

基于加权分簇的移动无线传感器网络拓扑维护算法

姚玉坤, 李小勇, 徐栋梁, 刘江兵

(重庆邮电大学 移动通信技术重庆市重点实验室, 重庆 400065)

摘 要: 针对移动无线传感器网络(mobile wireless sensor networks, MWSN), 本文提出了一种基于加权分簇的拓扑维护算法(Weighted clustering based topology maintenance algorithm, WCTMA), 该算法首先基于节点的移动性和剩余能量计算节点的权重值并对网络进行分簇; 然后利用节点间控制消息的发送和接收来判断网络拓扑是否发生变化; 最后根据引起网络拓扑发生变化的不同因素, 具体设计了簇内和簇间的拓扑维护策略。仿真结果表明, WCTMA 算法与 FDRC 算法在节点的平均剩余能量、网络的连通度、网络的生命周期等性能方面具有显著改善。

关键词: 移动无线传感器网络; 拓扑维护; 加权分簇; 节点的移动性; 剩余能量; 节点的权重值

中图分类号: TP393

文献标识码: A

文章编号: 1000-7180(2018)08-0055-05

Weighted Clustering Based on Topology Maintenance Algorithm of Mobile Wireless Sensor Networks

YAO Yu-kun, LI Xiao-yong, XU Dong-liang, LIU Jiang-bing

(Chongqing Key Lab of Mobile Communications Technology, Chongqing University of Posts and Telecommunications, Chongqing 400065, China)

Abstract: To deal with the problems of network topology change caused by node mobility, new nodes joining network, and node failure that impacted on network performance in mobile wireless sensor networks (MWSN). Weighted clustering based topology maintenance algorithm (WCTMA) was proposed in this paper. Firstly, the nodes' weights were calculated based on the mobility and surplus energy, the network was clustered according to the nodes' weights. Secondly, the control messages of transmission and reception between nodes was used to judge whether the network topology has changed. Finally, cluster topology maintenance strategy and inter-cluster topology maintenance strategy were designed according to the different factors that cause the network topology change. The simulation results show that the WCTMA algorithm has significantly improved compare with FDRC algorithm in average node residual energy, network connectivity degree and lifetime of network.

Key words: MWSN; topology maintenance; weighted clustering; the mobility of the nodes

1 引言

移动无线传感器网络(mobile wireless sensor networks, MWSN)是一种由大量移动的微小传感器节点构成的自组织网络^[1]。拓扑维护是指节点实时监测网络拓扑状态, 当网络拓扑发生变化时而对网络拓扑做出维护的过程^[2]。节点的移动速度与方向的不确定性以及节点故障等因素都会导致通信链

路频繁的中断, 甚至造成网络的分割。因此利用拓扑维护算法修复网络拓扑、提高通信可靠性、延长网络生命周期^[3-4]具有重要意义。

近年来, 在拓扑维护^[5-6]方面已经取得一些研究成果。文献[7]提出了自适应分簇算法(NACA, new adaptive clustering algorithm), 该算法在分簇阶段和拓扑维护阶段都存在较大开销。文献[8]针对网络中节点的移动、新节点加入网络以及网络的安全性

等因素提出基于分簇的拓扑维护算法(TCM, topology control algorithm for node mobility), 但该算法在拓扑维护过程中需额外的控制消息. 文献[6]提出的基于故障检测的簇恢复算法(FDRC, fault detection and recovery of cluster)该算法在选择监测节点时既带来了开销也增加了能耗, 且监测节点本身可能成为故障节点. 本文主要针对上述基于拓扑重构的维护策略会增大开销和能耗的不足提出一种基于加权分簇的拓扑维护算法(WCTMA).

2 系统模型

为提高拓扑维护的效率, 降低拓扑维护的开销, WCTMA 算法通过基于节点的移动性和能量计算节点的权重值并对网络进行分簇, 通过加权分簇算法构建一个层次型的系统模型.

2.1 拓扑维护使用的控制消息

WCTMA 算法主要通过控制消息驱动网络拓扑结构的维护. 本文 WCTMA 算法使用以下几种用于拓扑维护的控制消息:

- (1) Chead 消息: 节点成为簇头后用于广播的消息;
- (2) Response 消息: 簇头节点用于回复节点加入网络和重新加入簇时的请求消息;
- (3) Request 消息: 节点加入网络和重新加入簇的广播请求消息;
- (4) Renew 消息: 成员节点权重值更新消息, 用于回复簇头 Chead 消息;
- (5) jion 消息: 簇头广播 Chead 消息后, 节点同意加入簇内的回复消息.

