

# 基于动态流量的多拓扑链路权值优化算法

陈铎龙 孟相如 梁 霄 温祥西

(空军工程大学信息与导航学院 西安 710077)

**摘要** 针对网络流量动态变化和不同流量对拥塞控制和传输代价的要求不同的现象,提出一种基于小生境粒子群的多拓扑路由子层链路权值优化算法。该算法以适应网络流量动态变化为目标,设置了以时段划分的业务量矩阵和适应流量动态变化要求的权重因子。优化目标函数从拥塞代价影响和传输代价影响两方面进行了改进,并利用小生境粒子群算法对目标函数进行寻优,以解决一般优化算法存在的容易陷入局部最优的问题。实验结果表明,算法能够在网络中实现负载均衡。

**关键词** 流量工程,链路权值,小生境粒子群,多拓扑路由

**中图分类号** TP393 **文献标识码** A

## Multi-topology Link Weight Optimizing Algorithm Based on Dynamic Traffic

CHEN Duo-long MENG Xiang-ru LIANG Xiao WEN Xiang-xi

(Information and Navigation Institute, Air Force Engineering University, Xi'an 710077, China)

**Abstract** Considering the characteristic that network traffic change is dynamic and different traffics demand for congestion control and transmission cost are different, a link weights optimizing algorithm for multi-topology routing sub-layer based on niche particle swarm optimizer (NPSO) was proposed. The traffic matrix differentiated by periods and weight gene following different traffic demands were set to adapting the dynamic changing of network traffic. The congestion and transmission cost were considered in optimizing objective function, and NPSO was used to solve the optimization problem. The experimental result shows that the new algorithm can balance the load in network.

**Keywords** Traffic engineering, Link weight, Niche particle swarm optimizer, Multi-topology routing

## 1 引言

随着学者对网络可生存性研究的深入,重路由技术在网络可生存性增强中的应用越来越多,而多拓扑路由(Multi-Topology Routing, MTR)因为具有收敛速度快、网络振荡小、可处理多故障等优点,正逐渐成为重路由技术研究的热点。

多拓扑路由,就是将原始拓扑抽象成若干子层,每个子层都从原始拓扑中删除一些链路或节点,这些被删除的链路或节点,就是该子层能够保护的网络设备。通过这种方法,使网络不仅能够有效地避开网络故障设备将流量传输至目的节点,还因为每个子层都不止保护了一条链路而具备了抵抗多故障的能力。

随着网络承载业务量的增大,多拓扑路由的广泛应用也同时引发了一些问题,传统的 OSPF 协议以及基于 OSPF 协议的重路由技术虽然能够在网络故障的情况下为流量寻找一条新的路径,但这种寻找最短路径的路由方法可能会导致拓扑中某些链路经常被使用而某些链路几乎不被使用。在链路容量有限的条件下,这种现象会造成某些链路的拥塞,网络性

能急剧下降,因此,基于权值优化的负载均衡技术被提出。

早在 2000 年,Fortz 等人<sup>[1-3]</sup>就提出通过为链路设置权重来解决一部分负载均衡问题,但其仅限于解决流量固定的网络拥塞问题;文献[4]还提出通过对网络链路的权值进行优化来减少网络因拥塞造成的故障,同样增强了网络的可生存性,但没有考虑到除链路利用率以外的因素,如链路代价对网络的影响;文献[5]提出了针对动态流量情形下流量的表示方法,然而以流量期望为基础的业务流量表示方式无法准确反映网络流量的真实情况;文献[6]对权值优化的目标函数进行了改进,并利用智能算法来解决寻优当中的 NP 难问题。虽然目前关于负载均衡问题的研究很多,但专门针对 MTR 负载均衡问题的研究还很有限,文献[7]根据多拓扑路由对不同业务区分服务,提出了一种改进权值优化的启发式算法,但对动态流量的表示与适应动态流量变化要求的研究还是不足。基于此,本文提出一种专门针对流量动态变化的 MTR 子层链路权值优化算法。本文第 1 节针对链路权值优化问题中的业务量矩阵与优化目标函数进行改进;第 2 节提出一种 MTR 子层链路权值优化算法;第 3 节对该算法进行实验和性能分

