基于 Isolation Forest 改进的数据异常检测方法

徐 东 王岩俊 孟宇龙 张子迎
(哈尔滨工程大学计算机科学与技术学院 哈尔滨 150001)

摘 要 针对现有的基于隔离森林(Isolation Forest)的数据异常检测算法检测精度低、执行效率差和泛化能力弱等问题,提出一种改进的数据异常检测方法 SA-iForest。该方法基于模拟退火算法选择精度高和有差异性的隔离树来优化森林,同时去除冗余的隔离树,改进了隔离森林的森林构建。采用标准仿真数据集对所提方法进行验证,结果表明该方法与传统 Isolation Forest 和 LOF 方法相比,在准确率、执行效率和稳定性方面均有显著提高。

关键词 隔离森林,异常检测,SA-iForest,模拟退火
中图法分类号 TP306 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2018.10.029

Improved Data Anomaly Detection Method Based on Isolation Forest

XU Dong WANG Yan-jun MENG Yu-long ZHANG Zi-ying
(College of Computer Science and Technology, Harbin Engineering University, Harbin 150001, China)

Abstract An improved data anomaly detection method namely SA-iForest was proposed to solve the problem of low accuracy, poor execution efficiency and generalization ability of exsiting anomaly data detection algorithm based on isolated forest. The isolation tree with high precision and differences is selected to optimize the forest based on simulated annealing algorithm. At the same time, the redundant isolated trees are removed, and the forest construction of isdated trees is improved. The method of data anomaly detection based on SA-iForest was compared with the traditional Isolation Forest algorithm and LOF algorithm. The accuracy, execution efficiency, and stability of the proposed algorithm have significant improvement through the standard simulation data set.

Keywords Isolation forest, Outlier detection, SA-iForest, Simulated annealing

1 引言

数据异常检测作为数据挖掘的重要任务之一[1],通常会发现一些有价值的信息,因此受到该领域较多研究者的重视,一些好的异常检测方法应运而出。数据异常检测评价方法可分为:基于距离的方法[2]、基于密度的方法[3]、基于模型的方法[4-6]等。Liu 等[6]提出的隔离森林(Isolation Forest)算法通过计算待测数据的路径长度来进行异常检测。该算法与其他异常检测算法相比,减少了异常的掩盖和淹没效应,降低了计算开销,同时具备线性的时间复杂度,对高维数据的伸缩性较好,但该算法人为引入了随机因素,带来了精度低和稳定性差等问题[7-8]。文献[8]提出了 iForestASD 算法,该算法采用滑动窗口的框架处理流数据并且考虑了概念漂移现象,可以有效地检测流数据的异常实例。Yu 等[9]提出了 iForest 和 LOF 相结合的方法,该方法通过数据集的单次扫描生成一组异常候选,然后检测异常候选从而得出异常结果,提高了稳定性和精度,但是步骤繁琐,执行效率较低。文献[10]提出了一种基于相对质量改进的 iForest 算法,该算法基于路径长度的全局排名与局部排名,解决了局部异常被类似密度的正常数据簇掩蔽的问题,但是增大了计算开销。

与其他异常检测算法相比,Isolation Forest 没有利用距离或密度来检测异常,避免了基于距离和密度方法进行检测时的大计算量。该方法利用异常数据少且与众不同的特点,将它们与正常数据快速分离,具有较低的线性时间复杂度。虽然 Isolation Forest 算法在异常检测方面取得了较好的效果,但仍存在一些不足之处:

1)Isolation Forest 算法异常检测的精度与 iTree 的数目相关,构建大规模 iTree 需要耗费较多的内存空间,同时导致更大的计算量。

2)随着 iTree 数量的增加,iTree 之间的差异性将减小,导致泛化性能降低。同时,iTree 数量的增加会导致出现大量性能较差的 iTree,且其异常检测的精确度参差不齐。

