

自组网中一种增强分群稳定性的分群算法

吴 静¹ 鞠宏军² 田立勤¹ 赵蕴龙³

(青海师范大学计算机学院 西宁 810008)¹ (华北科技学院计算机学院 燕郊 101601)²

(哈尔滨工程大学计算机科学与技术学院 哈尔滨 150001)³

摘 要 在自组网中,MSWCA 是考虑运动相关性的分群算法中在分群稳定性度量方面考虑得最全面的典型算法。针对 MSWCA “只考虑群内稳定性,而忽视群间稳定性”的问题,提出一种增强分群稳定性的分群算法(CAECS)。该算法基于移动预测思想,综合考虑群内稳定性、群间稳定性和分群优化,通过调节权值使算法适用于不同的场景。仿真结果表明:当权值取值适当时,CAECS 在分群稳定性和分群维护开销等性能指标上都要优于 MSWCA。

关键词 通信技术,自组网,分群算法,稳定性,移动预测

中图分类号 TN925 文献标识码 A

Clustering Algorithm which Enhances Clusters' Stability in Ad hoc Networks

WU Jing¹ JU Hong-jun² TIAN Li-qin¹ ZHAO Yun-long³

(College of Computer, Qinghai Normal University, Xining 810008, China)¹

(College of Computer, North China Institute of Science and Technology, Yanjiao 101601, China)²

(College of Computer Science and Technology, Harbin Engineering University, Harbin 150001, China)³

Abstract In Ad hoc networks, MSWCA, which makes the most comprehensive consideration on clusters' stability, is a typical algorithm of the motion-correlation considered clustering algorithm. Aiming at MSWCA's problem that "it only considers intracluster stability, but neglects intercluster stability", a clustering algorithm which enhances clusters' stability (CAECS) was proposed. Based on mobility prediction idea, CAECS considers intracluster stability, intercluster stability and clusters' optimization comprehensively, and it's adapted to different scenarios by adjusting weights. The simulation shows that CAECS outperforms MSWCA on clusters' stability and clusters' maintenance overheads.

Keywords Communication technology, Ad hoc networks, Clustering algorithm, Stability, Mobility prediction

自组网可扩展性问题的主要解决方案是分级结构,而构建分级结构的方法是采用分群算法。目前,可以根据不同的分群策略将分群算法分为基于支配集的分群算法、考虑低维护开销的分群算法、考虑运动相关性的分群算法、考虑群尺寸的分群算法、考虑能量效率的分群算法、考虑负载均衡的分群算法和基于组合度的分群算法等 7 大类^[1],其中,有利于提高分群稳定性的是考虑运动相关性的分群算法。在对考虑运动相关性的分群算法的国内外研究现状进行深入分析的基础上,本文将该类分群算法进一步归纳总结为如下 3 大类:只考虑运动速度或运动位置的分群算法,如 MOBIC^[2]、WBA-CA^[3]、CEMCA^[4]、MCFA^[5]、MPBC^[6]、CBMD^[7]等;将运动速度和运动位置进行简单加权组合的分群算法,如 DWCA^[8]、文献[9]提出的高效稳定加权组合的分布式分群算法和文献[10]描述的分群算法;综合考虑运动速度和运动位置的分群算法,如 MSWCA^[11]、EMSWCA^[12]、MEACA^[13]。其中,综合考虑运动速度和运动位置的分群算法在分群稳定性度量方面

考虑得最全面,而该类算法中具有代表性的典型算法是 MSWCA。

分群稳定性包括群内稳定性和群间稳定性,它们都会对分群的性能产生重要影响,同时也应该得到关注。但是,MSWCA 只是通过群首与所有邻节点的连接保持时间累加和这个因素对群内稳定性进行了考虑,却忽视了对群间稳定性的考虑。针对上述问题,本文提出一种增强分群稳定性的分群算法 CAECS(Clustering Algorithm which Enhances Clusters' Stability)。

本文第 1 节描述了 CAECS 的基本思想;第 2 节描述了基于上述基本思想的 CAECS 分群算法;第 3 节对 CAECS 和 MSWCA 的算法性能进行了仿真比较分析;最后对全文进行总结。