到稿日期:2012-06-29 返修日期:2012-10-15 本文受国家自然科学基金(61003252,61201209),全军军事学研究生课题(2011JY002-524)资助。

陈铎龙(1988-),男,硕士生,主要研究方向为网络可生存性增强,E-mail:cdl120220@126.com;孟相如(1963-),男,教授,博士生导师,主要研究方向为宽带通信网络技术;梁霄(1988-),女,硕士生,主要研究方向为网络可生存性评估;温祥西(1984-),男,博士生,主要研究方向为人工智能与网络健康管理。

析;最后给出结论和下一步的研究方向。

## 2 负载均衡技术

负载均衡技术可分为先应式和反应式,其区别是对网络链路的权值优化是否是在故障发生之前进行的<sup>[4]</sup>。链路权值的变化会导致网络路由重新进行计算,这必然会引起网络振荡,因此,在网络发生故障后进行权值优化的反应式负载均衡技术应用并不广泛。本文研究的多拓扑路由属于主动式,即先应式重路由技术,故本文所论述的权值优化也在故障发生之前进行,这样可以有效地避免因链路信息发生变化而造成的网络振荡。

用一个无向加权图  $G=(V, E)$  表示网络拓扑,其中  $V=\{v_1, v_2, \dots, v_n\}$  为拓扑中节点的集合,  $E=\{e_1, e_2, \dots, e_m\} \subseteq V \times V$  表示拓扑结构中的所有链路。  $n=|V|$  表示节点数量,  $m=|E|$  表示边数量。  $C=\{c_1, c_2, \dots, c_m\}$  为链路容量集合,其中的每个元素表示相应链路的链路容量。下面讨论业务量矩阵的表示形式。

### 2.1 业务量矩阵

业务量矩阵是描述各节点间的流量需求大小的矩阵,为了避免个别链路拥塞的现象,负载均衡技术通常采用变更链路权值的方法来改变路由路径,以实现全网络流量均衡分布在各个链路上,各节点间的流量需求大小直接决定了网络拓扑中各条链路所承载流量的大小。

文献[5]建立了期望值模型来反映节点间流量的大小,它在一定程度上能够反映节点间的动态流量,但通常情况下,网络节点间的流量是随时间变化的,在不同的时间段,节点间流量变化差别可能非常大,因此并不能完全依据期望作为业务量的表示形式。本文将网络所经历的一个流量变化规律的周期分为  $T$  个时段,采用如下方式表示业务量矩阵:

$$D = \{D_1, D_2, \dots, D_k, \dots, D_T\} \quad (1)$$

式中,  $D_k$  为一个  $n \times n$  的矩阵,表示第  $k$  时段中所有节点间的流量,用节点之间的流量  $S_{i,j}^k$  作为  $D_k$  中的元素。

现实中网络在一天之内的流量变化是存在规律的,所以这种采取从时间上分割的方法可以动态地表示网络中流量的变化。

### 2.2 优化目标函数

结合业务量矩阵的表示形式可以得出,链路  $a$  在第  $k$  时段所承载的流量可以表示为:

$$l^k(a) = \sum_{i,j \in V} \delta_{i,j}^k(k) S_{i,j}^k \quad (2)$$

式中,  $\delta_{i,j}^k(k)$  表示节点  $i, j$  之间的流量是否经过链路  $a$ ,若经过则其值为 1, 否则为 0。

链路利用率表示为:

$$u^k(a) = l^k(a) / c(a) \quad (3)$$

文献[1]以链路利用率为基础,提出了拥塞对网络造成的代价函数  $\phi^k(a)$ :

$$\phi^k(a) = \begin{cases} u^k(a), & 0 \leq u^k(a) < 1/3 \\ 3u^k(a) - 2/3, & 1/3 \leq u^k(a) < 2/3 \\ 10u^k(a) - 16/3, & 2/3 \leq u^k(a) < 9/10 \\ 70u^k(a) - 178/3, & 9/10 \leq u^k(a) < 1 \\ 500u^k(a) - 1468/3, & 1 \leq u^k(a) < 11/10 \\ 5000u^k(a) - 19468/3, & 11/10 \leq u^k(a) < \infty \end{cases} \quad (4)$$

