

基于 0-1 规划的网络优化模型及其在信道分配中的应用

刘 蔚^{1,2}, 赵 宇¹, 陈 锐¹

(1. 中国科学院科技政策与管理科学研究所, 北京 100190; 2. 中国科学院大学, 北京 100049)

摘 要: 针对无线网络中的信道冲突和干扰问题, 提出一种应用于多射频多信道无线网络中的资源优化模型。该模型采用 0-1 线性规划方法, 以最小化网络干扰为目标, 利用分簇和构建冲突搜索树的方法收集干扰信息, 并通过信道分配算法达到消除干扰和资源优化的目的。应用于无线 Ad Hoc 网络信道分配中的实验结果表明, 与基于分簇的信道分配方法相比, 该模型使吞吐量最多提高 89.5%, 能降低待发信道队列长度和冲突节点数目, 并获得较好的网络性能。

关键词: 0-1 规划; 网络优化; 信道分配; Ad Hoc 网络; 冲突图

中文引用格式: 刘 蔚, 赵 宇, 陈 锐. 基于 0-1 规划的网络优化模型及其在信道分配中的应用[J]. 计算机工程, 2016, 42(5): 93-101.

英文引用格式: Liu Wei, Zhao Yu, Chen Rui. Network Optimization Model Based on 0-1 Programming and Its Application in Channel Allocation[J]. Computer Engineering, 2016, 42(5): 93-101.

Network Optimization Model Based on 0-1 Programming and Its Application in Channel Allocation

LIU Wei^{1,2}, ZHAO Yu¹, CHEN Rui¹

(1. Institute of Policy and Management, Chinese Academy of Sciences, Beijing 100190, China;

2. University of Chinese Academy of Sciences, Beijing 100049, China)

[Abstract] This paper proposes a resource optimization model on Multi-Radio Multi-Channel (MR-MC) wireless network for solving the problems of channel conflict and interference. The model uses 0-1 linear programming method to minimize network interference. The interference information between links is collected by the conflict searching tree structure in each cluster, and then the channel assignment algorithm is used to eliminate interference and optimize resources of the whole network. The optimization model is applied in the wireless Ad Hoc network, and experimental results show that this method performs much better than existing Cluster-based Channel Assignment (CCAS). It can improve the throughput by 89.5%, decrease of the channel queue length and reduce the conflict between nodes.

[Key words] 0-1 programming; network optimization; channel allocation; Ad Hoc network; conflict graph

DOI: 10.3969/j.issn.1000-3428.2016.05.016

1 概述

近年来, 无线通信技术发展迅猛, 渗透到日常生活的方方面面, 如蜂窝系统、无线局域网(WLAN)、无线广域网(WAN)、分组无线网(PNET)、无线自组织网络(Ad Hoc)等, 对于无线传输网络的研究是这些通信系统中最重要的部分。在无线网络的运行中, 网络节点的部署和有限带宽资源的分配利用, 都直接影响到网络质量。利用各种方案对无线网络的

有限资源进行管理和优化也成为当前研究的热点。Ad Hoc 网络是一类典型的无线自组织网络, 是由一组具有收发装置的移动节点自组织形成的无线通信网络。由于 Ad Hoc 网络的对称多跳性、分布式特性、自组织性等特性^[1], 使它可以缩小每个中间节点的发射范围, 降低设备功耗, 节省网络能量^[2], 为它在民用和军事通信领域的应用提供了优势; 但同时也存在网络带宽低、信道质量差, 以及无线信道传输产生的竞争冲突、隐藏节点和无线信号多径衰落等

基金项目: 国家科技支撑计划基金资助项目(2012BAK27B00); 国家质检公益性科技计划专项课题基金资助项目(Y300571801); 中国科学院重大科技任务专项基金资助项目(Y201161Z04)。

作者简介: 刘 蔚(1986-), 女, 博士研究生, 主研方向为网络优化、大数据; 赵 宇(通讯作者), 助理研究员、博士; 陈 锐, 研究员、博士。

收稿日期: 2015-02-02 **修回日期:** 2015-03-26 **E-mail:** takashimirai@163.com

问题。随着多媒体业务的迅猛增长及无线网络的商业化发展,用户对多样化业务的需求与日俱增,因此,如何更有效公平地利用有限的网络资源成为亟需解决的问题^[3]。在众多的解决方案中,网络分簇和多射频多信道(Multi-Radio Multi-Channel, MR-MC)都被证明是管理和优化自组织网络资源的有效方法。

分簇算法是无线网络中的一个活跃的研究领域。平面结构无线网络中所有节点地位相等,且网络规模庞大,在这样的网络环境中执行算法和协议,往往会产生较高的控制负载,以及拓扑变化频繁、链路中断等问题。针对这些问题,可以采取对网络结构进行分簇的方法,将网络中的节点划分为相对独立的簇结构,能缩减网络规模,减少节点能耗,更加有效、安全的运行路由协议、数据收集以及安全机制等。另外,在无线网络中配置多射频多信道也是一种有效的资源优化技术。已有多位学者通过不同的研究内容和研究方式证明,在无线网络中为节点配置多射频多信道环境,使节点通过转向多条不重叠信道进行通信,可显著增加网络容量^[4-6]。IEEE 802.11a、IEEE 802.11b以及IEEE 802.16等标准协议中对于多信道都做出了相应设定。目前,对于多射频多信道无线网络的研究,主要集中在如何合理地分配数量有限的可用信道,达到最大限度地减小无线链路之间干扰的目标^[5-8]。

针对无线网络中最关键的资源分配和网络优化问题,本文提出一种在分簇的多射频多信道 Ad Hoc 网络(MR-MC Ad Hoc)中适用的,且基于冲突搜索树结构零一规划的网络优化模型,并给出信道分配算法。

2 相关研究

由于无线网络具有独特的优势,同时也存在需要解决的问题,分簇和多信道技术已经成为优化无线网络资源的重要技术手段,同时,也是近年来无线网络的研究热点方向。已有大量国内外的研究人员在这方面做出了探索和研究。