1 CAECS 的基本思想

CAECS 涉及到一些概念,本文首先对它们进行了系统的

到稿日期:2012-08-16 返修日期:2012-11-22 本文受国家自然科学基金项目(61003235),973 计划项目(2011CB311809),中央高校基本科研业务费资助项目(DX1208B,3142013098),青海省自然科学基金项目(2012-Z-935Q,2012-Z-932Q,2012-N-525),哈尔滨市科技检才研究专项(2011RFQXG012)资助。

吴 静(1980—),女,博士,副教授,主要研究方向为移动自组网、无线传感器网络、无线 MESH 网络,E-mail:99061632@163.com;鞠宏军(1973—),男,硕士,副教授,主要研究方向为计算机网络;田立勤(1970—),男,博士后,教授,主要研究方向为无线传感器网络、信息监测、网络性能评价、网络服务质量、网络安全;赵蕴龙(1975—),男,博士,副教授,主要研究方向为计算机体系结构、计算机网络、移动计算、可信计算等。

定义和相关定量描述,然后再对该算法的基本思想加以描述。

1.1 相关概念定义和定量描述

下面,对 CAECS 涉及到的一些概念进行了如下定义和相关定量描述。除了连接保持时间(即定义 3)是已有的概念之外,其它概念均由本文提出。

定义 1(虚拟群) 是指在分群之前,每个节点都以其自身为群首,以其所有邻节点为群成员构建一个虚拟的群。

定义 2(总体连接度) 是指在群内或者群间,所有连接的总数。本文研究了虚拟群内总体连接度 VIATC(Virtual Intracluster Total Connectivity)和虚拟群间总体连接度 VIETC(Virtual Intercluster Total Connectivity),将它们作为虚拟群创建者 i 的属性。

定义 3(连接保持时间) 连接保持时间 LET(Link Expiration Time)是在移动预测思想中提出的,是根据节点的有效传输距离、位置和速度等来预测节点之间的连接所能维持的时间。其计算方法如式(1)所示:

$$LET(i, j) = \frac{-(ab+cd) + \sqrt{(a^2+c^2)r^2 - (ad-bc)^2}}{a^2+c^2} \quad (1)$$

式中, $a = v_i \cos\theta_i - v_j \cos\theta_j$, $b = x_i - x_j$, $c = v_i \sin\theta_i - v_j \sin\theta_j$, $d = y_i - y_j$, r 是 i 和 j 的有效传输距离, (x_i, y_i) 和 (x_j, y_j) 分别为 i 和 j 的坐标, v_i 和 v_j 分别为 i 和 j 的平均速度大小, θ_i 和 θ_j ($0 \leq \theta_i, \theta_j \leq 2\pi$) 分别为 i 和 j 的平均速度方向^[14]。当 $v_i = v_j$, $\theta_i = \theta_j$ 时,即 i 和 j 两个节点的运动方向一致, $LET(i, j) \rightarrow \infty$, 此时认为两个节点可以一直保持连接状态;如果 $LET(i, j) < 0$, 则认为两个节点不可连接;如果 $LET(i, j) > 0$, 则认为两个节点可以连接, $LET(i, j)$ 值越大,表示两个节点的连接保持时间越长^[14]。

定义 4(总体连接保持时间) 是指在群内或者群间,所有连接的保持时间总和。本文研究了虚拟群内总体连接保持时间 VIATLET(Virtual Intracluster Total LET)和虚拟群间总体连接保持时间 VIETLET (Virtual Intercluster Total LET),将它们作为虚拟群创建者 i 的属性。

定义 5(平均连接保持时间) 是指在群内或者群间,所有连接保持时间的平均值,它是总体连接保持时间与总体连接度的商。本文研究了虚拟群内平均连接保持时间 VIAALET (Virtual Intracluster Average LET)和虚拟群间平均连接保持时间 VIEALET (Virtual Intercluster Average LET),将它们作为虚拟群创建者 i 的属性。

