

自组网分簇算法仿真设计

郭晓莲¹,林志伟²,黄榕宁²

(1. 福建工程学院 计算机与信息科学系, 福建 福州 350014;
2. 福建师范大学 数学与计算机科学学院, 福建 福州 350007)

摘要:自组网是一种由移动节点自组织形成的、不需要任何基础设施的网络, 针对其随机的拓扑结构研究人员提出了基于分簇结构的拓扑机制, 用于网络路由优化和安全控制。然而, 这些算法在不同的移动环境中面临着不同的挑战, 因而所表现出来的性能也各不相同, 为进一步验证这些算法在不同移动环境中的有效性, 文中使用 Delphi 设计了自组网的几个典型分簇算法, 通过随机环境的仿真实验, 得到相关仿真数据, 分析比较了这些算法的性能, 为进一步的研究提供依据。

关键词:自组网; 分簇; 连通支配集

中图分类号: TP393

文献标识码: A

文章编号: 1673-629X(2007)09-0092-03

Simulation Design of Cluster Algorithm in Ad Hoc Network

GUO Xiao-lian¹, LIN Zhi-wei², HUANG Rong-ning²

(1. Department of Computer and Information, Fujian University of Technology, Fuzhou 350014, China;
2. School of Mathematics and Computer Science, Fujian Normal University, Fuzhou 350007, China)

Abstract: Ad Hoc network is a self-organized network, consisting of mobile nodes without any infrastructure. Cluster-based topology control strategies have been proposed to meet to routing optimization and security control due to the random environment of Ad Hoc network. But these algorithms vary in performance when facing different mobile environment. Some proposed cluster algorithms were designed by using the tool of Delphi to test them. The data were analyzed in a random simulation environment and the algorithm's performances were compared in this paper, from which the future study can benefit.

Key words: Ad Hoc network; cluster; connected dominating set

0 引言

自组网^[1]是在不需要预先部署任何基础设施的情况下, 无线终端以随机方式构成的一种分布式的自组织网络环境。自组网中的每个无线节点分享网络构造的任务, 它们以服务者和被服务对象的身份, 通过分布式点对点协同工作的方式实现数据传输功能, 因此, 自组网具有自组织、自管理和可配置的能力。由于自组网随机动态部署, 节点可移动, 无中心, 不需要任何的基础设施, 使得其在战地通信、抗震救灾、交通管理和环境监测等领域具有广阔的应用前景。

在自组网环境中, 网络拓扑扮演着重要的角色, 拓扑信息的准确性直接影响到网络中为进行传输、寻路和广播所使用的算法的性能, 并进而影响网络的连通

性、延时、吞吐量、能耗和冲突等。为了减小冲突, 提高网络容量和路由性能, 并进而达到节点能量保护和延长网络生存时间等目的, 研究人员提出了基于虚拟主干网技术的自组网拓扑控制策略^[2], 它通过分布式分簇算法把网络划分为各个独立的子网(也称为簇), 每个子网由一个簇头负责簇内成员的维护; 子网之间通过分布式网关选取策略, 选取尽可能少的节点作为网关, 连接不同的子网, 从而构造出具有分层结构的虚拟主干网, 实现网络全局连通。为了实现网络性能的优化, 通常要求分簇算法的开销尽可能小, 即通过尽可能少的节点消息交换, 在较小的时间复杂度保证下, 分布式地求解出尽可能少的主干节点形成虚拟主干网。

文中在文献[3]的基础上, 使用 Delphi 对 4 种典型的分簇算法进行仿真设计, 采集相关数据进行仿真性能的分析, 并在此基础上进行理论分析。

1 模型假设和算法描述

自组网中的无线节点等同于图论中的顶点, 分簇

收稿日期: 2006-11-14

基金项目: 国家自然科学基金(60502047); 福建省教育厅项目(JA05208); 福建工程学院科研发展基金(GY-200661)

作者简介: 郭晓莲(1978-), 女, 甘肃嘉峪关人, 助教, 研究方向为计算机网络。

所形成的虚拟主干网实质上就是图论中的极小连通支配集。为了更好地对算法加以数学描述和方便仿真设计,因此,首先给出相关的数学模型:

定义1 图 $G = (V, E, d)$ 为单位圆图(Unit Disk Graph),其中 V 和 E 分别代表顶点集和边集, d 表示以顶点集 V 中顶点为圆心的圆的半径 $\Leftrightarrow \forall v_1, v_2 \in V$,若边 $(v_1, v_2) \in E$,则 v_1, v_2 两顶点间的距离 $|v_1 - v_2| \leq d$ 。为了方便,以下将单位圆图 $G = (V, E, d)$ 简称为图 $G = (V, E)$ 。

定义2(连通支配集)对于给定的图 $G = (V, E)$,集合 S 是一个支配集当且仅当 $G - S$ 中的节点同 S 中至少一个节点相连接。而当删掉 S 中的任意一个节点时, S 不再是一个支配集,则此时称 S 为极小连通支配集。在 S 中添加有限个节点,使得 S 中的节点构成连通子图,此时 S 称为连通支配集。

基于上述定义,下面给出4种典型算法的描述:

(1)最小ID算法:最小ID算法是一种简单的分簇算法。每个节点分配唯一的ID,相邻节点中具有最小ID的节点作为簇头。这种分簇算法计算量小,实现方便,算法收敛较快。最小ID算法的簇头更新的频率较慢,维护簇所需花费的开销较小。该算法的缺点是较小ID的节点频繁作为簇头,将使这些节点能量很快耗尽,从而构成网络出现分割,该算法不具有扩展能力。

(2)最大连通度算法:该算法可以尽可能地减少簇的数目。节点之间通过交互信息掌握邻居节点的数目,即节点的度。再以最小ID式算法为基础,把具有最大度的节点备选为簇头,当度数相同时则选择ID最小的节点作为簇头。该算法的优点在于簇的数目较少,有利于减少分组时延。但是对于部分簇内成员密度较大,由于共享信道,就会造成每个用户可用的带宽非常有限,从而使整体的网络性能并不理想。

(3)基于节点权重分簇算法:每个节点按照约定计算自己的权重,并通告给其邻居,节点之间按照权重最大原则选取节点作为簇头。典型的计算权重方法是根据节点的移动速率,移动速度越快,分配的权重越低。基于移动速率的算法可以明显减少簇头更新频率。但此类算法的缺点是为了节点权重的更新,需要频繁地计算和更新权重,从而增加开销。

(4)自适应分簇算法:影响网络性能的因素是多方面的,比如带宽、能量等,因此,需要一种自适应的机制来实现联合的优化,使得节点能对网络状态的变化及时响

应,调整出最佳通讯状态。与上述三个分簇算法仅考虑单一的网络特性相比,文献[4]提出了自适应分簇算法,它把节点能量、移动速度等因素加以考虑,通过自适应机制,提高分簇的性能。

2 算法仿真设计和性能分析

2.1 移动模型

自组网仿真中,节点的移动策略有多种方案。文中选取通用的 Random Waypoint 模型^[5]。其思想是在一个给定的二维移动区域内,节点在到达一个目标位置后,随机选择下一个移动目标D,并在预先设定的最大速率(Max Speed)和最小速率(Min Speed)中随机选取一个速率,以此速率匀速前进,到达该目标D后随机等待 T 时间, T 是0与最大停留时间 Pause Time 之间的随机数。利用设定不同的最大、最小速率和最大停留时间这三个参数,可以调整出不同的移动模型来模拟自组网。

2.2 参数设置和评价参数

在对上述4个算法进行仿真测试中,设置的节点个数为30个,节点的最小移动速度为1(单位距离/单位时间),节点的最大移动速度为30(单位距离/单位时间),最大停留时间为5(单位时间),而节点的传输范围从5到80(单位距离)变化,程序运行100个单位时间后把采集到的簇头数 C 、单位时间内节点重入簇的次数 J 、单位时间内簇头构成的支配集更新的次数 U 记入数据库中。图1为仿真程序运行的界面。

