

异构 Ad hoc 网络骨干网络的建立与性能分析

郭攀红¹ 杨 扬¹ 李新友²

(北京科技大学信息工程学院 北京 100083)¹ (国家信息中心 北京 100045)²

摘 要 目前针对虚拟骨干网的研究都是基于最小连通支配集(MCDS)的方式,其目的是最小化骨干网中的节点个数,但没有考虑到骨干节点的实际通讯能力。这样,在异构 Ad hoc 网络中一些低性能骨干节点可能成为整个网络的性能瓶颈。充分考虑到 Ad hoc 网络的节点异构性问题,提出了基于高性能节点的最小连通支配集算法(MCDS-HNP),同时给出了算法的逼近度证明,并通过仿真试验表明在与 MCDS 算法相似的骨干节点个数的情况下,MCDS-HNP 算法能获得更好的骨干网性能。

关键词 Ad-hoc 网络,异构网络,骨干网,连通支配集,无线网络

Backbone for Heterogeneous Ad hoc Networks and their Performance Analysis

GUO Pan-hong¹ YANG Yang¹ LI Xin-you²

(School of Information Engineering, University of Science and Technology Beijing, Beijing 100083, China)¹

(State Information Center, Beijing 100045, China)²

Abstract Most of existing work are based on the concept of minimum connect dominating set(MCDS), which is only target for minimizing the number of backbone nodes, but not take the real characteristics of nodes into account when constructing the backbone. In the way, some low performance nodes could be the bottleneck of the backbone. Selecting more capable nodes as candidates to construct a high performance backbone, a minimum connected dominating set with high node performance(MCDS-HNP) algorithm was proposed, and the approximation ratio of the proposed algorithm was also presented. The simulation results demonstrate that the MCDS-HNP algorithm achieves better backbone performance while maintaining approximately the same backbone size.

Keywords Ad hoc networks, Heterogeneous networks, Network backbone, Connected dominating set, Wireless networks

1 引言

无线 Ad hoc 网络可以在没有网络基础设施的情况下,通过移动无线设备的多跳转发能力和自组织能力,搭建一个端到端的无线通信网络。这种网络不需要网络基础设施的支持和搭建的快速性与灵活性,因此适合应用在紧急救灾和敌方环境下的军事用途中。在目前的路由协议中,针对网络所采用的逻辑拓扑结构形式,路由协议可以分为平面型和层次型。层次型路由由于信息分层次转发,需要全局传播的路由信息较少,降低了大型网络的存储要求和通信负载,因此有更好的可扩展性,越来越受到研究学者的关注^[1]。目前,基于骨干网的体系结构^[2]已成为 Ad hoc 网络中层次型路由的典型技术。在搭建 Ad hoc 骨干网络时,大多采用图论中的最小连通支配集(minimum connected dominating set, MCDS)^[4,5]的方式。最小连通支配集是从基于支配集的概念衍生出来的。一个图的支配集定义为:图中任一节点隶属于该集合,或者至少与集合中的某一节点相邻,则称该集合为支配集。最小连通支配集定义为:支配集中任意两点相互连通,且所需要的节点个数最小。最小连通支配集节点称为骨干节点(backbone node,

BN),连接这些骨干节点构成网络中的一棵树,该树即为网络中的骨干网。基于最小连通支配集的方法能够有效地找出网络拓扑中的关键节点,使得其它节点可以单跳访问到这些关键节点,并使得所需要的关键节点个数最小。因此,本文同样基于最小连通支配集来构建 Ad hoc 网络中的骨干网。

