

模拟退火算法优化聚类头节点的 MANET 服务质量改进

鲜敏 郑翔

(四川工程职业技术学院计算机科学技术系 四川 德阳 618000)

摘要 针对移动自组织网络中移动节点和聚类头节点的通信负载问题,提出一种基于模拟退火算法优化聚类头节点的移动 Ad hoc 网络服务质量改进算法。首先,通过对移动 Ad hoc 网络聚类中存在的通信负载等问题的分析给出多目标优化问题;然后,利用模拟退火算法随机生成一个聚类头节点的有序列表;最后,结合主导解与概率决策思想解决目标优化问题,从而有效地平衡网络通信负载、功耗、聚类头节点生命周期及节点度。在移动 Ad hoc 网络上的仿真实验验证了所提算法的有效性及其可靠性。实验结果表明,所提算法在不同类型的随机场景中均取得了非常好的平衡聚类效果,相比其他几种较为先进的算法,具有更好的网络服务质量。

关键词 模拟退火算法 聚类头节点 移动自组织网络 通信负载 服务质量

中图分类号 TP399 文献标识码 A DOI:10.3969/j.issn.1000-386x.2015.04.076

CLUSTERING HEAD NODE OPTIMISED BY SIMULATED ANNEALING ALGORITHM FOR IMPROVING QUALITY OF SERVICE IN MANET

Xian Min Zheng Xiang

(Department of Computer Science and Technology, Sichuan Engineering Technical College, Deyang 618000, Sichuan, China)

Abstract For the communication load problem of mobile node and clustering head node in mobile ad hoc networks (MANETs), we propose a service quality improving algorithm of MANET, which is based on optimising the clustering head node by simulated annealing algorithm. First, we present the multiple objectives optimisation problem by analysing the communication load problem in MANET clustering. Then, we use simulated annealing algorithm to randomly generate an ordered list of clustering heads. Finally, we solve the objective optimisation problem in combination with leading solution and probabilistic decision thoughts so as to effectively balance the networks communication load, power consumption, clustering head node, lifetime and node degree. The effectiveness and reliability of the proposed algorithm has been verified by the simulation experiments on MANET. Experimental results show that the proposed algorithm achieves good balanced clustering effect in random scenes of different types, and has better networks services quality than several other quite advanced algorithms.

Keywords Simulated annealing algorithm Clustering head node Mobile ad hoc network Communication load Quality of service

0 引言

如今无线移动计算已经逐渐变成了主流,一些领先企业(如苹果)的主管们已经开始了后 PC 时代的设想^[1]。移动计算的一些应用不依赖于预先存在的基础设施,如有线网络中的路由器或者无线网络中的接入点,它们的通信通常采用移动自组织网络(MANET)^[2]。MANET 是一种自我配置的无基础设施的无线网络移动设备,如蓝牙、IEEE802.11(Wifi)、IEEE802.15.3(无线 PAN)、超宽带(UWB)等^[3]。移动自组织网络能建立一个临时移动网络以供即时通信,而无需任何固定的基础设施,它在各种工业应用领域如抢险、救灾、救济、移动会议、法律执行、战场通信等有很大的潜力^[4,5]。

已有文献表明大型 MANET 无路由方案中分层网络架构优于平坦型结构^[6,7],典型的分层架构是通过聚类结构实现的,优化选择聚类头节点是一个 NP 难的问题^[8]。文献[9]提出了一种高度聚类启发式算法,基于节点到其他节点的距离计算节点度,该方法存在一个很大的缺点,它没有在任何聚类中限制节点

数目的上限,严重影响了聚类的吞吐量和稳定性。文献[10]提出一个最低 ID 聚类启发式算法,它为每个移动节点安排一个唯一 ID,选择最低 ID 的节点作为聚类头节点,该方法的吞吐量比最高度方法好,然而,有较小 ID 的节点往往反复选为聚类头节点,这可能会很快就耗尽电池。文献[11]提出了分布式聚类算法和分布式移动自适应聚类算法,也称为节点加权启发式算法,启发式评估每个节点作为聚类头节点的适宜性,并安排对应的节点权重,然而,一个节点必须等到它所有邻居的应答才能确定是聚类头节点还是聚类成员。文献[12]中研究了负载均衡聚类,它相信一个聚类能处理的移动节点数目有个最优值,当聚类太大或太小时将相邻聚类合并在一起或者分离某个聚类。文献[13]提出一种加权聚类启发式算法,合并了聚类的各种指标,如连接到聚类头节点的节点数目、发送功率、移动性和节点电源能量,遗传算法^[14]和模拟退火算法^[15]已经改进了这个方法。