且 Chead 和 Response 消息中都包含包含了簇头节点的权重值、节点 ID、簇 ID 等信息; Request、Renew 和 jion 消息中包含了节点最新的权重值、节点 ID 等信息.

2.2 WCTMA 算法分簇过程

定义 1 节点移动因子 M . 移动因子定义为节点当前的移动速度与最大移动速率的比值, 如式(1)所示. 式中 (X_t, Y_t) 和 (X_{t-1}, Y_{t-1}) 分别表示两个不同时刻节点的坐标, $V_{\max}$ 表示节点的最大移动速率.

$$M = \frac{\frac{1}{T} \sum_{t=1}^T \sqrt{(X_t - X_{t-1})^2 + (Y_t - Y_{t-1})^2}}{V_{\max}} \quad (1)$$

定义 2 节点能量因子 E . 能量因子 E 定义为节点已消耗的能量与节点初始能量的比值, 如式(2)所示. 式中 $E_{\max}$ 表示网络中所有节点的初始能量, E

表示节点的剩余能量.

$$E = \frac{E_{\max} - E_{\text{res}}}{E_{\max}} \quad (2)$$

定义 3 节点权重值 W . 权重值 W 定义为节点 i 移动因子与能量因子的加权和, 如公式

$$W = aE + bM \quad (3)$$

式(3)中 a 和 b 分别是权重因子. 且 WCTMA 算法的具体的分簇步骤如下:

步骤 1: 节点根据公式(2)、(3)、(4)计算出权重值, 并将其添加到 Hello 包中, 然后广播 Hello 消息.

步骤 2: 节点 i 根据其邻居节点广播的 Hello 包创建权值列表 W -List, 然后根据 W -List 表中的权重值按由小到大进行排序.

步骤 3: W -List 表中权重值最小的节点 i 自发成为簇头节点, 随机生成簇 ID, 并广播 Chead 消息. 节点 i 的邻居节点 j 接收 Chead 消息后, 若同意加入该簇, 则回复 join 消息, 反之则不回复.

步骤 4 簇头节点将未回复 join 消息的节点从 W -List 表中删除, 从而得到更新后的簇成员列表 W -List. 然后簇头节点将 M -List 表信息发送给 sink 节点.

步骤 5 重复上述步骤, 保证所有节点只能成为簇头或成为某一个簇成员, 并确保所有的成员节点最多只能同时属于两个不同的簇.

3 WCTMA 拓扑维护策略

本文提出的 WCTMA 算法不同于拓扑重构的维护策略, 其主要从簇内以及簇间两个方面执行拓扑维护, 由于拓扑维护策略只在簇内或簇间执行, 不需洪泛控制消息至整个网络. 其主要思路是: 在分簇算法已成簇的基础上, 首先通过簇头与成员节点之间的控制消息判断网络拓扑是否发生变化, 然后判断拓扑变化是发生在簇内还是簇间, 从而决定执行簇内拓扑维护还是簇间拓扑维护策略.

3.1 簇内拓扑维护策略

引起簇内拓扑结构改变的因素有: 新节点加入网络; 簇头节点与成员节点之间链路中断; 簇内更换簇头节点等. 这些因素的产生都会触发簇内拓扑维护策略.

3.1.1 新节点加入网络

新节点加入网络后, 会改变已有的网络拓扑结构, 为减小节点在加入网络后对已有网络拓扑结构产生的影响, 在节点加入网络前后采取相应的处理

措施,具体过程如下:

步骤1:节点 i 在加入网络前若在 t_0 时间内收到某簇头节点 j 广播的 Chead 消息,且节点 i 愿意加入簇头节点 j 所在的簇,则回复 join 消息;反之,则不回复并继续侦听,若节点 i 在多个 t_0 时间内都未收到其它簇头节点广播的 Chead 消息则进入步骤2.

步骤2:节点 i 主动广播 Request 请求消息.若簇头节点某收到 Request 消息,则单播回复 Response 消息;若节点 i 接收了多个 Response 消息,则选择与 Response 消息信号强度最大的簇头节点比较权重的大小,然后进入步骤3;若节点 i 未收到 Response 消息则重复广播 Request.

步骤3:若簇头节点与节点 i 之间的权重值的差值大于阈值 W_{th1} ,则节点 i 随机生成一个新的簇ID,然后广播 Chead 消息,成为新的簇头节点,然后建立新的 M-List;反之则进入步骤4.

