

无线 Ad Hoc 网络中异构链表支配集算法

韩冰青 陈一飞

(南京审计大学工学院 南京 211815)

摘要 首先给出无线 Ad Hoc 网络的异构圆盘图模型 HDG,并分析 HDG 模型的不同形态;然后设计出一种新的节点双向链表结构,在此基础上,提出一种基于链表结构的异构连通支配集算法 C-LDS。该算法通过双向链表结构管理支配集,并通过节点引用的方式来提高支配集节点增加、删除及修改的时间效率,从而得到优化的连通支配集。将 C-LDS 算法与其他支配集算法进行对比测试,结果表明:在均匀分布以及随机分布的网络场景中,C-LDS 所生成的支配集尺寸是最小的;在随机移动的网络场景中,C-LDS 的分组投递率是最高的,展现出了较好的异构连通性并且提高了支配集节点的生成效率。

关键词 连通支配集,双向链表,异构圆盘,无线自组网

中图分类号 TP393 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2018.09.021

Heterogeneous Chain Dominating Set Algorithm in Wireless Ad Hoc Networks

HAN Bing-qing CHEN Yi-fei

(School of Technology, Nanjing Audit University, Nanjing 211815, China)

Abstract Firstly, the heterogeneous disk model HDG of Ad Hoc network was presented, and different forms of HDG model were analyzed. Secondly, a new two-way chain table structure was designed. Based on the chain structure, a connected dominating set algorithm C-LDS was proposed, which manages dominating set through two-way chain table structure, and improves the time efficiency of inserting, deleting and modify the dominating node through the node reference way. Thus, the connected dominating set of Ad Hoc network is optimized. Finally, the C-LDS algorithm was compared with other dominating set algorithms. The simulation results show that C-LDS algorithm generates minimal CDS size in both uniform and random scenarios, and can achieve highest packet delivery ratio in mobile network scenarios, which shows good heterogeneous connectivity and improves the efficiency generation of dominating set node.

Keywords Connected dominating set, Two-way chain table, Heterogeneous disk, Ad Hoc network

1 引言

无线自组网是由一组无线节点构成的自组织、多跳无线网络,它没有中央集中式的基础设施,被广泛应用于民用(如医疗行业、食品行业、农业等)和军事领域(如战场通信、搜救、灾难控制等)。与有线网络和蜂窝网络不同,无线自组网不需要物理骨干设施,通常由静态节点或者移动节点组成,每个节点包含一个全向天线,在其传输范围内可以将消息广播至其他节点。

虽然无线自组网可被应用于很多领域,但是无线网络节点会受到能源、计算能力、存储容量以及通信范围等资源的约束,在这些约束中能量消耗是至关重要的。无线自组网节点所需的能源主要由电池提供,而电池的能量是有限的,最先耗尽能源的节点通常决定了整个无线自组网的生存时间。为了延长网络的生存时间,需要通过拓扑控制的方法来优化无线网络的结构。如何对无线自组网进行拓扑构建、拓扑控制以及拓扑维护,成为了无线自组网设计的一个重要挑战。

拓扑控制的主要目标是通过调整网络拓扑结构的调整、精简与优化,达到节能并满足网络连通性、覆盖性的目的^[1-2]。近年来,基于支配集的网络拓扑控制方法得到了广泛关注,通常采用一种分层的拓扑控制方法,通过选择部分节点作为支配节点,在减少冗余的同时确保足够的覆盖率。若支配集中的节点是相互连通的,则该支配集成为连通支配集。在无线自组网中,由于没有固定的基础设施和集中式的管理,连通支配集常用于构建虚拟骨干网,以进行高效的路由和连通管理。一个小尺寸的连通支配集有利于简化路由,降低通信开销,并提高收敛速度,因此构造一个最小连通支配集很有意义,然而最小连通支配集的构建是一个 NP-Hard 问题。因此,对于无线自组织网络,在多项式时间复杂度内找出最小连通支配集的近似方法或者替代方法具有实际意义。

近年来,基于连通支配集的无线网络拓扑控制得到了学者们的广泛关注,成为了新的研究热点^[3-6]。文献[3]通过局部检测算法首先构建主 CDS,然后通过概率剪枝的方法生成

收到日期:2017-07-21 返修日期:2018-02-26 本文受国家自然科学基金项目(61402231),江苏省自然科学基金项目(BK2011692)资助。

韩冰青(1979-),男,博士,副教授,主要研究方向为 Ad Hoc 网络、网络协议、智能计算,E-mail:hbqsky@163.com(通信作者);陈一飞(1977-),女,博士,副教授,主要研究方向为数据挖掘、文本挖掘等。