3)Isolation Forest 算法在处理大数据时易受最大内存空间限制,因此处理海量数据较为困难,并且人为引入了随机因素造成精度低和稳定性差等问题。

针对上述问题,本文提出了一种新的 SA-iForest 的数据

到稿日期:2017-09-05 返修日期:2017-11-27 本文受国家自然科学基金项目(61502118)资助。
徐 东(1969-),男,博士,教授,主要研究方向为可信计算、网络与信息安全;王岩俊(1992-),男,硕士生,主要研究方向为可信计算、网络与信息安全;孟宇龙(1976-),男,博士,讲师,主要研究方向为可信计算、网络与信息安全,E-mail:mengyulong@hrbeu.edu.cn(通信作者);张子迎(1973-),男,博士,副教授,主要研究方向为人工智能与机器感知。

异常检测方法，该方法利用模拟退火(Simulated Annealing)算法优化 iForest，增强了 iForest 的泛化能力，通过去除冗余 iTree，减少了检测计算量，加快了检测速度，增强了稳定性，提高了数据异常检测性能。

2 Isolation Forest 算法及其改进

2.1 Isolation Forest 算法

Isolation Forest 算法的设计利用了异常数据的两个特征[11]：1)异常数据占数据集总体规模的比重较小；2)异常数据相比正常数据的属性值存在明显的差异。在仅包含数值类型的训练集中，对数据进行递归的划分，直至 iTree 将每个数据与其他数据区别开来。因为它们对隔离具有较强的敏感性，所以异常数据更接近于树的根节点，而正常数据离根节点较远，这样用少量的特征条件就能检测出异常数据。

Isolation Forest 算法的核心在于构建由 iTree 组成的森林(iForest)。为方便描述和计算，Isolation Forest 算法引入了隔离树、路径长度的定义。

定义 1(隔离树(iTree)) 令 T 是一棵二叉树，N 是 T 的节点，若 N 是叶子节点，则称其为外部节点，若 N 是一个具有两个孩子的节点，则称其为内部节点。

一棵 iTree 的构造过程如下：从数据集 $D=\{d_1,d_2,\cdots,d_n\}$ 中随机地选择属性 A 和分裂值 P，然后按照 A 的值(记为 $d_i(A)$)对每个数据 d_i 进行划分。如果 $d_i(A)<p$，则将数据 d_i 放在左子树，反之则放在右子树。以此递归地构造左子树和右子树，直至满足下列条件之一：1)D 中只剩下一条数据或多条相同的数据；2)树达到最大高度。

定义 2(路径长度) 在一棵 iTree 中，从根节点到外部节点所经历边的数目称为路径长度，记为 $h(d)$。

由于 iTree 与二叉查找树的结构等价，因此包含 d 的叶子节点的路径长度等于二叉查找树中失败查询的路径长度。给定数据集 D，文献[12]给出了二叉查找树中失败查询的路径长度：

$$C(n)=2H(n-1)-(2(n-1)/n) \tag{1}$$

其中，$H(i)=Ln(i)+\gamma$，γ 为欧拉常数；n 为叶子节点数；$C(n)$ 为给定 n 时 $h(d)$ 的平均值，用以标准化 $h(d)$。

Isolation Forest 算法构造一定数目的 iTree 来组成 iForest。具体地，随机采样提取 D 的子集来构造每棵 iTree，以保证 iTree 的多样性。通过遍历 iForest 中的每棵 iTree，计算数据 d 在每棵树中的路径长度，然后根据其路径长度计算 d 的异常分数，从而判断 d 是否异常。数据 d 的异常分数 $S(d,n)$ 如式(2)所示：

$$S(d,n)=2^{\frac{-E(h(d))}{C(n)}} \tag{2}$$

其中，$E(h(d))$ 为 iTree 集合中 $h(d)$ 的平均值。当 $E(h(d))\to C(n)$ 时，$S\to 0.5$，即当所有数据均返回的 $S\approx 0.5$ 时，全部样本中没有明显的异常值；当 $E(h(d))\to 0$ 时，$S\to 1$，即当数据返回的 S 非常接近于 1 时，它们是异常值；当 $E(h(d))\to n-1$ 时，$S\to 0$，即当数据返回的 S 远小于 0.5 时，它们有很大的可能为正常值。

