

一种基于端到端多路径的最优化流量控制算法

薛 森,董 平,张思东,张宏科

(北京交通大学下一代互联网互联设备国家工程实验室,北京 100044)

摘 要: 随着网络技术的发展,端到端多路径传输正在成为一个热点研究问题.本文研究了基于端到端多路径的最优化流量控制,提出了一种进行最优网络资源分配的分布式算法.该算法充分利用端到端多路径的路径多样性,达到有效的带宽利用和带宽聚合.通过证明,该算法在延时条件下是全局渐进稳定的.仿真结果证明了该算法的快速收敛性和实用性.

关键词: 端到端多路径;最优化流量控制;稳定性;效用最大化;路径多样性

中图分类号: TP393 **文献标识码:** A **文章编号:** 0372-2112 (2012) 03-0564-07

电子学报 URL: <http://www.ejournal.org.cn>

DOI: 10.3969/j.issn.0372-2112.2012.03.025

An Algorithm for Optimal Flow Control Based End-to-End Multipath

XUE Miao, DONG Ping, ZHANG Si-dong, ZHANG Hong-ke

(National Engineering Laboratory for Next Generation Internet Interconnection Devices, Beijing Jiaotong University, Beijing 100044, China)

Abstract: With the evolving of the network technologies, there has been a growing interest in the end-to-end multipath transfer. This paper investigates the optimal flow control based multiple end-to-end paths and proposes a distributed algorithm to optimally allocate the network resource. The algorithm achieves efficient bandwidth allocation by taking the path diversity into consideration. The algorithm proposed is shown to be asymptotically global stable in the absence of the propagation delay. The simulations demonstrate the rapid convergence and practicality of the algorithm.

Key words: end-to-end multipath; optimal flow control; stability; utility maximization; path diversity

1 引言

随着网络技术的发展,一方面网络接入技术不断多样化,有线和无线接入技术不断成熟并商用,例如 xDSL, 3G, 80211x 等;另一方面,接入设备成本不断降低,越来越多的终端开始配置多个网络适配器.例如笔记本电脑通常配有 LAN 接口和 WiFi 接口,也可以支持 3G 无线接入;而一般智能手机和平板电脑都支持 WiFi 和 3G 接入.因此,基于多种接入技术的多宿终端正在成为下一代互联网的主要特征.

尽管终端设备可以支持多种接入方式,但是传统的 TCP, UDP 等传输协议却只能使用由一种接入方式形成的端到端路径传输数据,不能充分利用多条路径带来的优势.因此,端到端的多路径传输技术受到研究人员的关注. mTCP^[1], pTCP^[2] 通过对 TCP 扩展, LS-SCTP^[3], CMT-SCTP^[4] 通过对 SCTP 扩展,初步实现了在端到端的多条路径上分发数据.

相比于单路径传输,端到端多路径可以有效的聚合多条路径的带宽,从而为用户提供更好的 QoS 保障;由于同时使用多条端到端的路径,单条路径失败不会影响服务的连续性,从而为终端提供网络层冗余;多条路径同时使用,可以根据网络中的拥塞状况动态的调整在不同路径的发送速率,从而实现在网络边缘处的负载均衡;所有的应用数据都从多条路径传输,对任一单条路径的监听嗅探无法有效的恢复初始数据内容,具有更好的安全性.

正是由于如上所述的众多优点,端到端多路径的研究逐渐成为热点. IETF 新成立了 MPTCP^[5] 工作组来研究如何在后向兼容的情况下开发新的 TCP 扩展,使其支持端到端多路径传输;欧洲的 Trilog^[6] 计划将端到端的多路径传输作为其资源池 (Resource Pooling)^[7] 概念的一部分;文献[8,9]提出的一体化网络架构下的多连接多路径 (Multi-Connection Multi-Path, MCMP) 也将端到端多路径作为多路径研究的一个方面.

虽然关于端到端多路径进行了很多研究,但是这些研究大多集中在协议实现方面,而关于端到端多路径的最优化的流量控制,资源分配以及使用端到端多路径的网络稳定性方面,相关研究却很少.

近年来,利用网络效用最大化(Network Utility Maximization, NUM)^[10,11]的思想进行拥塞控制算法设计和传输协议的设计,是实现网络资源的最优分配的一个主要方法. Kelly^[12]首次提出网络效用最大化模型.基于该模型,网络的资源分配被看作一个关于用户效用的函数,并引入“影子价格”(Shadow Price)用来调节资源的供需关系.在此基础上设计的源端拥塞控制算法,可以得到全局资源的最优分配. Low^[13]从网络效用最大化模型的对偶问题出发,利用最优化理论中的梯度算法设计了一种源-链路端的分布式算法,该算法能够收敛到模型的全局最优点,即能根据各个链路的最优价格,使网络最优的分配资源,达到最优的流量控制.

