

doi: 10.3969/j.issn.0490-6756.2017.03.011

基于反向传播神经网络的 WSN 节点定位方法研究

周春良, 王 明, 屈卫清, 陆正球

(宁波大红鹰学院信息工程学院, 宁波 315175)

摘要: 针对无线传感器网络存在的定位精度问题, 基于反向传播神经网络提出一种新的节点定位方法. 该方法首先结合时间差测距和信号强度给出了节点定位计算公式, 同时结合反向传播神经网络对上述参数进行快速求解. 最后结合 NS2 和 MATLAB 进行仿真实验, 深入研究了影响定位方法的关键因素. 通过对比其他定位算法, 本方法具有较好的适应性, 能够有效降低定位误差.

关键词: 无线传感器网络; 定位; 反向传播神经网络; 误差

中图分类号: TP393 **文献标识码:** A **文章编号:** 0490-6756(2017)03-0493-06

The study of WSN node localization method based on back propagation neural network

ZHOU Chun-Liang, WANG Ming, QU Wei-Qing, LU Zheng-Qiu

(College of Information Engineering, Ningbo Dahongying University, Ningbo 315175, China)

Abstract: In order to cut down the localization accuracy problem of wireless sensor network (WSN), a novel node localization method is proposed with back propagation neural network (BPNN). At first, the calculation of node localization is presented by ranging interval and signal strength, and the parameters are rapid solving base on BPNN. Finally, a simulation experiment is conducted to study the influence key factor with NS2 and MATLAB. The results show that, compared other localization algorithm, this method has good suitability, and it could effectively reduce the localization error.

Keywords: Wireless sensor network; Localization; Back propagation neural network; Error

1 引言

无线传感器网络(Wireless Sensor Network, WSN)是一种由大量廉价、具有通信和感知能力的传感器节点通过无线通信方式形成的多跳自组织网络系统, 被广泛用于环境监测、跟踪、军事国防和医学等各个领域^[1-6]. 而无线传感器网络领域的关键核心支撑技术之一就是节点定位^[7, 8]. 目前, 定位算法分为基于测距的定位算法与非测距定位算法^[9-12]. 基于测距定位算法必须获得节点间的具体

距离, 其优势在于定位精度较高, 但实现较为复杂, 成本较高. 而非测距定位算法不需要知道节点间的距离与角度信息, 实现过程比较简单, 应用较为广泛. 在非测距的 WSN 定位方法中, WSN 节点通常分为锚点和未知节点, 只需利用锚节点和未知节点间的关系来进行节点定位, 通常采用抛撒方式部署节点, 其地理位置分布具有随意性, 因此很难在初始化阶段为每个节点预设位置信息. 目前使用最广泛的定位服务是 GPS, 但 GPS 存在能耗等问题, 且有时在信号较差的地方也无法实现, 传感器网络在

收稿日期: 2016-12-20

基金项目: 国家自然科学基金(115050099); 浙江省公益技术应用研究项目(2016C31129, 2015C33236); 浙江省教育厅科研项目(Y201432666); 浙江省本科院校中青年学科带头人学术攀登项目(pd2013443)和宁波市自然科学基金项目(2012A610071)

作者简介: 周春良(1982-), 男, 硕士, 讲师, 研究方向为无线传感器网络. E-mail: yang_gl1978@163.com

部署之后需先采用一定的算法实现自身定位,找到节点初始位置信息,然后才能进入监测状态.为了获得初始的定位参照,WSN 需要部署一定比例的锚点,锚点可通过 GPS 系统或人工预设等方式实现自定位,并回送初始定位参照信息,对于未知的节点就利用测距装置或网络连通信息获取本节点与多个锚点的相对位置信息,然后使用三边法或最大似然估计法计算未知节点的位置坐标.但当未知节点的邻居锚节点的个数小于 3 时,则未知节点不能对自身进行准确定位.