最小 ID 分簇算法^[9-10]将相邻节点中具有最小 ID 的节点作为簇首,将一跳邻居节点作为簇成员节点。这种算法计算简单、易于实现,但没有考虑公平性因素和负载平衡问题。链路分簇算法^[11]是较早提出的分簇算法之一,在这种分簇算法中,选举一定范围内 ID 最高的节点作为簇首。这种方法简单易实现,且计算量小,但往往会产生较多的簇首。最高连接度分簇(Highest Connectivity Cluster, HCC)算法也是比较常用的分簇方法之一^[12]。在这种分簇算法中,具有最高连接度的节点会被选为簇首节点,簇

首将其覆盖范围内的节点选为其簇成员。另外,文献[13]提出一种按需加权的分簇算法以提高网络体系结构的稳定性。文献[14]提出一种分簇融合方法,采用最大熵聚类法和蚁群算法实现分簇和节点访问顺序规划。文献[15]则针对单一结构提出了一种兼容弱联通簇的分簇算法,选择更优节点作为簇首从而获取更好的稳定性和持续性。

多射频多信道(MR-MC)技术也是优化无线网络容量的重要手段之一。文献[5]提出一种支持多播无线网络的信道分配算法,文献[6]设计一种满足公平性的最小生成树信道分配算法。文献[16-17]通过设计最佳吞吐量优化的跨层多信道算法,实现网络容量最大化的目的,但这些算法都是高复杂度的集中式算法,对于以分布式为特点的 Ad Hoc 无线网络来说,很难执行,算法的复杂度较高。文献[18-19]算法则需要收集先前的网络拓扑信息,增加了网络的控制负载量。

3 网络优化模型和信道分配过程

3.1 网络模型设定

利用最高连接度分簇算法 HCC,将无线网络划分为逻辑上不交叠的独立簇结构。采用簇首的统筹管理特性,收集簇内链路拓扑构建冲突图,并在冲突图的干扰关系基础上,建立冲突搜索树结构,通过直系父-子节点关系体现实际链路拓扑中的干扰关系,从而对无线网络的拓扑结构和冲突关系进行优化,并为网络优化模型和执行信道分配算法奠定基础。

设定网络模型,以 Ad Hoc 网络为例。假设在本文所设定的 MR-MC Ad Hoc 网络中,共有 n 个实际节点,组成 l 条链路。其中,每条链路都由一个发送节点 $t(l)$ 和一个接收节点 $r(l)$ 组成。并且:

(1)网络中节点移动性不强,节点不会在短时间内频繁移动位置。这样,可以保持链路和网络拓扑在一段时间内稳定存在,缩短算法更新时间。

(2)网络中每个节点都配置有多个射频接口,并工作于全双工状态;每个节点的接口数目可以不同,但某一特定时刻,一个射频接口只能转向一条固定信道。

(3)设定节点的感知半径为传输半径的 2 倍,并且所有节点拥有相同的收、发功率,感知半径和传输半径。

(4)假设网络中已有合适的单径路由算法为需要通信的源节点和目的节点选择了路径,建立通信链路(例如动态源路由(Dynamic Source Routing, DSR)协议或者基于距离数组的按需路由(Ad Hoc On-demand Distance Vector Routing, AODV)协议)。

(5) 选用最大连接度分簇算法 HCC 对网络进行分簇,即选用连接度最高的节点作为簇首。由于簇范围和簇成员的数目会直接影响到簇内节点以及干扰链路的数目,因此在实验中设定了 2 种情况:当 $R = 1$ 时,所有节点不是簇首,即是离簇首最多一跳的距离;当 $R = 2$ 时,所有节点不是簇首,即是离簇首最多两跳的距离。在后续对比实验中,也采取了 2 种范围的分簇方法。

另外,由于 HCC 算法在对网络中节点进行分簇时会在地理位置上产生部分交叠,本文对交叠部分的节点进行了归属划分:将交叠部分的节点划分给交叠的簇中包含簇成员节点个数较少的簇。另外,在实际分簇过程中还会产生一些独立节点,本文在算法的实现过程中将独立节点划分为独立簇,并将它自己设定为簇首。

3.2 冲突图和搜索树

3.2.1 冲突图

在无线网络中,可以用冲突图的方法反映实际

网络中的冲突干扰关系^[17-18]。根据网络模型,在网络初始化过程结束后已经完成分簇,并形成相对稳定的通信链路和网络拓扑。每个独立簇结构的簇首节点收集所有簇成员的拓扑信息,并根据这些链路和拓扑信息,生成以簇为单位的冲突图 $G_C(V_C, E_C)$ ^[19]。

冲突图含义解释如下:(1) 顶点 V_C 表示拓扑图中的一条链路。 $V_C = \{l_{ij} | (i, j)$ 是拓扑图中一条已建立的通信链路}。(2) 边 E_C 表示在拓扑图中 2 条链路之间存在干扰关系。 $E_C = \{l_{ij}, l_{mn}$ 之间存在干扰 $|, (i, j), (m, n)$ 是已建立的通信链路}。

还需要注意分簇中出现的边界干扰链路问题。当一条链路的 2 个端点(即是发送节点和接收节点)分别属于 2 个不同的簇时,这条链路就是边界链路,体现在冲突图中就是边界节点。如图 1 所示,簇 CL1 和簇 CL2 中存在边界链路 bm ,边界链路会同时对簇 CL1 和簇 CL2 中的链路产生干扰,这种干扰关系在冲突图中用虚线表示。