2.2 改进后的 Isolation Forest 算法

虽然 Isolation Forest 算法在一定程度上减少了异常的掩盖和淹没效应，且具有较低的线性时间复杂度，但由于人为引入了随机因素，造成了精度低和稳定性差等问题。并且在构造iForest 时，存在冗余的 iTree，从而造成了内存空间浪费和计算量增大，影响了算法的执行效率。本文基于模拟退火算法改进了 Isdation Forest 算法中的森林构建过程，使得改进后的 Isdation Forest 算法在异常检测方面的性能有所提高。

首先，通过 Q-统计量法计算 iTree 之间的差异值。给定训练集 D_{Train}，如果树 T_i 能正确检测 d_k，则 $y_{k,i}=1$，否则 $y_{k,i}=0$，$i=1,2,\cdots,L$。T_i 与 T_j 的检测结果列联表如表 1 所列。

表 1 T_i 与 T_j 的检测结果列联表

Table 1 T_i and T_j test results listed in table

	$T_i=1$	$T_i=0$
$T_j=1$	N^{11}	N^{10}
$T_j=0$	N^{01}	N^{00}

表 1 中，$n=N^{11}+N^{10}+N^{01}+N^{00}$。

T_i 与 T_j 之间的差异值 $Q_{i,j}$ 如式(3)所示：

$$Q_{i,j}=\frac{N^{11}N^{00}-N^{10}N^{01}}{N^{11}N^{00}+N^{10}N^{01}} \tag{3}$$

$$Q=\begin{bmatrix} Q_{11} & Q_{12} & \cdots & Q_{1L} \\ Q_{21} & Q_{22} & \cdots & Q_{2L} \\ \vdots & \vdots & \ddots & \vdots \\ Q_{L1} & Q_{L2} & \cdots & Q_{LL} \end{bmatrix} \tag{4}$$

其中，N^{ab} 表示 T_i 和 T_j 检测 D_{Train} 中的 n 个样本，满足 $y_{k,i}=a$ 和 $y_{k,j}=b$ 的样本数目，$k=1,2,\cdots,n$；Q 表示 L 棵 iTree 的差异矩阵。统计学认为，如果两个 iTree 独立，那么这两个 iTree 的 Q 统计量的值为 0。Q 统计量的值在$[-1,1]$之间变化，值越大两棵 iTree 的差异度越小。

其次，用交叉验证法计算每棵 iTree 的精度值。首先将原始训练数据划分为数量相等且互不相交的 N 个子集，每次用 $N-1$ 个子集进行训练，用剩余的 1 个子集进行测试，将 N 份数据逐一作为测试集进行训练和测试，最终 N 个度量值的平均值为该棵 iTree 的精度值 X。该做法的目的在于：在保证各棵 iTree 特性不变的情况下，使其更有效地反映出它们各自的真实检测能力，比在原始训练集上直接测试更为真实客观。

最后，根据 iTree 的差异值和精度值计算每棵 iTree 的适应度值，利用模拟退火算法从 T 中选出 l 个较优秀的 iTree 组合成 iForest。适应度函数如式(5)所示：

$$F(T_j)=\frac{1}{W_1X_j+W_2Q_{ij}} \tag{5}$$

其中，$F(T_j)$表示 T_j 的适应度函数，X_j 表示 T_j 的精度值，W_1 和 W_2 分别表示精确度和差异性对应的权重。

模拟退火算法最早由 Kirkpatrick 等应用于组合优化领域[13]。随机选定一棵 iTree 的适应度值作为初始解，并借助控制参数 t 递减时产生的一系列 Markov 链，利用一个新解产生装置和接受准则，重复执行“产生新解→计算目标函数差→判断是否接受新解→接受或舍弃新解”，不断地对当前解进行迭代，算法终止时输出选出的最优 iTree。如此重复循环，从初始森林中选出检测性能高的 l 棵 iTree 构建成 iForest，减少了 iForest 的内存空间，降低了检测计算量，提高了精确度和执行效率。