目前在大多数 Ad hoc 网络应用中,普遍存在节点的异构性。而在现有的基于最小连通支配集来构建骨干网的方案中^[6],都是将最小化骨干节点的个数作为唯一目标,并没有考虑到骨干节点的性能。在实际应用中,如果某些低性能节点被选作骨干节点,该骨干网反而成为整个网络的瓶颈。本文将节点的性能考虑进来,提出了 MCDS-HNP(minimum connected dominating set with high node performance)算法。算法将节点的性能离散化为若干个等级,所选的骨干节点的性能等级大于或等于与之相邻的所有节点,并最小化所选的骨干节点的个数。虽然 MCDS-HNP 算法需要较多的骨干节点,但整体节点性能比 MCDS 算法要高。

在第 2 节中,将给出符号定义和问题的形式化定义;第 3 节对算法进行描述和分析;第 4 节对仿真结果和性能进行比较;最后总结全文。

2 符号定义与问题描述

2.1 符号定义

文中使用如下符号定义。对欧几里得平面上的两点 $x, y, [x, y]$ 表示连接该两点的线段, $\|xy\|$ 表示两点间的距离。给定 V 为平面上所有节点的集合 $\{v_1, v_2, \dots, v_n\}$ 。BN 为一组从 V 中选取的骨干节点, 记为 $\{v_{B_1}, v_{B_2}, \dots, v_{B_k}\}$, 其中 $\{B_1, B_2, \dots, B_k\} \subseteq \{1, 2, \dots, n\}, k \leq n$; 未被选取的节点 $NBN(v_i)$ 为普通节点, 即 $v_i \in V - BN$ 。假设节点设备使用全向天线模型, 即每个节点的通信覆盖区域是半径为 R 的圆, 两节点的通信只有当它们的距离在射程长度 R 之内才能发生。每个节点 (v_i) 有一个相应的权值 Q_i , 表示该节点的性能级别。本文目的是找出权值较大的点作为骨干节点, 并尽量使得所选的节点个数最小。整个网络拓扑可以表示成一个无向图 $G=(V, E)$, 其中 V 为包含所有节点的顶点集合, E 为图中的无向边集合。若图中两节点 v_i, v_j 之间的距离不超过节点的射程距离, 它们之间存在一条无向边, 即 $\|v_i v_j\| \leq R$ 。如果节点 $v_i, v_j \in V$ 且 $(v_i, v_j) \in E$, 称 v_i 和 v_j 互为邻居节点。 $N(v_i)$ 表示节点 v_i 的所有邻居节点的集合, $d_G(v_i)$ 表示节点 v_i 在图 G 中的度。 Δ 表示所有节点最大的度, 即 $\Delta = \max_{v_i \in V} d_G(v_i)$ 。

2.2 问题描述

在本文中, 用 MDS 表示最小支配集问题, HNP-MDS 表示较高节点性能的最小支配集问题, MCDS-HNP 表示较高节点性能的最小连通支配集问题。

问题 MDS: 给定图 $G(V, E)$, 其中 V 为顶点集, E 为图中无向边集合。给定节点的射频距离 R , 满足 $E = \{(v_i, v_j) | v_i, v_j \in V, \|v_i v_j\| \leq R\}$ 。MDS 问题为从 V 中找出最小的一组顶点子集 X , 满足 $\forall v_i \in V$, 有 $v_i \in X$ 或 $\exists v_j \in X, v_i \in N(v_j)$ 。

问题 MDS-HNP: 给定图 $G(V, E)$, 其中 V 为顶点集, E 为图中无向边集合。给定节点的射频距离 R , 满足 $E = \{(v_i, v_j) | v_i, v_j \in V, \|v_i v_j\| \leq R\}$ 。MDS-HNP 问题为从 V 中找出最小的一组具有较高性能的顶点子集 X , 满足

$$\forall v_i \in V, \text{有 } v_i \in X \text{ 或 } \exists v_j \in X, v_i \in N(v_j)$$