从式(4)和图 1 可以看出,  $\phi^k(a)$  是一个分段、线性、递增的凹函数,反映了链路利用率的大小对网络的影响。当链路利用率逐渐增高时,造成的网络代价也在逐渐增加,增速也越来越大,当链路利用率接近 1 时,网络拥塞代价急剧增加,网络性能恶化。在进行优化时,需要一组权值  $\omega^k = [\omega_1^k, \omega_2^k, \dots, \omega_m^k]$ , 将  $\omega^k$  代入网络中,使得拥塞造成的代价最小,而该权值即为所求权值。拥塞代价函数定义为:

$$f_\phi^k = \sum_a \phi^k(a) / m, a \in E \quad (5)$$

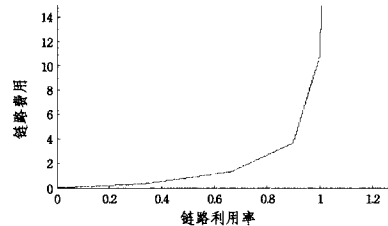


图 1 链路拥塞代价随利用率变化曲线图

在实际网络中,权值设置不仅能够影响网络的流量拥塞,还会影响网络传输的代价。因为权值设置的改变会改变网络的路由,而链路的物理长度不会因为权值设置的改变而发生变化,因此路由的改变必然会影响到源节点和目的节点之间的传输路径,进而影响流量的传输距离,即传输代价。传输代价函数定义为:

$$f_l^k = \max_a \sum_i \delta_{i,j}^k(k) l_e^k(a) / [\sum_a l_e^k(a) / m], a \in E, i, j \in V \quad (6)$$

式中,  $l_e^k(a)$  表示链路  $a$  的物理代价,即物理长度。式(6)表示的是权值为  $\omega^k = [\omega_1^k, \omega_2^k, \dots, \omega_m^k]$  时网络中所有路由路径中的最大代价。

综上所述,网络的负载均衡问题可以转化成如下问题:

$$\min f^k = \alpha f_\phi^k + (1 - \alpha) f_l^k \quad (7)$$

$$\text{s. t. } 0 \leq \Delta \omega^k \leq 65535 - L_e \quad (8)$$

$$L_e = [l_e^k(1), l_e^k(2), \dots, l_e^k(m)] \quad (9)$$

$$\Delta \omega^k = \omega^k - L_e \quad (10)$$

$$0 \leq \alpha \leq 1 \quad (11)$$

式(7)为优化目标函数,即要求寻找一组权值  $\omega^k$  使得拥塞代价与传输代价组成的目标函数最小化。  $\alpha$  为权重因子,表示环境要求网络更注重拥塞控制还是传输代价,当  $\alpha=0$  时,表示当前环境只要求网络传输代价最小,不考虑拥塞控制;当  $\alpha=1$  时,表示当前环境只重视拥塞控制而不考虑传输的代价。

式(8)~式(11)为式(7)的约束条件,其中式(8)表示链路权值调整的范围  $\Delta \omega^k$  应该为正且每条链路最终权值不应超过 65535。