步骤4:若两者的差值不大于阈值 W_{th1} ,则节点 i 回复 jion 消息.簇头节点收到 jion 消息则将节点 i 添加到 M-List,更新 M-List 表并结束.

3.1.2 链路中断

分簇后的网络模型是层级集中控制,因此链路中断主要发生在簇头节点和簇内成员节点之间.链路中断的判断方法如下:若某簇头节点 j 广播 Chead 消息却未收到 M-List 表中簇内成员节点 i 回复的 Renew 消息或节点 i 在间隔 t_0 时间后仍未收到簇头节点 j 广播的 Chead 包,则判定成员节点 i 和簇头节点 j 之间发生了链路中断.针对此种情况下的拓扑维护策略如下:

步骤1:簇头节点 j 更新 M-List 表,将节点 i 从表中删除,并将更新后的 M-list 表信息发送给 sink 节点;而与簇头节点 j 失去关联的成员节点 i 则尝试加入其它的簇.节点 i 首先等待时间 t_0 ,若在时间 t_0 内收到其它簇头广播的 Chead 消息,则直接回复 jion 消息,加入到该簇内;反之则进入步骤2.

步骤2:若节点 i 在时间 t_0 内未收到 Chead 消息,则节点 i 主动广播 Request 请求消息.若节点 i 收到了其它簇头节点或 sink 节点回复的 Response 消息则进入步骤3;反之则标记为失效节点.

步骤3:若节点 i 既收到了簇头节点回复的 Response 消息又收到了 sink 节点回复的 Response 消息,则只向簇头节点回复 join 消息,如果节点 i 只收到了 sink 节点回复的 Response 消息时,则节点 i 独立成为一个簇头节点,等待后续节点的加入.

3.1.3 簇内簇头节点变更

由于网络中每个节点移动速度的不同,节点能量的损耗量也不同,导致节点权重值的改变量也会不同,网络中的簇头节点在一段时间之后可能不再适合继续担任簇头节点的角色.因此,为了维护网络拓扑,此时簇内需更换簇头节点.簇头节点的 M-List 表中存有簇内成员节点的权重值,并每隔 t_0 时间都会更新.更新方法如下:

步骤1:簇头节点以周期 t_0 向簇内成员节点广播 Chead 消息,簇内的成员节点接收 Chead 消息后按式(3)计算自己当前的权重值,并将其添加到更新包 Renew 中,然后将 Renew 包发给簇头节点;

步骤2:簇头节点收到 Renew 包后更新 M-List,若簇头节点发现 M-List 中权重值最小的节点的权重值比自身的权重值更小,且两者的差值大于阈值 W_{th1} ,则说明该成员节点更适合做簇头节点,此时簇头节点通过发送消息推举该节点作为新的簇头节点,节点收到簇头节点发送的消息后广播新的 Chead 消息,宣布自己成为新的簇头节点,邻居节点则根据收到的 Chead 消息决定是否加入该簇,若需要则回复 jion 消息,反之则不回复.

3.2 簇间拓扑维护策略

引起簇间拓扑结构改变的因素有:一方面簇内成员节点从一个簇移动到另一个簇,引起两个簇结构的调整;另一方面则是簇头节点移动到了另一个簇头的通信范围内,导致两个簇出现了大范围内重叠的情况.

3.2.1 成员节点从一个簇移动到另一个簇

节点在不同的簇间移动的过程中,为了保证节点间链路的连通性,节点首先与其它簇头节点建立新的链路,其次再断开与原来簇头间的链路.如图1所示,节点 k 由簇1的A位置移动到簇2的B位置,在移动的过程中当节点 k 检测到与簇头节点 i 的链路即将要中断时,节点 k 首先与簇2的簇头节点 j 建立新的链路,其目的是保证数据传输的连续性,提前做出拓扑维护措施,降低网络收敛时间.具体步骤可以分为以下4步:

步骤1:节点 k 根据收到 Chead 消息的信号强度来判断彼此间的链路是否即将中断;若节点 k 接收 Chead 消息的信号强度小于阈值 $RSSI_{th}$,则表明链路即将中断,进入步骤2;反之则在收到下一个 Chead 消息时继续判断信号强度.

步骤2:若节点 k 在检测到与簇头节点间的链路即将中断前就已经与簇头节点 j 建立了新的链

路,则即使节点 k 与簇头节点 i 之间的链路即将中断,节点 k 也无需处理,簇头节点 i 在下一广播周期内更新 M -List,并将更新后的 M -list 表发送给 sink 节点.反之则进入步骤 3.