定义 6(属性优越度) 是指节点的属性在节点及其所有邻节点构成的集合中的优越程度,是由节点的某个属性值与节点及其所有邻节点相应属性的平均值进行比较得到的。本文研究了 VIAALET 的优越度 $SD_{VIAALET}$ (VIAALET Superiority Degree)、VIEALET 的优越度 $SD_{VIEALET}$ (VIEALET Superiority Degree)和 VIATC 的优越度 SD_{VIATC} (VIATC Superiority Degree)等 3 个属性优越度,将它们也作为虚拟群创建者 i 的属性。

定义 7(总体优越度) 是指节点在节点及其所有邻节点构成的集合中的总体优越程度,它由若干属性优越度综合而成。它基于组合度分群策略,根据多个属性优越度在不同场景中的重要程度设置相应的权值,并对权值进行综合控制。本文研究的总体优越度 TSD (Total Superiority Degree)由

$SD_{VIAALET}$ 、 $SD_{VIEALET}$ 和 SD_{VIATC} 3 个属性优越度综合而成,将 TSD 作为虚拟群创建者 i 的综合属性,其计算方法如式(2)所示。

$$TSD(i) = w_1 SD_{VIAALET}(i) + w_2 SD_{VIEALET}(i) + w_3 SD_{VIATC}(i) \quad (2)$$

式中, $0 \leq w_i \leq 1$ 且 $\sum_{i=1}^3 w_i = 1$ 。

1.2 算法基本思想描述

CAECS 的目标是力求提高分群稳定性,这也有利于降低分群维护开销;为了能进一步地改善网络性能,该算法同时考虑了分群优化的问题,它只是起到一定的辅助作用,本文不对其做过多研究。该算法对群内稳定性、群间稳定性和分群优化等问题给予了综合考虑,通过调节权值使算法适用于不同的场景。

本文将虚拟群作为研究对象,对虚拟群设立了多项评价指标,以虚拟群的综合评价指标数据来指导最终的分群。本文基于移动预测思想,对节点之间的连接保持时间 LET 进行预测,用虚拟群内平均连接保持时间 VIAALET 来反映群内稳定性,用虚拟群间平均连接保持时间 VIEALET 来反映群间稳定性,用虚拟群内总体连接度 VIATC 来反映分群优化程度。为了将上述 3 个属性统一到同一数量级上,本文引入属性优越度的概念,每个节点都有 $SD_{VIAALET}$ 、 $SD_{VIEALET}$ 和 SD_{VIATC} 3 个属性优越度;为了将上述 3 个属性优越度进行综合,又引入了总体优越度的概念,每个节点都有一个总体优越度 TSD。本文将 $SD_{VIAALET}$ 、 $SD_{VIEALET}$ 和 SD_{VIATC} 作为虚拟群的 3 项评价指标,将 TSD 作为虚拟群的综合评价指标,以最大 TSD 为标准选取群首进行分群,若节点的 TSD 相同,则选择 id 较小的节点作为群首进行分群。

2 CAECS 分群算法

本节描述了基于上述基本思想的 CAECS 分群算法,其中包括分群算法描述及其群维护策略的描述。

2.1 分群算法描述

在算法基本思想的基础上,本文接着对分群算法 CAECS 进行了设计,将算法的执行过程分为初始化过程和分群过程两个步骤进行。

(1) 初始化过程

初始化过程主要是完成 TSD 计算的过程,为分群提供依据。节点之间需要交换 3 轮控制包才能完成 TSD 的计算,最后再交换 1 轮控制包以获取邻节点的 TSD 信息。

(2) 分群过程

分群过程利用初始化过程计算的 TSD,以最大 TSD 为标准选取群首进行分群,若 TSD 相同,则选择 id 较小的节点作为群首进行分群。在分群过程开始和接收到各类消息之后的操作步骤分别描述如下。

(a) 节点 i 在分群过程开始的操作步骤如下:

$\Omega = N(i)$; // Ω 记录未加入群的邻节点集合

if TSD(i) = $\max_{j \in N(i) \cup \{i\}}$ TSD(j)