### 2.3 动态优化函数

根据第 2.1 节的论述,真实网络的流量应该是动态的,所以网络链路权值优化应该根据时间的变化动态进行。式(1)建立了业务量矩阵按照时间划分的模型,因为现实中,有的时段全网络流量极少,而有的时段全网络流量急速增加;又结合第 2.2 节所述,当全网络流量较少时,权值优化过程中应该更加注重减小网络的传输代价,尽可能提高服务质量;当网络流量过大时,权值优化则应当重视流量的负载均衡,尽量避免网络的拥塞。由此可知,根据网络流量的不同为网络设置不同的链路权值是很有必要的,故提出动态优化函数:

$$F=[F_\varphi, F_\gamma][\vec{\alpha}, \vec{1}-\vec{\alpha}]^T \quad (12)$$

式中,  $F_\varphi=[f_\varphi^1, \dots, f_\varphi^k, \dots, f_\varphi^T]$  表示各个时段的拥塞代价函数,  $F_\gamma=[f_\gamma^1, \dots, f_\gamma^k, \dots, f_\gamma^T]$  表示各个时段的传输代价函数,  $\vec{\alpha}$  表示各个时段的权重因子向量。因此, 优化问题就成为:

$$\min F=[F_\varphi, F_\gamma][\vec{\alpha}, \vec{1}-\vec{\alpha}]^T \quad (13)$$

约束条件仍然为式(8)~式(11)。

综上所述, MTR 的子层链路权值优化实质就是在故障发生之前, 根据网络业务量矩阵提前对各个子层进行动态权值优化。下面本文将结合小生境粒子群算法, 提出一种 MTR 子层的权值优化算法。

### 3 MTR 子层权值优化

MTR 为可能发生故障的链路都提供了相应的保护子层, 在该子层中流量可以不用经过故障链路仍然被传输至网络中的所有节点。但是, 相比于原始拓扑, 子层拓扑中的部分链路被保护, 即被移除, 可用链路数量少于原始拓扑, 因此更容易产生拥塞, 所以对 MTR 子层进行链路权值优化以实现子层负载均衡是很有必要的。

对 MTR 进行负载均衡, 就是对其各个子层进行权值优化。有学者已经证明链路权值优化是一个 NP 难问题<sup>[1]</sup>, 且一般的智能算法寻优容易陷入局部最优, 所以本文提出利用小生境粒子群算法来解决这一问题。

#### 3.1 小生境粒子群算法

小生境是来自生物学的一个概念, 指特定环境下的一种生态环境, 若干物种在一定区域内形成小生境, 小生境之间的基因由于自然隔离缺乏交流, 使得物种之间的差异得以保留。将这种概念运用到优化上来说, 就是将每一代的个体划分为若干个类, 依据某种规则从每个类中选择适应度较大的个体作为类的代表, 在类内和类间进行杂交、变异产生新一代个体。Brits 等人将小生境这一概念引入到粒子群算法中, 提出了小生境粒子群算法 (Niche Particle Swarm Optimizer, NPSO)<sup>[8]</sup>, 有效地解决了传统粒子群算法易陷入局部最优的问题。

小生境粒子群优化算法的简要步骤如下:

(1) 设置算法参数, 初始化主粒子群。

(2) 使用一般 PSO 对主粒子群进行一次搜索, 选出若干个适应值变化较小的粒子, 以这些粒子为中心, 距此粒子最近的粒子之间距离为半径构造圆形小生境。

(3) 对每个子粒子群有如下操作:

a. 使用一般收敛 PSO 更新粒子:

$$v_i^{k+1} = -x_{gbest}^k + x_{pbest,i}^k + \omega v_{gbest}^k + \eta^k (1-2r_2^k) \quad (14)$$

$$x_i^{k+1} = x_{pbest,i}^k + \omega v_{gbest}^k + \eta^k (1-2r_2^k) \quad (15)$$

式中,  $\eta$  为搜索范围最大值;

b. 计算每个子粒子群的适应值;

c. 更新子粒子群小生境半径。

(4) 根据各个子粒子群的范围与各个粒子的位置决定子粒子群是否合并或粒子是否被某子粒子群吸收。

(5) 重复步骤(3), 直到达到最大迭代次数或结果满足要求, 算法结束。

#### 3.2 子层权值优化算法

现有网络拓扑  $G$ , 其中  $L$  为网络  $G$  的一个子层,  $D$  为经过

统计得出的网络各个时段的流量期望, 即业务量矩阵,  $\vec{\alpha}$  为权重因子向量, 且所有元素取值均在  $[0, 1]$  之内,  $L_c$  为所有链路的物理代价, 则对该子层进行权值优化的实质是寻找一组权值调整向量  $\Delta\omega$ , 使得将权值  $\omega = \Delta\omega + L_c$  代入网络后, 满足目标函数式(13)。

