

基于 I-序列顶点着色算法的 N-时分多址动态时隙分配协议

王灵芝¹, 陈凌宇², 袁艾莎²

(1. 闽南师范大学物理与信息工程学院, 福建 漳州 363000;
2. 厦门大学信息科学与技术学院, 福建 厦门 361102)

摘 要: 针对移动自组网中介质访问控制层协议的特性, 结合实际工程需求, 基于 I-序列顶点着色算法, 提出一种 N-时分多址动态时隙分配协议。对协议的帧结构、交互过程及动态时隙分配算法进行分析。在 NS2 软件上的仿真结果表明, 该协议的时隙利用率、吞吐量、平均时延、公平性等均优于 IEEE802.11 和时分多址协议。

关键词: 移动自组网; 时分多址; 序列顶点着色; 动态时隙分配; NS2 仿真

中文引用格式: 王灵芝, 陈凌宇, 袁艾莎. 基于 I-序列顶点着色算法的 N-时分多址动态时隙分配协议[J]. 计算机工程, 2016, 42(1): 89-94, 102.

英文引用格式: Wang Lingzhi, Chen Lingyu, Yuan Aisha. N-time Division Multiple Access Dynamic Slot Allocation Protocol Based on I-Sequential Vertex Coloring Algorithm[J]. Computer Engineering, 2016, 42(1): 89-94, 102.

N-time Division Multiple Access Dynamic Slot Allocation Protocol Based on I-Sequential Vertex Coloring Algorithm

WANG Lingzhi¹, CHEN Lingyu², YUAN Aisha²

(1. School of Physics and Information Engineering, Minnan Normal University, Zhangzhou, Fujian 363000, China;
2. School of Information Science and Engineering, Xiamen University, Xiamen, Fujian 361102, China)

[Abstract] According to the characteristics of Medium Access Control (MAC) layer protocol in mobile Ad Hoc network, combining with the actual engineering requirements, N-time Division Multiple Access (TDMA) dynamic slot allocation protocol based on I-sequential Vertex Coloring (SVC) algorithm is proposed. The protocol frame structure, interaction and dynamic slot allocation algorithm are specifically analysed and described. This protocol is simulated under NS2, results show that the algorithm has better performance on high slot utilization, throughput, average latency, fairness and other parameters than IEEE802.11 and Time Division Multiple Access (TDMA) protocols.

[Key words] Ad Hoc network; Time Division Multiple Access (TDMA); Sequential Vertex Coloring (SVC); dynamic slot allocation; NS2 simulation

DOI: 10.3969/j.issn.1000-3428.2016.01.017

1 概述

在移动自组 (Ad Hoc) 网中, 介质访问控制 (Media Access Control, MAC) 层的性能对整个网络性能的提高具有重要影响, 对该层协议的设计是 Ad Hoc 面临的关键技术问题。随着时隙同步问题的解决, 时分多址 (Time Division Multiple Access, TDMA) 再次成为研究热点, 其相对于常见的 IEEE802.11 协议而言, TDMA 类的协议在工程上简单实用, 仪器设备使用寿命长^[1]。本文旨在为宽带无线自组织系统设计一个有效的 MAC 层协议。

不同的 TDMA 协议对帧结构的设计不同, 时隙调度方式也不同, 可以将 TDMA 方式下的协议分类如下^[2-3]: (1) 固定时隙分配方式。该方式下帧的长度由网络中节点总数决定, 接入延时和帧长度相关, 所以当节点业务量增大或者变小的时候, 无法动态满足变化, 会造成时隙浪费或者过载, 信道利用率低。但该方法的分配方式在工程中实现起来较容易, 节点公平性好, 适用于中小规模并且业务量较均匀的网络。(2) 动态时隙分配方式, 即时隙分配协议自适应于拓扑结构和瞬时业务的变化, 基于预约或竞争机制解决时隙分配问题, 其典型代表协议有 FFRP

基金项目: 厦门大学海西通信工程技术中心开放基金资助项目 (HXCXJJ2014-012)。

作者简介: 王灵芝 (1981 -), 女, 讲师、硕士, 主研方向为移动自组网、物联网; 陈凌宇, 助理教授、博士; 袁艾莎, 硕士研究生。

收稿日期: 2015-06-25 **修回日期:** 2015-08-24 **E-mail:** 64564254@qq.com

(Five Phase Reservation Protocol)^[4], E-TDMA^[5], HRMA^[6]等。虽信道利用率增加,但需要增加额外的系统开销,工程实现复杂。