精简的连通支配集,有效地减少了CDS的尺寸。文献[4]提出了一种分布式的贪婪算法来构造连通支配集,首先构造一个基础支配集,然后将基础支配集中的节点连接起来,构成连通支配集。文献[5]提出了一种可靠的、节能的连通支配集拓扑构建和维护协议Poly,该协议通过关闭一些不必要的网络节点,构建一种多边形的骨干网络,既保证了网络的连通性又达到了网络的覆盖率。最近,Ramesh等^[6]提出了一种分布式的算法来构建单位圆盘图的连通支配集,并给出了一种调度方案,该方案能够在线性无冲突时段处理干扰。

以上研究工作主要关注连通支配集的尺寸大小,较少涉及到路由、能量效率等因素。对于无线Ad Hoc网络而言,路由协议不仅需要较小的CDS尺寸,还需要支持节能等额外需求。为了这个目标,文献[7]提出了一种拓扑构造生成算法A3,该算法根据接收信号的强度来估算传感器节点间的距离,给予远离父节点并且能量充沛的子节点更高的优先级,从而生成一个次优的连通支配集近似解。文献[8]提出了一种能量有效的拓扑控制算法A1,其作为A3的扩展,通过选择高能量的活跃节点构成连通支配集,既保证了网络连通性又保证了能量有效性。

为了在无线网络中构建高质量容错虚拟骨干网,Namsu等^[9]提出了一种整数编程的优化算法来求解k-连通m-支配集问题,所构建的支配集既能保证路由的灵活性,又能提供容错性。文献[10]提出了一种多项式时间常数的近似算法来求解最小4-连通m-支配集问题。此外,文献[11]提出了一种性能保证的近似算法来构建无线网络容错连通支配集,其对于最小(3,m)连通支配集问题能提供多项式时间的近似算法,改善了同构无线传感网络的性能比率。文献[12]研究了k-连通d-支配的容错支配集问题,围绕顶点割给出了3种不同类型的整数编程公式,从而构建k连通的虚拟骨干支配集,通过小幅度增加容错参数来保证网络的鲁棒性,进而提高顶点集的整体最优解。

然而,以上大多数研究工作都是以单位圆盘图(Unit Disk Graph,UDG)^[6,13-14]为网络模型,该模型中的所有节点都是同构的,具有相同的传输范围,没有涉及到异构节点的情况。但是,在实际的无线网络部署中,无线节点的传输功率、传输范围以及能量等参数并不相同,存在异构节点的情形,从而形成异构的无线网络。而以单位圆盘图为基础的网络模型不再适用于异构无线网络,因此如何对异构无线网络进行拓扑控制并构建相应的支配集成为了新的挑战。

本文研究的目及意义主要有以下两个方面:1)提高无线Ad Hoc网络支配集的生成效率,改进对支配节点进行增加、删除及修改的时间效率;2)优化异构Ad Hoc网络连通支配集的尺寸,减少异构Ad Hoc网络连通支配集节点的数量,用较少的支配节点达到对Ad Hoc网络的覆盖,同时保证支配集节点之间是相互连通的,从而构建异构Ad Hoc网络的骨干拓扑结构。

本文首先对无线网络模型进行分析,给出无线Ad Hoc网络的异构圆盘图模型HDG;接着在分析HDG模型的基础上,提出一种基于链表结构的异构连通支配集算法C-LDS。本文的主要特点在于:首先,给出了异构圆盘图模型HDG的定义,并分析了HDG模型的不同形态,分别描述了节点直接

通信和转发通信的不同实例;其次,设计了一种新的节点双向链表结构来管理支配集,通过节点引用而不是新增节点的方式来提高对支配集节点进行增加、删除及修改操作的时间效率;最后,评测结果表明C-LDS算法具有有效性和稳定性,能够提高支配集节点的生成效率。

2 数学模型

2.1 无线网络模型

在同构无线网络 $G=(V,E)$ 中, V 为无线节点集, E 为边集,所有无线节点 $v \in V$ 具有相同的最大传输范围 r ,任意两个节点 u,v 之间存在一条边 $(u,v) \in E$,当且仅当这两个节点的距离小于传输范围,即 $d(u,v) \leq r$ 。通过一个适当的比例因子 k ,可以将所有节点的传输范围缩放为一个单位,从而构成单位圆盘图(UDG)模型。

在异构无线网络 $G_h=(V_h,E_h)$ 中, V_h 表示异构的无线节点集, E_h 表示异构边集。异构节点由于自身的功率、能量等因素,传输范围并不完全相同。假设所有节点分布在一个欧几里得平面中,每一个节点 $u \in V_h$ 具有传输范围 r_u ,两个节点 u,v 之间存在一条有向边 $(u,v) \in E_h$,当且仅当 $d(u,v) \leq r_u$,其中 $d(u,v)$ 表示节点 u,v 之间的欧氏距离。称边 (u,v) 为双向边,当且仅当 $d(u,v) \leq r_u$ 并且 $d(u,v) \leq r_v$,即 $d(u,v) \leq \min(r_u,r_v)$ 。令 $disk(u,v)$ 表示以节点 u,v 的距离为直径的圆盘, $disk(r_v)$ 表示以节点 v 为中心、 r_v 为半径的圆盘。

