

基于 Openflow 网络的高可靠性虚拟网络映射算法

蔡进科^{*①} 顾华玺^① 卢冀^② 余晓杉^①

^①(西安电子科技大学 ISN 国家重点实验室 西安 710071)

^②(通信网信息传输与分发技术重点实验室 石家庄 050081)

摘要: 该文基于 Openflow 网络提出了具有容错能力的虚拟网络映射模型, 并且采用蚁群算法对其进行求解。针对虚拟网络的故障恢复机制, 提出了区分用户优先级的故障恢复算法(Priority_Diff), 该算法为用户提供不同的网络可靠性级别, 对高级用户采用提前映射的备份路径替代故障链路, 对低级用户重新映射故障链路; 设计了故障备份链路重映射(BLRM)算法, 将故障链路中的备份资源迁移到相邻链路, 增强了备份链路的可用性。最后, 通过仿真实验, 从虚拟网络故障修复率、虚拟网络成功运行率和工作链路资源利用率 3 个方面验证了所提算法的优越性。

关键词: 虚拟网络; Openflow; 映射算法; 可靠性

中图分类号: TP393

文献标识码: A

文章编号: 1009-5896(2014)02-0396-07

DOI: 10.3724/SP.J.1146.2013.00367

Highly Reliable Virtual Network Mapping Algorithm Based on Openflow Network

Cai Jin-ke^① Gu Hua-xi^① Lu Ji^② Yu Xiao-shan^①

^①(State Key Laboratory of Integrated Service Networks, Xidian University, Xi'an 710071, China)

^②(Science and Technol. on Information Transmission and Dissemination in Communication Networks Lab., Shijiazhuang 050081, China)

Abstract: A fault tolerant virtual network mapping model based on Openflow network is proposed, and it is solved by the ant colony algorithm. In view of the virtual network fault recovery mechanism, a distinction user priority failure recovery algorithm named Priority_Diff is proposed, and the algorithm provides users different network reliability levels. The failed link is replaced by a backup path for advanced users, and remapped for low-level users. In addition, a failed Backup Link ReMapping (BLRM) algorithm is proposed, and the backup resources in the failed link are migrated to the adjacent link, which improves the availability of the backup link. Finally, the performance parameters, including virtual network failure repairing ratio, virtual network success running ratio, and working link resource utilization are evaluated by simulation experiments, and the results demonstrate the superiority of the proposed algorithms.

Key words: Virtual network; Openflow; Mapping algorithm; Reliability

1 引言

互联网在过去的几十年里飞速发展, 随着用户业务越来越丰富, 传统网络的“傻瓜式网络+智能终端”的构网方式无法满足不同用户业务的多样化需求, 与三网融合的趋势相违背。网络虚拟化技术^[1]的出现为打破传统的网络架构提供了一条有效的途径, 被认为是下一代互联网架构的发展趋势^[2,3]。源

自于斯坦福大学“Clean slate”研究计划的 Openflow 网络将网络的控制层和转发层分离, 是未来虚拟化网络的雏形^[4]。

Openflow 网络是由多个网络域相互连接构成, 每个网络域包括一个控制器, 一个或多个 Openflow 交换机以及若干主机。控制器承载的是该域内的控制资源, 主要负责运行不同的路由协议, 并且计算出相应的路由流表。Openflow 交换机承载的是转发资源, 通过运行在安全通道上的 Openflow 协议与控制器通信, 根据控制器产生的流表来转发数据。

Openflow 网络的主要思想是利用网络虚拟化技术将物理网络划分为多个虚拟网络, 这些虚拟网络都共享同样的底层物理网络资源, 各个虚拟网络

2013-03-22 收到, 2013-08-12 改回

国家自然科学基金(60803038, 61070046), 国家重点实验室专项基金(ISN1104001), 中央高校基本业务费项目(K5051301003), 高等学校学科创新引智计划(B08038)和通信网信息传输与分发技术重点实验室(ITD-U12002)资助课题