在动态分配算法上,有集中式和分布式 2 种方式^[7],对于集中式算法,选取一个节点作为网络的中心控制节点,该节点能够获取整个网络的所有节点信息,统一分配时隙。该方法时隙利用率高,但中心控制节点获取全网信息开销较大,若其工作不正常将导致全网的瘫痪。对于分布式算法,网络中的每个节点都将依据某些已经规定好的原则,逐一或者同时预留自己相应的传输时隙。如果增大网络规模,那么将增加为时隙分配而耗费的时间。除此之外,还要提前获取整个网络的规模、节点 ID 等一些网络信息,所以分布式算法不能很好地适用于规模较大或者规模变化的网络^[8-9]。本文拟将这 2 种方法相结合,提出一种符合工程应用的动态时隙分配算法。

2 N-TDMA 协议

2.1 设计目标

协议具备能够接入 4 个~32 个节点的能力,任意节点都可承担中心节点的功能,中心节点可通过交互手段获取全网拓扑状态,并根据 I-序列顶点着色 I-sequential Vertex Coloring (SVC)着色算法实现时隙的复用及动态分配。在设计与分析前,对网络模型提出假设:所有节点的时隙都同步于一个参考基准;每个节点有自己唯一的 MAC 地址;节点工作于半双工模式,即发送数据和接收数据不能同时进行;链路为对称链路。

2.2 帧结构

结合实际宽带自组织网络工程项目的需求,N-TDMA 协议的帧结构如图 1 所示。帧结构由控制子帧 CCH、接入子帧 RCH、业务子帧 TCH 组成。控制子帧 CCH 可分为 K 个子时隙,从 1~ K 编号。接入子帧 RCH 只有一个时隙。业务子帧 TCH 由 n 个时隙组成,从 1~ N 编号。



图 1 N-TDMA 协议的帧结构

2.3 控制信道

假设网络中包含 N 个活动节点,对节点从 1~ N 编号,控制信道采用固定分配方式,为每一个节点分配唯一的控制时隙,若控制时隙数 K 小于总节点数

N ,未分配到控制时隙的节点在下一帧依序分配,直到所有节点分配完毕。如果控制帧存在空闲时隙,不做任何分配,从新的一帧开始依序进行新一轮分配,并循环该过程。

网络中每个节点在其控制时隙都要向邻居节点广播自己保存的邻居状态表和转发接收到的时隙分配表,并且如果接收到其他节点序号发生变化的邻居信息表时要在该时隙转发出去。经过多次交互,中心节点的时隙分配表最终收敛。

2.4 接入信道

接入信道为公共信道,该信道用于节点入网时所用,当入网节点数目少的时候冲突概率较小,当入网节点较多时,入网节点就会竞争该时隙,本协议采用随机接入信道竞争抢占机制 S-ALOHA。该机制需要新节点发出接入请求,中心节点收到请求后会分配一个新的控制时隙给它,并将这个信息广播出去,如果一段时间后新节点没有收到该信息,那么就认为入网请求信息发送冲突,退避一段时间再发。

2.5 业务信道

业务信道是业务数据发送信道,各节点根据中心节点所发的时隙分配表确定自身的业务信道,每个节点至少被分配 1 个时隙,每个节点在业务时隙发送消息,在非业务时隙侦听并且接收消息。

2.6 控制交互帧结构

节点之间信息的交互在控制信道完成,交互的信息由 2 个部分组成:邻居状态表和时隙分配表。图 2 为邻居节点状态表,其中,字段 *Index* 表示节点号 ($1 \leq Index \leq 32$);字段 *seq* 代表邻居节点状态表的新旧程度,值越大代表越新;*Node i state* ($1 \leq i \leq 32$) 表示节点 *Index* 与节点 *i* 是否为一跳邻居节点,值为 1 表示为互为邻居节点,为 0 反之。图 3 为时隙分配表结构,其中,字段 *Table start* 表示中心节点发送时隙表的时间,即时隙分配表的生效时间;字段 *TTL* 表示该表生存期,即有效时间。若节点收到表的时间超过有效期,则该表作废。字段 *traffic_num* 表示业务时隙个数,协议中取 8。字段 *Slot_i* ($1 \leq i \leq 8$) 共 32 bit,表示各节点占用时隙 *i* 的情况。若节点 *n* 占用时隙 *i*,对应第 *n* 比特位置 1,反之为 0。每个节点都需要维护这 2 个表,在自己的控制时隙根据不同的情况进行发送。