2.2 异构圆盘图模型

定义1(异构圆盘图模型 HDG) 异构圆盘是一个四元组,由节点 u,v 以及传输范围 r_u,r_v 构成,为节点 u,v 所构成的圆盘 $disk(u,v)$ 与节点 u 的圆盘、节点 v 的圆盘3个圆盘合成的交集,即:

$$disk(u,v,r_u,r_v) = disk(u,v) \cap \{disk(r_u) \cap disk(r_v)\}$$

本文称其为异构圆盘图模型 HDG(Heterogeneous Disk Graph)。在HDG模型中,各个无线节点的传输范围 $r_v \in R$ 不同,根据拓扑控制的需要可以调整节点功率,传输范围可以随拓扑结构的变化而变化,具有较高的灵活性。

HDG模型有多种形态。第一种情况:节点 u,v 所构成的圆盘 $disk(u,v)$ 包含在传输范围 r_u,r_v 相交的圆盘内,即:

$$disk(u,v) \subseteq \{disk(r_u) \cap disk(r_v)\}$$

在这种情况下,节点 u,v 之间构建双向链路,可以直接通信。图1给出了HDG模型的示例,图中的箭头指出了HDG圆盘。

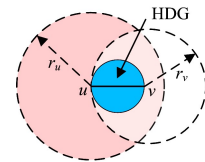


图1 异构圆盘图模型示例1

Fig.1 Example 1 of heterogeneous disk model

第二种情况:节点 u,v 所构成的圆盘 $disk(u,v)$ 不在传输范围 r_u 与 r_v 相交的圆盘内,即:

$$disk(u,v) \not\subseteq \{disk(r_u) \cap disk(r_v)\}$$

图2给出了一个示例,根据定义1,HDG圆盘如图中的箭头所指。在这种情况下,节点 u 和节点 v 不能直接通信,需要通过其他节点转发通信。

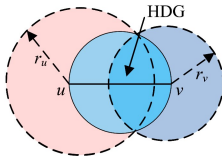


图 2 异构圆盘图模型示例 2

Fig. 2 Example 2 of heterogeneous disk model

定义 2(支配集 DS) 对于同构无线网络 $G=(V, E)$ 而言,如果存在子集 D 使得任意一个节点 $v_i \in V$, 都有 $v_i \in D$ 或者存在节点 $v_j \in D$ 且 $(v_i, v_j) \in E$, 则子集 $D \subseteq V$ 称为支配集。

定义 3(异构支配集 HDS) 对于异构无线 Ad Hoc 网络 $G_h=(V_h, E_h)$, 如果存在子集 D_h 使得任意一个节点 $v_i \in V_h$, 都有 $v_i \in D_h$ 或者存在节点 $v_j \in D_h$ 且 $(v_i, v_j) \in E_h$, 则子集 $D_h \subseteq V_h$ 称为异构支配集, 节点 $v_m \in D_h$ 称为支配节点。令 $B_h = V_h - D_h$, B_h 中的节点 $v_n \in B_h$ 称为被支配节点。

定义 4(邻居集 Neighbor) 令 $Neighbor(v) = \{u \in V | uv \in E\}$ 表示节点 v 的开放邻居集, 则其闭合邻居集为 $Neighbor[v] = Neighbor(v) \cup \{v\}$ 。

3 异构 Ad Hoc 网络的支配集生成算法

3.1 节点双向链表结构

为了提高支配集节点增加、删除及修改操作的时间效率, 设计了一种双向链表结构, 如图 3 所示。其中, WhiteList 为白色节点链表, 表示未处理的节点链表; BlueList 为蓝色节点链表, 表示支配节点链表; GrayList 为灰色节点链表, 表示被支配节点链表。3 个链表都以双向链表结构管理支配集, 链表元素包含对节点 v 的引用, 并指向原节点集 V , 没有额外生成节点, 节省了时间开销。

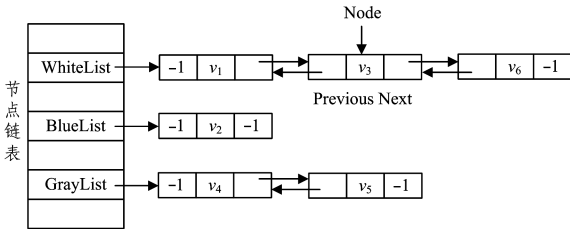


图 3 节点双向链表结构

Fig. 3 Two-way chain structure of node

3.2 基于链表的支配集生成算法

由于无线 Ad Hoc 网络没有基础设施, 节点具有移动性, 网络拓扑会动态变化, 而集中式的支配集生成算法需要全局位置信息, 因此该算法不适用于动态的、分布式的网络场景。下面给出一种基于链表结构的支配集生成算法 LDS, 该算法以分布式的方式实现, 适用于动态的无线网络场景。