上述提出的对偶算法在单路径条件下使用可以得到最优解,而且算法易于为协议设计采用,但是却无法有效扩展到多路径的情况.因为尽管在多路径情况下对偶变量可以有效收敛,但是对应的原变量却不一定唯一. Wang^[14]在求解多路径最大网络效用的对偶问题时,为最优化问题的目标函数引入了一个二次项因子,从而将目标函数转化为变量的严格凹函数,进而得到了一种多路径情况下的最优的资源分配的分布式算法.文献[14]中采用的算法与本文的思路类似,但是没有给出使用该算法的系统稳定性的证明. Lin^[15]分析了多路径网络效用最大化问题,为了得到唯一的最优资源分配,通过求解对偶问题,利用近似优化算法(Proximal Optimization Algorithms, POA)设计了一种两层迭代的分布式算法,并对该算法的收敛性进行证明,但算法忽略了多路径情况下路径重叠情况的考虑.

由于采用求解网络效用最大化的对偶问题的方法通常受到得限制条件较少而且研究问题易于分解,本文采用该方法来求解端到端多路径情况下网络资源最优分配.本文通过为原问题效用函数添加一个二次项因子,解决求解对偶问题时对偶变量原变量收敛不一致的问题.在资源分配过程中,我们充分考虑了端到端多路径的路径多样性.通过对偶问题的求解,本文提出了一种实现最优的网络流量控制的分布式算法,并证明在不考虑传输时延的条件下,使用该算法的系统是全局渐进稳定的.

2 端到端多路径效用模型

我们首先简要描述基于端到端多路径的效用模型.考虑网络资源是由一系列的单向链路组成,链路集合 $L = \{1, \dots, L\}$, 对应链路的容量为 $c_l, l \in L$. 网络资

源被分配到用户集合 $I = \{1, \dots, I\}$. 取用户 i 的效用函数为 $f_i(\cdot)$, 其代表当用户 i 从网络中分得一定的资源后用户的满意度,这里的资源主要是指带宽. 本文侧重研究端到端多路径承载弹性服务时的资源分配情况,因此假设 $f_i(\cdot)$ 为连续的增函数并且是严格的凹函数^[16]. 设每个用户 i 有 $J(i)$ 条端到端的路径, x_{ij} 为用户 i 在路径 j 上分得的网络资源. 则用户 i 获得的总的效用为 $f_i(\sum_{j=1}^{J(i)} x_{ij})$. 取 E_j^l 用于标识用户 i 的第 j 条路径是否经过了链路 l , 如果经过,则 E_j^l 为 1, 否则为 0.

根据上面的模型,得到端到端多路径的效用最大化问题的数学描述,

$$\begin{aligned} \max_{x_{ij} \geq 0} \quad & \sum_{i=1}^I f_i\left(\sum_{j=1}^{J(i)} x_{ij}\right) \\ \text{subject to} \quad & \sum_{i=1}^I \sum_{j=1}^{J(i)} E_j^l x_{ij} \leq c_l, \forall l \in L \end{aligned} \quad (1)$$

限制条件说明经过链路 l 的总的源速率不能超过其容量 c_l . $\mathbf{x} = \{x_{11}, x_{21}, \dots, x_{1j(1)}, x_{21}, \dots, x_{2j(2)}, \dots, x_{ij(i)}\}$ 为各条路径的速率分配向量.

为了保证路径多样性,本文采用开放性的假设(Wild Assumption),即每个源经过同一条链路只有一次,保证了本文中的分析是基于独立的端到端的多路径.

设定 $\sum_{j=1}^{J(i)} E_j^l = 1$, 从而避免了同一个源的多条路径经过重叠的链路.

在单路径的情况,对偶算法可以求得文献[13]中效用函数最大化的最优解. 但当我们把单路径情况下的效用最大化问题扩展到式(1)时,对偶算法就有些不适用.这主要是因为尽管式(1)中的目标函数是关于 $\sum_{j=1}^{J(i)} x_{ij}$ 严格凹的,但是其关于 x_{ij} 并不是严格凹的. 因此使得式(1)的目标函数最大化的最优解 \mathbf{x} 不是唯一的,而式(1)的对偶函数也不是处处可微的. 此时虽然可以通过次梯度算法求得最优的可收敛的对偶解,而由此得到的向量 \mathbf{x} 却不一定是最优解,使用 \mathbf{x} 进行资源分配的系统也可能不是全局稳定的.

本文采用对偶算法进行求解,因为对偶算法可以更容易设计出易于实际使用的分布式算法. 因此为了使对偶算法的最优解与原算法的最优解都使式(1)中的目的函数收敛到同一值,则必须使每个用户的效用函数 $f_i(\cdot)$ 是关于 x_{ij} 的严格凹函数. 我们为每一个 x_{ij} 引入一个辅助变量 y_{ij} , 为用户的效用函数添加一个二次项因子. 则用户 i 的效用函数变为

$$f_i\left(\sum_{j=1}^{J(i)} x_{ij}\right) - \sum_{j=1}^{J(i)} \frac{1}{2} (x_{ij} - y_{ij})^2 \quad (2)$$