$$\forall v_i \in V - X, \exists v_j \in X, \text{满足 } v_i \in N(v_j),$$

且 $Q_j \in \max_{v_k \in N(v_j)} (Q_k)$ (即所选节点的性能为它所支配的节点的所有邻居节点的最大值)。

问题 MCDS-HNP: 给定图 $G(V, E)$, 其中 V 为顶点集, E 为图中无向边集合。给定节点的射频距离 R , 满足 $E = \{(v_i, v_j) | v_i, v_j \in V, \|v_i v_j\| \leq R\}$ 。MCDS-HNP 问题为从 V 中找出最小的一组具有较高性能的顶点子集 X , 满足:

$$\forall v_i \in V, \text{有 } v_i \in X \text{ 或 } \exists v_j \in X, v_i \in N(v_j)$$

$$\forall v_i \in V - X, \exists v_j \in X, \text{满足 } v_i \in N(v_j)$$

且 $C_j \in \max_{v_k \in N(v_j)} (C_k); \forall v_i \in X, \exists v_j \in X, \text{满足 } v_j \in N(v_i)$ (即所选的子集中的节点是连通的)。

3 算法设计与分析

3.1 寻找支配集

本节讨论实现 MDS-HNP 问题的算法。MDS-HNP 的目的是寻找一组支配集, 使得所选的节点的性能在它所支配的节点的所有邻居节点中为最大。在此首先讨论 MDS-HNP 的一般形式, 即 MDS 问题。我们知道, MDS 问题等价于 MSC 问题 (minimum set cover, 最小集合覆盖问题)^[7]。

定义 1 在图 $G(V, E)$ 中, 每一个顶点及其所有邻居节点作为集合中的元素构成一个集合, 即每个节点 v_i 对应一个集合 S_i , 则 $S_i = \{v_i\} \cup \{v_j | v_j \in N(v_i)\}$ 。

定义 2 给定一个集合 U 及子集 $\Omega = S_1, S_2, \dots, S_n$, 满足这些子集的全集为 U (即 $S_1 \cup S_2 \cup \dots \cup S_n = U$), 最小集合覆盖问题 (MSC) 为找到最少个数的子集 $BN = S_{B_1}, S_{B_2}, \dots, S_{B_k}$, $BN \subseteq \Omega, k \leq n$, 使得 $\bigcup_{S_{B_i} \in BN} S_{B_i} = U$ 。

算法 1 MDS 问题

输入: $\Omega = S_1, S_2, \dots, S_n$ 子集族;

输出: $BN \subseteq V$ // 找出个数最少的点覆盖子集

第一步初始化

1.1 $NC = U$; // NC 为未覆盖的节点集合

1.2 $BN = \phi$; // BN 为支配集合

1.3 $NS = \{1, 2, \dots, n\}$; // 支配集的节点号集合

第二步循环, 直到所有集合中的元素被覆盖

2 while $NC \neq \phi$ do

2.1 选择 $i \in G$ 的子集 S_i , 且 $|S_i \cap NC|$ 个数最多 // 选择能够覆盖最多未被覆盖的节点集合。

2.2 $NC = NC - S_i$;

2.3 $BN = BN \cup \{v_i\}$;

2.4 $NS = NS - \{i\}$;

end

第三步输出

3 output BN ;

因为 MSC 问题为 NP 难度问题, 这里使用贪心启发式算法来解 MSC 问题, 如算法 1 所示。其主要思想是在每次挑选一个集合时, 总是选择能够覆盖最多的未被覆盖的节点的集合, 直到所有集合中的元素被覆盖。对于被选中的集合, 其对应的节点将构成一个支配集。根据文献[8], 我们知道对 MSC 问题的贪心算法是一个逼近度为 $H(k)$ 的算法, k 表示每个子集 S_i 最多有 k 个元素, H 表示调和级数。由于 $k = (\Delta + 1)$, 图 3 中算法的逼近度为 $H(\Delta + 1)$ 。由于采用贪婪算法, 该算法的时间复杂度为 $O(n)$ 。

