

无线 Mesh 网络的高实时业务调度方法*

王金涛^{1,2}, 曾鹏¹, 李栋¹, 郑萌¹

(1. 中国科学院沈阳自动化研究所 工业控制网络与系统研究室, 沈阳 110016; 2. 中国科学院大学, 北京 100049)

摘要: 为了解决无线 Mesh 网络实时业务的规划调度问题, 提出一种基于动态规划的实时业务调度算法。首先, 通过边集映射, 将网络中的边转换为映射节点进行阶段划分; 然后, 每个 Mesh 路由器节点根据缓存中的队列情况预测数据包的排队时间, 并结合业务截止期进行优先级重新分配, 得到映射边的接入权值; 最后, 利用动态规划算法, 根据映射边的接入权值选择最优的映射节点接入次序, 得到该业务在各阶段的最优调度顺序。仿真结果表明, 该算法能够兼顾高、低优先级业务的实时需求, 降低全网业务的平均时延和丢包率。

关键词: 无线 Mesh 网络; 实时性; 优先级; 调度

中图分类号: TP393.09 文献标志码: A 文章编号: 1001-3695(2016)08-2491-05

doi:10.3969/j.issn.1001-3695.2016.08.056

High real-time service scheduling method in wireless Mesh network

Wang Jintao^{1,2}, Zeng Peng¹, Li Dong¹, Zheng Meng¹

(1. Research Laboratory of Industrial Control Network & System, Shenyang Institute of Automation, Chinese Academy of Sciences, Shenyang 110016, China; 2. University of Chinese Academy of Sciences, Beijing 100049, China)

Abstract: In order to solve the real-time service scheduling problem in wireless Mesh networks, this paper proposed a dynamic programming based real-time service scheduling (DPRSS) algorithm. First, it transformed the network edges into mapping nodes through edge mapping. Then, it estimated the queuing time of each packet in the system according to the queue of the Mesh router and reassigned the priority of each packet with the consideration of the deadline of services in each node. After that, it obtained the access weights of the mapping edges. Finally, chose the optimal mapping nodes access sequence using dynamic programming algorithm to get the optimal scheduling sequence of the service in each stage. The simulation results show that, the DPRSS algorithm proposed can meet the real-time requirements of different network services from high priority to low priority with the advantage of decreasing the average time delay and packet loss rate in the whole network.

Key words: wireless Mesh network(WMNs); real-time; priority; scheduling

0 引言

随着无线数据服务和多媒体业务需求的快速增长, 人们对业务服务质量(quality of service, QoS)的需求逐渐提高, 新的无线网络架构因此不断涌现。无线 Mesh 网络(wireless Mesh networks, WMNs)作为“最后一公里”宽带无线接入的一种关键解决方案^[1], 与传统的无线网络相比, 在网络资源的调度方面存在许多新的问题^[2]。在 WiFi Mesh 的研究方面, 由于其遵循的 IEEE 802.11 系列规范在资源调度方面的局限性, 所以难以实现端到端的时延保障。在 WiMAX Mesh 的研究方面, 由于其遵循 IEEE 802.16 d/e 标准而提供无连接的服务^[3], 所以难以进行端到端时延保障。

近年来出现了很多从调度的角度对多跳无线网络进行实时性保障的方案, 主要可以分为集中式和分布式两类^[4]。总体来说, 集中式的调度方案能够提供端到端的 QoS 保障, 而分布式调度方案可以提供更好的灵活性和可扩展性。在对集中式调度的研究中, 文献[5]提出了一种结合网络层和 MAC 层的跨层集中式调度算法。该方法面向多信道、单收发器的

WiMAX 网络, 通过创建路由树的方法降低干扰, 从而提高网络有效性。文献[6]给出一种基于令牌的调度机制。只有持有令牌的节点才能进行传输, 通过调整竞争窗口 CW(contending windows)和帧间隔 IFS(inter frame space)的大小对持有令牌的不同类型的业务进行调度。文献[7]提出一种联合功率控制、调度及路由的集中式调度算法, 可以提高网络的吞吐量, 减少时延及功率的消耗。文献[8]提出了基于功率和码率控制的网络调度问题模型, 利用列生成与启发式算法来进行优化问题的求解。文献[9]同样基于列生成算法来求解无线 Mesh 网的调度优化问题。文献[10]基于操作系统的调度模型, 提出基于最早截止任务优先 EDF(earliest deadline first)的调度算法, 按照业务的剩余截止时间来调度, 业务的剩余截止时间越小, 该业务的紧急程度越高, 被赋予的优先级就越高, 从而得到优先调度。文献[11]提出基于比例时限单调算法 PD(proportional deadline monotonic)的调度方法, 并结合时限单调算法 DM(deadline monotonic)^[12]进行改进。算法中业务的优先级等于截止期大小除以整条路径的传输次数。