端到端多路径的效用最大化问题转化为

$$\begin{aligned} \max_{x_{ij} \geq 0} & \sum_{i=1}^l \left(f_i \left(\sum_{j=1}^{J(i)} x_{ij} \right) - \sum_{j=1}^{J(i)} \frac{1}{2} (x_{ij} - y_{ij})^2 \right) \\ \text{subject to} & \sum_{i=1}^l \sum_{j=1}^{J(i)} E_{ij}^l x_{ij} \leq c_l, \forall l \in L \end{aligned} \quad (3)$$

对比式(1)和式(3)可以看到,式(3)和式(1)的最优解是一致的.假设 \mathbf{x}^* 为式(1)的最优解,则令 $\mathbf{x} = \mathbf{x}^*$, $\mathbf{y} = \mathbf{x}^*$ 即为式(3)的最优解.同样,如果 \mathbf{x}^* 为式(3)的最优解,只需使 $\mathbf{y} = \mathbf{x}^*$, 则 \mathbf{x}^* 也是式(1)的最优解.

3 端到端多路径分布式算法

式(2)在 y_{ij} 固定的情况下,其是关于 x_{ij} 的严格凹函数,并且是连续可微的,因此原问题式(3)与其对偶问题没有对偶间隙^[17].通过求解对偶问题的最优解可以得到原问题的最优解,从而得到原问题的最优化流量控制和资源分配算法.

3.1 目的函数的凹性

定理 1 定义函数 $\Phi: \mathbf{R}^n \rightarrow \mathbf{R}$ 为

$$\Phi(\mathbf{x}) = f_i \left(\sum_{j=1}^{J(i)} x_{ij} \right) - \sum_{j=1}^{J(i)} \frac{1}{2} (x_{ij} - y_{ij})^2 \rho_l \quad (4)$$

其中 \mathbf{x}, \mathbf{y} 为向量,且 \mathbf{y} 为定值, $f_i(\cdot)$ 为严格凹函数,则 $\Phi(\mathbf{x})$ 是关于 \mathbf{x} 的凹函数,并且是连续可微的.

证明 为了证明 $\Phi(\mathbf{x})$ 是关于 \mathbf{x} 的凹函数,令 \mathbf{x}_1 和 \mathbf{x}_2 为 \mathbf{R}^n 中的任意两个向量, α 为在 $[0, 1]$ 中的一个标量.

由于 $f_i(\cdot)$ 为严格凹函数,而 $-\sum_{j=1}^{J(i)} (x_{ij} - y_{ij})^2 / 2$ 也是关于 x_{ij} 的凹函数.因此如式(5)所示,当 \mathbf{y} 为一定值时 $\Phi(\mathbf{x})$ 为关于 \mathbf{x} 的凹函数:

$$\begin{aligned} & \alpha \Phi(\mathbf{x}_1) + (1 - \alpha) \Phi(\mathbf{x}_2) \\ &= \alpha \left[f_i \left(\sum_{j=1}^J x_{1j} \right) - \sum_{j=1}^J \frac{1}{2} (x_{1j} - y_{1j})^2 \right] \\ &+ (1 - \alpha) \left[f_i \left(\sum_{j=1}^J x_{2j} \right) - \sum_{j=1}^J \frac{1}{2} (x_{2j} - y_{2j})^2 \right] \\ &= \alpha f_i \left(\sum_{j=1}^J x_{1j} \right) + (1 - \alpha) f_i \left(\sum_{j=1}^J x_{2j} \right) \\ &- \left[\alpha \sum_{j=1}^J \frac{1}{2} (x_{1j} - y_{1j})^2 + (1 - \alpha) \sum_{j=1}^J \frac{1}{2} (x_{2j} - y_{2j})^2 \right] \\ &\leq f_i \left(\sum_{j=1}^J \alpha x_{1j} + \sum_{j=1}^J (1 - \alpha) x_{2j} \right) \\ &- \left[\alpha \sum_{j=1}^J \frac{1}{2} (x_{1j} - y_{1j})^2 + (1 - \alpha) \sum_{j=1}^J \frac{1}{2} (x_{2j} - y_{2j})^2 \right] \\ &\leq f_i \left(\sum_{j=1}^J \alpha x_{1j} + (1 - \alpha) x_{2j} \right) \\ &- \sum_{j=1}^J \frac{1}{2} (x_{1j} + (1 - \alpha) x_{2j} - y_j)^2 \\ &= \Phi(\alpha \mathbf{x}_1 + (1 - \alpha) \mathbf{x}_2) \end{aligned} \quad (5)$$

而关于 $\Phi(\mathbf{x})$ 的连续可微性,由于 $f_i(\cdot)$ 是连续可

微的,而当 y_{ij} 为一定值时, $-\sum_{j=1}^{J(i)} (x_{ij} - y_{ij})^2 / 2$ 也是连续可微的,因此易证 $\Phi(\mathbf{x})$ 是连续可微的.

由于 $\Phi(\mathbf{x})$ 是凹函数,并且是连续可微的,则通过对偶方法可以解得式(3)的对偶最优解.