{ if { $\exists j | TSD(j) = TSD(i) \wedge j \in N(i) \}$ } = \emptyset

{

//如果不存在邻节点与 i 的 TSD 相同,则 i 创建群 C_i

i creates cluster C_i ;

```

i.member=creator;
i sends Cluster(i) to N(i);
}
else {
//否则,存在邻节点与 i 的 TSD 相同
if i.id=

$$\min_{j \in \{k | TSD(k) = TSD(i) \wedge k \in N(i)\} \cup \{i\}}$$
 j.id
{
//如果 i 在上述邻节点中 id 最小,则 i 创建群 Ci
i creates cluster Ci;
i.member=creator;
i sends Cluster(i) to N(i);
}
}
}
return;
(b)节点 i 接收到节点 j 的 Cluster(j)消息之后的操作步骤如下:
neighbor(j).member=creator;
i joins to Cj;
i.member=member;
i sends Join(i,Cj) to N(i);
//如果邻节点都已加入群,则 i 结束参与分群
 $\Omega \leftarrow \Omega - \{j\}$ ;
if  $\Omega = \Phi$ 
{ Finish clustering; }
return;
(c)节点 i 接收到节点 j 的 Join(j,Ck)消息之后的操作步骤如下:
neighbor(j).member=member;
if i has not joined to any cluster
/* 如果 i 还没有加入任何群,若所有 TSD 比 i 大或者所有 TSD 与 i 同为最大且 id 比 i 小的邻节点都作为成员加入群,则 i 创建群 Ci */
{ canCluster=true;
for p ∈ N(i) {
if (TSD(p) > TSD(i) ∧ p.member ≠ member) || [TSD(p) = TSD(i) ∧ TSD(i) =  $\max_{q \in N(i) \cup \{i\}}$  TSD(q) ∧ p.id < i.id ∧ p.member ≠ member]
{ canCluster=false; }
}
if canCluster=true
{ i creates cluster Ci;
i.member=creator;
i sends Cluster(i) to N(i);
}
}
 $\Omega \leftarrow \Omega - \{j\}$ ;
if  $\Omega = \Phi$ 
{ Finish clustering; }
return;

```

2.2 群维护策略描述

为了尽量维持分群结构的稳定,本文为 CAECS 设计了一种群维护策略,将其分为分群重构、分群调整和群首轮换 3 个部分。其中,分群重构是指当群首无法继续下去时,它放弃自己的群首身份,该群内所有节点成员将重新选择群首来重

新进行分群;分群调整是指节点成员的离开和加入都会引起分群内部成员的调整;群首轮换是指在群首可以与节点成员进行正常通信的情况下,为了延长整个网络的生存时间,将群首进行轮换,以更均衡地使用网络内各节点的能量。

3 算法性能的仿真比较分析

分群调整次数是指由分群重构、分群调整和群首轮换等事件引起的群内调整和群间调整的次数之和,它在一定程度上反映了分群稳定性;分群维护开销是指分群形成和分群维护的总开销。本文采用 NS2 仿真软件对 CAECS 和 MSWCA 的分群调整次数和分群维护开销等性能指标进行了分析,仿真所用参数如表 1 所列。

表 1 仿真参数

参数名称	参数取值
物理区域(米 ²)	1000×1000
传输范围(米)	150
仿真时间(秒)	600
移动模型	Random Way Point 模型
暂停时间(秒)	5
最小速度(米/秒)	0
最大速度(米/秒)	1,5,10,15,20,25
网络节点数目	100
MAC 协议	IEEE 802.11
CAECS 权值	(w ₁ =0.4, w ₂ =0.4, w ₃ =0.2), (w ₁ =0.5, w ₂ =0.3, w ₃ =0.2)
MSWCA 权值	(w ₁ =0.2, w ₂ =0, w ₃ =0.8)