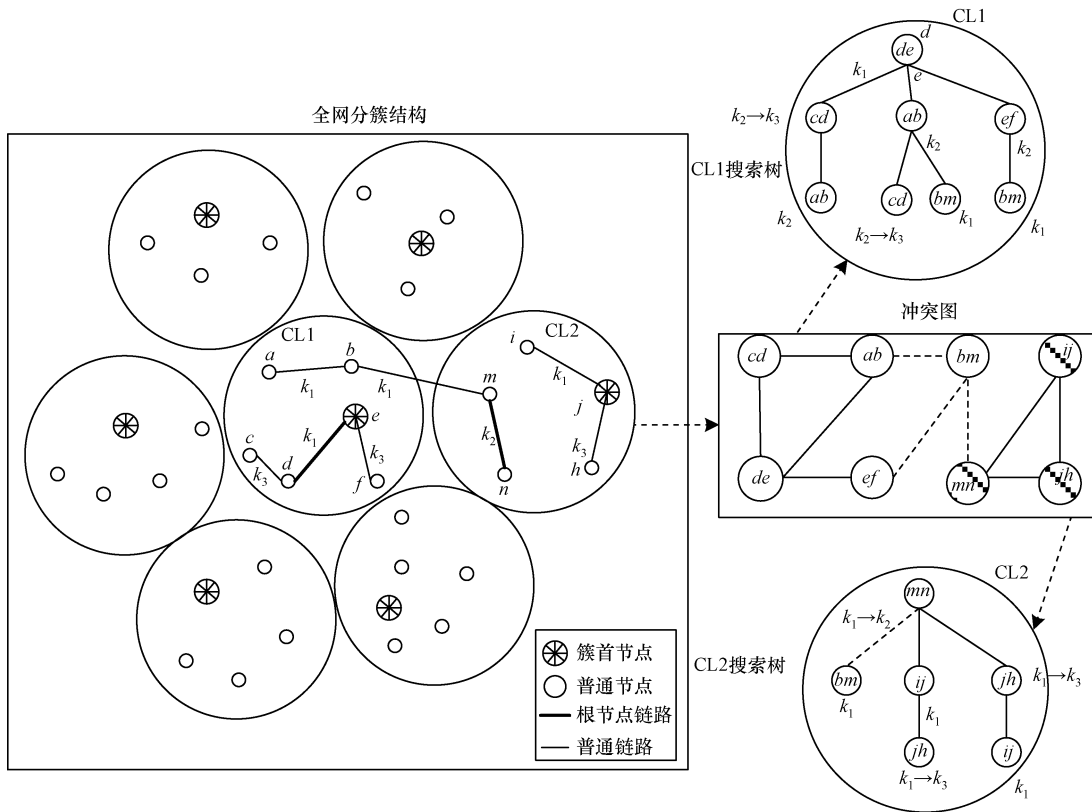


图 1 网络拓扑构建和搜索树的信道分配过程

3.2.2 搜索树的构建

簇首根据输入的冲突图信息构建冲突搜索树结构,其步骤如下:

(1) 构建过程从冲突图中干扰链路最多的节点开始,并将其作为根节点,即就是拥有干扰边最多的节点。例如,根据图 1 中的冲突图,簇 CL1 中的链路

de ,在感知范围内有 3 条干扰链路,干扰度最大,选择该节点作为搜索树结构的根节点。

(2) 将除它自己外所有与根节点有干扰边关系的节点作为其子节点展开。例如图 1 中,与根节点 de 存在干扰边的节点有 ab, cd 和 ef ,将这 3 个节点作为 de 的子节点展开。

(3) 找出除根节点 de 之外, 分别与节点 ab, cd 和 ef 存在干扰边的节点, 作为它们的子节点, 继续构建下层搜索树。按照这种方法向下延伸, 对于搜索树结构中的每一个节点, 它所有的子节点都应为除了父节点之外其余所有与它存在干扰边的节点(根节点只有子节点)。例如在图 1 中, 与节点 ab 存在干扰边的节点 cd 和 bm 都作为 ab 的子节点展开。

(4) 按照上述步骤中描述的过程逐层构建搜索树中所有的节点, 该过程一直持续到某个子节点已在上层父节点中出现过, 或者除其父节点以外, 该节点的所有子节点均不属于本节点所属的簇为止。例如图 1 中, 节点 cd 作为节点 ab 的子节点出现在搜索树中, 而它自身也只有 bm 干扰节点, 则节点不再展开; 节点 bm 有干扰边关系的节点 ab 和 mn 中的节点 ab 都是 bm 节点, 而 mn 属于簇 $CL1$ 将不再展开。

(5) 如果某条干扰链路是边界链路, 那么该链路的代表节点在构建冲突图时被“标记”为“边界节点”。这样的链路在通信时会影响到 2 个簇, 所以, 在构建搜索树结构时也会同时被其端点所属的 2 个簇管理, 在后续的信道分配过程中, 也会赋予边界节点更高的优先级。图 1 中, 显示了全网节点进行分簇, 簇首收集簇内链路冲突信息, 并构建冲突搜索树结构的过程。

3.3 网络优化模型

本节根据各个簇内建立的搜索树结构, 进行网络信道资源优化的建模过程。

假设在网络中, 共有 N 个节点构成网络拓扑, 形成 l 条链路, 经过分簇后被划分为 m 个独立簇结构, 全网可用信道为 K 条。在网络初始化后, 由各个簇首节点收集簇成员信息, 并构建冲突搜索树结构。

网络模型参数设定如下:

(1) 对于簇首节点

由簇成员节点的网络拓扑图生成冲突图, 再由簇首节点建立搜索树结构, 并将该结构作为模型输入参数。

(2) 网络优化过程

建立分配函数: $f: V_T \rightarrow \kappa$; 使得搜索树结构中的每一个节点都被分配可用信道集合 $\kappa = \{k_1, k_2, \dots, k_r\}$ 中的一条信道。

(3) 网络优化目的

在满足网络连通性和节点接口约束条件的前提下, 使处于各自干扰范围内的链路都能够使用唯一的信道进行通信。

(4) 接口约束条件

$$\forall i \in \mathbf{N}, \text{ 都有 } \{k | f(e) = k, k \in \kappa, e \in E(i)\} \leq R_i \quad (1)$$

(5) 目标函数(网络干扰函数 $I(f)$)

$$I(f) = \{(u, v) | (u, v) \in E_T, f(u) = f(v) = k | k \in \kappa\} \quad (2)$$

本文根据搜索树的节点干扰关系, 将整个网络优化模型设定为 0-1 线性规划问题, 用该理论模型表达网络资源优化过程目标和约束条件, 该模型的最优解就是本文设计算法所能够达到的网络优化性能边界值。以下设定优化模型中所使用的变量, 这些变量均为整数变量, 取值为 0 或 1。