在 MANET 中任何移动节点参与的目的都是与其他节点通

信,因此不管这个移动节点是不是聚类头节点,都有一个通信工作负载。此外,每个节点的通信数量是不同的,一些节点可能产生许多网络流量,而其他节点可能仅维持非常低的通信数量,如果一个节点变成聚类头节点,它将会产生比聚类中其他节点多的通信量,但只是它自身通信工作负载最大产生的开销,如果所有其他标准是相同的,则应该选择通信工作负载小的节点作为聚类头节点,这会最小化聚类头节点节点的总体通信工作负载。所有以前的研究都没有解决这个通信工作负载方面的负载平衡问题,因为他们没有优化网络中节点的通信需求。

基于上述分析,提出了一种基于模拟退火算法改进聚类头节点的 MANET 服务算法,有效地平衡了通信工作负载、节点度、能源消耗和聚类头节点的持续时间,实验结果表明本文提出的算法对不同随机场景集能产生更平衡的聚类。

1 移动自组织网络

MANET 聚类问题目的是划分 M 为一个聚类集, $C = \{c_1, c_2, \dots, c_N\}$, 有 $c_1 \cup c_2 \cup \dots \cup c_N = M$ 。对于每个聚类 $c \in C$, $h(c)$ 指的是头节点(聚类头节点),每个节点和它的聚类头节点之间的相对距离必须在传输范围 r 之内,即:

$$\forall c \in C \quad \forall n \in c \quad dist(n, h(c)) < r \quad (1)$$

令 $k_v^M(t)$ 表示 MANET 中在时间 t 时,非聚类头节点节点 v 与其他节点的正常通信量,这个通信工作负载是节点与网络中剩余节点之间典型的信息交换。令 $k_v^C(t)$ 表示聚类 C 的聚类头节点节点 v 的通信开销, $k_v(t)$ 为节点 v 在时间 t 的总体通信工作负载。

$$k_v(t) = \begin{cases} k_v^M(t) + k_v^C(t) & \text{聚类头节点} \\ k_v^M(t) & \text{聚类成员} \end{cases} \quad (2)$$

对于每个节点 v ,保持它在过去时间窗口 T 中与其他节点平均正常通信量的轨迹,表示为:

$$k_v^M(t) = \frac{1}{T} \sum_{t'=t-T+1}^t k_v^M(t') \quad (3)$$

最小化聚类头节点的最大通信工作负载,具体来说,就是最小化下式:

$$K = \max \{ K_{h(c)}^M(t) + k_{h(c)}^C(t) \mid c \in C \} \quad (4)$$

此外,优化前面章节提到的指标,最小化所有聚类头节点的总体能源消耗:

$$Dist = \sum_{c \in C} Dist_{h(c)} \quad (5)$$

最大化所有聚类头节点的最小寿命:

$$Life = \text{Min} \{ Life_{h(c)} \mid c \in C \} \quad (6)$$

最小化所有聚类头节点的总体度的差:

$$\Delta = \sum_{c \in C} \Delta_{h(c)} \quad (7)$$

2 算法提出

2.1 SA 改进聚类头节点算法

传统的聚类算法中,通过周期性地选取类内不同的节点充当聚类头节点能有效地平衡类内节点之间的能耗,但是,现有的各种能耗平衡技术主要侧重于局部的负荷分布,不平等聚类虽然能在一定程度上减轻重负荷和轻负荷区域之间的能耗不平衡,但无法有效地利用轻负荷区域的能量。

模拟退火算法是基于 Monte Carlo 迭代求解策略的一种随机寻优算法,其出发点是基于物理模拟过程与组合优化之间的相似性,在初始温度足够高、温度下降足够慢的条件下,能以概率 1 收敛到全局最优解,由于它以某种概率接受较差点,从而具有跳出局部最优解的能力,而且模拟退火算法通用性高,易于实现。