在保持网络环境的其它参数不变的情况下,通过改变节点最大速度这项参数来分别评测分群调整次数和分群维护开销等性能指标,其仿真结果如图 1 和图 2 所示。

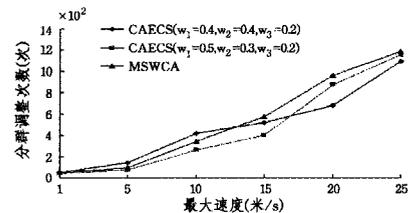


图 1 最大速度对分群调整次数的影响

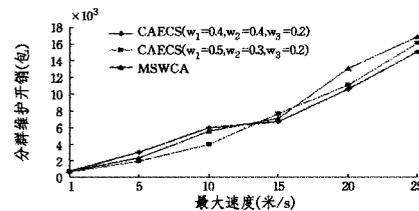


图 2 最大速度对分群维护开销的影响

在图 1 和图 2 中,与 MSWCA 相比,当 CAECS 取第一组权值时,从总体上来讲,性能上不具有什么优势;而当 CAECS 取第二组权值时,则表现出更低的分群调整次数和分群维护开销,具有比较明显的优势。这是因为:MSWCA 只考虑了群内稳定性,而 CAECS 在群内稳定性和群间稳定性之间进行均衡,当调节的权值适合特定的场景时,一定能比 MSWCA 表现出较低的分群调整次数和分群维护开销。这里,第二组权值比第一组权值更适合这个场景,故能表现出较好的效果,如果继续调节权值,可能会得到更好的效果。

仿真结果表明:(1)当权值取值适当时,CAECS 减少了分

群调整次数,这说明 CAECS 提高了分群稳定性;(2)当权值取值适当时,CAECS 降低了分群维护开销。以上两点说明:当权值取值适当时,CAECS 要优于 MSWCA。

结束语 在自组网中,考虑运动相关性的分群算法有利于提高分群稳定性,而 MSWCA 是该类分群算法中在分群稳定性度量方面考虑得最全面的典型算法。本文针对 MSWCA 存在的“只考虑群内稳定性,而忽视群间稳定性”的问题,提出一种增强分群稳定性的分群算法(CAECS),它以提高分群稳定性为目标,这也有利于降低分群维护开销。该算法基于移动预测思想,综合考虑群内稳定性、群间稳定性和分群优化,通过调节权值使算法适用于不同的场景。本文首先描述了 CAECS 的基本思想,然后描述了基于上述基本思想的分群算法,最后采用 NS2 仿真软件对 CAECS 和 MSWCA 的算法性能进行了仿真比较分析。仿真结果表明:当权值取值适当时,CAECS 在分群稳定性和分群维护开销等性能指标上都要优于 MSWCA。