3.2 分布式算法设计

原问题式(3)的拉格朗日函数如下

$$\begin{aligned} L(\mathbf{x}, \mathbf{y}, \boldsymbol{\rho}) &= \sum_{i=1}^l f_i \left(\sum_{j=1}^{J(i)} x_{ij} \right) - \sum_{l=1}^L \rho_l \left(\sum_{i=1}^l \sum_{j=1}^{J(i)} E_{ij}^l x_{ij} - c_l \right) \\ &- \sum_{i=1}^l \sum_{j=1}^{J(i)} \frac{1}{2} (x_{ij} - y_{ij})^2 \end{aligned} \quad (6)$$

根据文献[17],则对偶问题为

$$\min_{\rho \geq 0} D(\boldsymbol{\rho}, \mathbf{y}) = \min_{\rho \geq 0} \max_{x_{ij} \in \mathbf{x}} L(\mathbf{x}, \mathbf{y}, \boldsymbol{\rho}) \quad (7)$$

由于式(4)是严格凹的并且可微,因此式(7)关于 ρ 是可微的,并且存在极小值^[17].这里我们将 ρ_l 看作是链路 l 对分配的每单位带宽的成本价格,并将 ρ_l 的迭代更新过程称为链路端算法^[13].这里,我们通过梯度投影法(Gradient Projection Method)求解对偶问题的最小值.由于式(7)关于 ρ_l 的梯度为

$$\frac{\partial D}{\partial \rho_l} = c_l - \sum_{i=1}^l \sum_{j=1}^{J(i)} E_{ij}^l x_{ij} \quad (8)$$

则可以通过梯度法迭代得到链路价格 ρ_l 的更新,算法如下

$$\rho_l(t+1) = \left[\rho_l(t) + \gamma^l \left(\sum_{i=1}^l \sum_{j=1}^{J(i)} E_{ij}^l x_{ij}(t) - c_l \right) \right]^+ \quad (9)$$

其中 $\sum_{i=1}^l \sum_{j=1}^{J(i)} E_{ij}^l x_{ij}(t)$ 表示经过链路 l 的速率之和,可以理解为对链路 l 的资源需求; $[\cdot]^+$ 表示取值空间在 $[0, +\infty)$; $t+1$ 表示从 t 开始的 1 个单位迭代时间; γ^l 为迭代步进.

同样,式(7)关于 x_{ij} 的梯度为

$$\frac{\partial L}{\partial x_{ij}} = f_i' \left(\sum_{j=1}^{J(i)} x_{ij} \right) - \sum_{l \in \mathcal{J}} \rho_l - (x_{ij} - y_{ij}) \quad (10)$$

我们将 x_{ij} 的迭代过程称为源端算法,通过梯度迭代可以得到源端速率

$$\begin{aligned} x_{ij}(t+1) &= x_{ij}(t) \\ &+ \alpha^{ij} \left(f_i' \left(\sum_{j=1}^{J(i)} x_{ij}(t) \right) - \sum_{l \in \mathcal{J}} \rho_l(t) - (x_{ij}(t) - y_{ij}(t)) \right) \end{aligned} \quad (11)$$

其中 $\sum_{l \in \mathcal{J}} \rho_l(t)$ 为第 i 个用户的第 j 条路径上的链路代价之和; $\alpha^l > 0$ 为迭代步进,以确定 x_{ij} 向最优解靠近的速度.

为了简化算法, y_{ij} 在每次 x_{ij} 迭代时固定的表示为上一次 x_{ij} 的值.

$$y_{ij}(t+1) = x_{ij}(t) \quad (12)$$

3.3 目的函数的凹性

源端算法:

(1)源端从网络中接收每条端到端路径的价格反馈 $\sum_{l \in j} \rho_l(t)$;

(2)按照上述式(11)、式(12)对源端发送速率进行更新;

(3)使用更新后的各条端到端路径速率 $x_{ij}(t+1)$ 发送数据.

链路端算法:

(1)计算所有经过该链路的路径速率 $x_{ij}(t)$ 的和;

(2)按照式(9)更新该链路的价格;

式(3)将更新后的价格以其经过的链路价格和的形式 $\sum_{l \in j} \rho_l(t+1)$ 通知源端.

从算法描述可以看出,源端算法关于速率的调整只需知道该端到端路径的链路价格之和,而无需知道其他路径的价格^[12];链路端算法价格的调整只需知道经过该链路的速率总和,而无需考虑经过其他链路速率.因此该算法是一个分布式算法,它将一个全局求解最优问题分解成为每个用户参与的最优化问题,具有良好的可扩展性并且易于应用到实际网络协议中.