文献[13]对 EDF 算法进行改进, 提出了最小空闲时间优

收稿日期: 2015-05-11 修回日期: 2015-07-02 基金项目: 中国科学院战略性先导科技专项资助课题(XDA06020500)

作者简介: 王金涛(1986-), 男, 山东滨州人, 博士研究生, 主要研究方向为无线工业控制网络的实时性; 曾鹏(1976-), 男, 研究员, 博导, 主要研究方向为工业控制网络的关键技术及应用(zp@sia.cn); 李栋(1986-), 男, 博士, 主要研究方向为工业控制网络和工业无线传感器网络; 郑萌(1983-), 男, 副研究员, 硕导, 主要研究方向为无线自组织和传感器网络、认知无线网络等。

先(least slack first,LSF)调度算法。在 LSF 算法中,业务的优先级根据业务的预测完成时刻与截止时限间的空闲时间来确定,这一段空闲时间越小,该业务的优先级就越高。由于在等待队列中的业务空闲时间随着时间的增大严格递减,而当前已接入业务的空闲时间不变,所以等待业务的优先级有可能随时会超过当前已接入业务的优先级,从而造成业务的频繁抢占,使系统性能降低。

在分布式调度方法的研究中,文献[14]提出了在多跳无线 Mesh 网络下确保带宽和时延约束的多信道、多射频分布式链路调度算法。文献[15]证明了通过将 Mesh 网络合理划分为多个子网,在每个子网中都运用简单的调度算法来提高网络的实时性。由于与集中式调度算法相比,分布式调度方法本身在时延保障方面具有劣势,所以在此不作赘述。

另外,在某些情况下,在整个的网络环境中可能存在只有某段链路出现拥塞的情况。这时如果采用端到端的拥塞控制方法单纯地通过降低源节点的发送速率等方法来解决,则会降低网络的整体利用率,因此端到端的拥塞控制缺乏灵活性^[16]。如果通过在各个节点独立地进行队列管理等方法来进行拥塞控制,则又难以保证系统的端到端时延、吞吐量等性能。例如在如图 1 所示的网络结构中,对于一条要经过多个拥塞节点的数据流,假如前者 *a* 比后者 *b* 具有更高的优先级,但通常后者也要保障一定的时延需求,这时如果在节点 *A* 处给数据流 *a* 以较高的优先级,在 *B* 处也给其较高优先级,在这种模式下传输可能使数据流 *a* 比系统时延要求的截至时间提前很多到达,而数据流 *b* 超时,从而造成资源的不合理分配。如果能适当地在其中的某些链路处作一些优化调整,则可能会对两条数据流的端到端时延都提供保障。

从上面的分析可以看出,现有的 Mesh 网络调度方案大多是为了保证业务的公平性、提高网络吞吐量、减小端到端时延等方面,而根据业务的截止期限限制来提供高实时保障的调度方法还比较少。因此,本文采用逐跳调节与集中式调度协同的方式,并结合实时系统的任务调度方法进行数据包的调度,为不同优先级业务提供高实时保障。

由于可以将端到端的接入调度问题划分为若干个阶段,而每个阶段都需要对接入顺序进行决策,决策的确定既要根据当前所处的优先级,又要根据截止时间,所以是一个前后连续的过程。各个阶段依次进行决策,就可以得到一个决策序列,从而可以确定整个业务流的接入顺序。因此,业务流的调度问题可以利用动态规划方法来求解。

首先,通过边集映射,将网络中的边转换为映射节点,映射节点之间通过映射边连接。然后,求映射边的权值。映射边的权值用业务的接入优先级表示,计算方法为:根据 Mesh 路由器的队列模型预计数据包在系统中的逗留时间,结合业务截止期限限制,在业务流经过的各个节点处对具有不同优先级的数据包进行优先级的重新分配。重新分配的接入优先级权值是截止时间和数据包预期逗留时间的函数。最后,利用动态规划算法,从端到端的角度对映射节点进行接入次序的选择,从而得到最优的链路接入顺序。由于结合截止时间进行优先级的划分和调度,所以可以最大限度地满足不同优先级业务的实时性需求,保证了实时性与调度的公平性。