根据网络模型设定, V_T 是搜索树结构中所有节点的集合, E_T 是搜索树结构中所有干扰边的集合。其中, $u, v \in V_T$ 表示搜索树结构中任意一个节点(它们同时也代表实际拓扑中的一条链路); 假设 $k \in \kappa = \{k_1, k_2, \dots, k_K\}$, 是网络中所有可用的 K 条信道中任意一条; R 为节点 v 中所含网络真实节点的接口数目, 并且, $R = \min\{(i, j) | i, j \in v\}$, 定义:

(1) $X_{vk} \in \{0, 1\}$, $X_{vk} = 1$, 如果节点 $v \in V_T$ 被分配了信道 k , 那么 $X_{vk} = 1$ 。

(2) $Y_{uv} \in \{0, 1\}$, $Y_{uv} = 1$, 如果边 $(u, v) \in E_T$ ($u, v \in E_T$) 且节点 $u, v \in V_T$ 分配了相同信道, 那么 $Y_{uv} = 1$ 。

(3) $W_{ik} \in \{0, 1\}$, $W_{ik} = 1$, 如果对于节点 $i \in N$, 信道 $k \in \kappa$, $u \in E(i)$ 被分配了信道 k , 那么 $W_{ik} = 1$ 。

(4) $Q_{uv} \in \{0, 1\}$, $Q_{uv} = 1$, 如果 u, v 是同一层的节点, 那么 $Q_{uv} = 1$, 由于一个节点可以与多个节点之间产生干扰关系, 因此同一个节点可能会出现在不同层。

(5) M_u : 表示所有和节点 u 处于同一层的节点集合, 其中, $u \in V_T$ 。

根据变量设定, 构建网络优化模型:

$$\min I(f) = \min \left(\sum_{(u, v) \in E_T} Y_{uv} \right) \quad (3)$$

$$\text{s. t. } \sum_{k \in \kappa} X_{vk} = 1, \forall v \in V_T \quad (4)$$

$$Y_{uv} \geq X_{uk} + X_{vk} - 1, \forall (u, v) \in E_T, \forall k \in \kappa \quad (5)$$

$$W_{ik} \leq \sum_{v \in E(i)} X_{vk}, \forall i \in N, \forall k \in \kappa \quad (6)$$

$$M_u = \sum_{v \in V_T} Q_{uv} \quad (7)$$

$$\sum_{v \in V_T} Q_{uv} Y_{uk} \geq \sum_{M_u} Y_{M_u k}, \forall u \in V_T \quad (8)$$

$$\sum_{j=1}^k W_{ij} \leq R_i, \forall i \in N \quad (8)$$

其中, 式(3)目标函数中 $\min I(f)$ 表示网络中干扰量最小化的优化目标, 在簇结构中, 是使搜索树中使用同样信道的直系父-子节点对数最小化的目标过程; 约束条件式(4)的含义是, 所有搜索树结构中的节点 $\forall v \in V_T$ 都要被遍历到, 保证搜索树结构中每一个节点都被分配一条可用信道进行通信; 约束条件式(5)是表达搜索树结构中 2 个不同的节点 u 节点 v 间的

信道关系,2 个节点 u 和 v 之间的信道相同,则变量 Y 值为 1,如果不同,则取值为 0,对于 2 个节点,只知道它们之间的信道关系还不能够判断它们之间是否存在干扰,同时还要结合它们的位置关系才能够判定,因为只有使用同一条信道的直系父-子节点之间才会产生冲突;约束条件式(6)是信道约束条件,即网络拓扑中的节点所分配的总信道个数,不能多于由该节点组成的链路所分配的信道个数;约束条件式(7)是描述搜索树中同层节点之间的分配关系,即同层节点可以使用相同的信道, M_u 表示对每一个节点 $u \in V_T$,遍取所有的节点 v 使得节点 v 和节点 u 同层节点;约束条件式(8)是接口约束条件,即网络拓扑中的每个节点所分配的信道总数不能超过其拥有的射频接口数目。

3.4 基于搜索树的信道分配过程

本节将根据上文中网络优化模型设定的约束条件和目标函数,利用独立簇结构中收集的冲突搜索树信息,确立簇内各条链路之间的干扰关系,进行信道分配过程。信道分配过程在每个独立的簇结构内进行,由簇首的算法服务器在簇内进行相对集中式的信道分配过程。而对于全网来说,信道分配又是分散在各个独立簇结构中进行的,是相对分布式的算法。因此,本文设计的信道分配算法是对于全网分布且各簇相对集中的算法。

根据本文建立的网络优化模型,设计一个贪婪规则,遍历搜索树结构中的所有节点,并进行信道分配,具体如下:

(1) 整个信道分配过程首先从根节点开始,按照从顶至底的顺序逐层进行信道分配,每次分配时选取剩余信道中质量最好的一条进行分配。对于信道质量,有很多参数可以作为参照标准,常用的有可用剩余信道带宽、丢包率等。本文结合信道丢包率以及信道使用次数作为信道质量的衡量标准。对可用信道的丢包率 $P_i(C_i)$ 进行统计计算:

$$P_i(C_i) = \frac{Y(C_i)}{Y(C_i) + X(C_i)} \quad (9)$$

其中, $Y(C_i)$ 是丢失的数据包数目; $X(C_i)$ 表示接收到的数据包数目,这个值可以从 MAC 层获取,并根据其历史统计平均值求得。

而信道使用次数,则是指在干扰范围内信道被使用过的次数。每次为节点分配信道时,都从其剩余信道集合中选取丢包率 $P_i(C_i)$ 最小,使用次数 n 少的信道为节点进行分配。特别的,如果一条可用信道需要在其丢包率和使用次数之间进行协调时,则选择使用次数最少的信道,在实验中发现,选择这样的信道在消除干扰方面具有更好的性能。