算法 1 基于链表的支配集生成算法 LDS

1. 邻居发现。初始状态时, 网络中的各个节点没有邻居信息。首先, 启动邻居发现进程, 各个节点将自身的节点 ID 以及邻居数广播至其他节点。经过一段时间后, 基于各节点的通信范围, 每个节点都获得了 1-跳范围内的邻居信息。
2. 节点排序。令 $N(v_i)$ 表示节点 $v_i \in V$ 的开放邻居数, 对节点集 V 中的节点按邻居数由大到小排序。
3. 链表构建。

3.1 构建白色节点链表 WhiteList、蓝色节点链表 BlueList 和灰色节点链表 GrayList, 3 个链表都以双向链表结构存储, 并指向节点集 V 。

3.2 通过双向链表管理排好序的节点集, 令 $WhiteList = V$, $BlueList = \{\emptyset\}$, $GrayList = \{\emptyset\}$ 。因为节点集 V 已经排好序, 所以 WhiteList 链表的第 1 位必然是具有最大邻居数的节点, 紧接着是具有次最大邻居数的节点。

4. 节点填色。

4.1 将所有的节点 $v_i \in V$ 填为白色, 更新白色链表, 令

$$WhiteList = V$$

4.2 第 1 个节点 $v_1 = WhiteList[1]$ 变为蓝色, 并更新蓝色链表、白色链表:

$$BlueList = BlueList \cup v_1$$

$$WhiteList = WhiteList - v_1$$

接着通知 v_1 的所有邻居 $Neighbor(v_1)$, 收到蓝色通知的白色节点变为灰色, 并更新灰色链表和白色链表:

$$GrayList = GrayList \cup v_k$$

$$WhiteList = WhiteList - v_k$$

其中, $v_k \in Neighbor(v_1)$ 且 $v_k \in WhiteList$ 。

4.3 对第 2 个节点 $v_2 \in V$ 进行处理:

1) 若 v_2 为白色, 重复步骤 4.2。

2) 若 v_2 为灰色, 检查 v_2 是否有白色邻居, 若有, 则将 v_2 置为蓝色。 v_2 发送蓝色通知, 将所有收到蓝色通知信息的白色邻居置为灰色。

4.4 重复上述过程, 直到 $WhiteList = \{\emptyset\}$ 。

5. 所有的蓝色节点构成支配集, 令 $DS = BlueList$, 输出支配集 DS 。

LDS 算法区别于其他算法的主要特征如下:

1) LDS 算法以双向链表的方式管理支配集, 并以引用的方式指向原节点集 V , 没有额外生成节点, 节省了时间开销。双向链表结构如图 3 所示。

2) LDS 算法的链表以增量式的方式进行管理, 并且节点之间的消息传送采用事件驱动的机制进行处理。当蓝色链表增加一个节点 v_i , 即 $BlueList = BlueList \cup v_i$ 时, 将触发“节点变蓝色”事件, 白色邻居节点一旦收到该消息, 将启动相应的消息处理进程, 白色链表就会删除 v_i 节点, 即执行 $WhiteList = WhiteList - v_i | v_i \in V$ 。这种增量式的、事件驱动的处理方式使得增加、删除节点的速度加快, 能够提高支配集的生成速度。

3.3 优化的异构链表连通支配集算法

当 Ad Hoc 网络为异构网络时, 异构节点的功率、电池能量、传输范围等并不完全相同, 需要对 LDS 算法进行扩展, 以适用于异构 Ad Hoc 网络。本文中, 异构 Ad Hoc 网络与同构网络的区别主要在于节点传输范围的异构, 并不涉及电池能量、发送速率等的设定。我们将异构圆盘图模型 HDG 应用于异构 Ad Hoc 网络中, 在算法初始化时, 对所有异构节点配置 HDG 异构圆盘图模型, 在执行邻居发现进程以及数据通信时按 HDG 模型进行处理: 1) 若节点 u, v 所构成的圆盘 $disk(u, v)$ 包含在传输范围 r_u, r_v 相交的圆盘内, 则节点 u, v 之间构建双向链路, 可直接通信; 2) 若节点 u, v 所构成的圆盘 $disk(u, v)$ 不在传输范围 r_u 与 r_v 相交的圆盘内, 则节点 u, v 不能直接通信, 只能通过其他节点转发通信。

此外, 我们对 LDS 算法进行进一步优化, 通过生成候选节点的方法更新支配节点集, 并通过剪枝的方式消除冗余节

点,缩小支配集的尺寸,从而得到优化的基于链表结构的异构连通支配集算法 C-LDS。

C-LDS算法的总体流程图如图4所示。算法首先判断 Ad Hoc 网络的类型,若为异构 Ad Hoc 网络,则所有异构节点配置 HDG 模型,并启动邻居发现进程;若为同构 Ad Hoc 网络,则直接执行邻居发现进程。然后经过双向链表构建、节点填色后生成初始支配集。最后对支配集进行优化和剪枝,输出最终的支配集节点。

