

WMN 中基于循环补偿的公平调度算法

彭 军, 闫 军

(中南大学信息科学与工程学院, 长沙 410075)

摘 要: 针对无线 Mesh 网络(WMN)中数据包调度方案公平性欠缺的问题, 提出一种基于循环补偿的公平调度算法。该算法将 Mesh 节点中需要发送的数据包缓存在各自的数据流队列中, 轮询处理队列的调度请求。利用信道状态监测机制确定队列的信道特征, 通过循环补偿模型将队列调度过程划分为发送、补偿、休眠 3 个时期, 各时期循环执行实现通信数据流间的平衡调度。仿真结果表明, 该算法在公平性和吞吐量方面的有效性。

关键词: 无线 Mesh 网络; 补偿机制; 公平性服务; 数据包调度

Circular-Compensation-based Fair Scheduling Algorithm for WMN

PENG Jun, YAN Jun

(School of Information Science and Engineering, Central South University, Changsha 410075, China)

【Abstract】 In order to solve the problem of unfairness in current packet scheduling of Wireless Mesh Network(WMN), a Circular-Compensation-based Fair Scheduling Algorithm(CCFSA) is proposed. CCFSA works in a flow-based round-robin fashion. The packets that Mesh nodes need to send are buffered in their respective data flow queues. CCFSA ascertains channel characteristics of queues by means of channel status monitoring mechanism, and divides queue scheduling process into three periods including sending period, compensation period and dormancy period. Every period is looping execution to achieve a balanced scheduling among communication data flow and fairness enhancement of system. Simulation results prove the algorithm is effective on network throughput and fairness.

【Key words】 Wireless Mesh Network(WMN); compensation mechanism; fair service; packet scheduling

DOI: 10.3969/j.issn.1000-3428.2011.02.028

1 概述

无线 Mesh 网(Wireless Mesh Network, WMN)是一种动态多跳自组织网络, 具有容量大、组网灵活和成本低廉等优点, 已经成为下一代无线宽带接入研究的热点^[1]。在多跳的 WMN 中, 数据流会多次竞争有限的带宽资源, 合理调度成为 WMN 研究的一项重要内容, 是确保用户间“公平”共享 WMN 网络资源的关键。数据包调度不当将导致网络中瓶颈增多、吞吐量下降、公平性降低等不良后果。因此, 有人提出使用“公平调度”来提高无线网络带宽分配的合理性。文献[2]针对多通道组播调度中数据分配不均带来的带宽浪费问题, 提出 3 种动态带宽分配方案。根据数据项平均长度或数量改变分配优先级, 对不同方案进行性能分析; 文献[3]针对 WMN 中集中式和分布式 2 种调度机制分离的情况, 提出结合 2 种机制的方案, 其中, 集中式调度用于保证实时服务; 分布式调度确保 Mesh 内部节点的直接通信。

在已有的公平调度方法中, 基于补偿的公平调度算法是利用补偿时隙为不同数据流队列提供性能保障, 促进了用户间公平共享网络资源, 提高了系统公平性。但目前调度算法大多是将补偿模型建立在美国 UCLA 大学 Lu 定义的“无线流体公平排队”模型上, 该模型限制了对滞后数据流的补偿数量, 减弱了对超前数据流的惩罚力度。在滞后流损耗时隙远大于补偿时隙时, 整个数据包的调度过程出现了不公平。

因此, 本文在 WMN 中重新设计了一种循环补偿模型 CCM, 并根据该模型提出一种基于循环补偿的公平调度算法(Circular-Compensation-based Fair Scheduling Algorithm, CCFSA), 重点讨论了 CCFSA 如何实现数据流之间的循环补偿。CCFSA 算法在 IEEE802.16 集中式调度模式的基础上,

通过 CCM 模型将数据流的调度过程划分为 3 个时期, 利用补偿期实现超前流对滞后流的“完全”补偿, 从而增强了系统的公平性。

2 WMN 调度机制相关研究

无线公平调度的任务是在不同用户之间合理地分配无线通信信道, 保证带宽资源的公平分配, 减少丢包率, 使整个系统的数据吞吐率达到最优^[4]。

目前 WMN 的公平调度方案是按照 IEEE802.16 的 MAC(Media Access Control)层调度机制执行的, 分为 2 种类型: 分布式和集中式。前者为了保证数据时隙没有冲突, 数据时隙的分配是通过控制时隙的交换实现的, 节点在本次传输中选择下次传输的时间; 后者需要基站的参与, 每个用户站的传输调度都由基站决定, 用户站在获得基站分配的发送时隙后, 可以不通过基站直接与目地站进行通信^[5]。