因此,本文提出利用模拟退火算法改进聚类头节点以改进 MANET 服务质量,与主导解处理多目标优化问题的思想相结合。算法 1 给出了伪代码,第 1 行,随机生成一个聚类头节点的有序列表,它是一个解 s ,使用平均邻域的概念将剩余的移动节点安排到各自的聚类。给定一个节点 v ,使用 $\bar{N}(v)$ 表示节点 v 的邻域中的节点集:

$$\bar{N}(v) = \{v' \mid v' \in M \wedge dist(v, v') < r \wedge \overline{dist}(v, v') < r\} \quad (8)$$

其中, r 是节点 v 的传输范围, $dist(v, v')$ 是节点 v 和 v' 之间的距离, $\overline{dist}(v, v')$ 是过去时间窗口 T 的平均距离,为了安排移动节点到聚类,以顺序列表中的第一个聚类头节点开始,将在它平均邻域中的所有节点排列到这个聚类,然后进行到第二节聚类头节点,将它平均邻域中除了聚类头节点的所有节点排列到该聚类中,类似的,进行到第三个聚类头节点,直到把所有节点都排列到聚类中。

算法 1 基于模拟退火改进聚类头节点算法

```

1:  $s \leftarrow \text{RandomClusterheads}()$ 
2:  $G \leftarrow \text{GeometricMean}(s)$ 
3: repeat
4:   for  $i = 1$  to  $L$  do
5:      $s' \leftarrow \text{RandomClusterheads}()$ 
6:      $G' \leftarrow \text{GeometricMean}(s')$ 
7:     if  $s' < d^s$  then
8:        $s \leftarrow s'$ 
9:        $G \leftarrow G'$ 
10:    else
11:       $p \leftarrow [0, 1]$  中的随机数
12:      if  $p \leq e^{-\frac{|G'-G|}{T}}$  then
13:         $s \leftarrow s'$ 
14:         $G \leftarrow G'$ 
15:      end if
16:    end if
17:  end for
18:   $T \leftarrow a * T, a < 1$ 
19: until 无明显改进

```

算法 1 中第 1 行和第 5 行中的函数 $\text{RandomClusterheads}()$ 负责随机生成聚类头节点列表,大多数简单环境下,所有节点选择进入聚类头节点列表的机会是随机的,然而,有些环境下可能要求偏置的选择某些节点,例如,可能一些节点功耗较大,使得它们更适合做聚类头节点,另外一些节点可能有故障,使得它们不适合做聚类头节点。

概率决策依赖于两个解的几何平均和温度 T 之间的绝对差,初始化为 50,使用下式计算几何平均:

$$G = \sqrt[4]{K \cdot Dist \cdot \Delta \cdot \frac{1}{Life}} \quad (9)$$

2.2 算法实现过程

挑选节点 P (有足够大电池能源)运行 MANET 聚类算法,

然后广播结果到网络的其余部分,收集节点之间相对距离有如下几种方法:

方法 1 每个节点携带一台设备定期测量其他节点的距离。本地存储每个节点能保持前 N 次测量的距离轨迹,重新建聚类时,每个节点将发送距离到节点 P 计算聚类算法。

方法 2 每个节点携带一个 GPS 本地设备,定期记录它的位置,本地存储的每个节点能保持前 N 个位置的轨迹,重新建聚类时,每个节点发送它的前 N 个位置到节点 P 计算聚类算法, P 使用给定的位置计算相对距离。

方法 3 三个或更多节点(有足够大电池能源)携带一台设备定期测量其他节点的距离,它们的本地存储能保持前 N 次测量的距离轨迹,重新建聚类时,这三个(或多个)节点将发送距离到 P 计算聚类算法,节点 P 使用距离三角定位网络中每个节点的位置,然后计算各个节点之间的相对距离。