3　基于 SA-iForest 的数据异常检测算法

在一个数据集中,根据异常数据的定义可以看出,异常数据只是很少一部分的数据[14]。根据异常数据所占比例较少的特点,在训练集中随机选择属性以训练出多棵 iTree,通过交叉验证法计算检测精度,通过 Q-统计量法计算 iTree 之间的差异值,然后根据精度值和差异值选出较优秀的 iTree 构建成iForest,最后对测试集进行测试,从而统计出其异常分值。该算法的具体步骤如算法 1 所示。

算法 1　基于 SA-iForest 的数据异常检测算法

输入:数据集 D;子采样大小 n;生成 iTree 的数量 L

Step 1　设置 iTree 的最大高度,初始化 iForest。

Step 2　构建第一棵 iTree。

Step 3　重复 Step 2,再构建 L−1 棵树组成初始森林。

Step 4　用训练集 D_{Train} 对初始森林 T 进行训练,根据交叉验证法计算每棵 iTree 的精度值,用 Q-统计量法计算 iTree 之间的差异值。

Step 5　根据 iTree 的差异性和精确度,利用模拟退火算法从初始森林 T 中选出 L 棵适应度值较优的 iTree 组合成 iForest。

Step 5.1　初始化。取初始温度 t_0 足够大,令温度 $t=t_0$,任取初始解 T_1。

Step 5.2　根据当前温度 t,重复 Step 5.3−Step 5.6。

Step 5.3　对当前解 T_1 随机扰动产生一个新解 T_2。

Step 5.4　计算 T_2 的增量 $df=F(T_2)-F(T_1)$,其中 $F(T_1)$ 为树 T_1 的适应度值。

Step 5.5　若 $df<0$,则接受 T_2 作为新的当前解,$T_1=T_2$;否则按 Metropolis 规则,计算 T_2 的接受概率 p,随机产生(0,1)区间上均匀分布的随机数 rand,若 $p>rand$,则接受 T_2 作为新的当前解,即 $T_1=T_2$,否则保留当前解 T_1。

Step 5.6　若满足设定的终止条件,则输出当前解 T_1 为最优解,终止条件为新解 T_2 在连续若干个 Metropolis 链中都没有被接受或者是达到结束温度;否则按衰减函数衰减温度 t 后返回 Step 5.2。所述衰减函数为:

$$t_s=\frac{t_0}{\lg(1+\theta)},\theta=1,2,\cdots \tag{6}$$

其中,t_s 为第 θ 步时的温度值,t_0 为初始温度。

Step 5.7　重复 Step 5.3−Step 5.6,从初始森林 T 中选出 $l(l\leqslant L)$ 棵具有较优适应值的 iTree 构成 iForest。

Step 6　对待检测数据进行检测,计算数据 d 在每一棵 iTree 的路径长度 h(d),并得到异常分数 S(d,n),进而判断数据 d 是否异常。

4　实验及结果分析

本节通过实验对 SA-iForest,Isolation Forest,LOF 算法的执行效率、有效性和稳定性进行分析和对比。从 UCI 数据集中选取了 Breastw,Diabetes,Unbalanced,Http,Shuttle,Pendigits 和 Messidor_features 7 个数据集。参考文献[15]中的实验方法,对每个数据集选用属于稀少类的样本作为异常数据。表 2 列出了各个数据集的统计特征。实验平台为配有 3.60 GHz CPU 和 8.00 GB 内存的 Windows 主机,实验程序采用 Matlab 编写。SA-iForest 执行参数的默认值为:$l=100$,$m=256$,即子采样大小为 256。

表 2　实验数据集特征

Table 2　Experimental dataset features

数据集	元素数目	属性数目	异常点数目
Breastw	683	9	239
Diabetes	768	9	268
Unbalanced	856	33	12
Http	567497	3	2270
Shuttle	49097	9	3437
Pendigits	6870	17	156
Messidor_features	1151	20	540