结合到小生境粒子群算法中, 将  $\Delta\omega$  中的每个元素即每条链路应该调整的权值看作粒子的坐标, 各个粒子的适应度函数即为式(7)中所表示的优化目标函数。且存在如下约束:

$$\Delta\omega_i^k(t+1) = \begin{cases} 65535 - l_i^k(t), & \text{if } \Delta\omega_i^k(t+1) > 65535 - l_i^k(t) \\ 0, & \text{if } \Delta\omega_i^k(t+1) < 0 \end{cases} \quad (16)$$

优化目标函数为式(13)所示的动态优化函数。

子层权值优化算法的具体步骤如下:

(1) 依次为算法所划分的所有时段初始化一组规模为  $N=20$  的种群, 随机生成各个粒子的初始速度与位置, 并评价初始粒子的个体最优位置  $\Delta\omega_{pbest}^k(t)$  和全局最优位置  $\Delta\omega_{gbest}^k(t)$ , 其中  $t$  为迭代次数;

(2) 对主粒子群进行一次搜索, 选出适应度变化较小的 10 个粒子, 以这 10 个粒子为中心构造小生境;

(3) 根据式(15)、式(16)更新所有粒子的速度、位置;

(4) 根据各个小生境的半径范围进行吸收粒子或合并小生境的操作, 同时更新粒子群的个体最优位置和全局最优位置;

(5) 判断是否达到最大迭代次数或得到最优解, 若是, 则跳至步骤(6), 否则重复步骤(3);

(6) 得出的全局最优位置  $\Delta\omega_{gbest}^k(t)$  即为各个链路应该调整的权值大小, 将该权值调整保存作为该子层在  $k$  时段的权值调整方案;

(7) 重复步骤(1), 直至所有时段都获得各自合适的权值调整方案, 保存这些方案至  $\Delta\omega$  作为子层  $L$  的权值调整方案, 算法结束。

至此, 算法完成了对子层  $L$  的链路权值优化, 重复此算法可完成 MTR 中所有子层的链路权值优化。由于该方法为先应式负载均衡技术, 即在故障发生前就完成了对所有链路的权值优化, 且 MTR 本身就是先应式的重路由技术, 因此当故障发生后, MTR 只须将流量切换至可以保护故障链路的子层, 并且使用根据目前时段的权值调整方案产生的路由, 就可以既完成流量的保护, 也尽可能地减小网络拥塞的可能。

下面结合具体实例对该权值优化算法的性能进行分析。

## 4 实验分析

实验中, 粒子群算法的运行参数设置如下: 为使算法能够尽快寻到最优解, 惯性权重应较小, 故令  $\omega = 0.729$ ; 且算法搜索范围不得超出链路权值的最大约束, 故  $\eta = 65535 - l_i^k$ , 每次算法迭代次数设置为 200。

### 4.1 简单网络拓扑分析

实验原始拓扑如图 2(a) 所示, 该拓扑包含 8 个节点、14 条链路, 图 2(b)~(d) 分别为原始拓扑的 3 个子层, 分别保护被移除的链路, 所有链路容量均为 5Tbit/s。

实验中采用 dijkstra 算法寻找最短路径, 将网络工作时段分为 6 时到 12 时、12 时到 18 时、18 时到 0 时、0 时到 6 时

4个时段,表示各个时段的权重因子向量为 $\vec{\alpha}^T = \{0.3, 0.5, 0.8, 0.1\}$ 。在6时到12时,网络中的流量较少,因此网络较重视减少传输代价,故权重因子较低;在12时到18时,网络流量逐渐增加,拥塞问题逐渐凸显,权重因子适当调高;在18时到0时,网络流量达到峰值,拥塞控制显得最为重要,权重因子最高;在0时到6时,网络中流量骤减,网络此时更注重传输效率,故权重因子最低,这样设置是符合实际一天当中网络的运行情况的。实验分别统计未采用MTR子层权值优化算法和采用该多拓扑算法后,各个时段两种情况的代价函数数值。