2.3 算法分析

算法比较了两个解 s 和 s' ,检查 s' 是否为主导解,否则,做出第 11 行的概率决策。这是一个多目标优化问题,例如,如果在一个目标中 s 优于 s' ,在另一个目标中 s' 优于 s ,则它们两个都不是主导解。行 7 中比较 $s' <_d s$ 确保所有目标中 s' 均等于或优于 s ,也就意味着 s' 决定 s 。本文提出的算法使用第 11 行中的概率随机挑一个选择,在此之前的研究可能使用加权成本,这会有利于其中一个目标。由于目标测量中有权重偏置,使用 *Life* 的逆使其成为一个最小化目标,配合其他三个最小化目标,几何平均可以自动调整每个目标不同的数值范围,这比以前研究中的加权算术平均要好。对于相同的温度 T ,保持第 4 行和第 17 行之间循环中的解干扰,这个循环最大重复 L 步,之后会更新 T 。

方法 1 和方法 2 提供了节点之间非常精确的相对距离,方法 3 受每个节点电源的影响最小,因为仅用三个或多个节点测量距离。运行算法执行聚类,整个 MANET 意味着到网络中节点的某种开销,如果仅一个聚类有问题,可以尝试在相同聚类内重新选择一个聚类头节点,广播聚类头节点代替聚类的其余部分,模式退火算法的运用,使得聚类算法能以概率 1 收敛全局最优解,从而更加有效地解决了目标测量权重偏置问题。

3 实验

本文对一个有 N 个节点的移动自组织网络进行了仿真模拟,对各种聚类算法进行了评估,模拟在 $500 \text{ km} \times 500 \text{ km}$ 的区域内展开,假设每对节点的概率通信量呈正态分布,每个节点 v 的通信工作负载 $k_v^M(t)$ 与它的概率通信量成正比,如果节点 v 是聚类头节点,它的附加通信工作负载 $k_v^C(t)$ 与聚类大小成正比。本模拟中,随机划分移动节点为几个组,可能有的组中节点多,有的组仅有 1 个或者 2 个节点,整个模拟过程中可随机移动所有节点,每个组中的节点都有平均速度,遵循参考速度组移动模型,而且每个节点的速度会一直随机的偏离组的平均速度。

首先研究的是相对于移动节点数目的聚类性能,节点数目从 50 个到 100 个。对于每个配置的节点数目,使用参考速度组移动模型随机生成 10 个仿真场景,在 10 个随机场景上都运行算法,10 个场景在图上的数据点是平均的。本文将提出的算法与加权聚类算法(WCA)^[13]、遗传算法^[14]和模拟退火算法^[15]进行了比较,图 1 显示了仿真过程中随着节点数目变化得到的聚类头节点平均持续时间,进行 10 组实验,取平均结果, $r =$

$100 \text{ km}, \delta = 5$, 时间 = 1000 s。

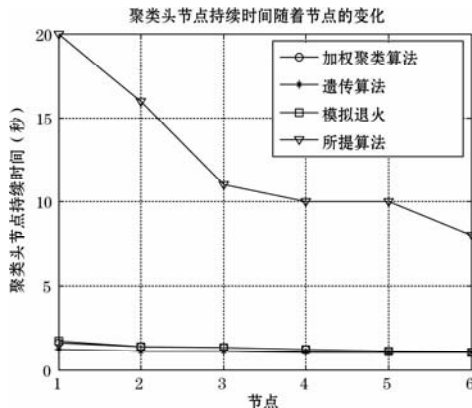


图 1 聚类头节点持续时间随着节点的变化

从图 1 可以看出,相比其他几种比较算法,本文提出的算法有更长的聚类头节点持续时间。

图 2 显示了整个仿真运行过程中聚类头节点的平均最大通信量,进行 10 组实验,取平均结果, $r = 100 \text{ km}, \delta = 5$, 时间 = 1000 s。

