0.001       | 0.000 1        | 1.041        |
| <b>100</b> | <b>0.01</b> | <b>0.000 1</b> | <b>0.888</b> |
| 100        | 0.01        | 0.001          | 1.117        |
| 100        | 0.001       | 0.000 01       | 1.219        |
| 100        | 0.001       | 0.000 1        | 0.990        |
| 110        | 0.01        | 0.000 1        | 1.020        |
| 110        | 0.001       | 0.001          | 1.029        |
| 120        | 0.01        | 0.001          | 1.174        |
| 120        | 0.001       | 0.001          | 1.042        |
| 130        | 0.01        | 0.001          | 0.978        |
| 130        | 0.001       | 0.001          | 1.054        |
| 140        | 0.01        | 0.001          | 1.114        |
| 140        | 0.001       | 0.000 1        | 1.121        |
| 150        | 0.01        | 0.000 1        | 1.199        |
| 150        | 0.001       | 0.001          | 1.082        |

根据寻优结果, 本文确定最优的参数组合为最大迭代次数 100, 学习率 0.01, 期望误差 0.000 1。

对于隐含层节点数选取如表 7 所示, 最终选择隐含层节点数为 12。

综上所述, 在实验场景 1 下 BP 神经网络的参数设置如表 8 所示。

在实验数据 2 上取 50 次预测结果(RMSE)如图 9 所示, 从图中可以看出 IGWO 算法在实验数据 2 上相较于其他 3 种算法, IGWO 算法曲线相较于其他算法靠下, 并且其

表7 不同隐含层节点数对应MAPE值

Table 7 Different numbers of hidden layer nodes correspond to MAPE values

| 隐含层节点数 | MAPE 值    | 隐含层节点数 | MAPE 值           |
|--------|-----------|--------|------------------|
| 3      | 0.088 421 | 8      | 0.079 817        |
| 4      | 0.141 454 | 9      | 0.140 771        |
| 5      | 0.115 143 | 10     | 0.102 030        |
| 6      | 0.102 475 | 11     | 0.084 016        |
| 7      | 0.111 173 | 12     | <b>0.061 654</b> |

表8 参数设置

Table 8 Parameter settings

| 参数名         | 值       |
|-------------|---------|
| 最大迭代次数      | 100     |
| 学习率         | 0.01    |
| 期望误差        | 0.000 1 |
| 输入层到隐含层传递函数 | tansig  |
| 隐含层到输出层传递函数 | purelin |
| 隐含层节点数      | 12      |

波动幅度相对较小,滤波后4种算法的误差都有不同程度的下降。

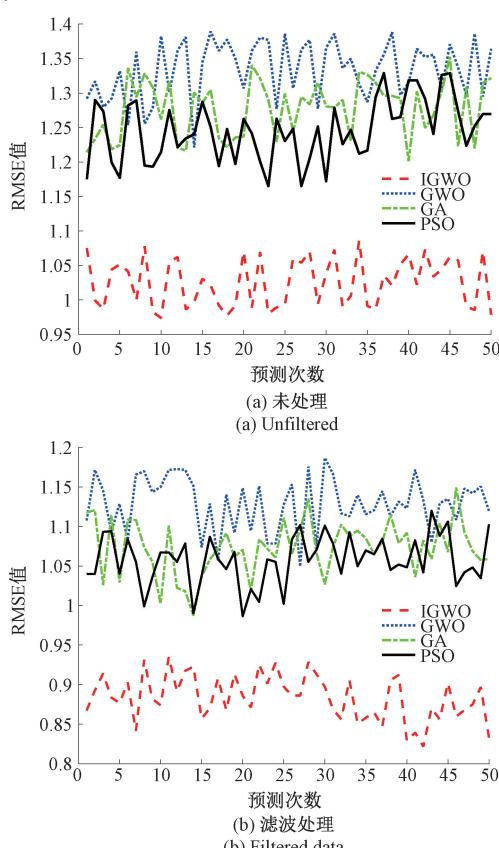


图9 实验数据2预测结果

Fig. 9 Experimental data 2 prediction results

结合实验数据2的RMSE指标标准差,如表9所示,通过精英反向学习和速度更新公式的引入,本文的IGWO对于预测数据在RMSE指标上标准差相较于GWO算法、GA算法、PSO算法有着更小的标准差,经过滤波处理标准差进一步降低,说明经过处理后的数据会降低BPNN的预测波动,提供更加稳定的定位。