<i>Index</i>	<i>seq</i>	节点1 状态	节点2 状态	...	节点32 状态
5 bit	16 bit	1 bit	1 bit		1 bit

图 2 邻居节点状态表结构

Table start	TTL	Slot num	Slot_1	...	Slot_8
16 bit	8 bit	5 bit	32 bit		32 bit

图 3 时隙分配表结构

2.7 N-TDMA 协议交互过程

交互过程在时帧的控制信道进行,协议中一帧包含 8 个控制时隙。N-TDMA 协议的交互过程如图 4 所示。每个节点维护一个发送队列,队列包含需要更新的邻居状态表记录。

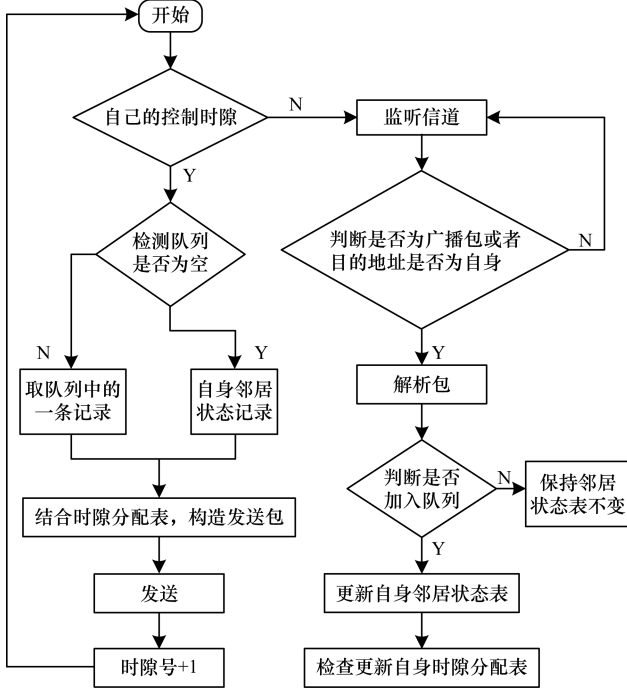


图 4 N-TDMA 协议的交互流程

若节点处于监听状态,接收到数据包后首先判断包的地址是不是自身或者广播地址,通过解析包头的序列号判断邻居状态是否发生变化。以此判断是否加入发送队列并更新自身邻居状态表;通过时隙分配表的生存时间判断是否更新自身时隙分配表。

若节点处于自身的发送时隙,首先检测发送队列是否为空,不为空则取出队列中的一条记录,并组合时隙分配表构造数据包并发送。否则,发送自身邻居状态记录和时隙分配表。

以 4 帧为一轮,各节点每隔一轮更新自身邻居状态记录,中心节点每隔 10 轮根据接收到的节点状态表得到全网拓扑信息,更新时隙分配表。由于各节点均可获得全网的拓扑信息,因此可以选择任意节点作为中心节点。

2.8 时隙动态分配

为了解决 TDMA 调度问题,可借鉴图论中图着色的方法。在图论中,图着色算法可实现任意相邻区域都具有不同的颜色。可以把时隙的分配看作为

图颜色的分配,这样可以在尽量避免冲突的前提下,得到较高的信道利用率^[10-12]。

图的着色问题是图论中一个经典的 NP 问题,可以把 AdHoc 网络的静态拓扑结构看作一个图 $G(V, E)$ 。其中,顶点 $V(1, 2, \dots, n)$ 表示网络中的节点; n 表示网络中节点的总数量;边 E 表示网络中所有链路的集合。假如网络中的节点 i 与节点 j 之间能够互相通信,那么就认为这两点之间存在一条无向边 $e_{i,j}$,且 $e_{i,j} \in E$,即节点 i 和节点 j 互为一跳邻居;互为一跳的邻居节点在同一时隙同一时刻传输数据会发生冲突;互为两跳邻居的节点在同一个时隙同时向同一个节点传输数据时也会发生冲突。互为两跳之外的节点相互间能够共用同一个时隙进行通信,这样就实现了空间的复用^[13-15]。

基于图论的典型时隙分配算法——序列顶点着色 (Sequential Vertex Coloring, SVC)^[16-18] 算法在确定最小帧长度的前提下,实现了信道利用率最大化。本文在 SVC 算法的基础上进行改进,并且结合实际工程需求,提出 I-SVC 算法。