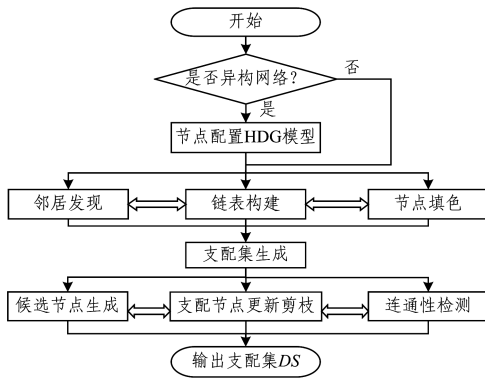


图4 C-LDS算法流程图

Fig. 4 Flowchart of C-LDS algorithm

C-LDS算法的具体描述如算法2所示。

算法2 异构链表连通支配集算法 C-LDS

输入:异构无线 Ad Hoc 网络 $G_h = (V_h, E_h)$

1. 异构节点配置 HDG 异构圆盘模型
2. 执行 LDS 算法,得到支配集 DS
3. 候选节点生成
 - 3.1 Candidate = $\{\emptyset\}$
 - 3.2 for each node $v_i \in \text{GrayList}$ do
 - 3.3 if (v_i 的蓝色邻居数 ≥ 2)
 - 3.4 将节点 v_i 变为绿色候选节点
 - 3.5 Candidate = Candidate $\cup v_i$
 - 3.6 GrayList = GrayList - v_i
 - 3.7 end if
 - 3.8 end for
4. 支配节点更新
 - 4.1 for each node $v_k \in \text{Candidate}$ do
 - 4.2 if(v_k 的邻居全部在蓝色节点的邻居集内)
 - 4.3 {将节点 v_k 变为灰色
 - 4.4 更新灰色链表与候选链表
 - 4.5 }
 - 4.6 else
 - 4.7 {将节点 v_k 变为蓝色
 - 4.8 更新蓝色链表与候选链表
 - 4.9 }
 - 4.10 end for
5. 支配节点剪枝
 - 5.1 对蓝色链表中的节点进行遍历
 - 5.2 if 蓝色节点有两个或两个以上蓝色邻居 and 该蓝色节点的邻居在相邻蓝色邻居的邻居集内
 - 5.3 将该蓝色节点置为灰色
 - 5.4 更新蓝色链表和灰色链表
 - 5.5 end if
 - 5.6 end 遍历
6. 连通性检测

- 6.1 若蓝色节点集未连通,则遍历灰色节点集,若有灰色节点能连接 2-跳范围内未连通的蓝色节点,则将该灰色节点变为蓝色并更新蓝色链表和灰色链表
- 6.2 重复步骤 6.1,直至蓝色节点集连通为止
7. 重复节点候选、剪枝、连通检测过程,进行 r 次迭代
8. 所有的蓝色节点构成优化的 CDS 并输出

C-LDS算法既适用于同构的 Ad Hoc 网络,又适用于异构的 Ad Hoc 网络。设网络中有 n 个节点,对于异构 Ad Hoc 网络而言,由于增加了 HDG 模型配置环节,该部分增加的运行时间复杂度为 $O(n)$;双向链表构建基于节点集 V 进行遍历,运行的时间复杂度为 $O(n)$;节点填色过程的运行时间与节点数及邻居节点最大度相关;支配集更新时间、剪枝的运行时间与网络拓扑结构及邻居节点度相关,并与迭代轮数 r 相关, $r(r \geq 1)$ 在算法运行时被初始化为一个正整数。

4 算法性质分析

定理 1 若网络 $G = (V, E)$ 是连通的,则通过 C-LDS 算法所构建的蓝色节点集合形成了一个连通支配集。

证明:通过对网络中节点集 V 使用归纳法来证明。首先,考虑节点数 $|V| = 1$ 的情况,此时整个网络只有 1 个节点 v_1 ,根据前述算法,该节点 v_1 初始时为白色,接着会被标识为蓝色,因此节点 v_1 形成了一个尺寸为 1 的连通支配集。

接着,通过归纳的方法来证明节点数 $|V| \geq 2$ 的情形。假设 $k-1$ 个节点被网络中的连通支配集 $DS(k-1)$ 覆盖,现在增加一个新节点 v_k ,有以下两种情况:1)若节点 v_k 存在一个蓝色邻居在支配集 $DS(k-1)$ 中,即 $Neighbor(v_k) \cap DS(k-1) \neq \emptyset$,则 $DS(k-1)$ 仍为新网络的连通支配集,此时, $DS(k) = DS(k-1)$;2)若节点 v_k 没有一个邻居为蓝色节点,即 v_k 的邻居全是灰色节点,且 $Neighbor(v_k) \cap DS(k-1) = \emptyset$,因为 v_k 的灰色邻居 $Neighbor(v_k)$ 必然被 $DS(k-1)$ 支配,根据算法,节点 v_k 的一个灰色邻居 $v_m (m \in \{1, 2, \dots, k-1\})$ 检测到白色节点 v_k ,则将节点 v_m 置为蓝色并将节点 v_k 置为灰色。此时,节点 v_k 被 v_m 支配, $DS(k) = DS(k-1) \cup \{v_m\} (m \in \{1, 2, \dots, k-1\})$ 构成了一个新的连通支配集。这证明了 $DS(k)$ 可由 $DS(k-1)$ 及新节点 v_k 推导出来。因此,通过 C-LDS 算法所得到的蓝色节点集合必然形成一个连通支配集。