(2) 对根节点分配完信道后,再对其子节点进行逐层分配。每一层的分配次序根据节点 ID 进行,并由 ID 值大的节点开始。该 ID 值在建立搜索树时根据节点连接度进行标记。按照这个原则,在每一层中按照 ID 值的降幂顺序对节点依次进行信道分配。

(3) 除根节点外,每个节点的信道分配原则是不能选取与自己直系父节点相同的信道。即对于子节点进行信道分配前,首先将其直系父节点所占用的信道从可用信道集合中剔除,然后在新的可用信道集合 κ' 中选取质量最好的一条信道。

(4) 信道更新过程,当一个节点在信道分配时,发现该节点已经标记有信道 k_{ch} ,这说明该节点在上层搜索树结构中已经出现过。这是由于一个节点会同时跟多个节点产生干扰关系所造成的。如果节点已被分配了信道 k_{ch} ,需要启动信道更新过程,即当子节点的已标记信道与父节点相同时,将子节点与父节点的 ID 值进行比较,然后将 ID 值较小的节点进行信道更新。

(5) 按照如上步骤逐层进行信道分配并遍历所有节点,直到搜索树结构中的每一个节点都被分配一条可用信道为止。特别的,如果算法执行到某一个节点时,已经没有信道可为其分配,就为该节点随机分配一条在干扰范围内使用次数最少的信道。

3.5 算法收敛规则

在进行信道分配时,需要考虑涟漪效应和振荡问题^[4]。本文在算法中补充 2 个收敛规则,来防止由于信道更新过程产生涟漪效应和信道振荡:

(1) 边界节点最优原则

为了防止边界节点的信道更新引起 2 个甚至更多簇的节点信道发生变更,因此,应在分配时首先给予边界节点更高的 ID 值,一旦边界节点分配了一条信道,就不应该轻易改动。如果该节点的信道与其他节点所分配的信道产生了冲突,则更改其他节点的信道来消除冲突。

(2) 干扰度最小原则

每一层的信道分配完成之后,可以得到该层节点与上层节点之间产生的干扰总值。在实际网络中,节点可用的正交信道数目是有限的,并且 2 条信道之间总会存在一定干扰,因此,在信道分配完成后,网络中干扰总会存在。如果更新信道后的干扰总值 μ' 大于更新前网络中的干扰总值 μ ,则此次更新作废;否则就进行更新。即每层的信道分配完成之后要保证网络中干扰度最小的原则。

4 对比实验

4.1 实验环境和参数设置

本文设计一种应用在无线网络环境中基于冲突

搜索树的零一规划的网络优化模型和信道分配过程(以下简称为 TBCA)。在这一节中,将使用 Matlab 仿真软件模拟 MR-MC Ad Hoc 无线网络环境,实现本文设计的信道分配算法和网络优化模型,并使用相关领域的对比协议基于分簇的信道分配(Cluster-based Channel Assignment, CCAS)算法^[20],通过干扰量、网络吞吐量等网络性能参数的对比,验证和进一步分析本文设计的算法性能。实验环境和相应参数设定如表 1 所示。

表 1 实验环境和相应参数设定

参数名称	参数取值
仿真网络范围	1 000 m × 1 000 m
节点总数	25 个 ~ 50 个随机
节点分布	均匀分布
MAC 层协议	IEEE 802.11b DCF
节点传输半径/m	250
载波侦听半径/m	500
数据包链路速率/(Mb · s ⁻¹)	1
最小、最大竞争窗口/bit	32, 1 024
数据流设定	UDP
数据包包长/Byte	1 000
实验场景	正交和非正交信道
信道设置	11 条信道, 3 条为完全正交, 8 条为部分交叠
解析效率下降值(非正交信道)	$\mu = [0.6, 0.4, 0.2, 0]$
信道选用频段	2.4 GHz, IEEE 802.11b
实验次数	每组 50 次, 95% 置信区间

在本节的实验中,选取常用的 3 个参数对网络性能进行对比分析,如下:

(1) 网络吞吐量,网络中所有节点在单位时间内传输的数据包比特总数。

(2) 信道队列长度,等待占有该条信道进行通信的所有节点队列中数据包的总长度。

(3) 干扰节点数量,在信道分配完成之后,网络中产生所有相互干扰的节点数量的统计平均值。

这 3 个参数分别从不同角度对网络性能进行评估,因此,综合使用这 3 个参数对算法性能进行分析。

本文采用了文献[20]提出的在 MR-MC 网络环境中执行的信道分配机制基于分簇的信道分配算法 CCAS 作为实验对比协议。该机制是一种不需要邻居节点位置信息,利用 2 个步骤(第 1 步通过为相邻簇分配完全正交的信道消除干扰;第 2 步通过在簇内划分分子干扰域进一步消除干扰)进行信道分配,从

而最小化网络干扰量的分配算法,主要机制如下:将网络划分为地理上不交叠的簇结构,每个簇使用一个与邻居簇完全正交的公共信道进行簇内通信,从而消除非协调干扰的存在,进而将簇划分为多个不同的干扰域,并充分利用信道多样性增加单个链路的容量。

4.2 网络拓扑优化实验结果

在网络初始化过程中,首先进行对整个网络进行拓扑优化。本文首先采用 HCC 分簇算法将网络划分为逻辑上不交叠的独立簇结构,并选用簇内连接度最高的节点作为簇首节点,簇半径范围设定,本文选用了一跳半径和两跳半径 2 种情况分别进行实验。每个独立簇结构的簇首节点掌控全簇节点冲突信息,在冲突图基础上构建搜索树结构,将簇内链路之间的干扰关系简化成直系父-子节点关系。

采用 HCC 分簇算法对网络原始节点进行分簇,簇半径分别选用一跳半径和两跳半径,网络分簇结果如图 2 和图 3 所示。在图中,黑色的实线圈表示传输半径一跳的范围;黑色虚线表示感知半径,也是传输半径两跳的范围;带星节点为簇首节点。