采用这 2 类调度方法的算法基本都应用补偿机制实现调度过程的公平性。但目前调度算法大多是将补偿模型建立在“无线流体公平排队”模型之上, 该模型将所有滞后流能补偿的数量限制为一个固定值 B , 对于任何一个滞后流 i , 允许的最大补偿量为 b_i , 计算公式如下:

$$b_i = B \frac{r_i}{\sum_{j \in \text{flows}} r_j}$$

其中, r_j 表示流 j 的权重。在该模型下, 滞后数目小于最大补

作者简介: 彭 军(1967 -), 女, 教授、博士生导师, 主研方向: 智能交通移动通信系统, 多智能体系统, 机器人技术; 闫 军, 硕士研究生

收稿日期: 2010-06-20

E-mail: yanjuncsu@yahoo.com.cn

偿值的滞后流能够获得相对公平的补偿,但如果滞后流所损失的数目远大于最大补偿值,该模型就未起到补偿的作用。因此,设计一种既能够完全补偿滞后流损失的时隙,又不明显造成其他流延迟增加的算法,具有重要的意义。

3 循环补偿的公平调度算法

公平调度算法从多个队列中选择下一个要传输的数据包,使所有接入业务流能按预定的方式共享节点的输出链路带宽,满足不同业务的服务质量要求。图 1 为无线环境下一种典型的调度器结构。

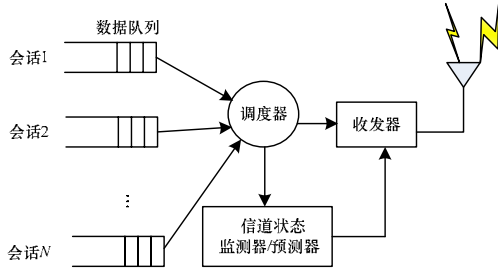


图 1 典型的无线调度器结构

定义 1(发送权限) 定义变量 Sh 为调度器当前时刻的发送权限:

$$Sh = ID_i \quad (1)$$

其中, ID_i 为节点 i 的数据流队列编号,只有发送权限与数据流队列的编号相同时,该队列才能发送数据包,当 $Sh = \text{null}$ 时,说明调度器尚未选择发送的数据流队列。

3.1 队列数据结构

在 CCFSA 算法中,为实现基于补偿的公平调度,WMN 节点需要维护一些缓存数据包的队列和补偿队列,这些数据结构如下:

(1) 数据流队列

在 IEEE802.16 中没有端到端的信令,因此 WMN 节点都需要维护一个数据流队列,节点的数据流队列存储了该节点需要发送的数据包以及其他节点需要转发的数据包。

以 A 表示所有节点的数据流队列总集, $A = \{\alpha_1, \alpha_2, \dots, \alpha_N\}$, α_i 代表第 i 个节点的数据队列, N 为数据流队列总数。记 $\alpha_i = \langle ID_i, W_i, D_i, F_i, C_i, H_i \rangle$, 表示 α_i 具有如下属性: 1) 编号 ID_i : 队列编号; 2) 权值 W_i : 队列的优先级大小; 3) 容量 D_i : 队列中数据包个数; 4) 补偿标记 F_i : 标记队列的发送权限是否由其他队列放弃而获得; 5) 发送容限 C_i : 记录由补偿期或休眠期进入发送期时队列 α_i 的数据包个数; 6) 休眠容限 E_i : 记录休眠期的长短。其中, ID_i 、 W_i 和 E_i 在队列初始化时将分配相应的固定值,各个队列的权值可以相同,表示这 2 个队列有相同的优先级; F_i 的初始值为队列编号 ID_i ; C_i 的初始值为 0。