步骤 3:节点 k 等待时间 t_0 ,若节点 k 在时间 t_0 内收到了簇头节点 j 广播的 Chead 消息,则节点 k 直接回复 jion 消息同意加入簇 2;反之进入步骤 4.

步骤 4:若节点 k 在时间 t_0 内未收到 Chead 消息,则节点 k 主动广播 Request 请求消息,等待其它节点簇头节点回应的 Response 消息,然后加入到其它的簇内.

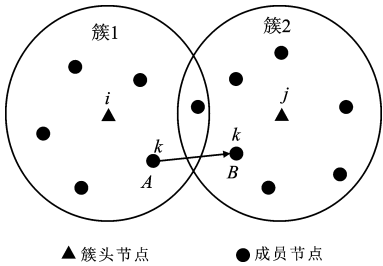


图 1 节点从一个簇移动到另一个簇

3.2.2 簇头节点移动到另一个簇头内

随着簇头节点的移动,易出现一个簇头节点移动到另一个簇头节点的通信范围内的情况,如图 2 所示.当簇 1 的簇头节点 i 收到了簇 2 的簇头节点 j 广播的 Chead 消息时,或簇 2 的簇头节点 j 接收了簇 1 的簇头节点 i 广播的 Chead 消息时,则表明簇头节点 i 和 j 都移动到了彼此的通信范围内,簇 1 和簇 2 可能有大部分的成员节点是公共的.为了避免因一个成员节点的改变触发两个簇头节点采取维护策略,同时减小簇头节点与成员节点间的控制消息的收发,采取拓扑维护措施,具体过程如下.

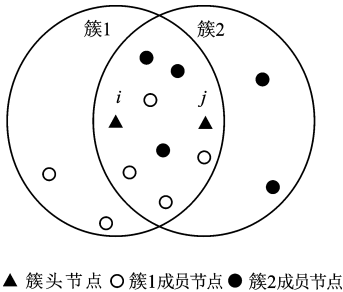


图 2 簇头点移动到另一个簇头内

当簇头节点 i 和簇头节点 j 都移动到了彼此的通信范围内时,簇头节点 i 和 j 比较接收的 Chead 消息中的权重值的大小,权重值大的簇头节点转化为普通的成员节点,权重值小的节点继续担任簇头,

并广播 Chead 消息,等待一跳范围内其它节点的回复 jion 消息,根据 jion 消息更新 M -List,并将更新后的 M -list 表发送给 sink 节点.

4 仿真实验及结果分析

本文采用 OPNET Modeler 14. 5 仿真软件对 WCTMA 算法性能进行仿真,并选取 FDRC 算法^[9]进行比较.通过仿真比较分析 FDRC 算法和 WCTMA 算法的节点平均剩余能量、网络的连通度以及网络生命周期等指标的性能差异.

4.1 仿真场景和参数设置

网络仿真的场景为分布在 $1\,000\text{ m} \times 1\,000\text{ m}$ 区域内,移动节点数目为 200,节点最大的通信半径为 100 m,节点初始能量为 5 J,节点发送或是接收单位比特数据的能量消耗为 50 nJ,节点的移动速度在 $[0, 5]\text{ m/s}$ 内取随机值,即节点的最大移动速度为 5 m/s,节点的最大发射功率为 50 mW,数据分组的大小为 256 bit,权值 a 和 b 的值都是 0. 5,仿真时间设置为 1 000 s.

4.2 仿真结果及分析

4.2.1 节点的平均剩余能量

从图 3 得知,WCTMA 算法的节点平均剩余能量比 FDRC 算法的节点平均剩余能量高.

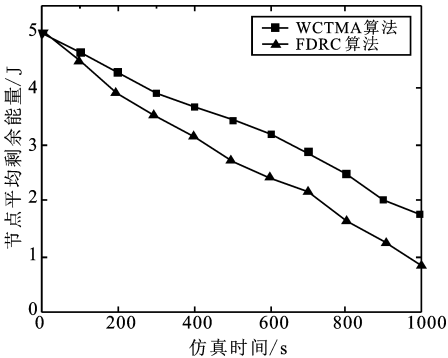


图 3 节点的平均剩余能量对比图

这是因为 FDRC 算法在分簇完成后,需要选取部分监测节点,选取过程中需要消耗一部分能量,且监测节点周期性发送探测消息,使监测节点能量消耗过快,且节点在选簇头节点时也会消耗一部分能量,所以节点的平均剩余能量较小.而 WCTMA 算法则利用簇头节点监测网络拓扑状态,不需额外的选择监测节点,且在分簇过程中网络中的节点只需广播一次 Hello 包就可快速分簇,既减小了开销,也降低了节点能耗.