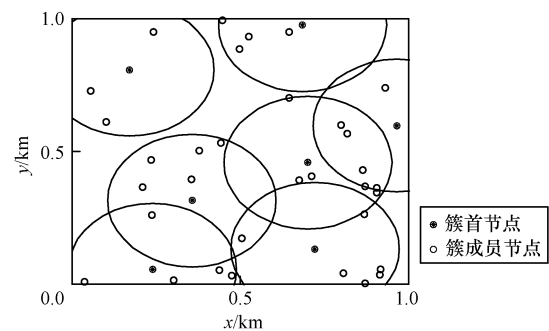


图 2 分簇半径为 1 跳范围的网络分簇结构

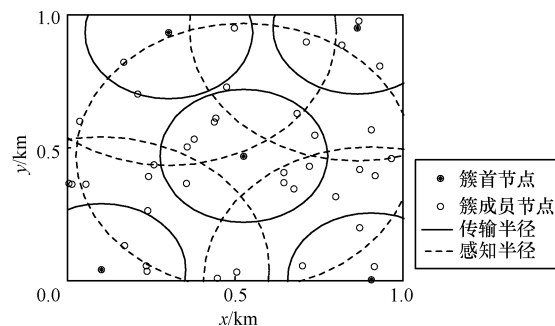


图 3 分簇半径为 2 跳范围的网络分簇结构

4.3 信道分配过程实验结果

本文考虑在网络环境中布置 3 条完全正交的信道和 11 条含有部分交叠信道的 2 种情况,分别实现对比 CCAS 算法、一跳分簇半径下的 TBCA 算法和 2 条分簇半径下的 TBCA 算法。统计参数为网络吞

吞吐量、信道队列长度以及干扰节点数量 3 项参数:

(1) 网络吞吐量统计结果

在正交与非正交信道条件下,信道分配算法能够达到的网络性能实验统计结果分别如图 4 所示。可以看出,单信道情况下的网络吞吐量始终处于比较低的状态,并且随着网络中节点数目的增加逐步下降。采用多信道的协议 CCAS 和 TBAC 在网络吞吐量方面都有很大提升,进一步说明了多射频多信道技术对于网络性能的提升作用。虚线表示对比协议 CCAS 的性能,无论是在正交信道还是非正交网络环境中,本文设置的 TBCA 算法都比 CCAS 的性能更加优越,而这种优势在非正交网络环境中体现的尤为明显。并且,无论是在正交还是非正交网络环境中,一跳分簇半径下运行的 TBCA 算法性能会略优于两跳分簇半径下运行的 TBCA 算法,说明在簇内节点较少的情况下,信道分配算法得到了更好的执行效果。

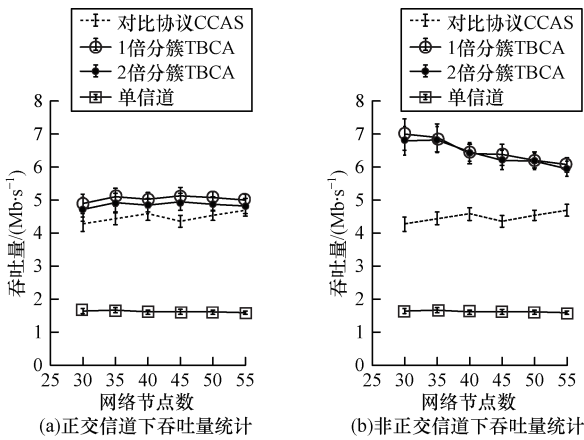


图 4 正交与非正交信道情况下的网络吞吐量统计

(2) 信道队列长度统计结果

图 5 中对正交与非正交信道网络环境下的信道队列长度进行了统计。可以看出,无论是正交还是非正交网络环境中,单信道情况下的信道队列长度都要比多信道算法中高很多,并且随着节点数目增多急剧增加。这是因为当网络中只有一条可用信道时,所有节点的传输都要抢占该信道,从而使得在该信道的等待队列不断增长。在正交信道环境下,CCAS 算法的信道队列长度略低于 TBCA 算法,这是由于 CCAS 算法在算法分配的第 1 步中首先将正交信道分配按照“顶点着色”分配给所有簇首节点,使得信道得到了充分利用。但另一方面,由于 CCAS 算法不能支持非正交信道的网络环境,因此在非正交信道条件下,CCAS 算法中信道队列长度要略高于本文设计的 TBCA 算法。

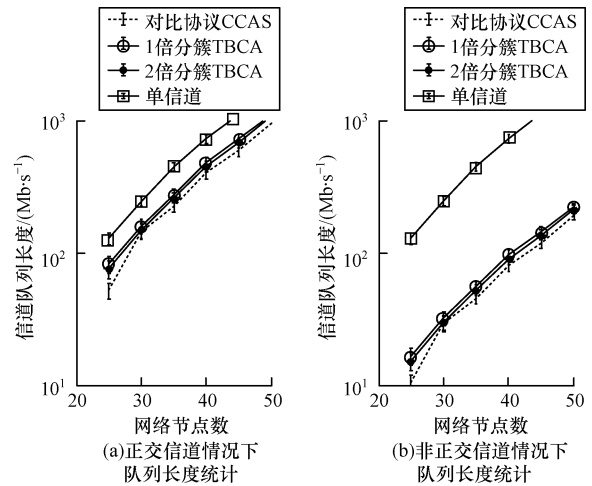


图 5 正交与非正交信道条件下的信道队列长度统计

(3) 干扰节点数量统计结果

图 6 中统计了在正交和非正交信道条件下,执行信道分配算法后网络中产生的干扰节点数量。理论上,在信道分配算法执行完后,网络中产生的总干扰节点数量应越小越好,这也是本文算法的优化目标。图 6 对于所有的信道分配算法,其产生的干扰节点数量都随着网络节点数目的增多而上升,其中,单信道条件下产生的干扰节点数目要远远大于多信道情况。另外,无论是在正交还是非正交信道环境下,对比协议 CCAS 要比 TBCA 算法产生的干扰节点数量略大,但这个优势并不明显。本文认为,如何在有限的信道资源下进行优化分配,已经被证明为是一个 NP-hard 问题,不同的信道分配算法采取了不同的思路求解,但都只能达到一个理论上的近似解,并不能完美地解决问题,这也说明在信道分配过程中可能存在一个受限于信道条数以及网络中其他因素的理论上限。