MDS-HNP 问题可以通过修改算法 1 来解决, 如算法 2 所示。在该算法中, 每次循环总是从性能级别最高 (L_{\max}) 的节点中选取覆盖节点数量最多的节点。当选取 L_{\max} 级别的节点不能在覆盖新的节点时, 算法从下一个级别 ($L_{\max} - 1$) 的节点中选取, 直到全集 U 中所有的节点被覆盖。

算法 2 MDS-HNP 问题算法

输入: $\Omega = S_1, S_2, \dots, S_n$ 是集合 U 的子集族;

节点权值集 $\{Q_1, Q_2, \dots, Q_n\}$

输出: $BN \subseteq V$

第一步初始化

1.1 $NC = U; BN = \phi$;

1.2 $L_{\max} = \max_{i \in \{1, 2, \dots, n\}} (Q_i)$; // 性能最高级别

1.3 for $level = 1$ to L_{\max} do

$NS_{(level)} = \phi$; // 初始化 L_{\max} 个支配集节点号集为空

end

1.4 for $i = 1$ to n do // 为每个级别的节点号集赋初值

$NS_{(Q_i)} = NS_{(Q_i)} \cup \{i\}$;

end

第二步循环直到所有集合中的元素被覆盖

2 while $NC \neq \phi$ do

2.1 选择子集 $S_i, i \in NS_{(level)}$ 且 $|S_i \cap NC|$ 在所有 $\{S_j | j \in NS_{(level)}\}$

中最大;

$$2.2 \quad NC = NC - S_i;$$

$$2.3 \quad BN = BN \cup \{v_i\};$$

$$2.4 \quad NS_{(level)} = NS_{(level)} - \{i\};$$

$$2.5 \quad \text{if } \forall i \in NS_{(level)} \&\& |S_i \cap NC| = 0 \text{ then}$$

$$level = level - 1;$$

end

end

第三步输出

3 output BN;

定理 1 算法 2 中对 MDS-HNP 问题的贪心算法所产生的支配集中的节点的个数最多为 $H(\Delta+1)|OPT|$ 。其中, OPT 是对 MDS-HNP 问题的最优化的最小支配集, Δ 是所有节点最大的度。

证明:我们已知,对于 MDS 问题,贪婪算法的逼近度为 $H(\Delta+1)$ 。当 $L_{max} = 1$ 时, MDS-HNP 问题等价于 MDS 问题,因此在此情况下算法 2 的逼近度为 $H(\Delta+1)$ 。当 $L_{max} > 1$ 时, OPT_i 表示从性能级别为 i 的节点中对 MDS 问题得到的最优解法, B_i 为从性能级别为 i 的节点中通过算法 2 所得到的结果,那么有 $|B_i| = H(\Delta+1)|OPT_i|$ 。因此,由算法 2 得到的支配节点的个数的总和为:

$$\sum_{i=1}^{L_{max}} |B_i|, \text{ 且不多于 } H(\Delta+1) \sum_{i=1}^{L_{max}} |OPT_i|$$

$$\text{由于 } |OPT| = H(\Delta+1) \sum_{i=1}^{L_{max}} |OPT_i|,$$

$$\text{有 } \sum_{i=1}^{L_{max}} |B_i| \leq H(\Delta+1) |OPT|.$$

证毕。

3.2 连接支配集

确定了一个无向图中的节点支配集之后,若所得的支配集的节点并不相互连通,需要增加新的节点到支配集中,使得支配集中的节点是相互连通的,即构成一个连通支配集。在选取新的节点加入到支配集时,从性能级别最高的节点开始寻找。选取的标准为邻居节点中包含支配集中节点最多的节点。假设支配集为 NB , 非支配集为 $\overline{NB} = U - NB$, 所选取的节点 v_i 满足

$$\text{Max}_{v_i \in \overline{NB}} (|N(v_i) \cap NB|) \text{ 且 } |N(v_i) \cap NB| \geq 2$$