目前,国内外学者关于传感器网络的定位方法提出了大量算法和模型.刘伟等^[13]针对无线传感器网络节点三维定位中可能会出现翻转模糊问题,提出并证明了节点三维定位的翻转模糊检测可以等价于判断是否存在一个平面和所有参考节点的测距误差球都相交的问题,并根据公切面法和正交投影法求解了 EIP 问题,实验结果表明大大降低计算复杂度.王福豹等^[14]详细介绍了无线传感器网络自身定位系统和算法的性能评价标准和分类方法,对于该领域内具有代表性的算法与系统的原理进行了详细的描述.文献^[15]提出了一种基于移动锚点的定位方法,即通过减少锚节点的数量,并基于三个移动锚节点的 Range-free 无线传感器的网络节点定位算法来降低网络开销,提高节点定位精度.姬靖等^[16]利用测距误差提出了一种定位算法,根据 WSN 中误差节点的邻居节点选择相对偏差最小的节点作为圆心,以误差节点的测距值作为半径来选择迭代的精度位置,该算法有效降低了多边定位模型中的节点位置误差.文献^[17]提出了一种构建位置校正矢量的算法,但该方法在测量成本以及能量消耗都超过了正常水平.

在上述工作的基础上,本文提出了一种基于反向传播神经(Back Propagation Neural Network, BPNN)^[18-20]的 WSN 节点定位算法.该算法通过分析未知节点与少量锚点之间的测量距离,建立以未知节点的位置为参数的定位模型,用 BPNN 节点定位算法对参数进行求解.同时通过仿真实验来验证该算法的有效性.

2 计算方法

假设在 WSN 网络中有 n 个节点 $\{1, 2, \dots, n\}$, 这些节点随机抛撒在一个大小为 S 的二维空间中,所有节点具有相同的计算能力和通信半径 r ,在 n 个节点中,有 k 个节点 $(1, 2, \dots, k)$ 为锚点,它们通过

GPS 预先获取自身的位置信息,其余 $n - k$ 个节点 $(k + 1, k + 2, \dots, n)$ 为未知节点,邻居节点间相互通信,非邻居节点间通过多跳转发进行通信.假定节点 i 到锚节点 j 的最短路径跳数为 $g(i, j)$,则节点到各锚点的最小通信路径可以组成跳数测量向量:

$$G_i = [g(s_i, s_1), g(s_i, s_2), \dots, g(s_i, s_k)] \quad (1)$$

令锚节点 A 节点的位置为 (x_1, y_1) , 锚节点 B 的位置为 (x_2, y_2) , d_{AD} 为节点 A 到节点 D 的距离, d_{BD} 为节点 B 到节点 D 的距离,那么根据式(2)可得节点 $D(x, y)$ 的位置信息:

$$\begin{cases} d_{AD} = 2(x - x_1) + 2(y - y_1) \\ d_{BD} = 2(x - x_2) + 2(y - y_2) \end{cases} \quad (2)$$

但距离的测量是求解坐标的关键,本文将采用基于到达时间差的测距,在抛撒节点的同时发送两种速度不同的传播信号,接收信号的节点根据这两种信号到达的时间差和他们的传播速度来计算节点间的距离.假如两种信号的传播速度分别为 v_1 、 v_2 , 到达的时间分别为 t_1 、 t_2 , 则两节点间的距离 d 为

$$d = \frac{(t_1 - t_2)v_1 v_2}{|v_2 - v_1|} (1 - p(d)) \quad (3)$$

其中, $p(d)$ 用来刻画信号衰减,因为信号在传播过程中会产生衰减.根据绝对自由空间衰减模型,则普通信号的衰减模型可定义为

$$p(d) = p(d_0) - 10\alpha \ln \frac{d}{d_0} \quad (4)$$

其中, α 为路径衰减因子; $p(d_0)$ 为参考距离 d_0 处的信号强度; $p(d)$ 为经过传播距离 d 后的信号强度.令 $b = [p(d_1), p(d_2), \dots, p(d_n)]$; $X = [p(d_0), \alpha, x_\zeta]$, d_n 为第 i 点到无线接入节点之间的距离, x_ζ 为置信系数.利用一组观测量 b 估计信号衰减中的参数 X , 则

$$\begin{bmatrix} 1 & -10 \cdot \alpha \cdot \log(d_1/d_0) & 1 \\ 1 & -10 \cdot \alpha \cdot \log(d_2/d_0) & 1 \\ \vdots & \vdots & \vdots \\ 1 & -10 \cdot \alpha \cdot \log(d_n/d_0) & 1 \end{bmatrix} X = b \quad (5)$$

为了快速获得节点的位置信息,这里结合 BPNN 来对上述方法进行求解. BPNN 是一种无反馈前向网络,具有较好的自学习功能,能够实现快速收敛.