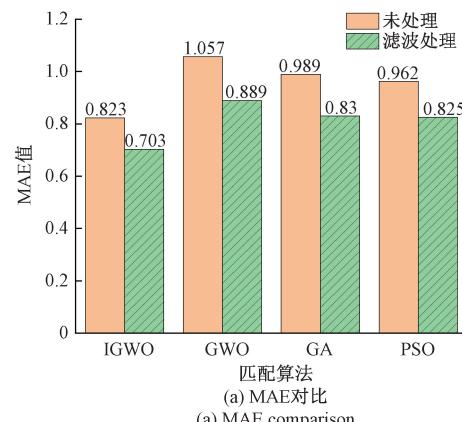
表9 实验数据2标准差对比

Table 9 Comparative analysis of experimental data 2 standard deviation

| 算法   | 未处理          | 均值滤波处理       |
|------|--------------|--------------|
| IGWO | <b>0.036</b> | <b>0.029</b> |
| GWO  | 0.043        | 0.036        |
| GA   | 0.041        | 0.034        |
| PSO  | 0.044        | 0.031        |

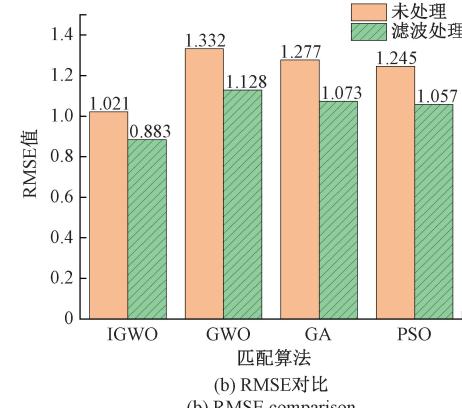
图10展示了4种算法在实验数据2上50次随机预测取均值的对比情况。

从图10可以看出,IGWO算法在3种评价指标上相比其他3种算法有着较明显的优势,通过精英反向学习扩充解的多样性和非线性收敛因子合理的收敛调整提高整体优化稳定性,计算滤波处理后RMSE指标差距,IGWO相比GWO算法、GA算法、PSO算法分别减少了21.7%、17.7%、16.9%,在MAE指标上,IGWO算法相比GWO



(a) MAE对比

(a) MAE comparison



(b) RMSE对比

(b) RMSE comparison

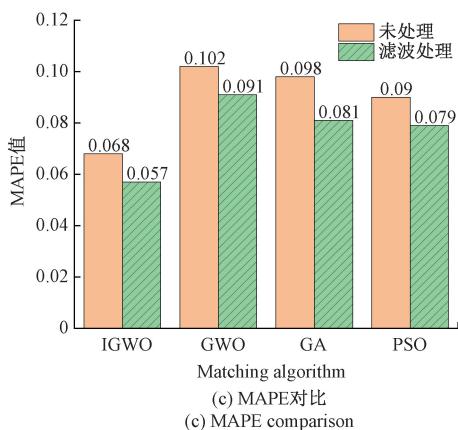


图 10 实验数据 2 随机 50 次预测误差均值

Fig. 10 Experimental data 2 mean prediction error of 50 random times

算法、GA 算法、PSO 算法分别减少了 21.3%、15.7%、14.6%，进一步证明了 IGWO 算法在优化 BPNN 做预测定位时准确性和鲁棒性方面的优势。通过计算滤波前后的误差，IGWO 算法在滤波处理后的数据误差表现比滤波处理前低 13% 左右，因此经过滤波处理的数据可以进一步提升定位的精度。

#### 4.2 场景 2 下实验

在桂林理工大学教五楼 4 楼 410 实验室采集实验数据 3 进行干扰较多的实验验证，其平面结构如图 11 所示，试验区域约为  $8 \text{ m} \times 10 \text{ m}$ ，将 4 个信号发射端设置在 4 个不同位置，共计取参考点 80 个，每个参考点采集十次数据，共计 800 条训练数据，测试点 40 个，每个测试点采集 10 次数据，共计 400 条测试数据。