*通信作者: 蔡进科 bestcaicai@126.com

可以运行不同的路由协议, 提供不同的服务。虚拟网络的构建请求包括虚拟节点请求和虚拟链路请求两部分, 为虚拟网络选择合理的映射算法不但关系到物理网络的资源利用率, 而且影响虚拟网络的性能。文献[5]将虚拟网络映射问题抽象为混合整数规划问题, 针对不同的应用环境分别提出了 D-ViNE 和 R-ViNE 两种算法。文献[6]基于负载平衡路由和小区划分结构的思想设计了一种具有优秀时延和吞吐量性能的虚拟网映射算法。文献[7]在 K 短路径的基础上改进了链路映射的过程, 并且提出了一种物理节点可重复映射的虚拟网络映射算法。但是在真实环境下, 物理网络经常由于地震、泥石流等自然原因或者恶意攻击等非自然原因发生故障, 影响用户业务的正常运行。而上述的这些算法都是针对物理网络的资源利用率最大, 没有考虑因为网络故障而导致的虚拟网络可靠性问题。

针对虚拟网络的可靠性问题, 文献[8]提出了一种虚拟节点和虚拟链路的重映射机制, 不但可以提高虚拟网络的接收率而且有利于网络的负载均衡。文献[9]通过预先构建备份路径, 将故障的虚拟链路迁移到备份路径。文献[10]通过构建统一的备份资源池, 为虚拟链路动态地分配备份资源, 提高了物理资源的利用率。上述这些算法分别采用不同的机制提高虚拟网络的可靠性, 但是它们或者故障发生时才进行链路重映射, 故障修复率低, 或者提前预留备份资源, 成本太高, 或者不能应对连续的多个网络故障。

本文针对 Openflow 网络提出了具有一定容错能力的虚拟网络映射模型, 并且采用蚁群算法对其进行求解, 同时针对已成功映射的虚拟网络设计了故障恢复机制。考虑到不同用户的网络可靠性需求不同, 银行、金融和军事机构等用户要求自己的业务 24 小时正常运行; 而像学校、社交论坛等大部分的普通用户对网络的可靠性需求相对较弱。通过为不同用户提供不同级别的网络可靠性保证, 提出了区分用户优先级的故障恢复算法(Priority_Diff), 即对受到故障影响的高优先级用户的流量迁移到预先设定的备份路径, 而对低优先级用户的故障链路进行重新映射。该算法可以有效兼顾物理网络资源利用率和网络的容错抗毁特性。同时, 由于物理链路上同时承载着工作资源和备份资源, 备份资源故障很可能导致后续的故障链路修复失败, 为了增强备份链路的可用性, 本文提出了故障备份链路重映射(BLRM)算法。最后通过仿真实验从虚拟网络故障修复率、虚拟网络成功运行率、工作链路资源利用率方面验证了所提算法的优越性。

2 基于 Openflow 网络的容错映射模型

2.1 Openflow 网络描述

Openflow 网络可以用一个有权无向图 $G_s = (N_s, E_s)$ 来表示, 其中 N_s 和 E_s 分别表示物理网络中节点和链路的集合。每个物理节点 $n_s \in N_s$ 的转发资源用 $N^{\text{fwd}}(n_s)$ 来表示, 控制资源用节点所在域 $g, n_s \in g$ 内的控制器资源 $N^{\text{ctrl}}(g)$ 来表示; 每条物理链路 $l_s \in E_s$ 的带宽资源用 $B(l_s)$ 来表示。

类似地, 虚拟网络可以用 $G_v = (N_v, E_v)$ 来表示, 其中 N_v 和 E_v 分别表示虚拟链路和虚拟节点的集合。每个虚拟节点 $n_v \in N_v$ 所需要的转发资源用 $R^{\text{fwd}}(n_v)$ 来表示, 所需要的控制资源用 $R^{\text{ctrl}}(n_v)$ 来表示; 每条虚拟链路 $l_v \in E_v$ 所需要的带宽资源用 $R(l_v)$ 来表示。