3 I-SVC 算法

3.1 SVC 算法

在由 N 个节点构成的 Ad Hoc 网络中信道利用率可以用式 (1) 表示。

$$\rho_i = \frac{\text{节点 } i \text{ 被分配的时隙数}}{\text{TDMA 时帧中时隙个数}} = \frac{\sum_{m=1}^M t_{mi}}{M} \quad (1)$$

其中, t_{mi} 代表节点 i 占用时隙 m 。整个网络的信道利用率可以用式 (2) 表示。

$$\rho = \frac{1}{N} \sum_{i=1}^N \rho_i = \frac{1}{MN} \sum_{m=1}^M \sum_{i=1}^N t_{mi} \quad (2)$$

这样,面向节点的广播时隙分配问题 (Broadcast Scheduling Problem, BSP) 就能够用式 (3) ~ 式 (6) 来描述。

$$\text{Minimize } M \text{ and Maximize } \rho \quad (3)$$

$$\text{s. t. } C1: \sum_{m=1}^M t_{mi} \geq 1 (i = 1, 2, \dots, N) \quad (4)$$

$$C2: c_{ik}t_{mi} + c_{kj}t_{mj} \leq 1 (i, j, k = 1, 2, \dots, N, i \neq j, j \neq k, k \neq i, m = 1, 2, \dots, M) \quad (5)$$

$$C3: c_{ij} + t_{mi} + t_{mj} \leq 2 (i, j = 1, 2, \dots, N, i \neq j, m = 1, 2, \dots, M) \quad (6)$$

在上述式中, M 表示时隙个数, t_{mi} 代表节点 i 占用时隙 m 的情况, c_{ij} 代表节点 i 与节点 j 是否为一跳邻居, 1 表示是, 否则为 0。

将上述内容概述为在一定的约束条件下找到最小的帧长和最大的信道利用率。约束条件即为: (1) 时帧内的任一节点都能分配到至少一个时隙, 如约束 C1; (2) 互为两跳邻居的节点不可以被分配到同一个时隙, 如约束 C2; (3) 互为一跳邻居的节点不

可以被分配到同一个时隙,如约束 C3。在 Ad Hoc 无线网络中,只要保证两跳距离范围内的节点能够分配不同的时隙就可以避免冲突发生。那么就可以将时隙调度问题转化为图着色问题,时隙等同于图中的颜色,用最少的颜色对图着色等同于用最少的时隙构成帧结构。

均域退火 (Mean Field Annealing, MFA) 算法^[19]是一种基于着色理论的动态时隙分配算法,但 MFA 算法有 2 个缺点:(1) 它不能充分最小化帧长度;(2) 缺少确定最优参数的方法。

SVC 算法在 MFA 基础上改进了该算法。按照某种规定将网络中的 N 个节点分别编号为 $1, 2, \dots, N$ 。为第一个节点着色,编号为颜色 1,然后向下扫描节点列表,将最小的颜色序号分配给非一跳邻居的节点,然后重复这个过程,直到所有的节点都被着色为止。在变量排序标准中,通过减少一跳和两跳邻居的数目来获取最好的性能,也就是说,节点中一跳和两跳邻居数目最多的节点应该最先被写入列表中。

该算法由 2 个阶段组成,第 1 阶段确定最小帧长度 M 的前提下,第 2 阶段实现信道利用率最大化。

定义以下符号。

N : 网络中节点个数。

$NH(i)$: 与第 i 个节点互为一跳邻居或者两跳邻居节点的集合。

$CHECK(m, i)$: 检查第 m 个时隙是否可以分配给第 i 个节点,在节点集合 $NH(i)$ 中,是否有节点被分配到 m 时隙,如果没有,则返回 1,有则返回 0。

M : 由阶段 1 获得的时隙长度。

x_{mi} : 表示节点 i 占用时隙 m 。

第 1 阶段算法伪代码如下:

Step1 对节点的一跳和两跳邻居进行递减排序。

Step2 $m = 1, i = 1$ 。

Step3 If($CHECK(m, i) = 1$) then $x_{mi} = 1$, go to Step4.
Else go to Step4。

Step4 If($i = N$) STOP。

Else $m = 1, i = i + 1$, go to Step2。

Step5 $m = m + 1$, go to Step2。

第 2 阶段算法伪代码如下:

Step1 递增的排序一跳邻居节点和两跳邻居节点数量。

Step2 $m = 1, i = 1$ 。

Step3 If($CHECK(m, i) = 1$ and $x_{mj} = 0$)
 $x_{mi} = 1$ 。

Step4 If($m = M$ and $i = N$) STOP。

Else if ($m = M$ and $i < N$) then $m = 1; i = i + 1$, go to Step3。

Else then $m = m + 1$, go to Step3。

SVC 算法对节点进行了 2 次排序,但未考虑对

具有相同邻居个数的节点排序的优先级算法,而是简单按照节点号进行先后排序。3 种场景的拓扑结构如图 5 ~ 图 7 所示。选择如图 7 场景 3 的随机拓扑结构,利用传统的 SVC 分配算法得到的时隙分配结果如图 8 所示。结合图表分析可知,节点 7 和节点 8 可以继续分配时隙并且不影响网络中其他节点的时隙分配,说明传统的 SVC 算法对节点的时隙分配不够充分,并且可变帧长的分配算法在工程实践上可行性差,由此对该算法进行改进,提出基于固定帧长改进的 SVC 算法。

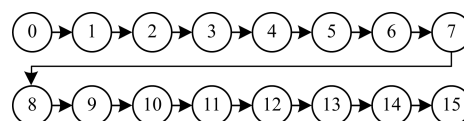


图 5 场景 1 拓扑结构

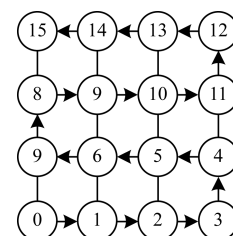


图 6 场景 2 拓扑结构

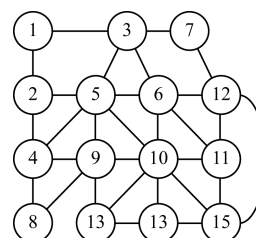


图 7 场景 3 拓扑结构

时隙 节点	1	2	3	4	5	6	7	8
1								
2								
3								
4								
5								
6								
7								
8								
9								
10								
11								
12								
13								
14								
15								

图 8 SVC 算法时隙分配表

3.2 I-SVC 时隙分配算法

针对 SVC 算法出现的问题,对第 2 阶段算法进行改进,在 Step1 中,根据 $NH(i)$ 将网络中各节点按照一跳邻居和两跳邻居节点总数进行递增排序。排序后若出现一跳邻居和两跳邻居节点总数相同的节点,再次根据一跳邻居总数递减的顺序重新排序,得到新的排序 i_1, i_2, \dots, i_N 。同样选择场景 3 的随机拓扑结构, I-SVC 分配算法得到的时隙分配结果如图 9 所示。可以明显看出,节点 7 和节点 8 再次被分配到新的时隙,有效解决了 SVC 算法的缺陷,着色更加充分,提高了网络的信道利用率。

节点 \ 时隙	1	2	3	4	5	6	7	8
1								
2								
3								
4								
5								
6								
7								
8								
9								
10								
11								
12								
13								
14								
15								

图 9 I-SVC 算法时隙分配表

表 1 对比了 I-SVC, SVC, MFA 3 种算法在图 5 ~ 图 7 的 3 种拓扑场景下得到的时隙总数与信道利用率。对比可以看出, I-SVC 算法在某些场景中优于 SVC 算法,整体高于 MFA 算法。

表 1 3 种算法的性能对比

场景	所需时隙总数			信道利用率		
	I-SVC	SVC	MFA	I-SVC	SVC	MFA
1	3	3	3	0.333 0	0.333 0	0.333 0
2	6	6	6	0.187 5	0.187 5	0.062 5
3	8	8	8	0.158 3	0.141 7	0.125 0

3.3 固定帧长的 I-SVC 时隙分配算法

上文介绍了可变的时隙分配算法,从工程实际角度出发,固定帧长度的时隙分配算法将更适用,选择固定业务时隙个数为 Z 个,那么该时隙分配问题即为在 Z 个时隙的情况下 N 个节点获得最大信道利用率的问题,此时的信道利用率修改如下:

$$\rho = \frac{1}{N} \sum_{i=1}^N \rho_i = \frac{1}{ZN} \sum_{m=1}^Z \sum_{i=1}^N t_{mi} \quad (7)$$

则该整数规划可以用式(8)表示。

$$\text{Maximize } \rho \quad (8)$$

以下是其对应的约束条件:

$$\text{s. t. C1: } \sum_{m=1}^Z t_{mi} \geq 1 (i = 1, 2, \dots, N) \quad (9)$$

$$\text{C2: } c_{ik} t_{mi} + c_{kj} t_{mj} \leq 1 (i, j, k = 1, 2, \dots, N, i \neq j, j \neq k, k \neq i, m = 1, 2, \dots, Z) \quad (10)$$

$$\text{C3: } c_{ij} + t_{mi} + t_{mj} \leq 2 (i, j = 1, 2, \dots, N, i \neq j, m = 1, 2, \dots, Z) \quad (11)$$