4.1　有效性验证

为了验证 SA-iForest 的精确度,分别采用 Breastw,Diabetes,Unbalanced,Http,Shuttle,Pendigits 和 Messidor_features 7 个数据集作为测试数据集,对 SA-iForest,Isolation Forest 和 LOF 进行精确度测试。图 1 和图 2 给出了 Breastw 和 Http 两个数据集的 ROC 曲线图。为了使结果更客观,对每一个算法分别运行 10 次,然后取平均值。在不同数据集上检测对应的 *AUC* 值,结果如表 3 所列。其中,*AUC* 为 ROC 曲线下方的面积,通过比较 *AUC* 值的大小来评判算法的性能[16-17]。*AUC* 的值为 0.5～1.0,*AUC* 越接近于 1.0,说明算法的精确度越高。

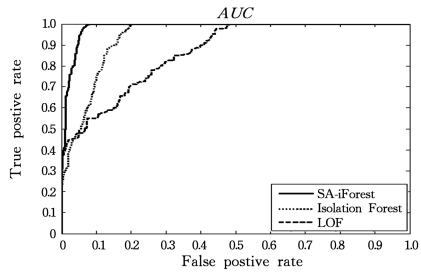

图 1　Breastw 数据集的测试结果

Fig. 1　Test results of Breastw dataset

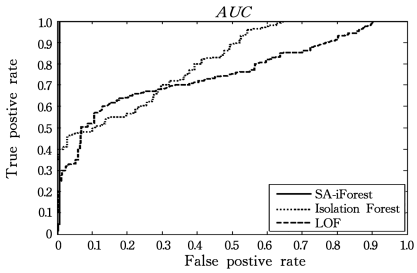

图 2　Http 数据集的测试结果

Fig. 2　Test results of Http dataset

表 3　在不同数据集上检测对应的 *AUC* 值

Table 3　Corresponding *AUC* values detected on different data sets

数据集	SA-iForest	Isolation Forest	LOF
Breastw	0.98	0.92	0.84
Diabetes	0.93	0.61	0.55
Unbalanced	0.87	0.57	0.62
Http	0.99	0.89	0.75
Shuttle	1.00	0.73	0.58
Pendigits	0.96	0.77	0.50
Messidor_features	1.00	0.85	0.52

从实验结果可知,SA-iForest 是可行的,其异常检测准确率比 Isolation Forest 和 LOF 的更好。这主要是因为SA-iForest 通过交叉验证计算出每棵 iTree 的异常检测精度值,同时采用 Q-统计量计算出 iTree 之间的差异值,利用模拟退火算法

筛选出异常检测精度相对较高的 iTree，提高了算法的检测精度。

由表 2 和表 3 可知，对于异常点较少的 Unbalanced 数据集，LOF 的检测精确度高于 Isolation Forest，且 SA-iForest 的检测精确度高于 LOF 算法。对于 Http 和 Shuttle 数据集，由于数据集较大且异常点较多，导致 LOF 算法的精确度相对较低，而 SA-iForest 和 Isolation Forest 是直接计算待测数据落在叶子节点的位置，通过其平均路径长度得到异常分数，最后根据异常分数的大小判断该数据是否异常，因此 SA-iForest 和 Isolation Forest 的检测精确度相对较高。并且 SA-iForest 在 Http 和 Shuttle 数据集上的 *AUC* 值分别为 0.99 和 1.00，说明 SA-iForest 算法在处理大数据集时更具优势。SA-iForest 在 Breastw，Diabetes，Pendigits，Messidor_features 数据集上的检测精确度也明显高于 Isolation Forest 和 LOF。

为了对比本文方法与 Isolation Forest，LOF 算法在稳定性上的差异，在表 2 中最大的数据集 Http 上进行实验，且连续运行 20 次，结果如图 3 所示。在其他数据集上的运行结果与此类似。