1 系统模型的建立

无线 Mesh 网络的拓扑用方向图 $G = (V, L)$ 来表示。 V 表

示 Mesh 网络所有的路由器节点集合,其中既有普通路由器节点,也有具有网关功能的节点; L 代表节点之间的所有无线链路集合。本文中设网络所有链路均为对称链路,即当链路 (i, j) 存在时,其反方向的链路 (j, i) 也存在,反之亦然。同时假设节点的移动性较低。由于对 Mesh 网络节点来说,每个节点可能对应于多个不同的链路,如图 2 所示。当进行数据传输时,就会对应多种不同的接入调度次序。本文将无线 Mesh 网络中不同业务调度问题建模成多阶段的决策问题,然后利用动态规划的方法进行问题的求解。

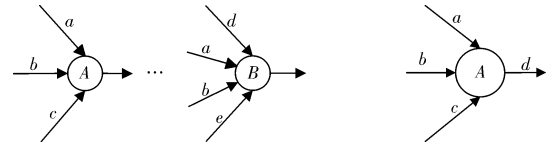


图 1 多条链路产生网络拥塞示意图 图 2 节点的链路连接模型

根据无线 Mesh 网络的多跳结构特点,将接入调度问题分解成若干个单跳接入问题。将相邻的节点之间一跳接入调度当做是动态规划求解方法中的一个阶段。该划分方式从源点开始,把与之直接相连的节点之间链路的选择顺序当做问题的一个子过程。这种划分依次进行,直至目的节点为止。在每一个子过程中,利用动态规划求解问题的依据是映射边的权值。因此,在节点映射完成以后要进行接入权值的分配。本章给出问题的数学模型,而接入权值的分配方法将在下一章进行阐述。

在本文建立的数学模型中,需要用到的术语定义如下:

定义 1 边集映射。将真实网络拓扑中的接入链路(分为入边和出边)映射成映射连接图中点的过程。

定义 2 映射连接图。将网络拓扑图进行边集映射得到连接关系图。

定义 3 映射节点。在进行边集映射后,由原来网络拓扑图中的链路映射得到映射连接图中的节点。

定义 4 映射边。在映射连接图中,连接映射节点的边。

为了便于利用动态规划进行求解,要对网络进行边集映射。映射的方法为:以目标节点为起点,源节点为终点,由同一节点相连的所有入边作为一级,将网络中的边进行映射得到映射连接图中的映射节点。同一节点连接的入边与出边为相邻的映射节点,通过映射边直接相连。这样若原来是同一节点连接的边,则进行边集映射后成为由映射边直接相连的映射节点。然后将出边作为入边的上一阶段节点,依次进行映射便可得到网络的映射连接图。

模型中用到的数学符号解释如下:

k : 动态规划过程的阶段序号 $k=1$ 代表第 1 阶段。

S_k : 第 k 阶段中可供选择的映射节点的集合, $|S_k|$ 为阶段 k 的映射节点的总数,其大小决定了待选决策的数量。

x_k^i : 第 k 阶段的映射节点 i 。

$L_k(i, j)$: 第 k 阶段中,从映射节点 i 到映射节点 j 的映射边的接入权值。

$u_k(x_k^i)$: 在阶段 k 映射节点 x_k^i 所作出的决策,显然有 $u_k(x_k^i) \in S_k$ 。

当在第 k 阶段,且作出的决策为 $u_k(x_k^i)$ 时,用 $L_k(u(x_k^i), x_k^j)$ 代表子过程的映射边接入权值,则各阶段的指标函数定义为

$$\varphi_k(x_k, \mu_k) = \sum_{j=k}^n L_j(u_j(x_j^i), x_j^j) \quad (1)$$

然后,取式(1)中的最小值,从而得到动态规划的优化目

标函数:

$$F_k^f(x_k) = \min_{u_k \in S_k} [L_k^f(u_k, x_k) + F_{k+1}^f(u_k)] \quad k = n, n-1, \dots, 1 \quad (2)$$

边界条件 $F_{n+1}^f(x_{n+1}) = 0$

$F_k^f(x_k)$ 为对数据流 f 的调度过程中最优目标函数。根据式 (2) 可得, 调度策略是由当前的链路状态之前的各个子过程中链路权值的累加和来决定。用 Γ 表示 Mesh 骨干网中端到端的业务流集合, 其中 $f \in \Gamma$ 表示任意一条业务流。对于每一个有实时性要求的数据包, 其时延截止期用 $D_{f\mu}$ 表示。记属于业务流 f 的数据包 μ 的端到端时延为 $T_{f\mu}$, 有 $T_{f\mu} \leq D_{f\mu}$ 。本文的设计目标是在现有的网络资源条件下, 通过合理的调度方法来为尽可能多的实时性业务提供时延保障。