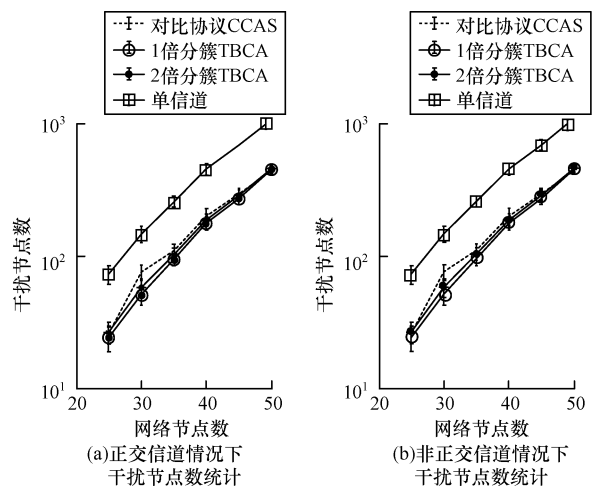


图 6 正交与非正交信道条件下产生的干扰节点数量统计

4.4 结果分析

通过在完全正交信道和非完全正交信道的网络环境下,分别运行单信道协议、对比协议 CCAS 算法以及本文设计的一跳 TBCA 算法和两跳 TBCA 算法,并对网络吞吐量、信道队列长度以及网络中的干扰节点数量进行了统计和性能对比。通过对比实验发现,单信道协议的网络性能会随着网络中节点数目的增多而逐步下降,对比多信道网络环境,它能够达到的网络性能非常低。这充分验证了多射频多信道技术在提升网络性能方面的优良特性。而本文设计的协议 TBCA 算法无论是在完全正交还是部分交叠的网络环境中,总体能够达到的网络性能都比 CCAS 算法要高;尤其在非正交网络环境中,TBCA 的性能要明显比 CCAS 性能好,统计结果如表 2 和表 3 所示。

表 2 单信道情况下的网络性能对比 %

对比算法	吞吐量 提升比率	信道队列 长度降低 比率	干扰节点 数量降低 比率
正交信道中 CCAS 算法对比 单信道协议	225	57.7	79.6
正交信道中 1 跳分簇的 TBCA 算法对比单信道协议	310	46.5	86.8
正交信道中 2 跳分簇的 TBCA 算法对比单信道协议	274	48.8	84.4

表 3 正交与非正交信道情况下的网络性能对比 %

对比算法	吞吐量 提升比率	信道队列 长度降低 比率	干扰节点 数量降低 比率
1 跳分簇 TBCA 算法对比 CCAS 算法	37.8	20.4	9.54
2 跳分簇 TBCA 算法对比 CCAS 算法	21.8	15.5	6.03
非完全正交信道中 1 跳分簇 TBCA 算法对比 CCAS 算法	78.6	44.3	15.40
非完全正交信道中 2 跳分簇 TBCA 算法对比 CCAS 算法	89.5	51.3	17.50

实验结果分析如下:

(1) 从实验结果的统计列表中可以看出,在网络中加入多射频多信道技术,并进行合理的信道分配之后,网络性能比单信道网络有了大幅度提升,CCAS 算法和 TBCA 算法对于网络吞吐量的提升分别达到了 225% 和 310%;信道队列长度平均降低了 50% 左右,干扰节点数量平均减少了 80% 左右。由此也验证了多射频多信道技术对于提升网络性能的重要作用。因此,选择一个合理的信道分配算法正是 MR-MC 无线网络提升网络性能的关键技术。

(2) 本文分别在完全正交信道条件和非正交信道条件 2 种网络场景下运行对比协议 CCAS 和本文

中提出的 TBCA 算法。在这 2 种场景下,TBCA 的效果均优于 CCAS 的效果。并且这种优势在非正交网络条件下体现的更加明显。如在表 3 中,正交信道条件下,一跳 TBCA 算法比 CCAS 算法达到的网络吞吐量提升了大概 37.8%;在非正交信道条件下,TBCA 算法比 CCAS 算法达到的吞吐量提升分别为 78.6% 和 89.5%。这是由于 CCAS 算法的执行,严格要求网络中所有的信道都是正交信道;而对于非正交信道网络环境,CCAS 算法能够达到的网络性能优势就不再明显。这也说明了本文设计的 TBCA 算法比 CCAS 算法更加适用于真实情况下的无线网络环境中。

(3) 为了分析簇范围大小对于算法效率和网络性能的影响,实验中同时运行了一跳分簇的 TBCA 算法和两跳分簇的 TBCA 算法,从整体性能看来,一跳分簇的 TBCA 算法都要略优于两跳分簇情况下的 TBCA 算法。如在表 2 中,一跳分簇的 TBCA 算法对比单信道网络,吞吐量提升了 310%,而两跳分簇 TBCA 提升了 274%。这是因为一跳分簇的簇范围较小,相对于每个独立簇结构,产生干扰的节点数目较少,搜索树结构较简单,所以信道分配算法的执行复杂度比较低。这个结果说明分簇会对 MR-MC 分簇无线网络的整体性能产生一定影响。另外,由于 CCAS 算法的分簇是基于节点地理位置,只适用于节点密度较低或者节点布置均匀的情况;如果节点密度较高或者布置不均匀,这种分簇方式就可能造成每个簇中的节点成员非常不均衡,也会对网络性能带来负面影响。

5 结束语

本文针对无线网络中的资源优化和信道分配问题,提出一种基于搜索树结构的 0-1 网络优化模型,并将其应用在 Ad Hoc 网络的信道分配中,从而达到在尽量减小链路干扰和冲突的条件下,优化网络信道资源分配,提高网络吞吐量的目的。通过仿真实验以及与相关算法对比分析,验证了本文网络优化模型和信道分配算法 TBCA 对网络性能提升的有效性。后续工作将在该理论优化模型的基础上增加数据,进一步逼近最优解和性能边界值,并考虑设计信道分配算法、链路调度机制,以进一步减少网络干扰,增加总体吞吐量。

参考文献

- [1] 易平,吴越. 无线自组织网络和对等网络——原理与安全[M]. 北京:清华大学出版社,2009.
- [2] 赵传信. 无线 Ad Hoc 网络跨层优化关键技术研究[D]. 苏州:苏州大学,2013.
- [3] 夏小霞,黄锐,粟根花,等. Ad Hoc 网络基于公平的带宽分配机制研究[J]. 数字通信,2014,41(3):44-47.