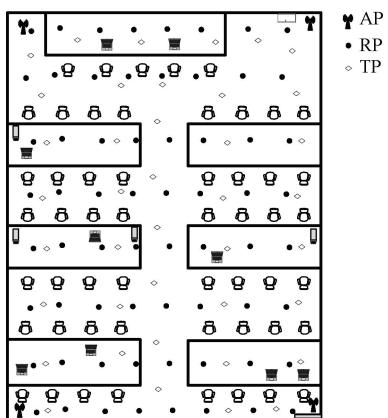


图 11 教五楼 4 楼 410 实验室

Fig. 11 Teach Lab 410 on the fourth floor of the fifth floor

场景 2 下对于 BPNN 的超参数调优，候选参数如表 1 所示，寻优结果如表 10 所示。

根据寻优结果，本文确定最优的参数组合为最大迭代次数 100，学习率 0.01，期望误差 0.000 01。

对于隐含层节点数选取如表 11 所示，选取最佳隐含层

表 10 参数寻优结果

Table 10 Results of parameter optimization

| 迭代次数       | 学习率         | 期望误差           | RMSE         |
|------------|-------------|----------------|--------------|
| 80         | 0.01        | 0.000 01       | 1.013        |
| 80         | 0.001       | 0.000 1        | 1.042        |
| 80         | 0.01        | 0.000 01       | 0.928        |
| 90         | 0.01        | 0.000 1        | 1.076        |
| 90         | 0.001       | <b>0.000 1</b> | <b>1.082</b> |
| <b>100</b> | <b>0.01</b> | 0.000 01       | 0.692        |
| 100        | 0.01        | 0.001          | 0.739        |
| 100        | 0.001       | 0.000 01       | 0.716        |
| 100        | 0.001       | 0.000 1        | 0.709        |
| 110        | 0.01        | 0.000 1        | 0.846        |
| 110        | 0.001       | 0.001          | 0.862        |
| 120        | 0.01        | 0.001          | 0.826        |
| 120        | 0.001       | 0.001          | 0.735        |
| 130        | 0.01        | 0.001          | 0.768        |
| 130        | 0.001       | 0.001          | 0.896        |
| 140        | 0.01        | 0.001          | 0.903        |
| 140        | 0.001       | 0.000 1        | 0.873        |
| 150        | 0.01        | 0.000 1        | 0.921        |
| 150        | 0.001       | 0.001          | 0.768        |

节点数 12。

表 11 不同隐含层节点数对应 MAPE 值

Table 11 Different numbers of hidden layer nodes correspond to MAPE values

| 隐含层节点数 | MAPE 值    | 隐含层节点数 | MAPE 值    |
|--------|-----------|--------|-----------|
| 3      | 0.094 742 | 8      | 0.129 140 |
| 4      | 0.129 760 | 9      | 0.075 718 |
| 5      | 0.099 784 | 10     | 0.105 220 |
| 6      | 0.101 060 | 11     | 0.067 887 |
| 7      | 0.097 631 | 12     | 0.042 401 |

综上所述，在实验场景 2 下 BP 神经网络的参数设置如表 12 所示。

表 12 参数设置

Table 12 Parameter settings

| 参数名         | 值        |
|-------------|----------|
| 最大迭代次数      | 100      |
| 学习率         | 0.01     |
| 期望误差        | 0.000 01 |
| 输入层到隐含层传递函数 | tansig   |
| 隐含层到输出层传递函数 | purelin  |
| 隐含层节点数      | 12       |

图12为实验数据3上取50次预测结果(RMSE),从图中可以看出IGWO算法表现出较明显的优势,与其他3种算法相比,IGWO的误差曲线整体位于较低位置,且波动范围相对较小,显示出更好的稳定性。此外,经过滤波处理后,四种算法的误差均有所降低,IGWO仍保持其相对优势。