(2) 补偿队列

如果节点 j 信道出错,节点 i 获得节点 j 放弃的发送权,则称节点 j 是需要“被补偿”的。为在适当的时间对节点 j 做出补偿,节点 i 维护一个补偿队列 β_i , 该队列存储节点 i “被补偿”节点的编号。记 $\beta_i = \langle H_i, T_i, R_i \rangle$, 表示 β_i 具有如下属性: 1) 头标记 H_i : 指向队列第 1 个数据; 2) 尾标记 T_i : 指向队列末尾数据; 3) 补偿容限 R_i : 记录由发送期进入补偿期时队列 β_i 中的数据个数。其中, $H_i = T_i$ 说明队列为空; $H_i - T_i$ 的值表示当前队列中需要补偿的数据个数。补偿队列的数据在入队、出队时与普通队列不同,需要遵守如下规则:

入队: 判断 $F_i = ID_i$ 是否成立,如果成立,则放弃入队,说明节点 i 这次的发送是正常的数据包调度;否则,将 F_i 按照先进先出的顺序插入到队列 β_i 中, T_i 加 1, F_i 置初值。

出队: 查询该编号节点的信道是否为“好”状态,若是,则将数据交出,同时 H_i 加 1;否则,将该数据保存到队尾, H_i 、 T_i 同时加 1,继续查询后面的数据,直到发现“好”状态的节点为止。

3.2 信道状态监测模块

CCFSA 算法采用二状态马尔可夫模型将信道状态分为 2 种: 好状态和坏状态,以此描述无线信道的差错情况。

定义 2(信道状态) 定义变量 S_i 为节点 i 的当前信道状态。

$$S_i = \begin{cases} \text{好状态} \\ \text{坏状态} \end{cases} \quad (2)$$

好状态: 当通信源端和目的端同时检测信道是无误码、无差错的状态。

坏状态: 当通信源端或目的端任何一端检测到信道有差错时信道的状态。

3.3 循环补偿模型 CCM

本文将补偿的思想引入 WMN 的公平调度算法中,关注如何在不同数据流之间公平调度。循环补偿模型 CCM 将数据包的调度过程分为 3 个时期,即发送期、补偿期和休眠期。记 θ_i 为节点 i 在各个时期的时间长度,则:

$$\theta_i = \begin{cases} L_i^1 & \text{发送期} \\ L_i^2 & \text{补偿期} \\ L_i^3 & \text{休眠期} \end{cases} \quad (3)$$

该节点中的 3 个时期按照循环补偿规则(Circular Compensation Rule, CCR)顺序执行,每个节点循环进入补偿期对其他节点做出补偿,补偿的时限以补偿容限为界,补偿容限越大,补偿期越长。不设置最大补偿值,而是按比例分配时隙给需要“被补偿”的节点。在每个时期中,调度器按照 CCR 规则分配发送权限,具体规则如下:

(1) 发送期

发送期的长短与各个节点数据流队列的权值大小有关,权值越大,发送期越长,反之越短。发送期的时长定义为:

$$L_i^1 = \frac{C_i \times W_i}{\sum_{j \in \text{flows}} W_j \times \gamma} \quad (4)$$

其中, γ 为数据包传输速率。在该时期中,当调度器轮询到队列 α_i 后,首先查询该队列的信道状态是否正常,然后按照如下步骤进行:

Step1 若 $S_i = 1$ 且 $D_i \neq 0$, 则节点 i 将数据流队列 α_i 中的数据包交给调度器发送,补偿队列 β_i 执行入队操作,令 $Sh = \text{null}$, 否则,执行 Step2;

Step2 若 $S_i = 0$ 且 $D_i \neq 0$, 则调度器选择权值最接近 W_i 的队列 α_j , 令 $Sh = ID_j$ 且 $F_j = F_i$, 否则,执行 Step3;

Step3 若 $D_i = 0$, 则说明数据流队列 α_i 没有数据需要发送,令 $Sh = \text{null}$, 调度器将发送权限交给其他节点。

(2) 补偿期

在该时期中,各个节点对自身补偿队列中的数据进行处理,完成对其他节点损耗时隙的补偿。补偿期的长短以补偿队列中所包含数据的个数为基准,数据越多,补偿期越长。补偿期的时长定义如下:

$$L_i^2 = \frac{R_i^2}{\sum_{j \in \text{flows}} R_j \times \gamma} \quad (5)$$

假设在当前时刻调度器轮询到节点 i , 则执行如下步骤:

Step1 节点 i 查询补偿队列 β_i , 若 $H_i=T_i$, 说明 β_i 为空, 则节点 i 直接进入休眠期, 令 $Sh=null$, 否则, 执行 Step2;

Step2 节点 i 执行补偿队列 β_i 的出队操作, 假设取出的数据为 ID_j , 执行 Step3;

Step3 若 $D_j \neq 0$, 说明数据流队列 α_j 不为空, 令 $Sh=ID_j$, 否则, 执行 Step4;

Step4 节点 j 查询补偿队列 β_j , 若 $H_j \neq T_j$, 说明 β_j 不为空, 执行 Step2; 否则, 节点 j 放弃节点 i 对它的补偿, 令 $Sh=null$.