2.3 性能分析

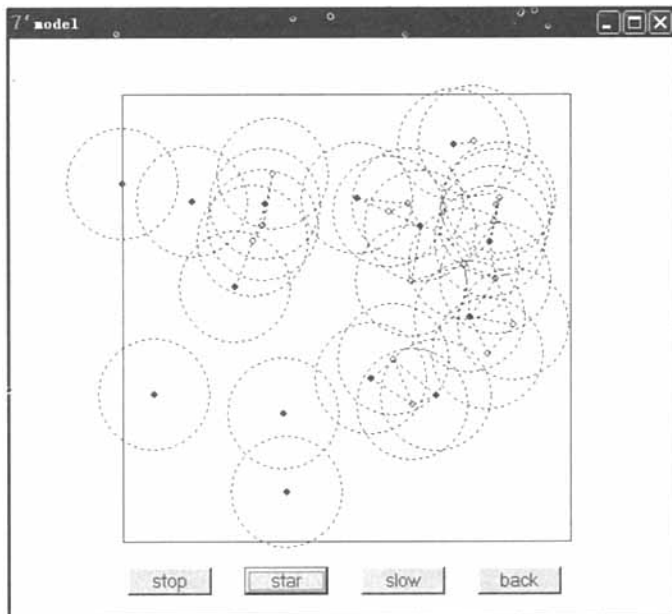


图1 仿真界面

参考文献[3]中采用的仿真性能评价参数,我们在仿真中同样考虑了分簇的簇头数 C 、单位时间内节点重入簇的次数 J 和单位时间内簇头构成的支配集更新的次数 U 。

通过设置随机种子,对程序进行了 50 次的仿真实验,并取平均值。从图 2 看出,簇头数随节点传输范围的增加而减少,并且当传输范围大于 80 后,簇头数的变化速度变慢,最终会收敛到 1。因为节点的传输范围越大,簇的覆盖范围也越大,最终一个簇头将覆盖所有节点。此外,MAXDEGREE 算法的簇头数明显低于其他算法。

再观测节点的传输范围对节点重入簇次数 J 的影响,如图 3 所示。从图中看出, J 随传输范围的增加先变大后变小。因为当节点传输范围较低时,簇数目较多,簇内节点很少,甚至只有一个簇头,此时节点离开原簇的概率较小。当传输范围增加并在传输范围为 15 ~ 25 之间时达到最大值。此时簇内节点的关联度较低,随后 J 开始下降。这是因为簇的覆盖范围逐渐增大,节点移出原簇的概率减少,该仿真结果同文献[3]在数值上有所不同,但其基本趋势是一致的。

从图 4 可以看出,在所有算法下单位时间内支配集更新次数 U 的变化规律相同:当节点传输范围很小时, U 很小,随着传输范围增加, U 也逐渐增大,当传输范围达到 5 ~ 15 之间时, U 达到最大值;当传输范围继续增加时, U 开始减小并逐渐趋向于 0。造成 U 这种变化的原因如下:当节点传输范围很小时,很大比例的节点成为簇头,并且簇内的节点数很少(很多簇只有一个簇头节点)。在这种情况下,节点移出原簇构成的统治集的覆盖范围(统治域)的概率很小,支配集更新频率很低。当传输范围增加时,簇头数减少,簇头能够覆盖所有节点的能力也随之降低,较容易移出支配集。但是随着传输范围的持续增加,尽管簇头数减少了,簇头的覆盖范围增加的更为明显,节点移出支配集的概率开始减小,并最终趋向于 0。

3 总 结

文中通过 Delphi 设计了自组网的一个典型分簇算法,即基于最小 ID、最大连接度、节点权值和自适应按需加权分簇算法。通过算法的设计,经过多次的仿真

运行,将最终结果进行平均,得到了几个算法的性能仿真结果,并进行了分析。研究也表明,分簇算法在自组网的性能优化、节能策略、密钥分配和管理等多个领域有重要的应用,我们将根据这些仿真数据开展下一阶段新的研究。