假如满足 L_{max} 级别的节点选择完毕,继续从 $L_{max} - 1$ 级别的节点选取。同时,该操作在第 $i (L_{max} - 1 \geq i \geq 1)$ 级别的节点选取完毕后,若该级别 i 的某节点存在一个性能级别大于它,且不在支配集中的邻居节点,则将将该节点及其邻居节点放入支配集中,这是由于两个不在支配集中的相邻节点必然连接两个分离的支配集^[8]。以上操作直到所有的支配集节点相互连通为一个整体(即 N_c 等于 1 时),如算法 3 所示。由于该算法仍然以贪心算法为基础,其时间复杂度为 $O(n)$ 。

算法 3 MCDS-HNP 问题求解算法

输入:图 $G(V, E)$, 支配集 $NB = \{v_1, v_2, \dots, v_k\}$, 节点权集 $\{Q_1, Q_2, \dots, Q_n\}$

输出:支配集 $NB' = \{v_1, v_2, \dots, v_j\}$

第一步初始化

$$1.1 \quad N_c = \text{分离的支配集个数};$$

$$1.2 \quad L_{max} = \max(Q_i);$$

$$i \in \{1, 2, \dots, n\}$$

$$1.3 \quad \text{for } level = 1 \text{ to } L_{max} \text{ do}$$

$$NS_{(level)} = \phi;$$

end

$$1.4 \quad \text{for } i = 1 \text{ to } n \text{ do}$$

$$NS_{(Q_i)} = NS_{(Q_i)} \cup \{i\};$$

end

第二步循环,直到所有支配集连通成整体($N_c = 1$)

2 while $N_c > 1$ do

$$2.1 \quad \text{if } \exists i \in NS_{(level)} \&\& |N(v_i) \cap NB| \geq 2$$

选择节点 $v_i, i \in NS_{(level)}$ 且满足 $\text{Max}_{v_i \in V - NB} (|N(v_i) \cap NB|);$

$$N_c = N_c - 1;$$

$$NB = NB \cup \{v_i\};$$

$$NS_{(level)} = NS_{(level)} - \{i\};$$

2.2 else

if $level = L_{max}$ then $level = level - 1;$

if $(\exists v_i \in V - X \&\& i \in NS_{(level)}) \&\&$

$(\exists v_j \in N(v_i) \&\& Q_j > Q_i \&\& v_j \in V - NB)$ then

$$N_c = N_c - 1;$$

$$NB = NB \cup \{v_i\} \cup \{v_j\};$$

$$NS_{(level)} = NS_{(level)} - \{i\};$$

else

$$level = level - 1;$$

end

end

end

第三步输出最小连通支配集

$$3.1 \quad X' = X;$$

3.2 Output X'

4 仿真与性能分析

为研究性能改进,本文在 MATLAB 下分别实现了 MCDS, MDS-HNP 和 MCDS-HNP 算法。在图 1 中,给出了一个运行结果的例子。图中有 250 个节点以同等概率随机分布在 $4000\text{m} \times 4000\text{m}$ 的网络中。图 1(a) 为网络的原始拓扑图。其中节点的性能划分为 5 个等级,以节点的大小来表示不同节点的性能等级。根据节点的性能等级,节点的射频范围在 400m 至 600m 之间。若两节点 v_i 与 v_j 之间的距离 $\|v_i v_j\|$ 同时小于 v_i 和 v_j 的射频范围,那么 v_i 和 v_j 互为邻居节点,中间有一条线段与之相连。图 1(b) 是 MCDS 算法运行的结果。图中红色的点为算法找出的骨干节点,这些骨干节点负责数据的转发。遍历这些骨干节点,构成了网络中的一棵树,即为整个网络的骨干网。MCDS 虽然以减少所需要的骨干节点个数为目标,但是并没有考虑骨干节点的性能。我们看到,图 1(b) 中选取了一些节点比较小(即性能较低)的节点作为骨干节点,而这些节点很可能成为骨干网瓶颈。图 1(c) 为 MDS-HNP 算法运行结果。该算法为构建骨干网的第一步,目的是找到图中的一个最小支配集,并且所选节点的性能为它所支配的节点的所有邻居节点的最大值。根据图 1(c) 的结果,尽可能地选择性能较高的节点来连接支配集中的节点,构成一个连通支配集。图 1(d) 为根据本文所提出的算法所构建的骨干网,其中红色节点为所选的骨干节点,绿色的连线构成了整个网络的骨干网。可看到,每一个普通节点都会找到一个最近的骨干节点与之相连,通过单跳的方式访问到骨干网。另外,所选的骨干节点平均来讲比 MCDS 算法所