算法的设计步骤如下:

a) 设置业务流 f 的边集调度集合 Ψ_f , 并令 $0 \leftarrow \Psi_f$ 。根据用户需求设置业务的用户优先级, 每一种业务的用户优先级都对应于相应的截止期 D_f 。

b) 对于每个业务流 f , 以与目的节点直接相连的边为起始边, 以与源节点直接相连的边为终到边进行边集映射, 从而得到从映射节点 l_D^f 到 l_S^f 的映射连接图。为了保持算法的一致性, 当与目的节点或与源节点直接相连的边大于一条时, 分别加入虚拟边, 形成虚拟映射节点 l_D^f 和 l_S^f 。

c) 从 l_S^f 开始, 计算 $L_{kl}^f(x_k^i, x_{k+1}^j)$ $k=1, 2, \dots, n; l=1, 2, \dots, A_i^f(k)$ 。其中 k 为阶段数, $A_i^f(k)$ 为第 k 阶段与映射节点 x_k^i 连接到下一阶段同一映射节点的映射节点个数的全排列。

d) 计算:

$$F_k^f(x_k) = \min_{u_k \in S_k} [L_{kl}^f(u_k, x_k) + F_{k+1}^f(u_k)]$$

e) 依据计算结果, 构建出数据流 f 的调度顺序向量 Ψ_f 。

为了清晰起见, 将本算法的执行过程通过图 3 进行说明。

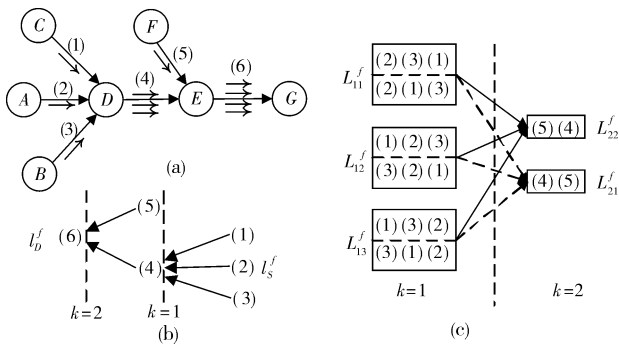


图 3 算法执行过程举例

假设网络的拓扑结构如图 3 (a) 所示。其中 A, B, \dots, G 为节点序号, (1), (2), \dots , (6) 为边的序号。经过 $A-D-E-G$ 的业务流记为 f 。图 3 (b) 是映射连接图。由于在图 3 (a) 中, 入边 (1) (2) (3) 和出边 (4) 与同一节点 D 相连, 所以在映射连接图中, 映射节点 (1) (2) (3) 与 (4) 都直接通过映射边相连。同理可以得到其他映射节点的关系。

在对网络进行边集映射以后就要进行链路权值的分配, 如图 3 (c) 所示。其中, 链路权值 $L_{kl}^f(x_k^i, x_{k+1}^j)$ 由下章的链路权值分配算法求得, 然后代入式 (2) 即可求得最优的调度次序。

2 各阶段的链路权值划分

现在考虑的情况是如果初始时链路的权值未知, 为了保证链路的接入顺序, 应如何为链路赋权值(即确定链路的接入优先级问题)?

根据业务的端到端时延需求, 链路的接入权值应该由业务的剩余时间所决定。而由于在传输过程中, 数据包所经历的时延主要是由排队引起的, 所以排队时延是影响端到端时延的关键因素。本文通过对数据包进行时延预测来预计数据包在不同优先级情况下的排队时延, 从而得到其在不同优先级情况下的接入权值, 为后续的动态规划求解提供依据。

为了不失一般性, 设 Mesh 网络路由器的数据处理系统符合 M/M/m/κ 排队模型, 其中 m 为路由器的并行处理器数量, 最大队列长度为 κ 。数据按照参数为 λ 的泊松流到达路由器, 当信息到达路由器后, 若有处理器空闲, 则直接进行转发; 否则, 数据要在缓冲区等待队列中等待, 直至处理器空闲时再发送。若一段时间内路由器缓存的数据超过其缓存容量, 则此后到达路由器的数据将会被丢弃。设路由器的转发速率为 μ , 数据包/时隙, 等待队列长度为 $\kappa - m$ 。令 $\rho = \lambda/\mu$, 由文献 [17] 可得数据的等待概率 P_n 和路由器的空闲概率 P_0 分别为

$$P_0 = \left(\sum_{n=0}^{m-1} \frac{1}{n!} \rho^n + \sum_{n=m}^{\kappa} \frac{m^m}{m!} \rho^n \right)^{-1} \quad (3)$$

$$P_n = \begin{cases} \frac{(m\rho)^n}{n!} P_0 & 0 \leq n \leq m-1 \\ \frac{m^m}{m!} \rho^n P_0 & m \leq n \leq \kappa \end{cases} \quad (4)$$