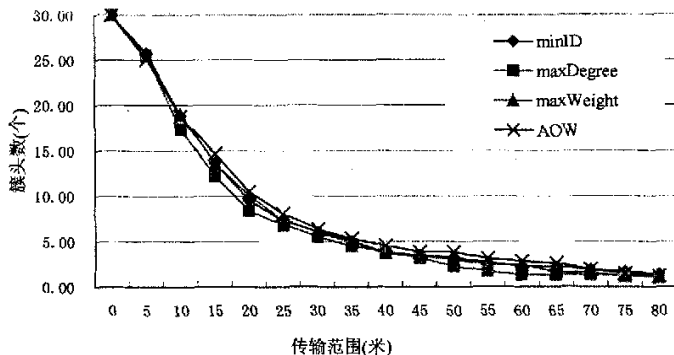


图2 簇头数随传输范围的变化

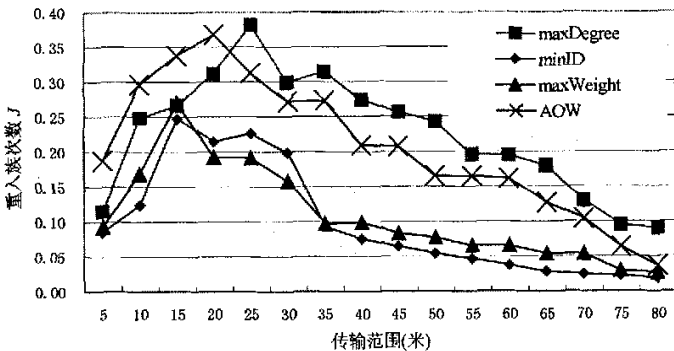


图3 重入簇次数随传输范围的变化

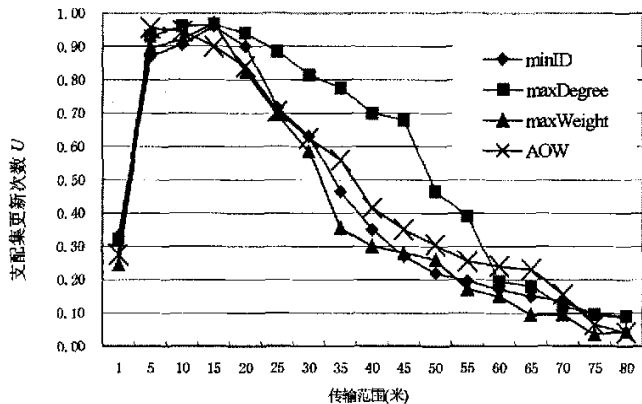


图4 支配集更新次数随传输范围的变化

参考文献:

- [1] Abolhasan M, Wysocki T, Dutkiewicz E, et al. A review of routing protocols for mobile ad hoc networks[C]//Ad Hoc Networks. [s.l.]: Elsevier Press, 2004: 1-22.
- [2] Rubin I, Behzad A, Ju H, et al. Ad Hoc Wireless Networks

(下转第98页)

表 3 基于 GA 算法的 SVM 参数选择结果

数据集	最优 C, γ	最优分类准确率	平均运行时间
banana	(496.000, 1.0633)	87.14%	836.5s
splice	(138.609, 0.1001)	61.93%	461s
tree	(277.382, 0.2485)	86.95%	1056.1s
waveform	(117.500, 0.1)	88.91%	103.5s
image	(154.500, 0.1198)	97.82%	1423.3s

群算法进行 SVM 模型选择取得了较好的效果,它能在较短的时间寻找到最优参数,而且最终所得的分类准确率与采用 GA 算法所得到的最优分类准确率相当,在大部分情况下还有所提高。

图 2 是在 tree 数据集上,利用遗传算法和蚁群算法 10 次寻参后,所得分类器分类准确率的比较图。

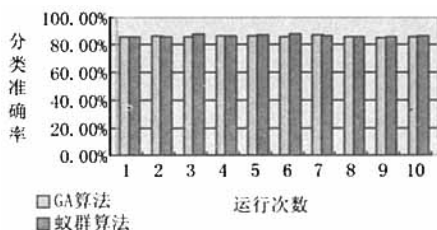


图 2 遗传算法和蚁群算法寻参比较图

从图中可以看出,10 次运行中利用蚁群算法所得的 SVM 分类器,最优分类准确率和平均分类准确率都要优于用遗传算法所得的结果。

除此之外,在实验当中还发现使用 Joachims 方法的 E_{γ, ξ_a} 作为目标函数估算错误率时较为准确,仍然以 tree 数据集为例,图 3 是在其上用蚁群算法寻找参数时的估算错误率与测试错误率(实际错误率)的曲线图。从图中可以看出两者的变化趋势基本一致。