4.2.2 网络连通度

从图 4 可得,WCTMA 算法的连通度比 FDRC 算法的大,主要原因有:(1)FDRC 算法从网络中随机选取监测节点,但监测节点不能检测成员节点的故障,故簇内成员节点发生故障时无相应的拓扑维护策略;(2)监测节点可能会成为故障节点,FDRC 算法同样无相应拓扑维护策略.而 WCTMA 算法通过簇头节点广播的 Chead 消息和成员节点回复的 Renew 消息判断网络拓扑是否发生变化,因此 WCTMA 算法能检测到本簇内所有节点,更有利于提高网络的连通度.

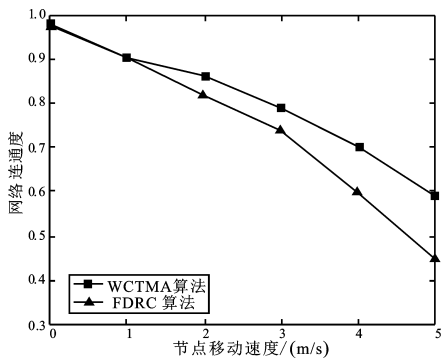


图 4 连通度对比图

4.2.3 网络生命周期

从图 5 得知,通过基于分簇维护算法可以更好的均衡节点的能量消耗,有利于延长网络生命周期.且 WCTMA 算法比 FDRC 算法的网络生命周期长.其原因主要是 WCTMA 算法直接利用簇头节点监测网络拓扑状态,且簇头节点根据节点的权重值自适应的更换,节点能量消耗更均匀,不会因为某一节点承担的业务较多导致能量消耗过快而过早死亡.

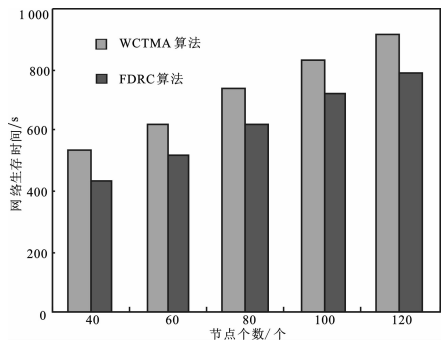


图 5 网络生命周期对比

5 结束语

本文提出一种拓扑维护算法 WCTMA,该算法利用簇头节点集中控制并利用控制消息驱动维护网络拓扑.相比现有拓扑重构的维护 FDRC 算法有较

大创新.仿真结果表明,WCTMA 算法在节点的平均剩余能量、网络连通度以及网络生命周期等性能方面均显著改善.下一步将研究根据移动传感器网络的特点,如何把分簇算法和节点功率控制相结合,进一步减小拓扑维护开销,延长网络生命周期,优化网络性能.

参考文献:

[1] Tomoki Y, Shojiro N. Data transmission techniques for mobile sensor networks[C]// 2014 28th international Conference on Advanced information Networking and Applications Workshops (WAINA). Taipei, Taiwan, 2014.

[2] Aparna M. A new method for controlling and maintaining topology in wireless sensor networks[J]. International Journal of Computer Networks & Communications, 2014, 6(4):91-98.

[3] Wightman P M, Labrador M. Topology maintenance: Extending the lifetime of wireless sensor networks [J]. Latin America Transactions IEEE, 2010, 8(4): 469-475.

[4] Shahid A, Qureshi H K. A survey on topology maintenance techniques to extend the lifetime of wireless sensor networks [M]. Springer Berlin Heidelberg, Wireless Sensor Networks for Developing Countries, 2013:96-107.

[5] 苏晓, 于洪. 移动自组织网中一种平均节点度分簇算法[J]. 重庆邮电大学学报自然科学版, 2010, 22(2): 237-241.

[6] Vijayanand K, Rajesh K Y. Prolonging network lifetime by electing suitable cluster head by dynamic weight adjustment for weighted clustering algorithm in MANET[C]// 2016 international Conference on Computing for Sustainable Global Development (india-com). [s. l]. 2016: 2915-2920.

[7] 章韵, 宋汝芸, 陈志, 等. 基于簇头选择的移动传感网拓扑控制算法研究[J]. 传感技术学报, 2011, 24(11):1602-1606.