2.2 映射描述

如图 1 所示, 图 1(a)所示为两个虚拟网络请求, 方框内为该节点所需要的控制资源和转发资源, 链路上的数字为所需要的带宽; 图 1(b)所示为 Openflow 网络模型, 节点旁的方框内为该节点的转发资源, 控制器旁的方框内为该域内的控制资源。

虚拟网络向物理网络的映射可以描述为 $M: G_v \mapsto G'_s = (N'_s, E'_s)$, 其中 $N'_s \subseteq N_s, E'_s \subseteq E_s$ 。映射过程一般分为两个阶段:

(1)节点映射: 在 Openflow 网络中, 不仅物理节点要能够满足所承载的虚拟节点的转发资源需求, 而且该物理节点所在域也必须能够满足虚拟节点的控制资源需求。除此之外, 采用的是最经典的映射方式, 即虚拟节点和物理节点是一一对一的映射^[11]。满足的约束条件如式(1)所示。

$$\left. \begin{aligned} N_r^{\text{fwd}}(n_s^j) &\geq R^{\text{fwd}}(n_v^i), & o_{ij} &= 1 \\ N_r^{\text{ctrl}}(g_k) &\geq R^{\text{ctrl}}(n_v^i), & o_{ij} &= 1, n_s^j \in g_k \\ \sum_{n_s^j \in N'_s} o_{ij} &= 1, & \forall n_v^i \in N_v \\ \sum_{n_v^i \in N_v} o_{ij} &\leq 1, & \forall n_s^j \in N'_s \end{aligned} \right\} \quad (1)$$

其中 $o_{ij} = 1$ 表示虚拟节点 n_v^i 成功映射在物理节点 n_s^j 上, $N_r^{\text{fwd}}(n_s^j)$ 表示物理节点 n_s^j 的剩余转发资源, $N_r^{\text{ctrl}}(g_k)$ 表示物理节点 n_s^j 所在域 g_k 内的剩余控制资源, 如式(2)所示。

$$\left. \begin{aligned} N_r^{\text{fwd}}(n_s^j) &= N^{\text{fwd}}(n_s^j) - \sum_{o_{ij}=1} R^{\text{fwd}}(n_v^i) \\ N_r^{\text{ctrl}}(g_k) &= N^{\text{ctrl}}(g_k) - \sum_{\substack{n_s^j \in g_k \\ o_{ij}=1}} R^{\text{ctrl}}(n_v^i) \end{aligned} \right\} \quad (2)$$