5 仿真实验

本文构建了 C++ 连通支配集仿真平台来评估所提算法,仿真平台可以对无线 Ad Hoc 网络场景进行参数化配置。将提出的算法与其他算法进行对比实验。一定数量的 Ad Hoc 网络节点分布在不同的平面区域,平面区域最小为 $200\text{m} \times 200\text{m}$,最大为 $600\text{m} \times 600\text{m}$ 。对不同类型的网络分别进行仿真:1)同构 Ad Hoc 网络场景;2)异构 Ad Hoc 网络场景。在同构 Ad Hoc 网络中,各节点的传输范围、发射功率相同。在异构网络场景中,各节点的传输范围、节点能量、发射功率不一定相同。在所有的仿真实验中,只计算双向链路,不统计单向链路。

对每种类型的网络设定不同的网络尺寸,无线 Ad Hoc 网络的节点数量为 $6 \sim 160$,每次增加 $10 \sim 20$ 个节点。仿真时对每个网络场景运行 50 次,取平均值作为仿真结果。

5.1 网络拓扑场景

为了全面评测 C-LDS 算法的性能,在仿真实验中选取了

不同类型的网络拓扑场景来测试。

(1) 均匀分布网络场景

在均匀分布网络场景中,网络节点是同构的,以指定的方式均匀分布在网络中,网络拓扑是静态的,节点数量为 6~100,网络中各节点的最大传输功率、最大传输范围相同。

(2) 随机分布网络场景

在随机分布网络场景中,网络节点随机部署在网络中的各个位置,节点数量为 20~160,网络节点可以是同构的,也可以是异构的。当网络节点同构时,各节点的最大传输范围相同;当网络节点异构时,各节点的最大传输范围有所不同。

图 5 为一张随机分布网络的示例图。该场景为异构 Ad Hoc 网络,30 个节点随机分布在 $400\text{ m} \times 400\text{ m}$ 的区域内,各节点的传输范围不完全相同,异构节点均配置了 HDG 异构圆盘图模型,最大传输范围在 $60\sim 90\text{ m}$ 之间动态变化,初始时所有节点均为白色。运行时期,所有的白色节点均被填为蓝色、灰色或绿色,图中蓝色节点构成当前的支配节点集,灰色节点和绿色节点共同组成被支配节点集,其中部分灰色节点有两个或两个以上的蓝色节点邻居,因此被选为绿色候选节点,从而进行下一轮支配节点的更新与剪枝。

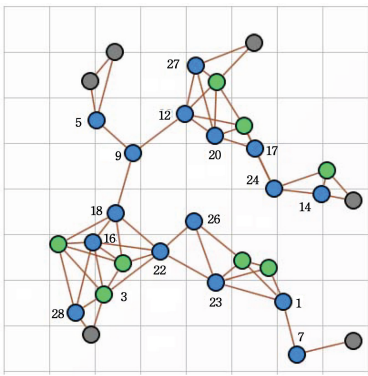


图 5 随机网络场景(30 个节点)

Fig. 5 Random network scenario(30 nodes)

5.2 算法性能

首先对 CDS 的尺寸进行测试,将 C-LDS 算法与其他几种算法^[3-4,15]进行评测,比较在不同的网络节点数量下生成连通支配集的大小。其中,Prunning 算法基于剪枝的启发式策略^[3]生成支配集,Greedy 算法基于贪心策略^[4]生成支配集,Random 算法^[15]通过随机化的近似方法来构建支配集。选取这 3 种算法作为支配集基准算法进行对比评测的主要原因在于:这 3 种算法分别表示贪心、剪枝以及随机化方法,都是典型的支配集求解方法,可以使得对比评测更全面,从而分析算法在不同指标参数下与不同类型算法的对比性能。

在对比实验中,既有均匀分布的网络场景,也有随机分布的网络场景,网络节点数从 20 逐步增加到 100,每次增加 20 个节点。

从图 6 可以看到,C-LDS 算法的性能超过了 Greedy 算法、Prunning 算法、Random 算法,所生成的 CDS 尺寸在仿真网络场景中是最小的。在网络节点数量小于 50 时,C-LDS 算法所得到的连通支配集是最小的,Prunning 算法比较接近 C-LDS 算法,而 Greedy 算法和 Random 算法所生成的 CDS 尺寸明显高于其他算法。当网络节点数逐渐增加从而使网络