3 基于反向传播神经的 WSN 节点定位

3.1 反向传播神经

BPNN 由输入层、隐含层、和输出层构成,是

典型的多层网络,层间多为全互连方式,同层单元之间不存在相互连接.学习过程包括正向传播和反向传播,在正向传播过程中,输入的信息经隐含层逐级处理,最终传出输出层,下一层神经元的状态会受到上一层神经元状态的影响,如果在输出层不能获得期望输出结果则转向反向传播,将误差信号按照原来的通路返回,把网络学习时出现的误差,归结为连接层中各节点间连接权及阈值的异常,通过把输出层节点的误差逐层向输入层逆向传播分给各连接节点,进而计算出各连接节点的误差值,并依据各连接权进行相应的调整.

具体的训练过程如下所述.

(1) 初始化输入层节点 α_r 到隐含层节点 β_r 的连接权 ω_{ir} , 隐含层节点 β_r 到输出层节点 θ_j 的连接权 ν_{rj} , 隐含层节点阈值 δ_r , 输出层节点阈值 σ_j .

(2) 假设有 p 个样本对, 每一个样本对 $(A^{(k)}, C^{(k)}) (k=1, 2, \dots, p)$ 进行下列操作: 将 $A^{(k)}$ 的值 $\alpha_i^{(k)}$ 输入到输入层节点中, 依据输入层节点激活值 α_i 依次正向计算隐含层节点阈值 δ_r 和输出层节点阈值 σ_j .

$$\beta_r = g\left(\sum_{i=1}^m \omega_{ir} \alpha_i + \delta_r\right) \quad (r=1, 2, \dots, u) \quad (6)$$

$$\theta_j = g\left(\sum_{r=1}^u \nu_{rj} \beta_r + \sigma_j\right) \quad (j=1, 2, \dots, n) \quad (7)$$

其中, u 和 n 分别为隐含层和输出层的节点个数; $g(\cdot)$ 为神经元输入与输出之间的转换函数, 该函数一般为可导函数, 在标准的 BP 算法中, 单元的实际输出是通过一个非线性的、可微的 S 型函数得到.

(3) 重复步骤 (2), 直到误差 $\Delta E = \frac{1}{2} \sum_{k=1}^p \sum_{j=1}^n (\theta_j^{(k)} - \theta_j)^2$ 变得足够小.

3.2 基于反向传播的节点定位

基于 BPNN 的 WSN 节点定位算法主要分为两个阶段, 即训练阶段和迭代定位阶段. 在网络训练阶段, 需要获取大量的实验数据计算真实目标距离值, 将真实目标距离值带入节点定位算法公式中计算得到真实的位置信息, 然后以实际测量值、误差方差作为 BPNN 的输入, 将计算得到的真实位置信息作为训练目标进行网络训练, 训练结束后, 就能得到表示测量值、误差方差和位置信息之间的关系的网络. 在迭代定位阶段, 以 BPNN 训练阶段的结果来计算未知节点的位置信息.

同时, 网络的权值是通过使网络输出与样本输

出之间的误差平方和达到期望值而不断调整网络的权值训练出来的, 而网络结构包括: 输入/输出神经元的个数、隐含层个数、隐含层中神经元数目以及每层传递函数的确定. 具体算法步骤如下所述.

Step 1 部署网络, 各锚节点全网广播包含节点 ID、位置的数据包, 未知节点接收到后保存并转发一次, 这样每个未知节点就能获得它 1 跳和 2 跳内的锚节点位置, 并依据式(2)和式(3)计算出未知节点 1 跳内邻居节点的距离.