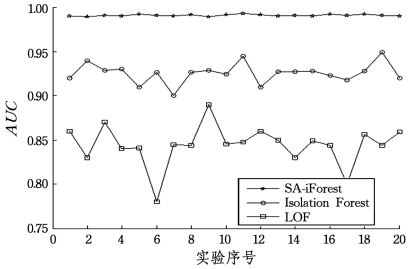

图 3 异常检测的稳定性

Fig. 3 Stability of anomaly detection

由图 3 可知，SA-iForest 在同一数据集上的多次实验结果的波动相对较低，而 Isolation Forest 算法和 LOF 算法的波动较大。SA-iForest，Isolation Forest 和 LOF 在 Http 数据集上经过 20 次异常检测的标准差分别为 $\sigma_1=0.0015$，$\sigma_2=0.01148$，$\sigma_3=0.0231$，这说明 SA-iForest 具有更好的稳定性。

4.2 执行时间对比分析

首先，将表 2 中的 7 个数据集作为测试集，对 SA-iForest，Isolation Forest 和 LOF 进行执行时间对比。为了使结果更具客观性，分别执行上述 3 种方法 10 次。表 4 列出了 SA-iForest，Isolation Forest 和 LOF 在不同数据集上的运行时间。

表 4 3 种方法在不同数据集上的执行时间

Table 4 Execution time of three methods on different data sets

数据集	SA-iForest/s	Isolation Forest/s	LOF/s
Breastw	0.11	0.23	1.14
Diabetes	0.24	0.29	2.07
Unbalanced	0.27	0.38	3.63
Http	8.83	32.67	9186.15
Shuttle	3.21	9.36	736.08
Pendigits	1.40	4.88	376.12
Messidor_features	0.89	1.16	4.54

从表 4 可以看出，SA-iForest，Isolation Forest 的执行时间明显低于 LOF 的执行时间，这是因为 SA-iForest 算法和 Isolation Forest 算法没有利用距离或密度测量来检测异常，这样消除了基于距离和密度方法的计算成本。通过时间复杂度进一步分析，LOF 的时间复杂度为 $O(n^2)$。而 Isolation Forest 在训练阶段的时间复杂度为 $O(Lm\log(m))$，在测试阶段的时间复杂度为 $O(nL\log(m))$。SA-iForest 在训练阶段的时间复杂度为 $O(L(m\log(m)+1))$，在测试阶段的时间复杂度为 $O(nl\log(m))$。对于 Http 数据集，当 $L=500$，$m=256$，$l=100$ 且检测 567497 条数据时，SA-iForest 的总处理时间为 8.83 s，Isolation Forest 的处理时间为 32.67 s，是 SA-iForest 的 3.7 倍，这表明 SA-iForest 在计算复杂度方面具有较低的常数。由于 SA-iForest 是通过模拟退火算法筛选出检测性能较好的树构建森林，而不是保留所有的树，而 Isolation Forest 是将构建的所有的树都用来异常检测；并且 SA-iForest 去除了一些检测性能差的树，减少了计算量，还利用模拟退火优化算法的快速收敛性提高了异常数据检测效率。因此，SA-iForest 的执行时间低于 Isolation Forest 的执行时间，同时还提高了泛化性能和检测性能。

结束语 针对 Isolation Forest 算法异常检测精度低、执行效率差和泛化能力弱等问题，本文提出了一种新的 SA-iForest 的数据异常检测方法，通过大规模数据集进行验证并与 Isolation Forest 算法及 LOF 算法进行比较。实验结果表明，SA-iForest在执行效率和精确度方面得到了显著提高，并且增强了稳定性。此外，所提算法的结构实现简单、执行时间短且开销小，在大数据时代数据无法存储时尤其有用。

参 考 文 献

[1] AGGARWAL C C. Outlier analysis [M]. Berlin: Springer, 2013.

[2] KNORR E M, NG R T. Algorithms for minning distance based outliers in large datasets[C]// Proceedings of the 24th International Conference on Very Large Databases. 1998: 392-403.