- [4] Raniwala A, Tzi-cker C. Architecture and Algorithms for an IEEE 802.11-based Multi-channel Wireless Mesh Network [C]//Proceedings of the 24th Annual Joint Conference of IEEE Computer and Communications Societies. Washington D. C., USA: IEEE Press, 2005: 2223-2234.
- [5] 邱振谋,姚国祥,官全龙,等.多信道无线 Mesh 网络的多播信道分配算法[J].计算机工程,2011,37(6): 107-109.
- [6] 邓雪波,王小强,陈曦,等.基于 QoS 和吞吐量公平的信道分配算法[J].计算机工程,2012,38(6):89-91.
- [7] 张彤芳. MIMC 网络中基于多径路由的跨层信道分配技术研究[D].南京:南京理工大学,2014.
- [8] Liu Lu, Cao Xianghui, Cheng Yu, et al. Energy-efficient Capacity Optimization in Wireless Networks [C]//Proceedings of INFOCOM'14. Washington D. C., USA: IEEE Press, 2014:1384-1392.
- [9] 赵志峰,郑少仁. Ad Hoc 网络体系结构研究[J].电信科学,2001,17(1):14-17.
- [10] Lin C R, Gerla M. Adaptive Clustering for Mobile Wireless Networks[J]. IEEE Journal on Selected Areas in Communications, 1997, 15(7):1265-1275.
- [11] Baker D J, Ephremides A. The Architectural Organization of a Mobile Radio Network via a Distributed Algorithm[J]. IEEE Transactions on Communications, 1981, 29(11):1694-1701.
- [12] 王海涛,张学平. Ad Hoc 网络中的分簇算法[J].数据通信,2003,32(4):32-35.
- [13] 杨卫东.考虑节点能量状态的 Ad Hoc 网络分簇算法[J].计算机工程,2010,36(12):119-122.
- [14] 王雪,王晟,姜爱国,等.无线传感网络中的分簇融合决策方法[J].控制与决策,2007,22(11):1208-1212.
- [15] 肖磊,符云清,钟明洋,等.兼容弱连通簇的 Ad Hoc 网络分簇算法[J].计算机工程,2013,39(6):107-110.
- [16] Bahl P, Chandra R, Dunagan J. SSCH: Slotted Seeded Channel Hopping for Capacity Improvement in IEEE 802.11 Ad-hoc Wireless Networks[C]//Proceedings of the 10th Annual International Conference on Mobile Computing and Networking. New York, USA: ACM Press, 2004:216-230.
- [17] Neely M J, Modiano E, Rohrs C E. Dynamic Power Allocation and Routing for Time-varying Wireless Networks [J]. IEEE Journal on Selected Areas in Communications, 2005, 23(1):89-103.
- [18] Alicherry M, Bhatia R, Li L. Joint Channel Assignment and Routing for Throughput Optimization in Multi-radio Wireless Mesh Networks [C]//Proceedings of the 11th Annual International Conference on Mobile Computing and Networking. New York, USA: ACM Press, 2005: 58-72.
- [19] Kodialam M, Nandagopal T. Characterizing the Capacity Region in Multi-radio Multi-channel Wireless Mesh Networks [C]//Proceedings of the 11th Annual International Conference on Mobile Computing and Networking. New York, USA: ACM Press, 2005:73-87.
- [20] Naveed A, Kanhere S S. Cluster-based Channel Assignment in Multi-radio Multi-channel Wireless Mesh Networks[C]//Proceedings of the 34th IEEE Conference on Local Computer Networks. Washington D. C., USA: IEEE Press, 2009: 53-60.

编辑 刘冰

(上接第 84 页)

- [8] Weil S A, Brandt S A, Miller E L, et al. Ceph: A Scalable, High-performance Distributed File System [C]//Proceedings of the 7th Symposium on Operating Systems Design and Implementation. Washington D. C., USA: IEEE Press, 2006:307-320.
- [9] Donovan S, Symposium L, Kleen A, et al. Lustre: Building a File System for 1,000-node Clusters [C]//Proceedings of Linux Symposium. Washington D. C., USA: IEEE Press, 2003:9-17.
- [10] Harter T, Borthakur D, Dong S, et al. Analysis of HDFS Under HBase: A Facebook Messages Case Study [C]//Proceedings of the 12th USENIX Conference on File and Storage Technologies. Washington D. C., USA: IEEE Press, 2014:199-212.
- [11] Siegel E H, Okasaki M E, Kumar P, et al. Coda: A Highly Available File System for a Distributed Workstation Environment [J]. IEEE Transactions on Computers, 1990, 39(4):447-459.
- [12] Rabin M O. Fingerprinting by Bandom Polynomials, TR-CSE-03-01[R]. Cambridge, USA: Harvard University, 1981.
- [13] Standard S H. FIPS 180-1 Secure Hash Standard [S]. Department of Commerce, National Institute of Standards and Technology, 1995.
- [14] 林雅榕,侯整风.对哈希算法 SHA-1 的分析和改进[J].计算机技术与发展,2006,16(3):124-126.
- [15] Szeredi M, Szeredi M. FUSE: Filesystem in Userspace [C]//Proceedings of the Usenix Winter Technical Conference. Washington D. C., USA: IEEE Press, 2009: 159-166.
- [16] 龚辉,徐学洲,曹荣禄.基于 Ramdisk 的全内存式 Linux 系统的设计与实现[J].微机发展,2005,15(4): 75-77.
- [17] IOzone [EB/OL]. [2015-03-20]. <http://www.iozone.org>.

编辑 索书志