Step 2 利用除趋于边界外的网格定点 $j (j=1, 2, \dots, n)$ 到锚节点的 $i (i=1, 2, \dots, n)$ 的距离向量 d_{ij} , 与 j 的坐标 (x_j, y_k) 构造训练集, 用于定位算法的学习阶段. 汇聚节点保存 BP 网络输出的绝对误差值和 E 的最小权值和阈值. 隐含层单元 j 的输出.

$$H_r = g\left(\sum_{i=1}^S \omega_{ir} x_i - \delta_r\right), \quad r=1, 2, \dots, p \quad (8)$$

输出层单元 k 的实际输出:

$$\hat{y}_k = \sum_{i=1}^p H_{ij} \nu_{rj} - b_k, \quad k=1, 2, \dots, t \quad (9)$$

其中, $g(x) = 1/(1+e^{-x})$ ω_{ir} 为输入层代隐含层的权值; x_i 为某一模式下输入层单元的输出; δ_r 为隐含层单元的阈值; ν_{rj} 为隐含层到输出层的阈值; σ_j 为输出层的阈值.

Step 3 对样本采样, 采样结束后, 对所有样本求平均值, 得到预估初始位置 (x_i, y_i) .

$$(sx_i, sy_i) = \left(\frac{1}{n} \sum_{i=1}^n x_i(s_i), \frac{1}{n} \sum_{i=1}^n y_i(s_i)\right) \quad (10)$$

Step 4 利用 BPNN 优化初始位置和, 在该阶段, 每个未知节点将自己的位置信息传递给相邻的邻居节点, 通过相邻节点的位置信息和邻居节点间的测量距离, 利用 BPNN 算法对初始位置进行优化. 目标函数为

$$\min E(\omega, \nu, \delta, \sigma) \sum_{r=0}^N \sum_{i=1}^t |y_{ri} - \hat{y}_{ri}| \quad (11)$$

Step 5 将 Step2 中的所有采样点作为初始种群, 计算每个采样点的适应度值找出最优个体, 适应度函数如下

$$\begin{cases} f(i, j) = [\sqrt{(x_i(s_i) - sx_j)^2 + (y_i(s_i) - sy_j)^2} \\ - R_{ij}]^{-2} \\ f(i, j) = [\sqrt{(x_i(s_i) - Tx_j)^2 + (y_i(s_i) - Ty_j)^2} \\ - R_{ij}]^{-2} \\ Fi = \sum \tau f(i, j) \end{cases} \quad (12)$$

Step 6 令输出函数如下

$$y = \frac{1}{1 + e^{-F_i}} \tag{13}$$

其中, y_j 为当前层单元 j 的实际输出,同时设置收敛标准函数:

$$RMS = \frac{1}{P} \sum_{k=1}^P \left(\frac{1}{n} \sum_{j=1}^n \overline{(\theta_j^{(K)} - \theta_j^{(K)})^2} \right) \tag{14}$$

其中, n 为输出单元的个数; $\theta_j^{(K)}$ 为训练模式为 k 时单元 j 的理想输出; $\theta_j^{(K)}$ 为训练模式为 k 时单元 j 的实际输出. 计算输出层理想输出 $\theta_j^{(K)}$ 与实际输出值 θ_j 的误差 $\Delta\theta_j$, $\Delta\theta_j = \theta_j(1 - \theta_j) \cdot (\theta_j^{(K)} - \theta_j)$; 向隐含层节点反向分配误差 Δb_r , $\Delta b_r = b_r \cdot (1 -$

$b_r) \cdot (\sum_{j=1}^n \nu_{rj} \cdot \Delta\theta_j)$; 将隐含层节点和输出层节点间的连接权 ν_{rj} 和输出层节点阈值 σ_j 进行如下调整: $\nu_{rj} = \nu_{rj} + \alpha \cdot \beta_r$, $\sigma_j = \sigma_j + \alpha \cdot \Delta\sigma_j$, $\alpha \in \text{rand}[0, 1]$; 将输入层节点和隐含层节点间的连接权 ω_{ir} 和隐含层节点阈值 δ_r 进行如下调整: $\omega_{ir} = \omega_{ir} + \alpha \cdot \alpha_i \cdot \Delta b_r$, $\delta_r = \delta_r + \alpha \cdot \Delta b_r$.

Step 7 判断训练网络是否满足要求,如满足则跳转到 Step 8,否则返回 Step1.

Step 8 输出当前最优解,算法结束.