密度变大时,C-LDS 算法的 CDS 尺寸始终保持在一个较低的范围,而 Random 算法、Greedy 算法的 CDS 尺寸则增加较快。与 Random 算法相比,C-LDS 算法的 CDS 尺寸减小了约 36%,与贪心算法 Greedy 相比减少了约 27%,与 Prunning 算法相比减少了约 17%。在不同网络尺寸的场景中,C-LDS 算法所得到的 CDS 尺寸都是最小的,因此该算法保证了网络的连通性,具有较好的性能。

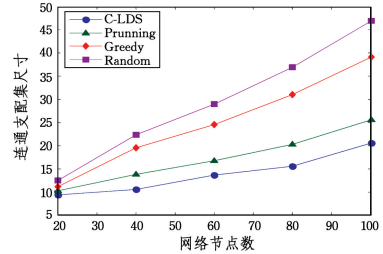


图 6 连通支配集的尺寸

Fig. 6 Size of connected dominating set

接着,对冗余节点缩减比率进行实验,比较 C-LDS 算法与 Prunning 算法、Greedy 算法在不同网络尺寸下冗余节点的缩减比率,其中,缩减比率=支配节点缩减数/网络尺寸。

缩减比率反映了算法对冗余节点的剪枝性能(即能够在初始支配集上进一步缩减的程度)。本文对 3 种算法进行了适应性的修改以进行评测,缩减比率越高,说明算法性能越好,越能够精简支配集的尺寸。

从图 7 可以看到,随着网络节点从 20 个增加到 160 个,支配节点的缩减比率逐步减小,在节点数超过 100 后,缩减比率趋于稳定。在各种网络尺寸下,C-LDS 算法平均缩减了 12.7% 的支配集尺寸,而 Prunning 算法平均缩减了 9.5% 的支配集尺寸,Greedy 算法平均缩减了 7.6%。在 3 种算法中,C-LDS 算法的缩减比率是最高的。

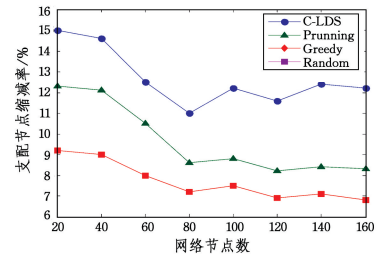


图 7 节点缩减

Fig. 7 Node reduction

最后,对移动 Ad Hoc 网络场景进行评测。在移动网络环境中,设置不同比例的移动节点,移动节点占总节点的比例从 0 增加到 100%,各节点的移动速度为 $0\sim 6\text{ m/s}$,节点的移动方向是随机的,各节点的传输范围有所不同,存在异构的网络节点。

图 8 给出了分组投递成功率随移动节点比例变化的情况,随着可移动节点占总节点比例的增加,分组投递率逐步下降。当移动节点为 0 时,各算法的分组投递率都是 100%,因为此时网络中各节点处于静止状态,网络的拓扑结构不会变化。而当移动节点比例增加时,网络中移动节点越来越多,链路的断开与重置也越来越频繁,网络的拓扑结构发生变化,分组投递率有所下降。

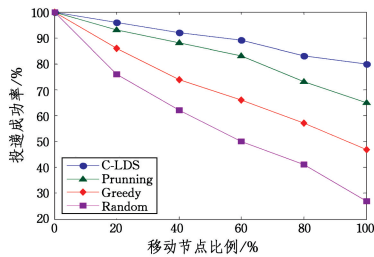


图8 随机场景投递成功率

Fig. 8 Packet delivery rate in random network

当移动节点的比例超过50%时,网络中节点的移动更加频繁,网络拓扑结构动态变化,C-LDS算法的分组投递率下降得比较缓慢,投递成功率始终保持在80%以上,而Pruning算法、Greedy算法以及Random算法的分组投递率下降得比较明显,这是因为当网络中存在异构节点时,这些算法不能有效地构建动态的连通支配集,从而引起分组投递率的大幅下降。当移动节点比例增大到100,即所有节点都在移动时,Greedy算法只有不到50%的投递成功率,Random算法仅有不到30%的投递成功率,这意味着在高动态的移动场景中,Greedy算法和Random算法丢失了一半以上的分组数据。在各种随机移动场景中,C-LDS算法的分组投递率都是最高的,在高动态的移动场景中,C-LDS算法能够适应动态的移动场景,展现了较好的异构连通性和整体稳定性。

在C-LDS算法中,异构节点配置HDG异构圆盘图模型可以有效构建异构连通支配集,能够适应Ad Hoc网络拓扑结构的动态变化。为了缩减连通支配集的尺寸,C-LDS算法会发送控制消息以管理双向链表,并更新链表节点,随着支配集尺寸的减少,C-LDS算法的通信开销会增大。同时,在移动网络拓扑场景中,随着移动性的增加,链路的断开与重置也更加频繁,为了维护链路保证分组投递成功率,C-LDS算法需要检测更多的邻居节点来转发通信,因此,C-LDS算法的消息复杂度也会随之增加,从而高出其他3种算法。