定义如下符号:

M :使用 I-SVC 算法后得到的时隙个数。

Z :业务时隙个数。

Step1 将 I-SVC 算法得到的时隙分配结果按照各时隙分配的节点数依次递减排序,得到新的排序 i_1, i_2, \dots, i_M 。

Step2 If $M > Z$, 跳到 Step3; else go to Step4。

Step3 将 $M-Z$ 部分的时隙分配到下一帧,依次循环,直至整数个业务时帧被填满。

Step4 对于 $Z-M$ 部分的时隙,重复 Step0 的排序结果,直到该业务时帧被填满。

4 基于 NS2 的 N-TDMA 协议仿真

利用 NS2 软件对设计的 N-TDMA 协议进行仿真验证, NS2 是目前网络模拟领域最广泛的开源软件之一,采用 C++ 与 OTcl 2 种开发语言进行开发工作, NS2 包括模拟事件调度器、网络组件对象库等结构。网络组件用来模拟网络设备或者节点之间的通信,可以通过搭建不同的模拟环境进行仿真^[20]。

仿真中协议分别采用可变帧长和固定帧长 2 种方式。并与 IEEE802.11 协议、TDMA 协议进行对比。仿真拓扑如图 7 所示,网络包含 15 个随机节点 Node1 ~ Node15, N 号节点向 $N+1$ 号节点发送数据。仿真中各节点数据流产生概率为泊松数据分布,数据包大小为 1 200 B,路由协议为 AODV,仿真时间 100 s。分别对饱和吞吐量、平均时延、接入公平性^[21] 3 个主要指标进行仿真。

图 10 为系统吞吐量随着业务量变化的仿真图,随着业务量的增加,吞吐量整体趋于饱和。可变帧长的 N-TDMA 协议和固定帧长的 N-TDMA 协议由于采用了 I-SVC 算法,性能最优。IEEE802.11 协议由于网络中业务量的增加导致交互信息碰撞增多,吞吐量中等而传统的 TDMA 因为不存在时隙复用所以吞吐量最低。

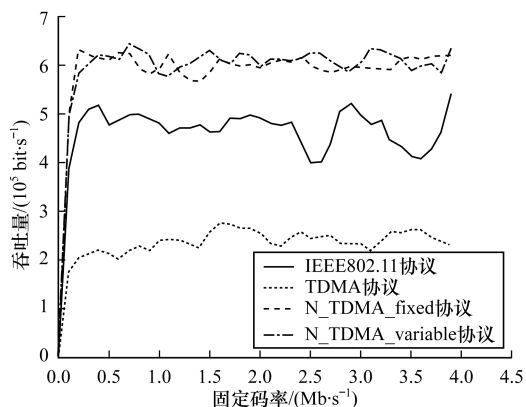


图 10 吞吐量仿真

从图 11 可以看出,随着业务量的增长,网络的平均时延趋于饱和。由于场景为随机拓扑结构,所有协议的时延均较大。固定帧长的 N-TDMA 协议和可变帧长的 N-TDMA 2 种协议具有相当平均的时延,IEEE802.11 协议由于退避机制,时延抖动得比较厉害。而原始的 TDMA 协议,由于时隙个数与节点个数相同,帧周期最长,导致时延最大。