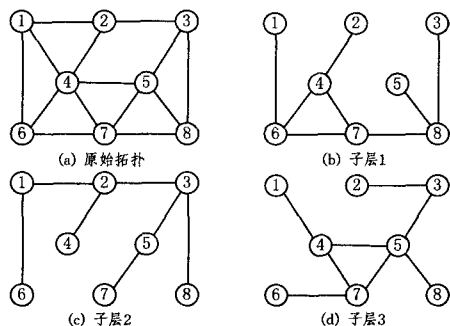


图2 小规模网络拓扑

图3为图2(d)所示子层3的优化效果示意图,从图中可以看出,子层3由于保护了部分链路,拓扑结构远比原始拓扑结构稀疏,因此链路承载的流量与传输代价都要大于原始拓扑,这是符合实际情况的;又当网络工作在18时到0时这个时段时,网络流量极大,网络拥塞代价极高,而使用子层链路权值优化算法可以适当地减少代价,实现了流量的负载均衡。

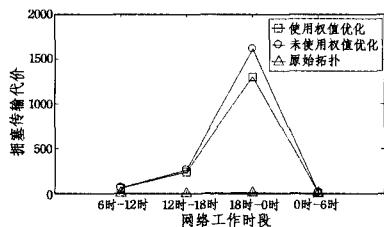


图3 小规模网络优化效果示意图

虽然该实验反映出网络负载均衡的性能有所改善,但由于网络拓扑结构较为稀疏,对网络中传输代价和链路权值的改善对网络流量控制造成的影响反映不够明显,因此接下来使用德国电信网络拓扑<sup>[9]</sup>对该算法进行验证。

#### 4.2 链路优化效果

图4(a)所示为德国电信网络的原始拓扑,图4(b)~图4(d)分别为其MTR子层,设每条链路容量为5Tbit/s,网络工作时段也分为0时到6时、6时到12时、12时到18时、18时到0,权重因子向量设置与第4.1节设置相同。实验统计图4(b)所示为子层链路权值优化前后以及使用文献[4]中优化方法时各个链路的链路利用率与传输代价,以观察算法对链路负载的改善情况。

图5所示为子层4个不同时段各个链路的利用率,图6所示为各个链路在不同时段的链路利用率方差。从图中可以看出,根据各个时段业务量矩阵不同与网络对拥塞代价和传输代价重视程度不同的特点,分时段对子层进行链路权值优化之后各条链路的利用率比优化前更加均衡,除个别负载较

大的链路作为割边无法进行优化外,负载较高的链路负载均得到了分担,一些利用率较低的链路也参与均衡了一部分流量,提高了资源利用率。由此可见,该链路权值优化算法在实现网络负载均衡方面是有效的。