3.4 系统稳定性

由于当 \mathbf{y} 一定时, $\Phi(\mathbf{x})$ 是关于 \mathbf{x} 的严格凹函数.因此,式(3)存在最优向量 \mathbf{x}^* , 此时 $\mathbf{y} = \mathbf{x}^*$,

$$f_i' \left(\sum_{j=1}^{J(i)} x_{ij} \right) = \sum_{l \in j} \rho_l.$$

由式(11)得

$$x_{ij}(t+1) - x_{ij}(t) =$$

$$\alpha^{\bar{y}} \left(f_i' \left(\sum_{j=1}^{J(i)} x_{ij}(t) \right) - \sum_{l \in j} \rho_l(t) - (x_{ij}(t) - y_{ij}(t)) \right) \quad (13)$$

当每次迭代的时间间隔趋向于 0 时,则式(13)成为

$$\frac{dx_{ij}(t)}{dt} = \alpha^{\bar{y}} \left(f_i' \left(\sum_{j=1}^{J(i)} x_{ij}(t) \right) - \sum_{l \in j} \rho_l(t) - (x_{ij}(t) - y_{ij}(t)) \right) \quad (14)$$

定理 2 令

$$V(\mathbf{x}) = \sum_{i=1}^I \left(f_i \left(\sum_{j=1}^{J(i)} x_{ij}(t) \right) - \sum_{l \in j} \rho_l(t) - (x_{ij} - y_{ij})^2 \right),$$

取采用式(14)算法的连续时间系统的利雅普诺夫函数为 $W(\mathbf{x}) = V(\mathbf{x}^*) - V(\mathbf{x})$, 其中 \mathbf{x}^* 为式(3)的最优解,且当且仅当 $\mathbf{x} = \mathbf{x}^*$ 时, $W(\mathbf{x}) = 0$. 则式(14)描述的系统是渐进稳定的,并且式(1)和式(3)的最优解都收敛到 \mathbf{x}^* .

证明 由于 \mathbf{x}^* 是 $V(\mathbf{x})$ 的最优解,因此 $W(\mathbf{x}) > 0$,

$\forall \mathbf{x} \neq \mathbf{x}^*$. 对 $W(\mathbf{x})$ 微分,得

$$\begin{aligned} & \frac{\partial W(\mathbf{x}(t))}{\partial t} \\ &= - \frac{\partial V(\mathbf{x})}{\partial x_{ij}} \frac{\partial x_{ij}(t)}{\partial t} \\ &= - \left(f_i' \left(\sum_{j=1}^{J(i)} x_{ij}(t) \right) - \sum_{l \in j} \rho_l(t) - (x_{ij}(t) - y_{ij}(t)) \right) \\ & \quad * \alpha^{\bar{y}} \left(f_i' \left(\sum_{j=1}^{J(i)} x_{ij}(t) \right) - \sum_{l \in j} \rho_l(t) - (x_{ij}(t) - y_{ij}(t)) \right) \\ &= - \alpha^{\bar{y}} \left(f_i' \left(\sum_{j=1}^{J(i)} x_{ij}(t) \right) - \sum_{l \in j} \rho_l(t) - (x_{ij}(t) - y_{ij}(t)) \right)^2 \\ &< 0 \end{aligned} \quad (15)$$

由上可得, \mathbf{x}^* 既是式(3)的最优解也是式(3)所描述系统的平衡点,因此系统是渐进稳定到最优点的.进而说明采用 3.3 节的离散算法也可以收敛到 \mathbf{x}^* .

令 $\mathbf{y} = \mathbf{x}^*$, 此时任意 $-\sum_{j=1}^{J(i)} (x_{ij} - y_{ij})^2 / 2 = 0$, 则 \mathbf{x}^*

同样是式(1)的最优解,并且采用 3.3 节的算法式(1)同样可以收敛到 \mathbf{x}^* . 但是由于式(1)的目的函数关于 x_{ij} 不是严格凹的,虽然我们可以得到其最优解 \mathbf{x}^* , 而其不一定是唯一的,但却可以使式(1)得到唯一的最大效用.

4 仿真与分析

4.1 流层面的仿真实验

我们用数值仿真实验验证上述分布式算法在流层面的应用.在流层面我们只考虑抽象的数据流.仿真实验采用如图 1 所示的简单网络作为数值试验的仿真拓扑.该拓扑有 3 个发送端和 3 个接收端,其中 S1 是采用端到端的单路径传输,传输路径为

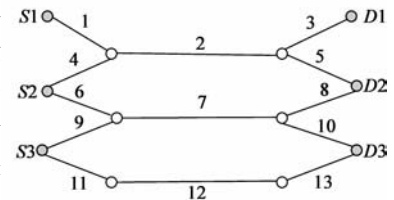


图1 数值仿真拓扑

$\{1, 2, 3\}$, S2、S3 采用端到端的两条路径传输,传输路径为 $\{4, 2, 5; 6, 7, 8\}$, $\{9, 7, 10; 11, 12, 13\}$. 每条端到端路径的速率为 $\mathbf{x} = \{x_1, x_{21}, x_{22}, x_{31}, x_{32}\}$. 由图 1 可以看到,链路 2 由 x_1 和 x_{21} 共享,链路 7 由 x_{22} 和 x_{31} 共享,链路 12 由 x_{32} 使用. 设定拓扑中单向路径的传输能力按标示的顺序为 $\mathbf{C} = \{3, 4, 3, 2, 2, 2, 3, 2, 2, 2, 2, 1, 5, 2\}$, 源 $\mathbf{S} = \{S1, S2, S3\}$ 对应的效用函数为 $a_1 \log(x_1 + 1)$, $a_2 \log((x_{21} + x_{22}) + 1)$, $a_3 \log((x_{31} + x_{32}) + 1)$, 其中 a_1, a_2, a_3 分别为 4, 6, 2. 取式(10), 式(12)中链路端和源端的步进速率分别为 $\gamma^l = 0.05$, $\alpha^{\bar{y}} = 0.1$.