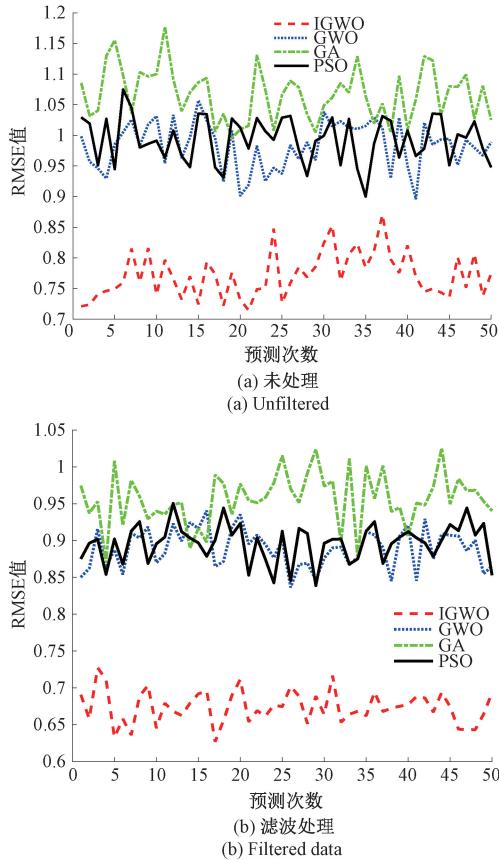


图12 实验数据3预测结果

Fig. 12 Experimental data 3 prediction results

表13为实验数据3在RMSE指标上的标准差对比,在未处理数据的情况下,IGWO算法和PSO算法预测稳定性相对较好,在经过均值滤波处理后,所有算法的标准差都有所下降,滤波处理较明显地提高了所有算法的预测稳定性。

表13 实验数据3标准差对比

Table 13 Comparative analysis of experimental data 3 standard deviation

| 算法   | 未处理          | 均值滤波处理       |
|------|--------------|--------------|
| IGWO | <b>0.037</b> | <b>0.022</b> |
| GWO  | 0.038        | 0.026        |
| GA   | 0.043        | 0.036        |
| PSO  | 0.037        | 0.026        |

图13展示了4种算法在实验数据3上取50次随机预测均值的对比情况。

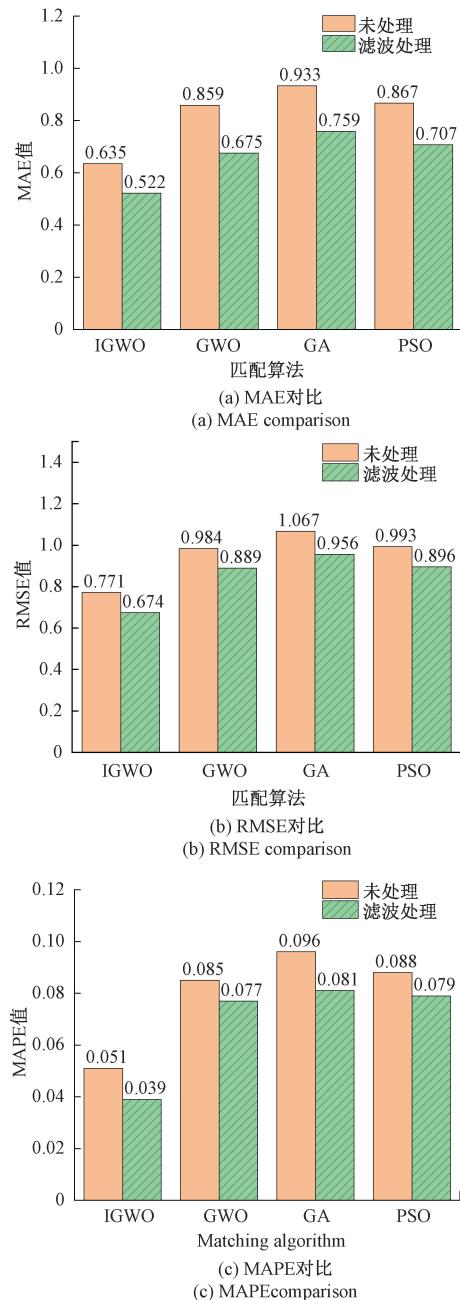


图13 实验数据3随机50次预测误差均值

Fig. 13 Experimental data 3 mean prediction error of 50 random times

从图13可以看出,IGWO算法在3种评价指标上相比其他3种算法有着较明显的优势,通过计算误差得到,IGWO算法在RMSE指标上,相较于其他算法分别低24.1%、29.1%、24.7%,在MAE指标上,相较于其他算法分别低22.6%、31.2%、26.1%,进一步表明其在优化BP神经网络进行室内定位任务中具有较高的精度和稳定性。此外,针对环境中存在的一些噪声经过滤波处理后,所有算