4 仿真实验

为了验证所提算法的有效性,这里采用结合 NS2 和 MATLAB 进行仿真实验,并与其它算法进行比较. 在 $S=100 \times 100$ 的区域随机抛撒 $n=100$ 个传感器节点,其中 $k=25$ 个锚节点,75 个未知节点,抛撒之后的节点均不发生移动,以此构建一个传感器网络. 假设通信距离 $r=10$,最大迭代次数为 $T=300$,每次采取 20 个有效样本,允许采样的最大次数为 $p=300$,设未知节点具有测量自身到邻居节点的能力,且测距误差设置为 0.01. 图 1 给出了 GA、POS、BPNN 算法的节点定位误差与算法迭代次数之间的关系曲线. 由图可知,在前 10 次迭代次数以内,节点定位误差急剧降低,说明这种优化算法具有良好的收敛性,但随着迭代次数的增加,定位误差的值减小的越来越少,当对定位误差精度要求不高时,可以适当的减少算法的迭代次数. 此外,从图中得出,在 GA、POS、BPNN 三种算法中,BPNN 的定位误差较其它两种算法更为优化,由此验证了该算法在节点定位算法中的定位精准性.

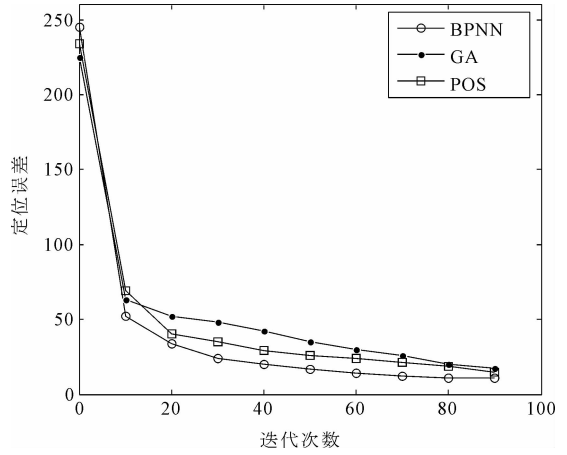


图 1 迭代次数对定位误差的影响
Fig. 1 The influence of iteration times on the location error

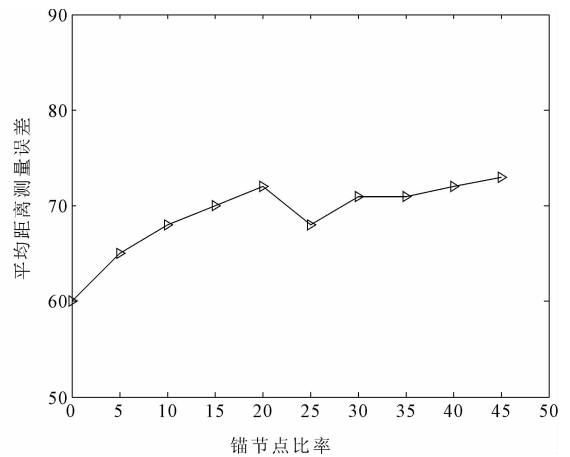


图 2 锚节点比率对平均距离测量误差的影响
Fig. 2 The influence of anchor node ratio on average distance measurement error

其次,图 2 给出了不同锚节点比率与平均距离测量误差的关系曲线. 由于平均距离测量对寻找最优路径和节点的定位至关重要,由图可得,锚节点比率对平均距离测量误差的影响并不是很大,在考虑平均测量误差的时候可以忽略锚节点比率的影响. 锚点可以获取自身的位置信息,以此来测量未知节点的位置信息,因此锚节点的比率是节点定位中的一个重要性能指标,适中的锚点比例有利于节点的定位.

同时,图 3 为不同传感器节点数对平均定位误差的影响,在 100×100 的区域内,假定通信距离为 10 m,这里不断的改变传感器的节点个数,使其分别为 10/20/30/40/50/60/70/80/90/100,观察平均定位误差的变化,结果如图 3 所示. 从图中可知,传感器的节点数密度越大,平均定位误差就越小. 但随着节点个数的不断增加,节点距离测量会受到

影响,使误差变大,不利于定位,因而应该多方面考虑选择适中的节点个数.