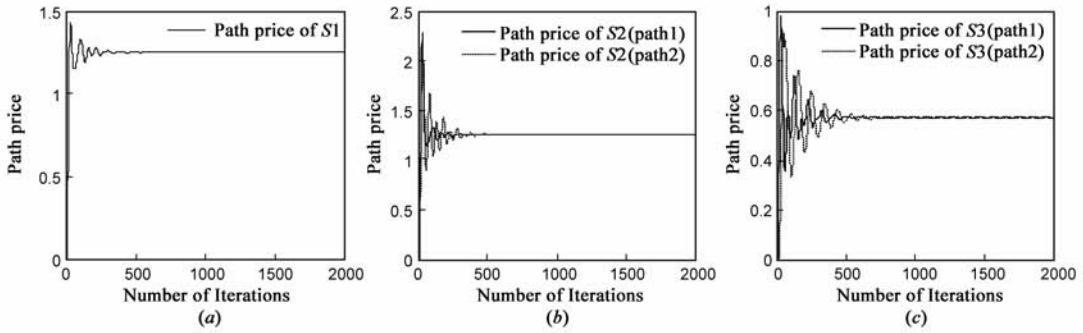


图2 路径价格的收敛

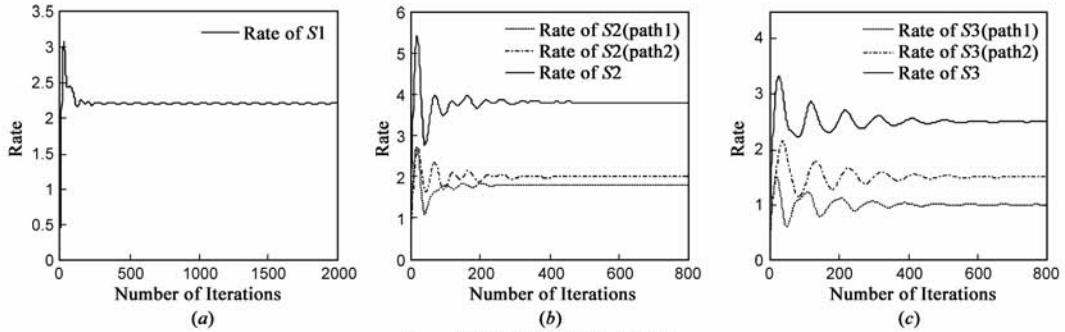


图3 端到端路径速率的收敛

从图2、图3可以看到,端到端路径的速率能够快速收敛到平衡点, x 收敛到 $\{2.2, 1.8, 2.0, 1.0, 1.5\}$, 而从源端观测到的路径价格也收敛到 $\{1.25, 1.25, 1.25, 0.57, 0.57\}$. 从图2(a)和图3(a)中可以看到,尽管多路径和单路径共同在同一个网络中使用,但最终都收敛到收敛,据此我们可以判断在实际网络中端到端多路径和端到端单路径的混合使用不会引起网络的不稳定. 而由于使用了没有重叠链路的多路径进行数据传输,可以看到 S2、S3 都分到了更多的网络资源,因此也提高了应用效用.

虽然图1的拓扑较为简单,但是分析式(9)和式(11)可以发现, ρ_l 的求解是要求对所有经过链路 l 的流量进行加性操作,而 x_{ij} 的求解是要求对所有流量经过的端到端路径的链路价格进行加性操作. 因此,图1所示拓扑可以产生包含更多节点的复杂拓扑的相同仿真效果.

4.2 包层面的仿真验证

本小节采用 ns2 对建议的分布式算法在包层面进行验证,其中端到端多路径采用基于流控制传输协议(SCTP)的 E2EMPT 扩展^[18]. 分析式(9),由于实际链路中超过链路容量的流量会被缓存在路由器的缓冲区中,根据文献[19],我们将式(9)修改如下:

$$\rho_l(t) = \gamma^l \frac{r_l(t)}{b_l} \quad (16)$$

其中, $r_l(t)$ 为 t 时刻进入链路 l 的节点缓存中的数据量, b_l 为缓存区大小.