当 $\rho \neq 1$ 时, 即数据转发速率和数据到达速率不等, 此时会造成处理器空闲而白白浪费资源或收到的数据量超过缓存的容量使数据丢失, 这样路由器空闲的概率表示为

$$P_0 = \left[\sum_{n=0}^{m-1} \frac{(m\rho)^n}{n!} + \frac{(m\rho)^m}{m!} \left(\frac{1}{1-\rho} - \frac{\rho^{\kappa-m+1}}{1-\rho} \right) \right]^{-1} \quad (5)$$

当 $\rho = 1$ 时, 数据的转发速率与数据到达速率相等, 可以得出路由器此时的空闲概率为

$$P_0 = \left[\sum_{n=0}^{m-1} \frac{m^n}{n!} + \frac{m^m}{m!} (\kappa - m + 1) \right]^{-1} \quad (6)$$

由于 M/M/m/κ 路由器存在缓存容量的限制, 若收到的数据量大于路由器的缓存容量, 必然会造成数据包出现丢弃。数据损失的概率用 P_κ 表示, 即若缓存中已存在 κ 个数据包, 则新到来的数据包不得不丢弃的概率为

$$P_\kappa = \frac{m^m}{m!} \rho^n P_0 \quad (7)$$

路由器中的等待队长设为 N_d , 当 $\rho \neq 1$ 时:

$$N_d = \frac{m^m}{m!} \frac{\rho^m}{(1-\rho)^2} P_0 \cdot \rho^{m+1} [1 - (\kappa - m + 1) \rho^{\kappa-m} (\kappa - m) \rho^{\kappa-m+1}] \quad (8)$$

由于路由器队列采用混合制容量有限的排队模型, 所以可以不必利用 $\rho < 1$ 来进行系统稳定性的保证。为了推导方便, 现可假设 $\rho = 1$, 从而可以得到

$$N_d = \frac{m^m}{2m!} P_0 (\kappa - m + 1) (\kappa - m) \quad (9)$$

当结合数据优先级进行考虑时, 则可将到路由器的包按其优先级排队。如图 4 所示, 对数据队列, 分为按到达顺序排队和按优先级排队两种情况。若优先级为 4 的数据包为最高优先级的数据包, 则从图 4 中可以看出, 相对于不排队时, 优先级为 4 的数据包在路由器中的逗留时间和等待时间都有相应的减少, 使实时通信的效率得以提高。

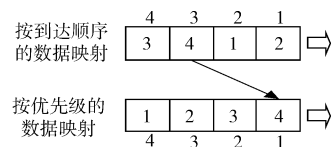


图 4 路由器队列中的数据映射

对任意的数据包 μ ,计算路由器此时的空闲概率为

$$P_{0\mu} = \left[\sum_{n=0}^{m-1} \frac{m^n}{n!} + \frac{m^m}{m!} (\kappa_\mu - m + 1) \right]^{-1} \quad (10)$$

μ 在路由器中的预计逗留时间为

$$W_\mu = \begin{cases} \frac{(m\rho)^m \rho P_{0\mu}}{m! (1-\rho)^2 \lambda (1-P_{0\mu})} [1 - (\kappa_\mu + 1) \rho^{\kappa_\mu} + \kappa_\mu \cdot \rho^{\kappa_\mu + 1}] & \rho \neq 1 \\ \frac{m^m P_{0\mu}}{m! \lambda (1-P_{0\mu})} (\kappa_\mu + 1) \kappa_\mu & \rho = 1 \end{cases} \quad (11)$$

对于第 k 阶段的映射节点 x_k^i ,设与其直接相连的下一阶段映射节点的个数为 N_{ki} ,其值也等于与其直接相连的到下一阶段的映射边个数。当到目的节点的跳数相同时 ,数据包的截止期剩余时间越小 ,说明越紧急 ,需要的优先级就越高 ;而当截止期的剩余时间相同时 ,数据包到目的节点的跳数越多 ,在后续传输中经历的时延可能就越大 ,所以 ,所需的优先级也应越高。因此 ,出于对业务的紧急性和公平性的考虑 ,对于映射节点 x_k^i 到 x_{k+1}^j 之间的映射边 ,将其接入次序为 l 时的权值定义为

$$L_{kl}^i(x_k^i, x_{k+1}^j) = \frac{\Delta\tau_{f_{kl}}^i - W_{f_{kl}}^l}{\Delta\tau_{f_{kl}}^i} / \sum_{j=1}^{N_{ki}} \frac{\Delta\tau_{f_{kl}}^i - W_{f_{kl}}^l}{\Delta\tau_{f_{kl}}^i} \quad (12)$$