[3] BREUNING M M, KRIEGEL H P, NG R T, et al. LoF: Indentifying density-based local outliers[J]. ACM SIGMOD Record, 2000, 29(2): 93-104.

[4] HE Z, XU X, DENG S. Discovering cluster-based local outliers [J]. Pattern Recognition Letters, 2003, 24(9): 1641-1650.

[5] ROUSSEEUW P J, DRIESSEN K V. A fast algorithm for the minimum covariance determinant estimator[J]. Technometrics, 1999, 41(3): 212-223.

[6] LIU F T, TING K M, ZHOU Z H. Isolation-based Anomaly Detection[J]. ACM Transactions on Knowledge Discovery from Data, 2012, 6(1): 1556-4681.

[7] VINH N X, CHAN J, ROMANO S, et al. Discovering outlying aspects in large datasets[J]. Data Mining & Knowledge Discovery, 2016, 30(6): 1520-1555.

[8] DING Z. An anomaly detection approach based on isolation forest algorithm for streaming data using sliding window[C]// Proceedings of the 3rd IFAC International Conference on Intelligent Control and Automation Science. 2013: 12-17.

[9] YU X, TANG L A, HAN J W. Filtering and refinement: a two-stage approach for efficient and effective anomaly detection[C]// IEEE International Conference on Data Mining. 2009: 617-626.

[10] ARYAL S, TING K M, WELLS J R, et al. Improving iForest with Relative Mass [J]. Advances in Knowledge Discovery and Data Minning, 2014, 8444(2): 510-521.

[11] HOU Y X, DUAN L, QIN J L, et al. Parallel Detection Design

Based on Isolation Forest [J]. Journal of Computer Engineering and Science,2017,39(2):236-244. (in Chinese)
侯泳旭,段磊,秦江龙,等. 基于 Isolation Forest 的并行化异常探测设计[J]. 计算机工程与科学,2017,39(2):236-244.

[12] PREISS B R. Data Structures and Algorithms with Object Oriented Design Patterns in Java[M]. New Jersey:Wiley,1999.

[13] WANG B,WANG S Y. A New Test Case Generation and Reduction Algorithm Based on Simulated Annealing[J]. Computer Applications and Software,2013,30(2):78-81. (in Chinese)
王博,王曙燕. 一种新的基于模拟退火的测试用例生成与约简算法[J]. 计算机应用与软件,2013,30(2):78-81.

[14] WANG C Y,LIU Z,WANG H B. An Algorithm for Exception Data Mining Based on OPTICS and IncLOF [J]. Journal of Tianjin University of Science and Technology,2015,6(31):14-18. (in Chinese)
王传玉,刘震,王怀彬. 一种基于 OPTICS 和 IncLOF 的异常数据挖掘算法[J]. 天津理工大学学报,2015,6(31):14-18.

[15] KELLER F,MULLER E,BOHM K. HiCS:High contrast subspaces for density-based outlier ranking [C]// Proceedings of IEEE International Conference on Data Engineering. 2012:1037-1048.

[16] BRADLEY A P. The Use of the Area under the ROC Curve in the Evaluation of Machine Learning Algorithms [J]. Pattern Recognition,1997,30(7):1145-1159.

[17] FAWCETT T. An Introduction to ROC Analysis[J]. Pattern Recognition Letters,2006,27(8):861-874.

(上接第 119 页)

小,且本文算法的均方根误差稳定在 1.8 m 左右。Chan 氏算法适应于测距方差处于一定范围内的场景,本文中 Chan 氏算法适应于测距方差在 0.4～1.6 范围内的情景。