法的位置预测性能得到了提升,在经过计算后得到 IGWO 算法在滤波处理后的数据上误差表现比滤波处理前低 12% 左右,进一步验证了滤波在降低噪声干扰、提高定位精度方面的有效性。通过对对比实验数据 2 和实验数据 3 的实验结果,本文改进的算法在较复杂的环境中相比其他算法有着更为明显的提升,在较复杂的环境中表现出更好的鲁棒性。

## 5 结论

本文提出一种基于改进灰狼算法优化 BP 神经网络的指纹定位方法,在初始化种群时结合混沌映射进行初始化,使得种群位置更均匀的分布在搜索空间。在灰狼算法进行位置更新时通过结合 PSO 算法,并在寻找最优解时通过反向学习寻找反方向的解,使得更具有多样性的解。改进了原始的线性收敛因子,平衡整体的寻优。通过实验,在两个数据集上进行均值滤波前后的定位对比。在实验数据 1 上验证了 IGWO 算法在 RMSE 指标上的误差分别比 GWO、PSO、GA 低 18.1%、12.7%、11.1%,IGWO 算法表现出了较好的预测性能,经过均值滤波处理,IGWO 算法的误差进一步降低了 8% 左右;在实验数据 2 上,本文提出的 IGWO 算法在均方根误差 RMSE 上相比 GWO 算法、GA 算法、PSO 算法在滤波处理后分别减少了 21.7%、17.7%、16.9%,IGWO 算法表现出了较好的预测性能,相较于滤波前,IGWO 算法误差降低了 13% 左右。滤波处理后的数据会进一步提升定位的精度和稳定性。在实验数据 3 上,IGWO 算法在 RMSE 指标上相较于其他算法分别低 24.1%、29.1%、24.7%。实验结果表明,IGWO 算法通过引入 PSO 速度更新、反向精英学习等优化策略,显著提升了滤波后的定位精度,相较于简单的环境,在更复杂的环境相较于其他算法有着更突出的表现,具有更好的定位效果。

## 参考文献

- [1] 秦波,唐秋楠,王美琪. 基于位置隐私保护研究综述[J]. 华东师范大学学报(自然科学版),2015(5): 14-27.
- [2] QIN B, TANG Q N, WANG M Q. Review on location privacy protection research[J]. Journal of East China Normal University (Natural Science Edition), 2015(5): 14-27.
- [3] 陈锐志,王磊,李德仁,等. 导航与遥感技术融合综述[J]. 测绘学报,2019,48(12):1507-1522.
- [4] CHEN R ZH, WANG L, LI D R, et al. A survey on the fusion of the navigation and the remote sensing techniques [J]. Acta Geodaetica et Cartographica Sinica, 2019, 48(12): 1507-1522.
- [5] PAU G, ARENA F, GEBREMARIAM Y E, et al. Bluetooth 5.1: An analysis of direction finding capability for high-precision location services [J]. Sensors, 2021, 21(11): 3589.
- [6] ALARIFI A, AL-SALMAN A M, ALSALEH M, et al. Ultra wideband indoor positioning technologies: Analysis and recent advances [J]. Sensors, 2016, 16(5): 707.
- [7] MOTRONI A, BUFFI A, NEPA P. A survey on indoor vehicle localization through RFID technology[J]. IEEE Access, 2021, 9: 17921-17942.
- [8] 周家鹏,汪云甲,李昕,等. 地磁室内定位技术研究[J]. 测绘通报,2019(1):18-22.
- [9] ZHOU J P, WANG Y J, LI X, et al. Research on geomagnetic indoor positioning technology[J]. Bulletin of Surveying and Mapping, 2019(1): 18-22.
- [10] DANG X, TANG X, HAO Z, et al. A device-free indoor localization method using CSI with Wi-Fi signals [J]. Sensors, 2019, 19(14): 3233.
- [11] LONG K, NSALO KONG D F, ZHANG K, et al. A CSI-based indoor positioning system using single UWB ranging correction[J]. Sensors, 2021, 21(19): 6447.
- [12] BESJEDICA T, FERTALJ K, LIPOVAC V, et al. Evolution of hybrid LiFi-WiFi networks: A survey[J]. Sensors, 2023, 23(9): 4252.
- [13] LIU W, ZHANG Y, DENG Z, et al. Low-cost indoor wireless fingerprint location database construction methods: A review [J]. IEEE Access, 2023, 11: 37535-37545.
- [14] 宋跃才,林海涛,卞媛,等. 基于移动信标的水下无线传感器网络定位算法[J]. 电子测量技术,2023,46(5): 44-49.
- [15] SONG Y C, LIN H T, BIAN Y, et al. Localization algorithm for underwater wireless sensor networks based on mobile beacons[J]. Electronic Measurement Technology, 2023, 46(5): 44-49.
- [16] 倪云峰,石小红. RSSI 的室内人员卡尔曼滤波定位算法[J]. 西安科技大学学报,2020,40(1):167-172.
- [17] NI Y F, SHI X H. Indoor staff kalman filter location algorithm based on RSSI [J]. Journal of Xi'an University of Science and Technology, 2020, 40(1): 167-172.
- [18] 姚军,甄梓越,马宇静. 基于 BP 神经网络的 RSSI 测距优化算法[J]. 电波科学学报,2022,37(4):663-669.
- [19] YAO J, ZHEN Z Y, MA Y J. RSSI ranging optimization algorithm based on BP neural network [J]. Chinese Journal of Radio Science, 2022, 37(4):663-669.
- [20] 朱清山,王伟. 基于 SAGA-BP 神经网络室内定位算法研究[J]. 电子测量技术,2021,44(9):100-104.
- [21] ZHU Q SH, WANG W. Research on indoor positioning algorithm based on SAGA-BP neural