其中: $W_{f_{kl}}^l$ 表示当映射节点 x_{k+1}^j 的接入次序排到第 l 位时 ,属于业务 f 的数据包 μ 的预计逗留时间; $i = 1, 2, \dots, |S_k|$; $j = 1, 2, \dots, N_{ki}$; $\Delta\tau_{f_{kl}}^i$ 为经过该映射边的数据包 μ 到规定的端到端时延截止期的剩余时间; $\Delta\tau_{f_{kl}}^i$ 为经过该映射边的数据包 μ 到目的节点所剩余的跳数。由于用户定义的业务优先级越高 ,其对应的时延截止期就越小 ,所以此参数也是对用户定义的业务优先级的体现。

3 算法仿真与分析

为了验证本文提出的优化方法 DPRSS (dynamic programming based real-time service scheduling) 的有效性 ,将其分别与参考文献 [10 ~ 13] 提出的一系列典型实时调度策略进行比较。仿真中 ,在一个 $1\ 000 \times 1\ 000$ 单位长度的区域中随机生成一个 25 节点的无线 Mesh 骨干网络拓扑 ,如图 5 所示。

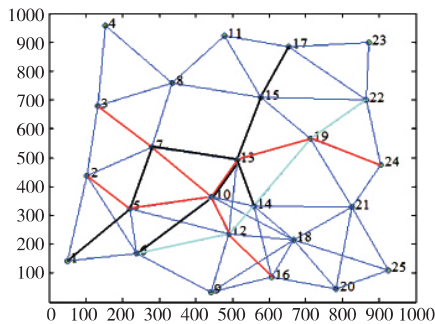


图 5 随机生成的网络拓扑 (25 个节点)

任意两点间的距离在 [50, 300] 内随机选取。由于本文关注的是调度问题 ,对于网络的连通性和业务的路由已知 ,所以在拓扑图中通过线段对两个节点进行连接来表示节点的连通性。为计算简便 ,文中假设各节点间信号覆盖 / 干扰的范围都相对保持固定 ,并且各无线链路容量都相等。设用户设置的优先级分为四种 ,截止期分别为 400 (优先级为 1) ,500 (优先级为 2) ,600 (优先级为 3) ,700 (优先级为 4) ,单位为时隙。网络中生成五条数据流 ,分别为 2-5-10-13-19-24、1-5-7-13-14、3-7-10-12-16、6-10-13-15-17、6-12-14-19-22 ,在图 5 中分别用五

种不同颜色的链路来表示。五条数据流的初始用户优先级随机分配。假设网络中的数据流服从参数为 λ 的泊松分布 , λ 的取值为 [0, 100] ,单位为时隙。仿真过程分别统计了 λ 在 0 ~ 100 的过程中 ,数据流 2-5-10-13-19-24 在优先级分别为 2 和 4 时各调度算法的端到端时延和全网的平均时延与丢包率情况 ,如图 6 ~ 9 所示。

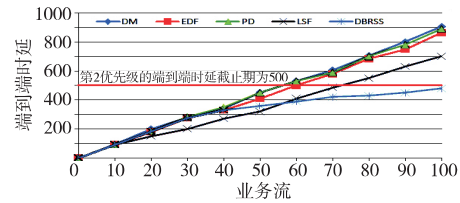


图 6 流 2-5-10-13-19-24 在业务优先级为 2 时的端到端时延

从图 6 中可以看出 ,当业务的优先级较高时 ,本文提出的 DPRSS 算法和传统的典型实时调度方法都能够具有较低的端到端时延。并且当业务流较少时 ,传统的典型实时调度方法比 DPRSS 算法时延更小。这是由于 DPRSS 算法是根据时延截止时间的约束进行调度的 ,不是单纯地追求时延越低越好 ,这样就可以在满足业务时延要求的前提下 ,节约更多的网络资源。但随着网络业务流的增多 ,传统的典型实时调度方法的时延优势逐渐消失 ,而 DPRSS 算法能够更最大限度地满足截止时间要求。