(2)链路映射: 以映射成功的物理节点为端点, 利用最短路径算法, 将每条虚拟链路都映射到一条物

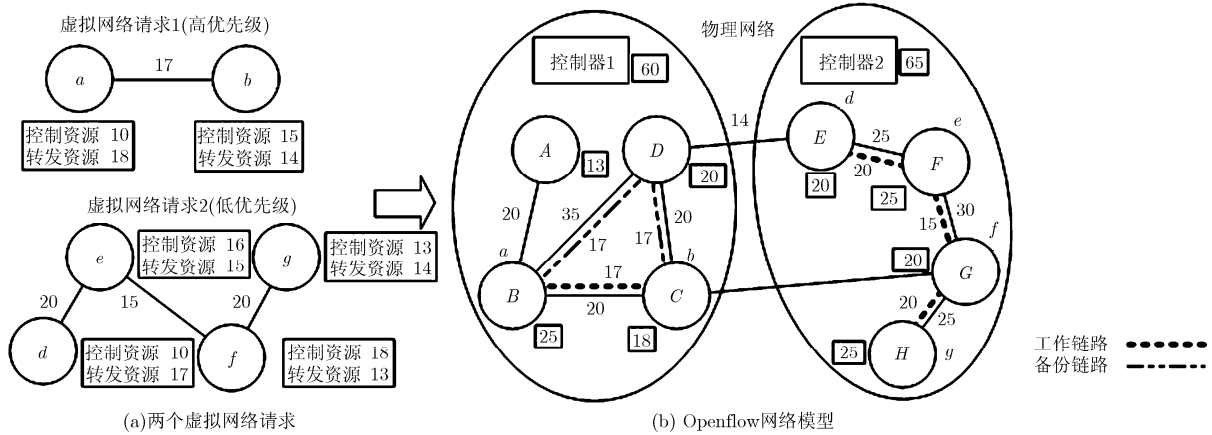


图 1 Openflow 网络映射实例

理路径上, 该物理路径上的每一段物理链路都必须能够满足所承载虚拟链路的带宽需求, 约束条件如式(3)所示。

$$B_r(l_s^m) \geq R(l_v^i), \quad \forall l_s^m \in \text{Ph}_{S,D}, \text{Ph}_{S,D} = M(l_v^i) \quad (3)$$

其中 $\text{Ph}_{S,D}$ 表示承载虚拟链路的一条物理路径, 其端点为 S 和 D 。 $B_r(l_s^m)$ 表示物理链路 l_s^m 的剩余带宽资源, 如式(4)所示。

$$B_r(l_s^m) = B(l_s^m) - \sum_{l_v^i \in M(l_s^m)} R(l_v^i) \quad (4)$$

对于高优先级用户的虚拟网络请求, 在映射工作路径的同时, 进行备份路径的构建。在寻找备份路径时, 不仅要满足链路的带宽需求, 而且还要保证同一条虚拟链路对应的工作路径和备份路径没有重合, 设 $\text{Ph}'_{S,D}$ 表示工作路径 $\text{Ph}_{S,D}$ 的备份路径, 约束条件如式(5)所示。

$$\begin{cases} B_r(l_s^m) \geq R(l_v^i), \quad \forall l_s^m \in \text{Ph}'_{S,D}, \text{Ph}'_{S,D} = M(l_v^i) \\ \text{Ph}'_{S,D} \cap \text{Ph}_{S,D} = \text{NULL} \end{cases} \quad (5)$$

根据上述映射规则, 在图 1 所示的实例中, 高优先级用户的虚拟网络请求 1 的两个虚拟节点 a, b 分别映射到物理节点 B, C 上, 虚拟链路 (a, b) 对应地映射到物理链路 (B, C) 上, 同时创建备份路径 (B, D, C) ; 低优先级用户的虚拟网络请求 2 的 4 个虚拟节点 d, e, f, g 分别映射到物理节点 E, F, G, H 上, 虚拟链路 $(d, e), (e, f), (f, g)$ 对应地映射到物理链路 $(E, F), (F, G), (G, H)$ 上。

2.3 映射目标

为了提高物理网络的资源使用效率, 即利用尽量少的物理资源承载尽量多的虚拟网络, 本文引入底层物理网络资源剩余价值 $S(G_s)$, 如式(6)所示。

$$S(G_s) = \alpha \left[\sum_{n_s^j \in N_s} N_r^{\text{fwd}}(n_s^j) + \sum_{g_k \subseteq G_s} N_r^{\text{ctrl}}(g_k) \right] + \beta \sum_{l_s^m \in E_s} B_r(l_s^m) \quad (6)$$

其中 α, β 分别表示节点资源和链路资源的价值转换权重。 $S(G_s)$ 越大表示网络中可用的物理资源越多, 可承载的虚拟网络就越多。因此, 将优化目标设置为 $\max \{S(G_s)\}$ 。

3 虚拟网络求解算法

虚拟网络映射问题由于数学模型复杂、约束条件多是公认的 NP-hard 问题[12], 求解算法需要满足以下条件: 很强的收敛性, 保证算法能够在有限次迭代之后找到优化目标的近似解; 低算法复杂度, 保证算法在较短的时间内运行完成; 高鲁棒性, 保证求得的解为全局最优解, 避免陷入局部最优解问题。