(下转第 126 页)

- [5] Niyato D, Hossain E. Radio resource management games in wireless networks: an approach to bandwidth allocation and admission control for polling service in IEEE 802. 16[J]. IEEE Wireless Communication, 2007, 14(1): 27-35
- [6] Lee J W, Mazumdar R R, Shroff N B. Joint resource allocation and base-station assignment for the downlink in CDMA networks[J]. IEEE/ACM Transactions on Networking, 2006, 14(1): 1-14
- [7] Bacci G, et al. Energy efficient power control in impulse radio UWB wireless networks[J]. IEEE Journal of Selected Topics in Signal Processing, 2007, 1(3): 508-520
- [8] Low S H, Lapsley D E. Optimization flow control- I ; basic algorithm and convergence[J]. IEEE/ACM Transactions on Networking, 1999, 7(6): 861-874
- [9] 陶军, 吴清亮, 吴强. 基于非合作竞价博弈的网络资源分配算法的应用研究[J]. 电子学报, 2006, 34(2): 241-246
- [10] 蒙文武, 朱光喜, 谭力, 等. UWB无线通信系统接入控制流程设计[J]. 华中科技大学学报: 自然科学版, 2008, 36(9): 55-58
- [11] Liu K-H, Cai L, Shen X. Multiclass utility-based scheduling for UWB networks[J]. IEEE Transactions on Vehicular Technology, 2008, 57(2): 1176-1187

(上接第 103 页)

选的节点的性能要高。

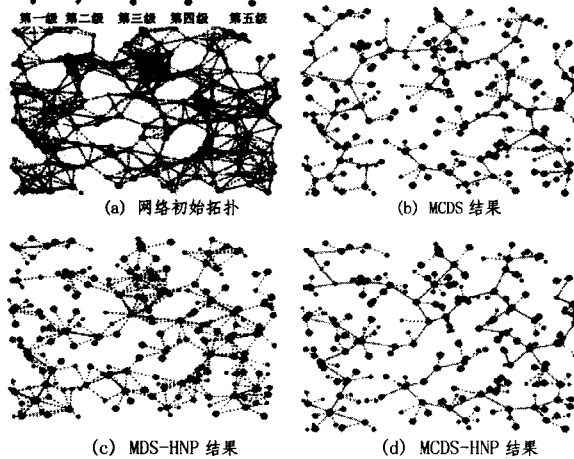


图 1 不同算法构建的无线骨干网络

为了测试采用本文所提算法后的骨干网络的性能, 图 2 给出了一组实验结果。本文定义骨干节点平均性能级别为 $\frac{\sum_{i=1}^k Q_i}{k}$ 。其中 k 为骨干节点的个数, Q_i 为骨干节点 x_i 的性能级别。骨干节点的平均性能级别能最好地体现所构建的骨干网性能。在图 2(a) 中, 设置最大节点性能级别为 5, 节点个数在 200 至 700 之间变化。在图 2(b) 中, 设置节点个数为 350, 最大节点性能级别在 2 至 8 之间变化。仿真结果表明, 我们提出的算法 MCDS-HNP 比原有的 MCDS 算法获得了更高的平均性能级别, 并且随着最大节点性能级别的增大, 两种算法的差别也增大。