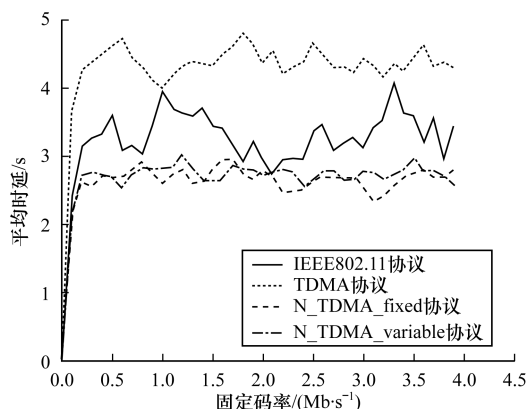


图 11 平均时延仿真

图 12 为 4 种协议的公平性比较,其中传统的 TDMA 协议每个节点拥有时隙个数均为 1,因此公平性最好。

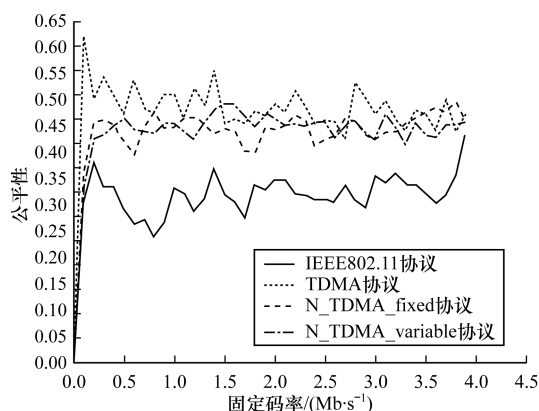


图 12 公平性仿真

固定帧长的 N-TDMA 协议和可变帧长的 N-TDMA 2 种协议存在时隙复用,公平性稍差,IEEE802.11 协议由于采用竞争机制,事先并不能给每一个节点分配信道,所以业务量的增大,碰撞增多,公平性降低。

5 结束语

本文在传统着色算法的基础上提出了适合 Ad Hoc 网络的 I-SVC 动态时隙分配算法。从实际工程需求出发,协议帧具有固定帧长结构,包含控制、接入及业务信道。根据全网拓扑结构对节点进行动态时隙分配。本文利用 NS2 软件对 3 种不同场景下系统的吞吐量、平均时延、公平性等性能指标进行仿真,实验结果表明本文提出的协议时隙利用率高,优于 IEEE802.11 和 TDMA 协议。如何进一步提高系统的收敛时间及随机选择中心节点是后续研究需要解决的问题。

参考文献

- [1] Lin Jing, Wu C, Ohzahata S, et al. A QoS Supporting Ad Hoc Network Protocol Combining Admission Based TDMA and 802.11 DCF[C]//Proceedings of the 16th Asia-Pacific Symposium on Network Operations and Management. Washington D. C., USA: IEEE Press, 2014:1-4.
- [2] Sgora A, Vergados D J, Vergados D D. A Survey of TDMA Scheduling Schemes in Wireless Multihop Networks[J]. ACM Computing Surveys, 2015, 47(3):53.
- [3] 曲金鑫. Ad Hoc 网络 TDMA 动态时隙分配算法研究[D]. 哈尔滨:哈尔滨工业大学, 2013.
- [4] 葛松虎. 基于 TDMA 的协同 MAC 协议设计与实现[D]. 长沙:国防科学技术大学, 2011.
- [5] Zhu Chenxi, Corson M S. A Five-phase Reservation Protocol(FPRP) for Mobile Ad Hoc Networks[J]. Wireless Networks, 2001, 7(4):371-384.
- [6] Nasipuri A, Zhuang Jun, Das S R. A Multichannel CSMA MAC Protocol for Multihop Wireless Networks[C]//Proceedings of Wireless Communication and Network Conference. Washington D. C., USA: IEEE Press, 1999:1402-1406.
- [7] Tang Z. Hop-reservation Multiple Access (HRMA) for Ad-hoc Networks[C]//Proceedings of IEEE INFOCOM'99. Washington D. C., USA: IEEE Press, 1999:231-236.
- [8] Zhang Rongqing, Cheng Xiang, Yang Liuqing, et al. A Novel Centralized TDMA-based Scheduling Protocol for Vehicular Networks[J]. IEEE Transactions on Intelligent Transportation Systems, 2015, 16(1):411-416.
- [9] 靳俊峰. 无线宽带网络中的资源分配和调度算法研究[D]. 合肥:中国科学技术大学, 2011.
- [10] 冷爱莲, 杨恢先, 李利. 基于固定和动态时隙相融合的 TDMA 分配算法[J]. 计算机工程与应用, 2015, 51(13):92-95.

(下转第 102 页)