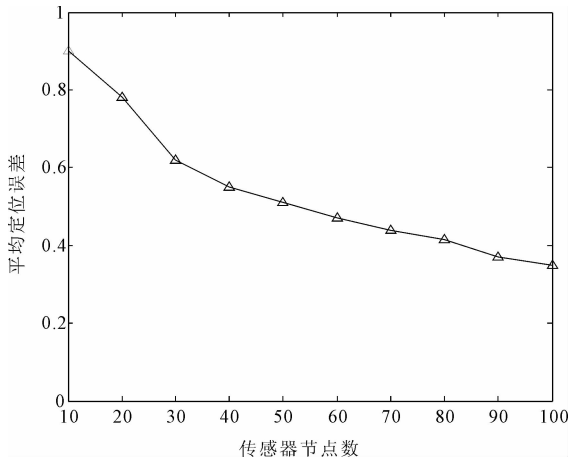


图 3 传感器节点数与平均定位误差的关系

Fig. 3 The relationship between the number of sensor nodes and the average location error

图 4 为锚节点通信半径对定位误差的影响,前提条件是在锚节点数量一定时,从图中可以看出,随着锚节点的通信半径的变大,定位误差的值也在减小,由于锚节点的位置信息已知,要提高定位误差,可以适当地减少锚节点的个数以提高锚节点间的通信半径,进而降低定位误差.但如果锚节点通信距离过大,也有可能影响测量误差,进而影响定位误差,因此锚节点的个数和锚节点的通信半径最好保持在一定范围内以保证最小化定位误差.

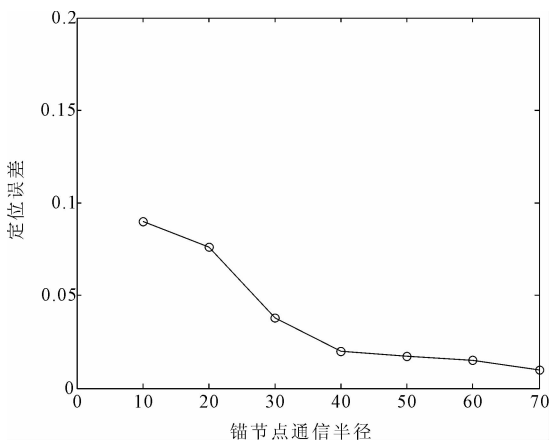


图 4 锚节点通信半径对定位误差的影响

Fig. 4 The influence of anchor node communication radius on location error

最后,图 5 比较了三种算法的锚点比例与定位误差的关系.从图中可以看出,随着锚节点个数的增加,BPNN 算法的定位误差在不断较少,在锚节点比例为 0.1 的时候,定位误差急剧下降,由此验

证了该算法的收敛性,当锚节点比例超过 0.8 之后,定位误差基本不变,由此可以看出,该算法只有在一定的锚节点比例范围内才能降低定位误差,此外也能从图中看出,BPNN 算法在网络节点定位中具有较好的优势,能很大程度上提高定位精度.

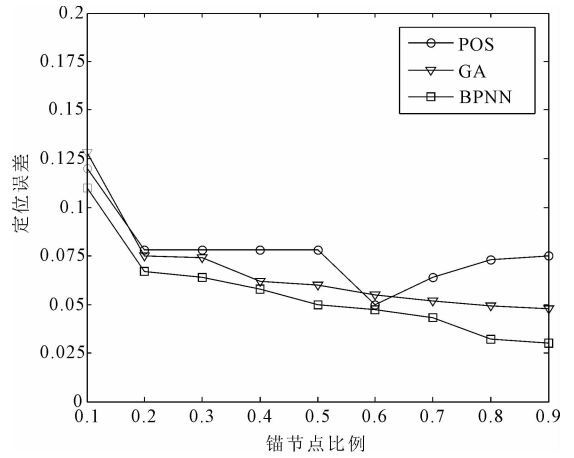


图 5 三种算法定位误差的比较

Fig. 5 The comparison of location error between three algorithms

5 结 论

为了有效解决无线传感器网络中存在的定位精度问题,本文利用反向传播神经网络提出一种新的节点定位方法.该方法首先结合时间差测距和信号强度给出了节点定位计算公式,同时结合反向传播神经网络对上述参数进行快速求解.最后结合 NS2 和 MATLAB 进行仿真实验,深入研究了影响定位方法的关键因素.通过对比其他定位算法,本方法具有较好的适应性,能够有效降低定位误差.在后续研究中,可以结合三维定位的翻转模糊检测、压缩感知多目标定位方法等建立一套完善的无线传感器网络定位模型.