(3) 休眠期

在该时期中, 节点 i 处于休眠状态, 其 S_i 并不根据信道状态监测器发回的信道状态信息进行判断, 而是被强制保持为“脏”状态, 不进行任何数据包的发送, 也不对其他队列进行补偿, 但可以接收其他节点发送来的数据包。休眠期的时间长度是一常量, 可以根据情况自行设置, 定义如下:

$$L_i^3 = E_i$$

3.4 基于循环补偿的公平调度算法流程

调度器首次轮询到节点 N 即 $Sh=ID_N$ 时, 其调度过程如下:

(1) 更新本节点处的数据流队列和补偿队列, 将 2 个队列中的属性值进行初始化, 然后跳转到(6)。

(2) 查看发送权限 Sh 的当前值, 选择相应的节点 N , 首先查询数据流队列 α_N 中的发送容限 C_N 是否为 0。如果 $C_N \neq 0$, 说明节点处于发送期, 则跳转到(3); 否则, 查询补偿队列 β_N 中的补偿容限 R_N , 如果 $R_N \neq 0$, 说明节点处于补偿期, 则跳转到(4), 否则, 跳转到(5)。

(3) 将发送容限 C_N 自减 1, 按照循环补偿模型 CCM 中发送期规则对数据流队列 α_N 进行调度, 执行结束则判断 $L_N^1=0$ 是否成立, 若成立就跳转到(7), 否则, 跳转到(8)。

(4) 将补偿容限 R_N 自减 1, 按照循环补偿模型 CCM 中补偿期规则对补偿队列 β_N 进行调度, 执行结束则判断 $L_N^2=0$ 是否成立, 若成立就跳转到(6), 否则, 跳转到(8)。

(5) 判断 $L_N^3 \neq 0$ 是否成立, 若成立, 则按照休眠期规则执行, 同时 E_i 减小 t (t 为一固定时间), 跳转到(8)。

(6) 获取数据流队列 α_N 的容量 D_N , 计算发送期容量 C_N 的大小 ($C_N=D_N$), 跳转到(8)。

(7) 更新补偿队列 β_N , 获取头标记和尾标记的值, 计算补偿容限 R_N 的大小 ($R_N=H_N-T_N$), 跳转到(8)。

(8) 调度器判断发送权限 $Sh=null$ 是否成立, 若成立, 则选择下一个节点 M 发送并令 $Sh=ID_M$, 然后跳转到(2), 否则, 直接跳转到(2)。

在步骤(8)中, 当调度器选择下一个发送节点时, 会选择当前时刻权值最大的节点, 如果 2 个节点的权值相同, 则逐个进行选择。需要说明的是, 在处理队列中的数据时, 无论是补偿队列还是数据流队列, 调度器每次只处理 1 个数据包, 而不是循环一次就将队列中所有的数据包全部处理完。

4 仿真实验结果与分析

为评估本文提出的分期补偿算法的性能, 采用 NS-2 网络仿真器对其进行仿真实验, 利用 NTNU 大学提供的 MW-Node 补丁使 NS-2 具有多接口多信道无线网络仿真能力。本文利用 NS-2 实现 Blind-WRR 算法, 将其作为本文算法的对比方案。

本文的仿真场景中共有 11 个节点, 其中, 10 个为发送

节点, 带宽为 2 Mb/s, 用于数据包的发送; 1 个接收节点, 带宽为 10 Mb/s, 用于数据包的接收。中间的基站用于 CCFSA 算法功能的实现。具体网络拓扑结构如图 2 所示。