- Proceedings of CISS'08. New York, USA; IEEE Press, 2008;197-202.
- [14] Hoang A T, Wong D, Liang Yingchang. Design and Analysis for an 802.11-based Cognitive Radio Network[C]//Proceedings of WCNC'09. Washington D. C., USA; IEEE Press, 2009;1-6.
- [15] Bae Y H, Alfa A, Choi B D. Performance Analysis of Modified IEEE 802. 11-based Cognitive Radio Networks[J]. IEEE Communications Letters, 2010, 14(10):975-977.
- [16] Khabazian M, Aissa S, Tadayon N. Performance Modeling of a Two-tier Primary-secondary Network Operated with IEEE 802. 11 DCF Mechanism[J]. IEEE Transactions on Wireless Communications, 2012, 11(9):3047-3057.
- [17] Kumar S, Shende N, Murthy C, et al. Throughput Analysis of Primary and Secondary Networks in a Shared IEEE 802. 11 System[J]. IEEE Transactions on Wireless Communications, 2013, 12(3):1006-1017.
- [18] Bian K, Park J, Chen R. A Quorum-based Framework for Establishing Control Channels in Dynamic Spectrum Access Networks[C]//Proceedings of MobiCom'09. New York, USA; ACM Press, 2009;25-36.
- [19] Chang G Y, Teng W H, Chen H Y, et al. Novel Channel-hopping Schemes for Cognitive Radio Networks[J]. IEEE Transactions on Mobile Computing, 2014, 13(2):407-421.
- [20] Lee C M, Lin J S, Feng K T, et al. Design and Analysis of Transmission Strategies in Channel-hopping Cognitive Radio Networks[J]. IEEE Transactions on Mobile Computing, 2012, 11(10):1569-1584.
- [21] Liu Quan, Wang Xiaodong, Zhou Xingming. CH-CSMA/CA: A MAC Protocol for Asynchronous Channel Hopping Rendezvous in 802. 11 DCF Based Cognitive Radio Networks[EB/OL]. (2014-02-09). <http://www.inderscience.com/info/ingeneral/forthcoming.php?jcode=ijsaacs>.
- [22] Gast M. 802. 11 Wireless Networks: The Definitive Guide[M]. 2nd ed. [S.l.]: O'Reilly Media, Inc., 2006.
- [23] DaSilva L, Guerreiro I. Sequence-based Rendezvous for Dynamic Spectrum Access[C]//Proceedings of DySPAN'08. Washington D. C., USA; IEEE Press, 2008;1-7.
- [24] Liu Hai, Lin Zhiyong, Chu Xiaowen, et al. Jump-stay Rendezvous Algorithm for Cognitive Radio Networks[J]. IEEE Transactions on Parallel and Distributed Systems, 2012, 23(10):1867-1881.
- [25] Zhang Yifan, Li Qun, Yu Gexin, et al. ETCH: Efficient Channel Hopping for Communication Rendezvous in Dynamic Spectrum Access Networks[C]//Proceedings of INFOCOM'11. Washington D. C., USA; IEEE Press, 2011;2471-2479.

编辑 金胡考

(上接第 94 页)

- [11] 刘 芳. 移动自组织网络动态时隙分配算法研究[D]. 西安: 西安电子科技大学, 2011.
- [12] 叶林睿. 基于 TDMA 的 Ad Hoc 网络 MAC 协议研究[D]. 成都: 电子科技大学, 2011.
- [13] 曲金鑫. Ad Hoc 网络 TDMA 动态时隙分配算法研究[D]. 哈尔滨: 哈尔滨工业大学, 2013.
- [14] 葛松虎. 基于 TDMA 的协同 MAC 协议设计与实现[D]. 长沙: 国防科学技术大学, 2011.
- [15] 张 弛. 基于 TDMA 的 Ad Hoc 网络 MAC 协议比较[D]. 西安: 西安电子科技大学, 2007.
- [16] Yeo J, Lee H, Kim S. An Efficient Broadcast Scheduling Algorithm for TDMA Ad-hoc Networks[J]. Computers & Operations Research, 2002, 29(13):1793-1806.
- [17] Ning H J. Vertex Coloring Model and Algorithm Research of Computer Distribution[C]//Proceedings of the 5th International Conference on Intelligent Systems Design and Engineering Applications. Washington D. C., USA; IEEE Press, 2014;1113-1116.
- [18] Kim G, Kim S, Kim I, et al. An Efficient Simulated Annealing with a Valid Solution Mechanism for TDMA Broadcast Scheduling Problem[J]. International Journal of Innovative Computing Information and Control, 2011, 7(3):1181-1191.
- [19] Wang Gangsheng, Ansari N. Optimal Broadcast Scheduling in Packet Radio Networks Using Mean Field Annealing[J]. IEEE Journal on Selected Areas in Communications, 1997, 15(2):250-260.
- [20] 符 琦, 陈志刚, 蒋云霞. 基于 NS-2 的多接口多信道仿真模型扩展研究[J]. 计算机工程, 2013, 39(4):113-117.
- [21] Vergados D J, Sgora A, Vergados D D. Fair TDMA Scheduling in Wireless Multihop Networks[J]. Telecommunication Systems, 2012, 50(3):1793-1806.

编辑 顾逸斐