[8] Ying Z, Wei C, Liang J, et al. A network topology control and identity authentication protocol with support for movable sensor nodes[J]. Sensors, 2015, 15(12):29958-29969.

[9] Paranjape S, Sutaone M, Mukherji P. Fault detection and recovery of cluster heads in mobile wireless sensor networks[C]// TENCON 2015 IEEE Region 10 Conference. Penang, Malaysia, IEEE, 2015:1-5.

本次实验虽然提高了邮件分类的效率,但存在不足之处. 应用的 spark 集群框架对邮件分类加速,在相同的数据集上分类时,数据的正确率等会有不同程度的起伏甚至下降,分析问题原因为不同的邮件分类时结果不同因此邮件分类的精度还需提高. 虽然此次实验虽然在精度上并没有下降,但在 NB 算法精度方面有待提高. 后续研究可以考虑在 spark 上运用特征提取和增加权值等方面增加分类的精度^[12];另一方面,朴素贝叶斯算法主要是研究整体邮件的合法与垃圾邮件的分类,是一种大众化的分类. 但是在邮件的具体分类上个体的看法不一定与众人相同,意味着可以考虑根据每个人兴趣情况取舍邮件,研究可以考虑根据个人情况进一步提高邮件的过滤精度.

参考文献:

- [1] 方言. 邮件安全不容忽视[J]. 中国信息安全, 2017, 7(3):100-100.
- [2] 张铭锋, 李云春, 李巍. 垃圾邮件过滤的贝叶斯方法综述[J]. 计算机应用研究, 2005, 22(8):14-19.
- [3] Ding Q, Boykin R. A framework for distributed nearest neighbor classification using Hadoop[J]. Journal of Computational Methods in Sciences & Engineering, 2016(17):1-9.
- [4] 陶永才, 薛正元, 石磊. 基于 MapReduce 的贝叶斯垃圾邮件过滤机制[J]. 计算机应用, 2011, 31(9):2412-2416.
- [5] 曾青华, 袁家斌, 张云洲. 基于 Hadoop 的贝叶斯过滤 MapReduce 模型[J]. 计算机工程, 2013, 39(11):57-60.

- [6] Davoodi M, Segal S, Peretz N K, et al. Semi-automated program for analysis of local Ca²⁺, spark release with application for classification of heart cell type[J]. Cell Calcium, 2017(64):83-90.
- [7] 刘月峰, 苑江浩, 张晓琳. 改进 NB 算法在垃圾邮件过滤技术中的研究[J]. 微电子学与计算机, 2017, 34(4):115-120.
- [8] 于苹苹, 倪建成, 姚彬修, 等. 基于 Spark 框架的高效 KNN 中文文本分类算法[J]. 计算机应用, 2016, 36(12):3292-3297.
- [9] Ramírez-Gallego S, Krawczyk B, García S, et al. Nearest neighbor classification for high-speed big data streams using spark[J]. IEEE Transactions on Systems Man & Cybernetics Systems, 2017(99):1-13.
- [10] Geng Y, Zhang J. Optimized parallelization of binary classification algorithms based on spark[J]. IEEE Robotics & Automation Magazine, 2006, 13(4):11-13.
- [11] 王雯, 赵衍衍, 李翠平, 等. Spark 平台下的短文本特征扩展与分类研究[J]. 计算机科学与探索, 2017, 11(5):732-741.
- [12] 李涛, 刘斌. Spark 平台下的高效 Web 文本分类系统的研究[J]. 计算机应用与软件, 2016, 33(11):33-36.

作者简介:

刘月峰 男, (1977-), 博士研究生, 副教授. 研究方向为机器学习、文本分类.

张亚斌(通讯作者) 男, (1992-), 硕士研究生. 研究方向为大数据、文本分类. E-mail: 1551255025@163.com.

苑江浩 男, (1992-), 硕士研究生. 研究方向为机器学习、数据挖掘.

(上接第 59 页)

作者简介:

姚玉坤 女, (1964-), 硕士, 教授. 研究方向为宽带无线自组织网络、网络编码.

李小勇(通讯作者) 男, (1992-), 硕士研究生. 研究方向为无线宽带自组织网络路由、网络编码.

E-mail: lixy_hbjz_cqupt@163.com.

徐栋梁 男, (1990-), 硕士. 研究方向为无线传感器网络拓扑维护算法.

刘江兵 男, (1989-), 硕士研究生. 研究方向为无线自组织网络路由算法.