参考文献:

[1] 肖竹, 陈杰, 王东, 等. 严重遮挡非视距环境下的三维定位方法[J]. 通信学报, 2015, 36(8): 68.

[2] 马祖长, 孙怡宁, 梅涛. 无线传感器网络综述[J]. 通信学报, 2004, 25(4): 114.

[3] 赵小敏, 张海洋, 金言, 等[J]. 信标节点漂移情况下的无线传感器网络节点定位机制[J]. 通信学报, 2015, 36(2): 38.

[4] 张睿敏, 陈钟, 李晓斌. 多目标量化变分滤波贝叶斯 WSN 跟踪定位算法[J]. 四川大学学报: 自然科学版, 2015, 52(6): 1237.

- [5] 刘菁原, 刘云. WSN 中可容定位误差的地理位置路由算法优化研究[J]. 四川大学学报: 自然科学版, 2015, 52(2): 338.
- [6] Tseng P H, Ding Z, Feng K T. Cooperative self-navigation in a mixed LOS and NLOS environment [J]. IEEE Trans Mobile Comput, 2014, 13(2): 350.
- [7] Chang C T, Chang C Y, Lin C Y. Anchor-guiding mechanism for beacon-assisted localization in wireless sensor networks[J]. IEEE Sensors J, 2012, 12(5): 1098.
- [8] Kuo S P, Kuo H J, Tseng Y C. The beacon movement detection problem in wireless sensor networks for localization applications[J]. IEEE Trans Mobile Comput, 2009, 8(10): 1326.
- [9] 熊竹林, 刘策伦, 安建平, 等. 低复杂度高动态低信噪比环境下的 GPS 信号捕获算法[J]. 通信学报, 2015, 36(3): 175.
- [10] 姚英彪, 姜男澜. 无线传感器网络定位的分布式求精算法[J]. 通信学报, 2015, 36(1): 179.
- [11] Chen W P, Meng X F. A cooperative localization scheme for zigbee-based wireless sensor network [C]//Proceeding of the 14th IEEE International Conference on Networks. Singapore: IEEE, 2006.
- [12] He T, Huang C D, Blum B M. Range-free localization schemes for large scale sensor networks[C] // Proceeding of the 9th Annual International Conference on Mobile Computing and Networking. San Diego, CA, USA: ACM, 2003.
- [13] 刘伟, 董恩清, 宋洋. 无线传感器网络节点三维定位的翻转模糊检测[J]. 电子学报, 2016, 44(2): 374.
- [14] 王福豹, 史龙, 任丰原. 无线传感器网络中的自身定位系统和算法[J]. 软件学报, 2005, 16(5): 857.
- [15] 史庭俊, 桑霞, 徐力杰, 等. WSN 中一种基于移动锚节点的节点定位算法[J]. 软件学报, 2009, 20(Suppl.): 278.
- [16] 姬靖, 刘桂雄, 郁文生. 传感网中的四锚点节点定位实解个数分析[J]. 软件学报, 2014, 25(11): 2627.
- [17] 王驭风, 王岩. 基于矢量的无线传感器网络节点定位综合算法[J]. 通信学报, 2008, 29(11): 227.
- [18] 韩红桂, 乔俊飞, 薄迎春. 基于信息强度的 RBF 神经网络结构设计研究[J]. 自动化学报, 2012, 38: 1083.
- [19] 邓万宇, 郑庆华, 陈琳, 等. 神经网络极速学习方法研究[J]. 计算机学报, 2010, 33(2): 279.
- [20] 杨雅辉, 黄海珍, 沈晴霓, 等. 基于增量式 GH-SOM 神经网络模型的入侵检测研究[J]. 计算机学报, 2014, 37(5): 1216.