基于包层面的仿真验证采用图4所示的“哑铃型”拓扑,该拓扑能较好的模拟网络中一个源端资源需求受到其他源端资源需求扰动的情況. 图中 $S \#$ 为数据源, $D \#$ 为数据接收端, $R \#$ 为路由器. 其中端到端单路径数据源采用 TCP 传输数据. 我们用参数向量(链路带宽,单向链路时延,链路丢包率)描述链路状态. 设定 Link1 和 Link2 的参数向量分别为 $(10\text{Mbps}, 20\text{ms}, 0.001)$, $(10\text{Mbps}, 25\text{ms}, 0.002)$. 端到端多路径的接入链路参数向量为 $(6\text{Mbps}, 10\text{ms}, 0)$, 其它端到端单路径的接入链路参数向量为 $(7\text{Mbps}, 5\text{ms}, 0)$. 所有仿真持续时间为 250s. 仿真先开启源 S3 传输 SCTP 数据包,经过 100s,启动源 S1 和 S4 传输 TCP 数据包,再经过 100s,启动 S2 和 S5 传输 TCP 数据包. 仿真中设定 S2 的效用函数为 $\log((x_{31} + x_{32}) + 1)$, 其它数据源的效用函数为 $\log(x + 1)$. 由于 Link1 和 Link2 的参数设置类似,我们只关注经过 Link1 的流量变化情况,将 S1 的源速率标识为“TCP1”, S2 的源速率标识为“TCP2”. “E2EMPT-Path1”表

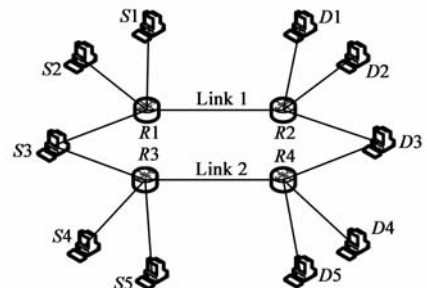


图4 包层面仿真拓扑

示 S_2 在 Link2 上的速率,“E2EMPT”表示 S_2 总的速率。

从图 5 中可以看出,端到端多路径的速率可以收敛到稳定值.虽然经过两次扰动,但可以迅速收敛到新的稳定值.因此,建议的端到端多路径的流量控制算法是稳定的,而且不会影响端到端单路径的系统稳定性.同时可以看到, S_2 采用 E2EMPT 可以充分聚合网络资源,达到更大的源端速率.图 6 和图 7 分别显示了 Link1 和 Link2 的链路价格变化情况.由于接入链路不是传输的

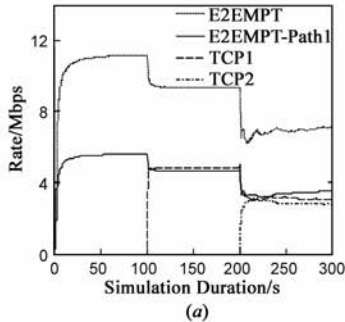


图5 源端发送速率

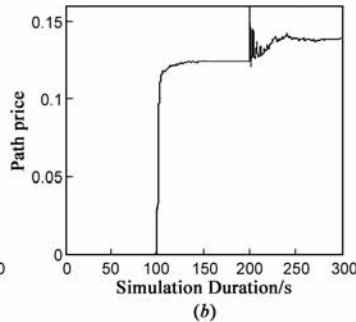


图6 Link 1 价格

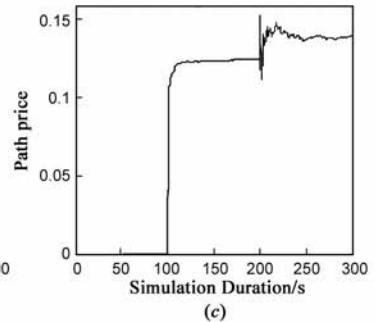


图7 Link 2 价格

5 结论和未来工作

本文设计了一种用于在端到端多路径数据传输中进行最优化流量控制和资源分配的分布式算法,并通过仿真验证了该算法的快速收敛性和实用性.本文中,对端到端多路径的路径多样性进行了描述,并在分析和仿真中均设定条件以保证使用无重叠链路的端到端多路径.通过仿真模拟端到端多路径和端到端单路径的混合网络,结果显示网络可以维持在稳定状态运行,并且端到端多路径的数据源在任一路径的速率都可以收敛。

由于本文模型是建立在无网络时延的条件下,因此下一步我们对有网络时延情况下基于网络效用最大化的端到端多路的最优化流量控制进行分析,同时研究非严格凹的效用函数下的数学模型。

参考文献

- [1] M Zhang, J W Lai, A Krishnamurthy, L Peterson, R Wang. A transport layer approach for improving end-to-end performance and robustness using redundant paths [A]. Proceedings of the Annual Conference on USENIX Annual Technical Conference [C]. Berkeley, CA, USA: USENIX Association, 2004. 8 – 8.
- [2] H Y Hsieh, R Sivakumar. pTCP: An end-to-end transport layer protocol for striped connections [A]. Proceedings of the 10th IEEE International Conference on Network Protocols [C]. Washington DC, USA: IEEE Computer Society, 2002. 24 – 33.
- [3] Al AE, T Saadawi, M Lee. LS-SCTP: A bandwidth aggregation technique for stream control transmission protocol [J]. Computer Communications, 2004, 27(10): 1012 – 1024.