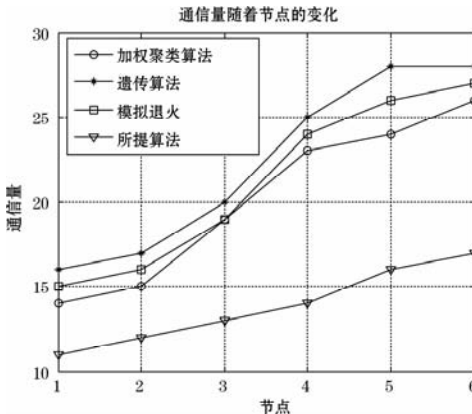


图 2 通信量随着节点的变化

从图 2 可以看出,相比其他几种比较算法,本文提出的算法聚类头节点的通信工作负载最低。

接下来的实验研究相对于组数目的聚类性能,由参考速度组移动模型建模。图 3 所示为仿真过程中随着组数目的变化得到的平均聚类头节点持续时间,实验参数 $N = 50, r = 100 \text{ km}, \delta = 5$, 时间 = 1000 s, 组数分别取 5、10、15、20、25。

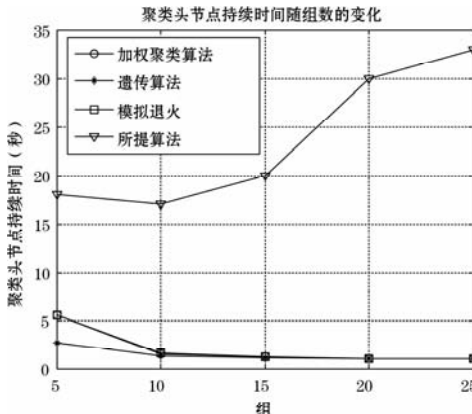


图 3 聚类头节点持续时间随组数的变化

从图 3 可以看出,相比其他几种比较算法,本文提出的算法有更长的聚类头节点持续时间。

图 4 所示为每个仿真过程聚类头节点的平均最大通信量随

组数的变化,实验参数 $N = 50, r = 100 \text{ km}, \delta = 5$, 时间 = 1000 s, 组数分别取 5、10、15、20、25。

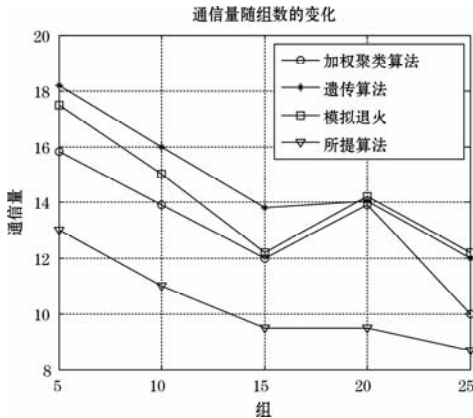


图4 通信量随组数的变化

从图4可以看出,相比其他几种算法,本文提出的算法聚类头节点的通信工作负载最低。

然后研究了相对于传输半径 r 的聚类性能,图5所示为每个移动节点随着传输半径变化得到的聚类头节点平均持续时间,进行10组实验,取平均结果, $N = 80, \delta = 5$, 时间 = 1000 s, 传输半径分别取 25、50、75、100、125、150 km。

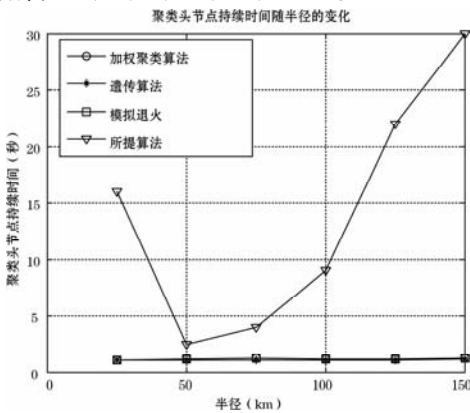


图5 聚类头节点持续时间随传输半径的变化

从图5可以看出,相比其他几种算法,本文提出的算法有较长的聚类头节点持续时间。

图6所示为每次仿真过程中聚类头节点随着传输半径变化得到的平均最大通信量,进行10组实验,取平均结果, $N = 80, \delta = 5$, 时间 = 1000 s, 传输半径分别取 25、50、75、100、125、150 km。