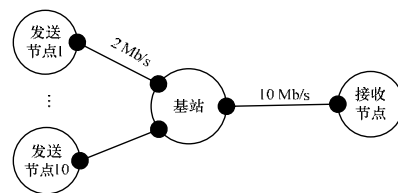


图 2 网络拓扑结构

在 NS-2 中设置不同的随机值让每种仿真场景运行 10 遍, 取结果平均值作为最后对比结果。具体仿真参数如表 1 所示。

表 1 NS-2 仿真参数设定

仿真参数	参数值
节点个数	10
节点编号	1~10
节点权值	10~100
休眠容限/ms	10~100
信道带宽/(Mb·s ⁻¹)	2
“好”信道时间/ms	30
“坏”信道时间/ms	10
业务类型	CBR 实时业务
仿真时间/s	100

为了对算法进行性能评价, 统计了以下性能参数:

节点端到端吞吐量: 没有 MAC 层过载时目的节点每秒钟收到的 bit 数。

数据包平均延迟: 数据包从发送节点的网络层到接收节点的网络层之间的时间间隔。

公平性指标: 判定公平性的一般性方法由公式 $(\sum_{i=1}^n x_i)^2 / \sum_{i=1}^n x_i^2$ 决定。其中, n 表示通信流的数目; x_i 表示第 i 个通信流的吞吐量。

图 3 是无线发送节点的端到端吞吐量比较, 从图中可以看出, 算法 Blind-WRR 和 CCFSA 由于区分了服务, 随着权值的减小, 节点的吞吐量也相应呈下降趋势。但由于 CCFSA 算法具有信道状态监测机制, 当信道为“坏”状态时, 会将发送权限交给其他数据流, 不会影响整个调度进程。与此相反, 算法 Blind-WRR 缺少信道监测机制, 若其中的某个节点出现信道故障, 调度器仍会将发送权限交给它, 从而严重影响低优先级节点的吞吐量。

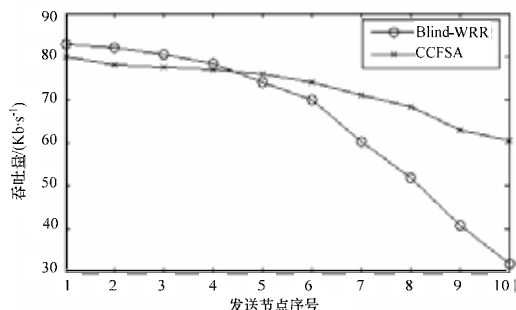


图 3 节点端到端吞吐量比较

图 4 为无线节点的数据包平均延时。从图中可以看出, 算法 Blind-WRR 中节点延时的高低对优先级的依赖性很大, 如图中高优先级节点 1~4 平均延时接近于 0.05 s, 而低优先级节点 5~10 的平均延时相对高得多, 这是因为在算法 Blind-WRR 中, 只有在高优先级节点的数据流队列为空时,

后面的节点才可以发送。而算法 CCFSA 中这种差别小很多,这主要是 CCFSA 采用了循环补偿机制,使得高优先级节点在休眠期放弃部分发送数据包的权利。并且在补偿期,对信道出错而丧失了发送数据包机会的节点加以补偿。图 5 为节点端到端吞吐量的公平性指标仿真结果。

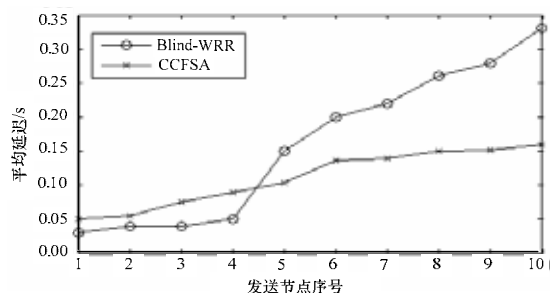


图 4 数据包平均延迟比较

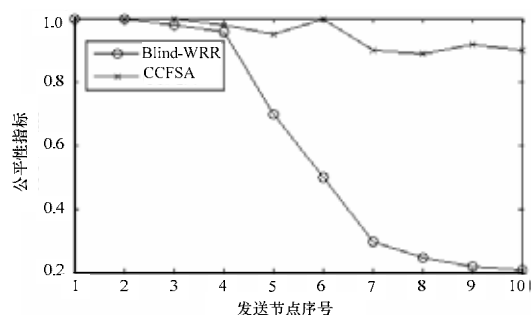


图 5 公平性指标比较

从图中可以看出, Blind-WRR 算法中优先级高的节点能够达到公平性的要求,而低优先级节点的公平性差得多。这主要是前者较长时间占用带宽,使后者无法公平地获得应有