瓶颈链路,接入链路的价格始终为 0,因此 Link1 和 Link2 的价格即为对应路径的价格.从图中可以看出,两条链路的价格开始均为 0,这是因为 S_3 的每条接入链路带宽只有 6Mbps,小于中间链路 10Mbps 的带宽.而随着 S_1 和 S_4 的启动,链路价格开始增加.链路价格作为一种调节源速率的手段,导致 S_3 速率下降.而 S_2 和 S_5 的启动导致 Link1 和 Link2 的价格进一步增加, S_1 , S_3 和 S_4 的速率也进一步下降,但最终都达到收敛。

- [4] J R Iyengar, P Amer, R Stewart. Concurrent multipath transfer using SCTP multihoming over independent end-to-end paths [J]. IEEE/ACM Transactions on Networking, 2006, 14(5): 951 – 964.
- [5] RFC6182. Architectural Guidelines for Multipath TCP Development [S]. 2011.
- [6] Trilogy consortium. Trilogy project [EB/OL]. <http://www.trilogy-project.org/>, 2009-11-28.
- [7] D Wischik, M Handley, M B Braun. The resource pooling principle [J]. ACM SIGCOMM Computer Communication Review, 2008, 38(5): 47 – 52.
- [8] 杨冬,李世勇,王博,张宏科.支持普适服务的新一代网络传输层架构 [J]. 计算机学报, 2009, 32(3): 359 – 370.
Yang Dong, Li Shi-yong, Wang Bo, Zhang Hong-ke. New transport layer architecture for pervasive service [J]. Chinese Journal of Computer, 2009, 32(3): 359 – 370. (in chinese).
- [9] 杨冬,周华春,张宏科.基于一体化网络的普适服务研究 [J]. 电子学报, 2007, 35(4): 607 – 613.
Yang Dong, Zhou Hua-chun, Zhang Hong-ke. Research on pervasive services based on universal network [J]. Acta Electronica Sinica, 2007, 35(4): 607 – 613. (in Chinese)
- [10] S Shakkottai, R Srikant. Network optimization and control [J]. Foundations and Trends in Networking, 2007, 2(3): 271 – 379.
- [11] N Bonmariage, G Leduc. A survey of optimal network congestion control for unicast and multicast transmission [J]. Computer Networks, 2006, 50(3): 448 – 468.
- [12] F Kelly, A Maulloo, D Tan. Rate control in communication networks: Shadow prices, proportional fairness and stability [J]. Journal of the Operational Research Society, 1998, 49

(3):237 – 252.

- [13] S H Low, D E Lapsley. Optimization flow control, I: Basic algorithm and convergence [J]. IEEE/ACM Transactions on Networking, 1999, 7(6): 861 – 874.
- [14] W H Wang, M Palaniswami, S H Low. Optimal flow control and routing in multi-path networks [J]. Performance Evaluation, 2003, 52(2): 119 – 132.
- [15] X J Lin, N B Shroff. Utility maximization for communication networks with multi-path routing [J]. IEEE/ACM Transactions on Automatic Control, 2006, 51(5): 766 – 781.
- [16] P Dharwadkar, H J Siegel, E P Chong. A heuristic for dynamic bandwidth allocation with preemption and degradation for prioritized requests [A]. International Conference on Distributed Computing Systems [C]. Washington DC, USA: IEEE Computer Society, 2001. 547 – 556.
- [17] D P Bertsekas, J N Tsitsiklis. Parallel and Distributed Computation [M]. Englewood Cliffs, NJ: Prentice-Hall, 1989.
- [18] 薛森, 高德云, 张思东, 张宏科. 面向下一代网络的端到端多路径传输层架构 [J]. 通信学报, 2010, (10): 26 – 35. Xue Miao, Gao De-yun, Zhang Si-dong, Zhang Hong-ke. End-to-end multipath transport layer architecture oriented the next generation network [J]. Journal on Communications, 2010, (10): 26 – 35. (in Chinese)
- [19] S H Low. A duality model of TCP and queue management algorithms [J]. IEEE/ACM Transactions on Networking, 2003, 11(4): 525 – 536.

作者简介



薛森 男, 1983 年生于河北, 北京交通大学博士生, 主要研究方向为下一代网络服务理论, 端到端传输协议和拥塞控制.

E-mail: xmkyller@gmail.com



张思东 男, 1945 年生于山东, 北京交通大学教授、博士生导师, 主要研究方向为下一代互联网与无线传感器网络路由、资源分配与管理、网络安全等.



董平 男, 1979 年出生于河北冀州, 现为北京交通大学讲师. 主要研究方向为 IP 网络的路由理论与组播技术、下一代网络理论.



张宏科 男, 1957 年生于山西, 北京交通大学教授, 博士生导师. 主要研究方向为下一代信息网络关键理论与技术, 下一代网络服务理论, 新一代移动互联网络路由, 协议理论与技术等.