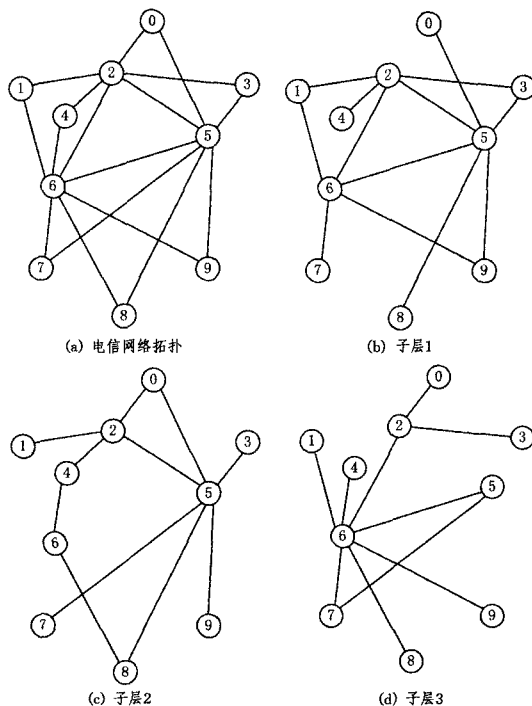


图4 德国电信网络拓扑结构及子层拓扑结构

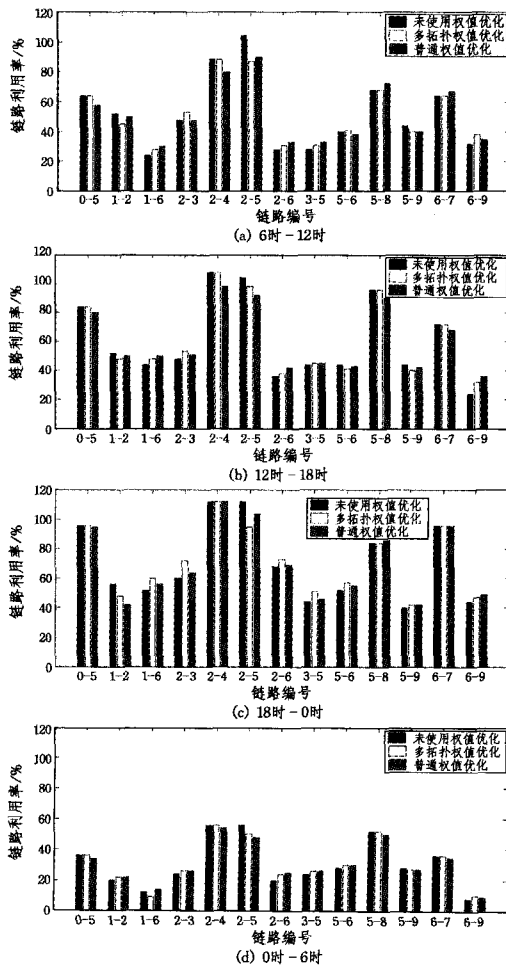


图5 子层不同时段链路利用率

文献[4]中所提方法由于只重视链路的拥塞,且没有依据时间与拓扑的动态变化,因此如图6、图7所示,当网络流量较少时,普通优化由于只重视拥塞控制,因此在负载均衡上的性能优于本算法,但当网络流量逐渐增加时,本算法也逐渐重视拥塞,双方负载均衡能力相当,且本算法的传输代价一直小于普通算法。因此本算法在处理多拓扑动态多目标负载均衡上有一定优势。

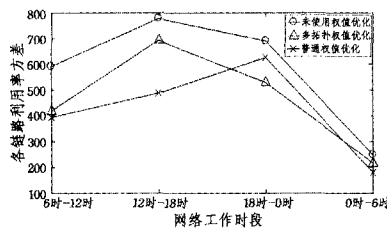


图6 链路利用率方差

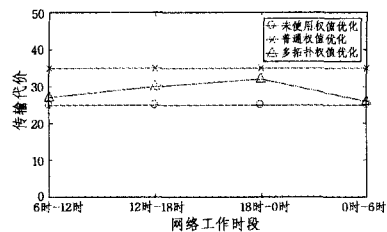


图7 各时段传输代价

**结束语** 本文根据网络流量动态变化的特点,提出一种MTR子层链路权值优化算法。该算法根据时段不同设置相应的业务量矩阵,且因为网络流量不同,对拥塞和传输代价的偏重程度也不同,因此该算法为不同的时段设置了不同的拥塞、传输代价权重因子,使得网络MTR子层可以在不同时段拥有一套适应该时段业务需求的链路权值。为了避免一般优化算法容易出现陷入局部最优的问题,该算法使用小生境粒

子群算法进行寻优,以保证搜索的范围能够覆盖全局。实验证明,该算法能够实现子层中流量的负载均衡。在后续研究中,将对拥塞造成的代价函数进行优化,使其更加适应网络环境。