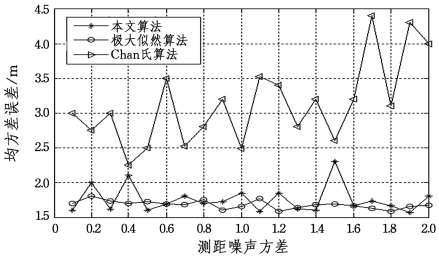

图 6　测距噪声对定位算法性能的影响

Fig. 6　Effect of ranging noise on performance of locating algorithm

结束语　本文提出了一种基站选择方法来减小室内定位的误差,其可以广泛地应用于各类蜂窝网的电信运营商级的室内定位场景,具有广阔的应用前景。该方法通过引入 TOA 消除 TDOA 中的虚定位点,并对初步定位结果进行二次聚类,以优选基站,达到减小室内定位误差的目的。下一步计划将服从高斯分布的定位误差应用于实际中,以降低不同定位场景可能带来的不同程度的定位差异。

参 考 文 献

[1] WU C S. Crowd sensing Based Wireless Indoor Position Location[D]. Beijing:Tsinghua University,2015. (in Chinese)
吴陈沭. 基于群智感知的无线室内定位[D]. 北京:清华大学. 2015.

[2] ROSA F D,PELOSI M,NURMI J,et al. Indoor Positioning in WLAN. Mobile Positioning and Tracking:From Conventional to Cooperative Techniques[M]// Mobile Positioning and Tracking. 2017:261-282.

[3] DE ANGELIS G,MOSCHITTA A,CARBONE P. Positioning techniques in indoor environments based on stochastic modeling of UWB round-trip-time measurements[J]. IEEE Transactions on Intelligent Transportation Systems,2016,17(8):2272-2281.

[4] WU C X,ZHANG G G,YANG Y H. channeled trend from the network hierarchy architecture operators[J]. Digital Communication,2013,40(1):48-51. (in Chinese)
吴翠先,张功国,杨映红. 从网络层次架构论运营商被管道化趋势[J]. 数字通信,2013,40(1):48-51.

[5] WANG X,QIU J,FAN J,et al. MDS-based localization scheme for large-scale WSNs within sparse anchor nodes[C]// 2015 IEEE International Conference on Communications (ICC). IEEE,2015:6609-6614.

[6] AHMADI H,VIANI F,POLO A,et al. An improved anchor selection strategy for wireless localization of WSN nodes[C]// 2016 IEEE Symposium on Computers and Communication (ISCC). IEEE,2016:108-113.

[7] HAN M F. Multilateration Algorithm in WSN Based on K-means Clustering and Data Consistency [D]. Changchun:Jilin University,2012:45-49. (in Chinese)
韩梦飞. 基于 K-means 聚类和数据一致性的 WSN 多边定位算法[D]. 长春:吉林大学,2012:45-49.

[8] CHEN S J,WANG H Q,CHEN Q,et al. Research and Implement of High Precision Indoor Location Simulation System[J]. Electronic Science &Technology,2016,3(6):710-715. (in Chinese)
陈诗军,王慧强,陈强,等. 一种高精度室内定位仿真系统的研究与实现[J]. 电子科学技术,2016,3(6):710-715.

[9] LIU Q,CHEN S J,WANG H Q,et al. Carrier-class Oriented High Precision Indoor location STandards,Systems and Technology[M]. Beijing:Publishing House of Electronics Industry,2017:16-25.

[10] YU P,ZHAN Y W. Moving weighted localization algorithm based on RSSI[J]. Application Research of Computers,2016,33(5):1450-1453. (in Chinese)
余刨,战荫伟. 基于 RSSI 的移动权值定位算法[J]. 计算机应用研究,2016,33(5):1450-1453.

[11] LI J R,WANG W L,JIE J,et al. Localization Algorithm for Wireless Sensor Networks Based on MDS-MAP Integrated with Maximum Likelihood Estimating[J]. Chinese Journal of Sensors and Actuators,2016,29(4):572-577. (in Chinese)
李津荣,王万良,介婧,等. 结合极大似然距离估计的 MDS-MAP 节点定位算法[J]. 传感技术学报,2016,29(4):572-577.

[12] LEE S,MORTARI D. Quasi-equal area subdivision algorithm for uniform points on a sphere with application to any geographical data distribution[J]. Computers and Geosciences,2017,103:142-151.