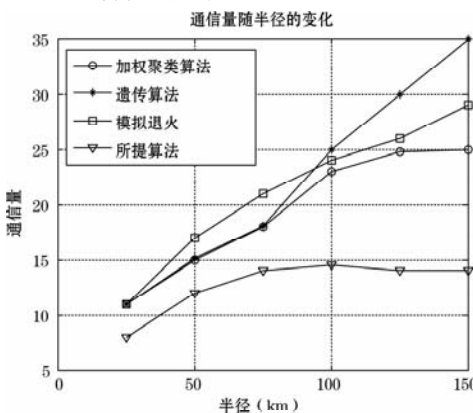


图6 通信量随传输半径的变化

从图6可以看出,相比其他几种算法,本文提出的算法聚类头节点的通信工作负载最低。

4 结 语

本文考虑了聚类成员的典型通信工作负载和移动自组织网络中聚类头节点的附加通信工作负载,提出了一个新的聚类方案,获取通信工作负载和其他聚类目标之间的平衡。实验结果表明,本文提出的聚类算法对不同随机场景集产生了有效的平衡聚类,相比其他几种较为先进的算法,取得了更好的平衡效果。未来会结合其他先进技术,进行大量的实验,进一步平衡聚类算法的各个因素。

参 考 文 献

- [1] 吴越,李建华,林闯. 机会网络中的安全与信任技术研究进展[J]. 计算机研究与发展,2013,50(2):278-290.
- [2] 夏辉,贾智平,张志勇. 移动 Ad Hoc 网络中基于链路稳定性预测的组播路由协议[J]. 计算机学报,2013,36(5):926-936.
- [3] 葛永明,朱艺华,龙胜春,等. IEEE802.11 移动自组织网络节点竞争窗口长度的概率分布[J]. 电子学报,2010,38(8):1841-1844.
- [4] 吴大鹏,武穆清,甄岩. 移动自组织网络可用带宽估计方法研究进展[J]. 通信学报,2010,31(4):103-115.
- [5] 张鹏,崔勇. 移动自组织网络路由选择算法研究进展[J]. 计算机学报,2010,37(1):10-21.
- [6] 牛晓光,崔莉,黄长城. 移动自组织网络中基于优化分簇的混合路由协议[J]. 通信学报,2010,31(10):58-67.
- [7] 何明,陈国华,赖海光,等. 物联网感知层移动自组织网可靠性评估方法[J]. 计算机科学,2012,39(6):104-106.
- [8] 刘刚,郭建立,崔刚,等. 移动自组织网络环境中负载均衡策略研究[J]. 无线电工程,2010,40(7):1-3.
- [9] 王安保,胡小明. 基于 GPS 的启发式 Ad hoc 路由算法研究[J]. 计算机应用研究,2010,27(12):4708-4710.
- [10] Baker D, Ephremides A. The architectural organization of a mobile radio network via a distributed algorithm[J]. Communications, IEEE Transactions on, 1981, 29(11):1694-1701.
- [11] 王博,黄传河,杨文忠. TRQ: Ad hoc 网络中基于 QoS 的可信路由算法[J]. 小型微型计算机系统,2011,32(7):1249-1254.
- [12] 甄岩,武穆清,吴大鹏,等. MANET 多路径负载均衡方法[J]. 北京邮电大学学报,2010,33(2):64-68.
- [13] 霍金海,王钺,徐赞新,等. 基于负载和优先级的 MANET 优化策略[J]. 清华大学学报:自然科学版,2012,52(9):1270-1274.
- [14] 朱莉,沈未名,李锐,等. 利用遗传算法的网络 GIS 集群服务器动态负载均衡算法[J]. 武汉大学学报:信息科学版,2011,36(6):721-725.
- [15] 张国印,李军. 移动对等网络覆盖网[J]. 软件学报,2013,24(1):139-152.

(上接第 316 页)

- [11] 张宇飞,苏锦海,张传富. 按空间网络中基于身份的 TLS 协议设计与仿真[J]. 计算机工程与设计,2013,34(7):2517-2521.
- [12] 马克,辛阳,李忠献. 基于 RSA 算法的 TLS 协议在智能卡上的设计与实现[C]//第八届中国通信学会学术年会论文集. 武汉,2011:534-540.
- [13] Blake-Wilson S, Nystrom M, Hopwood D, et al. TLS Extension [EB/OL]. 2002. <http://www.ietf.org/internet-drafts/draft-ietf-tls-extensions-05.txt>.