## 参考文献

- [1] Fortz B, Thorup M. Internet traffic engineering by optimizing OSPF weights[C]// Proceedings of IEEE infocom 2000. Te-Aviv, Israel, 2000; 519-528
- [2] Fortz B, Thorup M. Optimizing OSPF/IS-IS weights in a changing world[J]. IEEE Journal on Selected Areas in Communications, 2005, 20(4): 756-767
- [3] Karthik L, Matthew C, Murali R, et al. Achieving convergence-free routing using failure-carrying packets[J]. Proceedings of ACM SIGCOMM, 2007, 37(4): 241-252
- [4] 于涛, 陈山枝, 李昕, 等. 一种通过优化链路权值来增强网络生存性的方案[J]. 高技术通讯, 2008, 18(7): 661-665
- [5] 吴静, 郭成城, 晏蒲柳, 等. 基于动态流量矩阵的网络链路权值调整方法[J]. 华中科技大学学报, 2007, 35(7): 76-79
- [6] 袁荣坤, 孟相如, 李明迅, 等. 基于粒子群权值优化的网络可生存性增强方法[J]. 计算机应用, 2012, 32(1): 127-130
- [7] 黄赫, 王晟. 多拓扑路由实现IP网络区分服务的优化算法[J]. 计算机应用研究, 2010, 27(12): 4735-4737
- [8] Brits R, Engelbricht P, Bergh F D. A niching particle swarm optimizer[C]// Proceedings Conf. on simulated evolution and learning. Singapore, 2002; 1037-1040
- [9] Scheffel M C, Gruber C G, Schwabe T, et al. Optimal multi-topology routing for IP resilience[J]. International journal of electronics and communications, 2006(60): 35-39
- [10] 徐明伟, 杨莹, 李琦. 域内自愈路由研究综述[J]. 电子学报, 2009, 37(12): 2753-2761

(上接第68页)

- [2] Zhang Z, Lian Q, Lin S, et al. BitVault: A highly reliable distributed data retention platform[J]. ACM SIGOPS Operating Systems Review, 2007, 41(2): 27-36
- [3] Kubiatowicz J, Bindel D, Chen Y, et al. OceanStore: An Architecture for Global-Scale Persistent Storage[J]. ACM SIGPLAN Notices, ACM, 2000, 35(11): 190-201
- [4] Costa P, Picco G. Publish-subscribe on sensor networks: a semi-probabilistic approach[C]// Mobile Adhoc and Sensor System Conference. Washington, DC; Costa P, 2005; 323-332
- [5] Carzaniga A, Hall C P. Content-based communication: a research agenda[C]// Proceedings of the 6th international workshop on Software engineering and middleware. 2006; 2-8
- [6] Baldoni R, Virgillito A. Distributed Event Routing in Publish/Subscribe Communication Systems: a Survey[J]. Communications Surveys & Tutorials, 2010, 12(1): 39-58
- [7] Barham P, Dragovic B, Fraser K, et al. Xen and the art of virtualization[C]// Proceedings of the nineteenth ACM symposium on Operating systems principles. 2003; 164-177
- [8] Kivity A, Kamay Y, Laor D, et al. kvm: the Linux virtual machine monitor[C]// Proceedings of the Linux Symposium. 2007; 225-230
- [9] Moreno-Vozmediano R, Montero R, Llorente I. IaaS Cloud Architecture: From Virtualized Data Centers to Federated Cloud Infrastructures[J]. Computer, 2012(99): 1
- [10] Bhardwaj S, Jain L. Cloud computing: A study of infrastructure as a service (IAAS)[J]. International Journal of engineering and Information Technology, 2010, 2(1): 60-63
- [11] Knuth D. The art of computer programming [M]. Volume 3, 1973
- [12] Bentley J L, Sedgewick R. Fast algorithms for sorting and searching strings[C]// Proceedings of the eighth annual ACM-SIAM symposium on Discrete algorithms. 1997; 360-369
- [13] Jerzak Z, Fetzter C. Bloom filter based routing for content-based publish/subscribe [C]// Proceedings of the second international conference on Distributed event-based systems. 2008; 71-81