参 考 文 献

- [1] 陈嘉宁. 基于备份的移动自组织网络分簇策略研究[D]. 长沙: 湖南大学, 2007
- [2] Basu P, Khan N, Little T D C. A Mobility Based Metric for Clustering in Mobile Ad Hoc Networks[C]//2001 International Conference on Distributed Computing Systems Workshop, 2001. 2001:413-418
- [3] Dhurandher S K, Singh G V. Stable Clustering with Efficient Routing in Wireless Ad Hoc Networks[C]//2nd International Conference on Communication Systems Software and Middleware, 2007. 2007:1-12
- [4] Tolba F D, Magoni D, Lorenz P. Connectivity, Energy and Mobility Driven Clustering Algorithm for Mobile Ad Hoc Networks [C] // IEEE Global Telecommunications Conference, 2007. 2007:2786-2790
- [5] Torkestani J A, Meybodi M R. A Mobility-based Cluster Formation Algorithm for Wireless Mobile Ad-Hoc Networks[J]. Computer Science, 2011, 14(4):311-324
- [6] Zhong Z D, Zhao D M. MPBC: A Mobility Prediction-based Clustering Scheme for Ad Hoc Networks[J]. IEEE Transactions on Vehicular Technology, 2011, 60(9):4549-4559
- [7] Hussein A R, Yousef S, Al-Khayatt S, et al. An Efficient Weighted Distributed Clustering Algorithm for Mobile Ad Hoc Networks[C]//2010 International Conference on Computer Engineering and Systems (ICCES), 2010. 2010:221-228
- [8] Choi W, Woo M. A Distributed Weighted Clustering Algorithm for Mobile Ad Hoc Networks[C]//Advanced International Conference on Telecommunications-International Conference on Internet and Web Applications and Services, 2006. 2006:73
- [9] Hwang Y C, Jeong Y S, Lee S H, et al. Advanced Efficiency and Stability Combined Weight based Distributed Clustering Algorithm in MANET[C]//Future Generation Communication and Networking, 2007:478-483
- [10] Kawai Y, Sasase I. A Stable Clustering Scheme by Prediction of the Staying Time in a Cluster for Mobile Ad hoc Networks[C]//Proceedings of 14th Asia-Pacific Conference on Communications, 2008. 2008:1-5
- [11] 黄卫红, 李仁发, 彭献武. 基于移动保持时间的无线自组网分簇算法[J]. 网络与通信, 2007, 23(3):95-96, 119
- [12] Tao Y, Wang J, Wang Y L, et al. An Enhanced Maximum Stability Weighted Clustering Algorithm in Ad Hoc Network[C]//4th International Conference on Wireless Communications, Networking and Mobile Computing, 2008. 2008:1-4
- [13] Xu Y, Wang W Y. MEACA: Mobility and Energy Aware Clustering Algorithm for Constructing Stable MANETs[C]//Military Communications Conference, 2006. 2006:1-7
- [14] 蒋毅, 史浩山. 一种基于移动预测的自适应 Ad Hoc 网络分簇算法[J]. 计算机科学, 2007, 34(3):28-29
- [9] 李文, 郭立, 袁红星, 等. 一种高阶平滑表面并行提取方法[J]. 中国科学院研究生院学报, 2012, 29(2):251-256
- [10] NVIDIA Corporation. CUDA programming guide 4.0 [EB/OL]. <http://www.developer.nvidia.com/>, 2011-06-27
- [11] Cremeters D, Kolev K. Multiview stereo and silhouette consistency via convex functionals over convex domains [J]. IEEE Transactions on pattern analysis and machine intelligence, 2011, 33(6):1161-1174
- [12] NVIDIA Corporation. CUDA SDK C Samples[EB/OL]. <http://developer.nvidia.com/cuda-cc-sdk-code-samples>, 2012-06-27
- [13] Culess B, Levoy M. A volumetric method for building complex models from range images [C] // Proceedings of ACM SIGGRAPH, 1996. New Orleans, LA, USA: ACM Press, 1996:303-312
- [14] Yasutaka F, Jean P. 3D Photography Dataset [EB/OL]. <http://www.cs.washington.edu/homes/furukawa/research/mview/index.html>, 2011-06-27
- [15] Steve S, Brian C, James D, et al. The multi-view stereo evaluation[EB/OL]. 2011-06-27

(上接第 31 页)

- [2] 梁秀霞, 张彩明. 拓扑结构正确的三线性插值曲面的三角片逼近[J]. 计算机研究与发展, 2006, 43(3):528-535
- [3] Lopes A, Brodlie K. Improving the robustness and accuracy of the marching cubes algorithm for isosurfacing[J]. IEEE Transactions on Visualization and Computer Graphics, 2003, 9(1):16-29
- [4] 薛强, 蔡文立, 石教英. Marching Boxes: 一个多精度等值面抽取算法[J]. 计算机辅助设计与图形学学报, 1998, 10(1):7-14
- [5] 朱经纬, 蒙培生, 王乘. 一种改进的 MC 算法[J]. 中国图象图形学报, 2008, 13(7):1359-1366
- [6] 吕理伟, 顾耀林. 移动立体体算法的三重线性插值研究[J]. 计算机工程与应用, 2005, 32:41-44
- [7] Lempitsky V, Boykov Y. Global optimization for shape fitting [C]//Proceedings of Computer Vision and Pattern Recognition, 2007. Minneapolis, USA: IEEE Press, 2007:1-8
- [8] Li Wen, Guo Li, Yuan Hong-xing, et al. Parallel implementation and optimization of the Sebvhos algorithm[J]. Journal of Electronics, 2011, 28(3):277-283