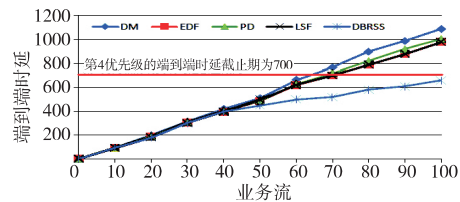


图 7 流 2-5-10-13-19-24 在业务优先级为 4 时的端到端时延

从图 7 中可以看出 ,随着网络中业务流的增多 ,传统的典型实时调度方法的端到端时延呈线性增加 ,当业务流速度达到 70 个每时隙时 ,所有的传统调度算法都已经无法对低优先级的业务进行实时性保障 ,而 DPRSS 算法则可以将低优先级业务的时延控制在端到端截止时间的范围之内。这是由于 DPRSS 算法始终以截止时间为参考进行资源的调度 ,所以可以根据业务的实时性需求动态地调整调度顺序。在保证端到端截止时间的前提下 ,通过牺牲部分高优先级业务的时延来使更多的低优先级业务能够满足时延需求。

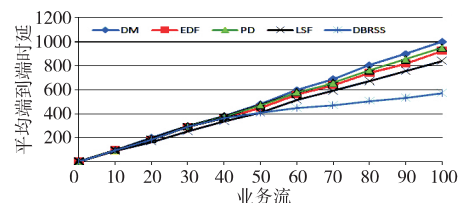


图 8 全网业务的平均端到端时延

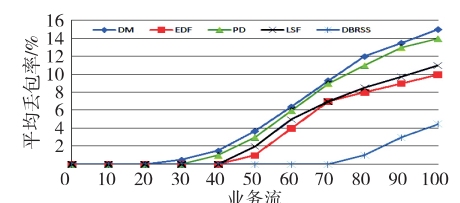


图 9 全网业务的平均丢包率

从图 8 和 9 中可以看出, DPRSS 算法对全网业务的平均端到端时延与丢包率的保障都比传统的调度机制有明显的提升。虽然在业务量比较小的情况下, DPRSS 算法与传统的经典调度方法的性能差不多, 都可以对时延有较好的保障, 但是考虑到 Mesh 骨干网的情况, 由于网络中存在着大量的业务要进行传输, 所以队列长度会很大, 拥塞情况时有发生, 此时随着队列长度的增加, DPRSS 算法的优势就会越来越明显。

4 结束语

本文针对具有较低节点移动性和对称性链路的无线 Mesh 网络, 提出了一种基于动态规划的实时业务调度算法。通过边集映射和阶段划分, 将端到端的调度问题转换为多阶段规划问题。根据队列时延的预测, 对链路的接入优先级进行重新划分, 最后通过优化目标函数求得在各个节点处业务的最优调度顺序。与以往的调度算法相比, 该算法的不同在于: 将动态规划的思想引入 Mesh 网络的调度中, 通过多阶段决策, 选择最优链路接入顺序, 为实时性业务提供保障。另外, 由于采用动态的多阶段决策方法, 可以兼顾不同优先级业务的实时性, 保证多实时性业务的公平性。最后的仿真结果证明了 DPRSS 算法的有效性。

参考文献:

- [1] Su Yisheng, Su S L, Li J S. Joint topology-transparent scheduling and QoS routing in mobile Ad hoc networks[J]. *IEEE Trans on Vehicular Technology*, 2014, 63(1): 372-389.
- [2] 何之栋, 张伟宁, 王海凤, 等. 工业无线网络路由及调度联合极值优化[J]. *信息与控制*, 2014, 43(2): 152-158.
- [3] 伸桂林, 张新有. WiMAX Mesh 网络中一种集中式 QoS 调度算法[J]. *计算机应用研究*, 2012, 29(11): 4269-4272.
- [4] Wellons J, Xue Yuan. The robust joint solution for channel assignment and routing for wireless Mesh networks with time partitioning[J]. *Ad hoc Networks*, 2014, 13(1): 210-221.
- [5] Afolabi R O, Dadlani A, Kim K. Multicast scheduling and resource allocation algorithms for OFDMA-based systems: a survey[J]. *IEEE Communications Surveys & Tutorials*, 2013, 15(1): 240-254.
- [6] Rhee I, Warrier A, Min J, et al. DRAND: distributed randomized TDMA scheduling for wireless Ad hoc networks[C]//Proc of the 7th ACM International Symposium on Mobile Ad hoc Networking and Computing. New York: ACM Press, 2006.
- [7] Li Yun, Ephremides A. A joint scheduling, power control, and routing algorithm for Ad hoc wireless networks[J]. *Ad hoc Networks*, 2007, 5(7): 959-973.
- [8] Capone A, Carello G. Scheduling optimization in wireless Mesh networks with power control and rate adaptation[C]//Proc of the 3rd Annual IEEE Communications Society on Sensor and Ad hoc Communications and Networks. [S. l.]: IEEE Press, 2006: 138-147.
- [9] Fu Liqun, Liew S C, Huang Jianwei. Fast algorithms for joint power control and scheduling in wireless networks[J]. *IEEE Trans on Wireless Communications*, 2010, 9(3): 1186-1197.
- [10] Socci D, Poplavko P, Bensalem S, et al. Mixed critical earliest deadline first[C]//Proc of the 25th Euromicro Conference on Real-time Systems. [S. l.]: IEEE Press, 2013: 93-102.
- [11] Rivas J M, Gutierrez J J, Palencia J C, et al. Optimized deadline assignment for tasks and messages in distributed real-time systems[C]//Proc of the 8th International Conference on Embedded Systems and Applications. 2010: 1-14.
- [12] Prajapati V, Shah A, Balani P. Design of new scheduling algorithm LLF_DM and its comparison with existing EDF, LLF, and DM algorithms for periodic tasks[C]//Proc of International Conference on Intelligent Systems and Signal Processing. 2013.
- [13] Qamhieh M, Midonnet S, George L. Dynamic scheduling algorithm for parallel real-time graph tasks[J]. *ACM SIGBED Review*, 2012, 9(4): 25-28.
- [14] Hu Yi, Li Xiangyang, Chen Haiming, et al. Distributed call admission protocol for multi-channel multi-radio wireless networks[C]//Proc of IEEE Global Telecommunications Conference. 2007: 2509-2513.
- [15] Brzezinski A, Zussman G, Modiano E. Distributed throughput maximization in wireless Mesh networks via pre-partitioning[J]. *IEEE/ACM Trans on Networking*, 2008, 16(6): 1406-1419.
- [16] Yan Mao, Lam K Y, Han Song, et al. Hyper graph based data link layer scheduling for reliable packet delivery in wireless sensing and control networks with end-to-end delay constraints[J]. *Information Sciences*, 2014, 278(10): 34-55.
- [17] 孙荣恒. 随机过程及其应用[M]. 北京: 清华大学出版社, 2004.
- [4] Donoho D L, Tsai Y, Drori I. Sparse solution of underdetermined systems of linear equations by stagewise orthogonal matching pursuit[J]. *IEEE Trans on Information Theory*, 2012, 58(2): 1094-1121.
- [5] Needell D, Vershynin R. Signal recovery from incomplete and inaccurate measurements via regularized orthogonal matching pursuit[J]. *IEEE Journal of Selected Topics in Signal Processing*, 2010, 4(2): 310-316.
- [6] Davenport M A, Needell D, Wakin M B. CoSaMP with redundant dictionaries[C]//Proc of Asilomar Conference on Signals, Systems and Computers. 2012: 263-267.
- [7] Dai Wei, Milenkovic O. Subspace pursuit for compressive sensing signal reconstruction[J]. *IEEE Trans on Information Theory*, 2009, 55(5): 2230-2249.
- [8] Do T T, Lu Gan, Nguyen N. Sparsity adaptive matching pursuit algorithm for practical compressed sensing[C]//Proc of Asilomar Conference on Signals, Systems and Computers. 2008: 581-587.
- [9] 范海宁, 郭英. 一种新的跳频信号参数盲估计算法[J]. *信号处理*, 2009, 25(11): 1754-1758.
- [10] 白凌云, 梁志毅, 徐志军. 基于压缩感知信号重建的自适应正交匹配追踪算法[J]. *计算机应用研究*, 2011, 28(11): 4060-4063.
- [11] 杨真真, 杨震, 孙林慧. 信号压缩重构的正交匹配追踪类算法综述[J]. *信号处理*, 2013, 29(4): 486-496.
- [12] 张宗福. 基于压缩感知和改进自适应正交匹配的稀疏信号重构[J]. *计算机测量与控制*, 2014, 22(5): 1568-1571.
- [13] 牛亚坤, 玉振明, 李陶深, 等. 一种压缩采样硬阈值追踪压缩感知重构算法[J]. *计算机应用研究*, 2015, 32(8): 2286-2288.
- [14] 王芳星, 刘顺兰. 一种改进的正则化自适应匹配追踪算法[J]. *杭州电子科技大学学报: 自然科学版*, 2015, 35(1): 79-83.

(上接第 2485 页)