结束语 本文在分析无线Ad Hoc网络异构圆盘图模型的基础上,提出了一种基于链表结构的连通支配集算法C-LDS。该算法以双向链表的方式生成支配集,通过节点引用而不是新增节点的方式来提高支配集节点增加、删除及修改操作的时间效率,并通过剪枝的方法消除冗余节点,能得到优化的连通支配集。最后,本文将C-LDS算法与Pruning算法、Greedy算法、Random算法进行了对比测试,结果表明:在均匀分布的网络场景以及随机分布的网络场景中,C-LDS算法所生成的CDS尺寸是最小的;在随机移动的网络场景中,C-LDS算法的分组投递率是最高的,保证了支配节点的覆盖,展现了较好的异构连通性,提高了支配集节点的生成效率。

本文中,异构Ad Hoc网络场景主要指节点传输范围的异构,忽略了节点能量、发送速率的异构,未来将进一步分析节点能量、发送速率等异构场景;此外,如何减少通信开销、降低消息复杂度也值得进一步研究。

参考文献

[1] MALEKI E N, MIRJALILI G. Fault-tolerant interference-aware topology control in multi-radio multi-channel wireless mesh networks[J]. *Computer Networks*, 2016, 110: 206-222.

[2] KONG S S, LIU L F, CHEN H. Low Delay Topology Control Algorithm Based on Delivery Ratio Constraint in Wireless Sensor Networks[J]. *Computer Science*, 2016, 43(2): 144-147. (in Chinese)
孔姗姗, 刘林峰, 陈行. 基于送达率约束的无线传感器网络低时延拓扑控制算法研究[J]. *计算机科学*, 2016, 43(2): 144-147.

[3] GHASEMI V, HASHEMI S N, MOZAFFARI M. Connected Dominating Set Construction Using an Efficient Pruning Method in Ad Hoc Networks[C]// *Proceeding of the 5th Annual ICST Wireless Internet Conference (WICON)*. Singapore, 2010: 1-8.

[4] DHAWAN A, TANCO M, SCOVILLE N. A distributed greedy algorithm for constructing connected dominating sets in wireless sensor networks[C]// *Proceedings of the 3rd International Conference on Sensor Networks*. Lisbon, Portugal, 2014: 181-187.

[5] QURESHI H K, RIZVI S, SALEEM M, et al. Poly: a reliable and energy efficient topology control protocol for wireless sensor networks[J]. *Computer Communications*, 2011, 34: 1235-1242.

[6] JALLUA R K, PRASADB P R, DASA G K. Distributed construction of connected dominating set in unit disk graphs[J]. *Journal of Parallel and Distributed Computing*, 2017, 104: 159-166.

[7] WIGHTMAN P, LABRADOR M. A3: a topology control algorithm for wireless sensor networks[C]// *Proceedings of IEEE Global Communications Conference (GLOBECOM)*. New Orleans, USA, 2008: 1-6.

[8] RIZVI S, QURESHI H K, KHAYAM S A, et al. A1: an energy efficient topology control algorithm for connected area coverage in wireless sensor networks[J]. *Journal of Network and Computer Applications*, 2012, 35(2): 597-605.

[9] NAMSU A, PARK S. An optimization algorithm for the minimum k-connected m-dominating set problem in wireless sensor networks[J]. *Wireless Networks*, 2015, 21(3): 783-792.

[10] WANG W, LIU B, KIM D, et al. A New Constant Factor Approximation to Construct Highly Fault-Tolerant Connected Dominating Set in Unit Disk Graph[J]. *IEEE Transactions on Networking*, 2017, 25(1): 18-28.

[11] ZHANG Z, ZHOU J, MO Y C, et al. Performance-Guaranteed Approximation Algorithm for Fault-Tolerant Connected Dominating Set in Wireless Networks[C]// *IEEE INFOCOM*. 2016: 1-8.

[12] BUCHANAN A, SUNG J S, BUTENKO S, et al. An integer programming approach for fault-tolerant connected dominating sets[J]. *Inform Journal on Computing*, 2015, 27(1): 178-188.

[13] JIN W L, CHEN G S. Improved broadcasting scheduling algorithm in cognitive radio networks[J]. *Application Research of Computers*, 2015, 32(3): 860-865. (in Chinese)
金伟林, 陈国顺. 认知无线网络中一种改进的广播调度算法[J]. *计算机应用研究*, 2015, 32(3): 860-865.

[14] KAMEI S, IZUMI T, YAMAUCHI Y. An asynchronous self-stabilizing approximation for the minimum CDS with safe convergence in UDGs[J]. *Theoretical Computer Science*, 2016, 615: 102-119.

[15] DHAWAN A, TANCO M, YEISER A A. Randomized algorithms for approximating a Connected Dominating Set in Wireless Sensor Networks[C]// *International Conference on Computing and Network Communications (CoCoNet)*. 2015: 589-596.