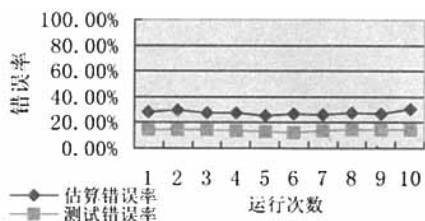


图 3 tree 数据集上预测误差与测试误差的曲线图

4 总 结

采用蚁群算法来进行 SVM 模型的参数选择。从实验的结果可以看出,蚁群算法较之遗传算法具有更好的效果,不论是分类准确率还是效率上都有一定的提高。下一步的工作是尝试采用其他的性能估计方法如:GACV, Radius - Margin bound 等作为目标函数,进行进一步的比较研究。

参考文献:

- [1] Chen Peng - Wei, Wang Jung - Ying, Lee Hahn - Ming. Model Selection of SVMs Using GA Approach[C]//Proceedings of the International Joint Conference on Neural Networks 2004. Hungary: IEEE, 2004: 2035 - 2040.
- [2] Xu P, Chan A K. Support vector machine for multi - class signal classification with unbalanced samples[C]//Proceedings of the International Joint Conference on Neural Networks 2003. Portland: IEEE, 2003: 1116 - 1119.
- [3] Xu P, Chan A K. An efficient algorithm on multi - class support vector machine model selection[C]//Proceedings of the International Joint Conference on Neural Networks 2003. Portland: IEEE, 2003: 3229 - 3232.
- [4] 黄景涛, 马龙华, 钱积新. 一种用于多分类问题的改进支持向量机[J]. 浙江大学学报: 工学版, 2004, 38(12): 1633 - 1636.
- [5] 董春曦, 饶 鲜, 杨绍全, 等. 支持向量机参数选择方法研究[J]. 系统工程与电子技术, 2004, 26(8): 1117 - 1120.
- [6] Joachims T. Estimating the generalization performance of an SVM efficiently[C]//Proceedings of the 17th International Conference on Machine Learning. San Francisco: Morgan, 2000: 431 - 438.
- [7] 贺益君, 俞欢军, 陈德钊. 基于募集机制的连续蚁群系统及其应用[J]. 浙江大学学报: 工学版, 2006, 40(5): 748 - 752.
- [8] 孙学勤, 刘 丽, 付 萍, 等. 一种连续空间优化问题的蚁群算法及应用[J]. 计算机工程与应用, 2005, 41(34): 217 - 220.
- [9] Keerthi S S. Benchmark datasets[EB/OL]. 2001 - 09 - 12. <http://guppy.mpe.nus.edu.sg/mpessk/comparison.shtml>.
- [10] Joachims T. SVMlight Support Vector Machine[EB/OL]. 2004 - 02 - 09. <http://svmlight.joachims.org>.

(上接第 94 页)

with Mobile Backbones[C]//Proceedings of IEEE International Symposium on Personal, Indoor and Radio Communications (PIMRC). Barcelona, Spain: IEEE Press, 2004: 566 - 573.

[3] 王海涛. 移动 Ad hoc 网络的簇算法及性能比较[J]. 北京邮电大学学报, 2004, 27(1): 93 - 97.

[4] Chatterjee M, Das S K, Turgut D. A weighted clustering algo-

rithm (WCA) for Ad hoc networks[C]//Proceedings of IEEE GLOBECOM 2000. San Francisco: IEEE Press, 2000: 1697 - 1701.

[5] Bettstetter C, Resta G, Santi P. The Node Distribution of the Random Waypoint Mobility Model for Wireless Ad Hoc Networks[J]. IEEE Transactions on Mobile Computing, 2003, 2(3): 257 - 269.