的带宽,以至吞吐量大幅下降。而算法 CCFSA 中采用的循环补偿机制正好弥补了这一缺陷,低优先级节点可以在高优先级节点处于休眠期时发送数据包,也可以在高优先级节点处于补偿期时获得损失的时隙,满足了该范围内用户的性能要求。因此,算法 CCFSA 中各节点都很好地达到了公平性的指标要求。

5 结束语

本文提出了一种实时业务下基于循环补偿机制的公平调度算法 CCFSA。该算法根据源节点组织数据流队列和补偿队列,根据循环补偿模型 CCM 将数据流的调度过程划分为 3 个时期,以实现超前流对滞后流的“完全”补偿。仿真结果表明,CCFSA 算法提高了实时业务的端到端吞吐率,降低了延时,增强了节点之间数据流调度的公平性,能够在一定程度上保证实时业务的服务质量。

参考文献

- [1] 方旭明. 下一代无线因特网技术 无线 Mesh 网络[M]. 北京: 人民邮电出版社, 2006.
- [2] 孙显军, 徐桐发, 刘卫东, 等. 实时数据组播调度动态带宽分配方案[J]. 计算机工程, 2009, 35(19): 7-9.
- [3] 赵元韬, 杨寿保, 滕 达, 等. WiMAX Mesh 网络混合调度下的 QoS 保障[J]. 计算机工程, 2010, 36(5): 86-88.
- [4] Holma H, Toskala A. WCDMA for UMTS-radio Access for Third Generation Mobile Communications[M]. New York, USA: John Wiley & Sons, 2003.
- [5] Loscri V. MAC Protocols over Wireless Mesh Networks: Problems and Perspective[J]. Journal of Parallel and Distributed Computing, 2008, 68(3): 387-397.

编辑 张正兴

(上接第 77 页)

- [3] 刘 泉, 任 平. 基于 MS/TP 协议的无线 Ad Hoc 网络[J]. 计算机工程, 2009, 35(10): 89-91.
- [4] 李文杰, 刘 斌. 输入排队中抢占式短包优先调度算法[J]. 电子学报, 2005, 33(4): 577-583.
- [5] McKeown N, Izzard M, Mekkittikul A, et al. Tiny Tera: A Packet Switch Core[J]. IEEE Micro, 1997, 17(1): 26-33.
- [6] Kallgren D G, Smaal J W, Gerbrands M, et al. An Architecture for Internet Protocol over HF: Allied High-Frequency Wide-Area

Networking Using STANAG 5066(AHFWAN66)[C]//Proc. of MILCOM'05. Atlantic City, New Jersey, USA: IEEE Press, 2005.

- [7] 徐 刚, 丁泉龙. 基于预测的最长队列优先调度算法[J]. 计算机工程, 2008, 34(1): 7-9.
- [8] Johnson E E, Anaya G, Tang Zibin, et al. Performance of the HF Token Protocol[D]. Las Cruces, New Mexico, USA: Klipsch School of Electrical and Computer Engineering, New Mexico State University, 2004.

编辑 金胡考

(上接第 80 页)

参考文献

- [1] Lim Hyojun, Kim Chonkwon. Flooding in Wireless Ad hoc Networks[J]. Journal of Computer Communications, 2001, 24(34): 353-363.
- [2] Laouiti A, Qayyum A, Viennot L. Multipoint Relaying: An Efficient Technique for Flooding in Mobile Wireless Networks[C]//Proc. of the 35th Annual Hawaii International Conference on System Sciences. Hawaii, USA: IEEE Press, 2002: 298-307.

- [3] Cheng Xiuzhen, Du Dingzhu. Virtual Backbone-based Routing in Multihop Ad hoc Wireless Networks[J]. IEEE Transactions on Parallel and Distributed Systems, 2003, 15(5): 35-42.
- [4] 卢先领, 于继明, 孙亚民. 一种能量高效的 Ad hoc 无线网络广播算法[J]. 计算机工程与应用, 2007, 43(31): 119-122.
- [5] 魏亚青, 李 杰, 郑 丁. Ad hoc 网络中一种基于传输功率的分簇算法[J]. 计算机工程, 2009, 35(21): 111-113.
- [6] 殷剑宏, 吴开亚. 图论及其算法[M]. 合肥: 中国科学技术大学出版社, 2004.

编辑 张正兴