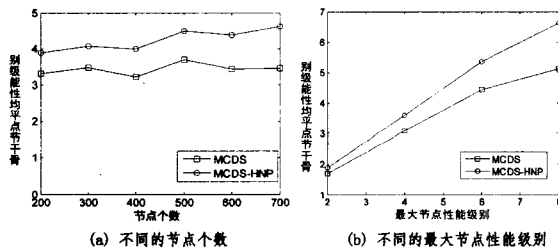


图 2 骨干节点平均性能级别

图 3 为骨干节点个数的仿真结果。可以看到, 虽然 MCDS-HNP 算法在寻找骨干节点时比 MCDS 算法增加了一些限制条件, 但是所选的骨干节点个数并没有增加很多。在不同的节点个数以及不同的最大节点性能级别的情况下, MCDS-HNP 算法所选择的骨干节点个数不超过 MCDS 算法的 10%。

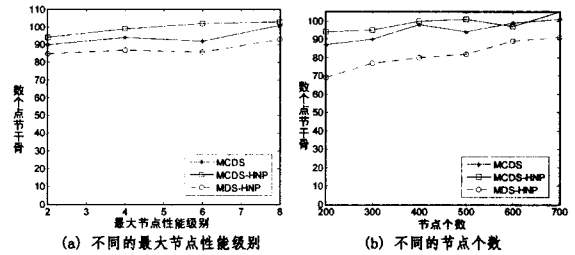


图 3 骨干节点个数

结束语 本文所做的主要工作有以下几点: 1) 讨论了在异构 Ad hoc 网络环境下通过搭建骨干网来进行层次型路由的策略; 2) 对选取高性能节点的最小连通集问题 (MCDS-HNP) 做了形式化定义; 3) 提出了针对异构 Ad hoc 网络中构建骨干网的多项式时间内估计算法; 4) 对算法的逼近度和时间复杂度进行了讨论; 5) 通过仿真对所提出的算法与现有算法进行了比较。结果表明, MCDS-HNP 算法由于考虑了选取节点本身的性能, 因此比 MCDS 算法能够获得更高的骨干网性能, 且没有过大地增加骨干网节点数。

目前, 我们所提出的算法是基于集中式的控制算法。集中式算法相对于分布式算法的优点是可以对整个网络拓扑进行全局优化; 对节点发现、路由维护等操作控制信息所需的开销比较小; 骨干网络的搭建速度快等。但是在某些特殊的应用场合如每个节点并不知道自己当前位置的情况下, 只能使用分布式算法来构建。在今后的工作中, 我们将研究 MCDS-HNP 问题的分布式算法实现。

参考文献

- [1] Saha A K, Johnson D B. Self-organizing hierarchical routing for scalable ad hoc networking [R]. TR04-433. Department of Computer Science, Rice University, 2004
- [2] 田明军, 赵丹, 王靖轩, 等. Ad hoc 网络中基于骨干网的相对定位算法研究[J]. 计算机研究与发展, 2008, 45(5)
- [3] Khuller S G. Approximation algorithms for connected dominating sets [J]. Algorithmic, 1998, 20(4): 374-387
- [4] 彭伟, 卢锡城. 一个新的分布式最小连通支配集近似算法[J]. 计算机学报, 2004, 24(3)
- [5] Daiand F, Wu J. An extended localized algorithm for connected dominating set formation in Ad hoc wireless networks[J]. IEEE Transactions on Parallel and Distributed System, 2004, 15: 908-920
- [6] Cheng X, Ding M, Duand H, et al. Virtual backbone construction in multihop ad hoc wireless networks [J]. Wireless Communications and Mobile Computing, 2006, 6: 183-190
- [7] West D B. Introduction to Graph Theory[M]. Prentice Hall, 1996
- [8] Chvatal V. A Greedy Heuristic for the Set Covering problem [J]. Mathematics of Operations Research, 1979, 4: 233-235