- network [J]. Electronic Measurement Technique, 2021, 44(9): 100-104.
- [15] 余振宝,卢小平,陶晓晓,等.一种GA-BP神经网络模型的RSSI测距算法[J].导航定位学报,2020,8(2): 63-68.  
YU ZH B, LU X P, TAO X X, et al. RSSI ranging algorithm based on GA-BP neural network model[J]. Journal of Navigation and Positioning, 2020, 8 (2): 63-68.
- [16] 林贻若,余科根,朱飞洋,等.一种基于GWO-BP神经网络的RSSI测距算法[J].测绘学报,2024,53(8): 1564-1573.  
LIN Y R, YU K G, ZHU F Y, et al. A RSSI ranging algorithm based on GWO-BP neural network[J]. Acta Geodaetica et Cartographica Sinica, 2024, 53 (8): 1564-1573.
- [17] GUIDARA A, FERSI G, JEMAA M B, et al. A new deep learning-based distance and position estimation model for range-based indoor localization systems[J]. Ad Hoc Networks, 2021, 114: 102445.
- [18] MIRJALILI S, MIRJALILI S M, LEWIS A. Grey wolf optimizer[J]. Advances in Engineering Software, 2014, 69: 46-61.
- [19] 陈嘉兴,刘扬,刘晓茜,等. Logistic混沌映射与差分进化改进人工蜂群优化水下定位[J].工程科学与术, 2025, 57(1): 57-67.  
CHEN J X, LIU Y, LIU X Q, et al. Improved artificial bee colony optimization underwater localization algorithm by logistic chaos mapping and differential evolution [J]. Advanced Engineering Sciences, 2025, 57(1): 57-67.
- [20] ZHAO S, DING Y, LIU X, et al. HKAM-MKM: A hybrid kernel alignment maximization-based multiple kernel model for identifying DNA-binding proteins[J]. Computers in Biology and Medicine, 2022, 145: 105395.

### 作者简介

刘伟,副教授,主要研究方向为室内定位、无线传感器网络。

E-mail:lwsdjnyz@163.com

李艾龙,硕士研究生,主要研究方向为室内定位、深度学习。

E-mail:1553916593@qq.com

李卓(通信作者),实验师,主要研究方向为无线传感网絡、室内定位。

E-mail:234162414@qq.com

王智豪,硕士研究生,主要研究方向为室内定位、深度学习。

E-mail:240910782@qq.com