本文首次采用了蚁群算法^[13]来求解在 Openflow 网络模型下的虚拟网络映射问题。蚁群算法是通过模拟现实世界中蚂蚁觅食的寻路过程而提出的一种智能算法, 它的基本原理是: 蚂蚁在寻找食物的过程中会在经过的路径留下信息素作为标记。信息素具有挥发性, 由于蚂蚁在离食物较短的路径来回的频率更快, 因此在该路径留下的信息素更多, 后续蚂蚁选择该条路径的概率就越大, 通过这种正反馈蚂蚁可以找到从蚁巢到食物的最短路径。利用蚁群算法求解虚拟网络映射问题的步骤如下:

```

00 Initialization(); //对物理网络和仿真参数进行初始化
01 VN_Creation(); //按照泊松分布产生虚拟网络构建请求
02 For  $i=1$  to  $N$  //迭代  $N$  次
03 { Update_probability(); //根据信息素更新虚拟节点对物理节点的映射概率
04 For ant= $1$  to  $M$  //每次迭代产生  $M$  只蚂蚁
05 { Node_Map(); //根据映射概率进行节点

```

映射

```

06 Link_Map(); //根据最短路算法进行链
路映射
07 if(S(local_solution)>S(global_solution))
//比较当前解和最优解的资源剩余量
08 global_solution=local_solution; //将剩
余资源量最大的解作为最优解保存
09 }
10 Update_info(); //更新虚拟节点对物理节
点的信息素
11 }

```

根据上述算法,在每一次迭代过程中,会根据设定的优化目标 $S(G_s)$ 选择底层物理网络资源剩余价值最大的可行解作为本次循环最优解,然后根据该次最优解更新虚拟节点对物理节点的信息素。经过 N 次迭代,初始化产生的 M 个可行解最终可以收敛到全局近似最优解。

虚拟节点 n_v^i 对物理节点 n_s^j 的信息素更新如式(7)所示。

$$\tau_{ij}(t) = \rho \cdot \tau_{ij}(t-1) + \Delta \quad (7)$$

其中 ρ 表示信息素的持久程度, Δ 表示信息素的增量,如式(8)所示。

$$\Delta = \begin{cases} \frac{1}{Z(p^{bsf})} \cdot \sigma, & (i, j) \in p^{bsf} \\ 0, & \text{其它} \end{cases} \quad (8)$$

其中 p^{bsf} 表示到本次循环为止的最优解, $Z(p^{bsf})$ 表示最优解对应的目标函数值, σ 表示最优解对信息素的影响因子。

虚拟节点 n_v^i 对物理节点 n_s^j 的映射概率按照公式 $p_{ij} = \frac{\tau_{ij}}{\sum_{k \in N_s} \tau_{ik}}$ 更新,若信息素越大则映射概率越大。

4 虚拟网络故障恢复机制

在骨干网中单链路故障占全部故障的70%^[14,15],是最易发生和最常见的问题。为了提高Openflow网络的可靠性,本文针对单链路故障问题,提出了区分用户优先级的故障恢复算法,以及增强备份资源有效性的故障备份链路重映射算法,构建了一套有效的虚拟网络故障恢复机制。

4.1 区分用户优先级的故障恢复算法

高优先级用户的虚拟网络由于在映射时就创建了备份路径,当工作链路发生故障时可以迅速将流量迁移到备份路径;低优先级用户的虚拟网络由于在映射时没有构建备份资源,当网络发生故障时,可以通过重新映射故障链路来修复虚拟网络。设网

络中发生故障的物理链路为 $l_{s,D}$,其中 S 和 D 表示该链路的两个端点, $W(l_{s,D})$ 表示该链路所承载的任一虚拟网络的链路带宽,算法的具体描述如下:

输入: $l_{s,D}, W(l_{s,D})$

输出: $P_{s,D}$

- (1)判断故障链路上所承载虚拟网络的优先级;
- (2)若为高优先级用户,判断相应备份路径 $P_{s,D}$ 的可用情况;若为低优先级用户,跳转到(4);
- (3)若备份路径 $P_{s,D}$ 不可用,置 $P_{s,D}$ 为NULL,跳转到(7);若备份路径 $P_{s,D}$ 可用,跳转到(7);
- (4)去除物理网络 G_s 中的故障链路 $l_{s,D}$;
- (5)利用最短路算法寻找满足链路带宽约束条件(9)的路径 $P_{s,D}$,该路径的两个端点与故障链路的端点相同。若找不到这样的路径,则置 $P_{s,D}$ 为NULL,跳转到(7);

$$B_r(l_s^m) \geq W(l_{s,D}), \quad \forall l_s^m \in P_{s,D} \quad (9)$$

- (6)更新路径 $P_{s,D}$ 上各链路的剩余带宽资源如式(10)所示;

$$B_r(l_s^m) = B_r(l_s^m) - W(l_{s,D}), \quad \forall l_s^m \in P_{s,D} \quad (10)$$

- (7)若 $P_{s,D}$ 为NULL,虚拟网络恢复失败,否则返回恢复路径 $P_{s,D}$ 。

对受该故障影响的所有虚拟网络逐一运行该算法,故障修复完成。

4.2 故障备份链路重映射算法

物理链路故障除了影响工作链路,还会造成已映射成功的备份链路不可用,而该算法可以为故障的备份链路寻找替代路径。设网络中的故障链路为 $l_{s,D}$,其中 S 和 D 表示该链路的两个端点, $W_b(l_{s,D})$ 表示该链路上备份资源所占的带宽, G_l 表示以该链路为备份链路的工作链路的集合,算法的具体描述如下:

输入: $l_{s,D}, W_b(l_{s,D}), G_l$

输出: $P_{s,D}$

- (1)去除物理网络 G_s 中的故障链路 $l_{s,D}$;
- (2)利用最短路算法寻找满足式(11)的替代备份路径 $P_{s,D}$,该替代路径的两个端点与故障链路的端点相同,若找不到这样的路径,则置 $P_{s,D}$ 为NULL,跳转到(4);

$$\left. \begin{aligned} B_r(l_s^m) &\geq W_b(l_{s,D}), \quad \forall l_s^m \in P_{s,D} \\ P_{s,D} \cap G_l &= \text{NULL} \end{aligned} \right\} \quad (11)$$

- (3)更新路径 $P_{s,D}$ 上各链路的剩余带宽资源如式(12)所示;

$$B_r(l_s^m) = B_r(l_s^m) - W_b(l_{s,D}), \quad \forall l_s^m \in P_{s,D} \quad (12)$$

- (4)若 $P_{s,D}$ 为NULL,找不到故障备份链路的替代路径,否则返回故障备份链路的重新映射结果 $P_{s,D}$ 。

5 实验结果及分析

5.1 实验设定

仿真实验在配置 Pentium(R) Dual-Core CPU E6500 @2.93 GHz, 1.96 GB 内存的 PC 机上进行, 实验按照 GT-ITM 规则^[16], 参照文献[5,11], 在 25×25 的空间下生成了 50 个节点, 每一对物理节点以 0.5 的概率参数相互连接, 并且根据 Openflow 网络模型, 将这 50 个节点随机分为 5 个域。每个节点的控制资源和转发资源以及物理链路的带宽资源分别服从[50,100]的均匀分布, 其中每个域内的控制资源集中在一起形成统一的资源池。虚拟网络构建请求的到达过程服从时间单位为 1 个时间窗, 强度 $\lambda_r = 4$ 的泊松过程, 并且每个虚拟网络的生存时间服从均值为 10 个时间窗的指数分布。每个虚拟网络的节点数服从[2,10]的均匀分布, 虚拟节点之间以 0.5 的概率参数相互连接。每个虚拟节点需要的控制资源和转发资源分别服从[0,20]的均匀分布, 每条虚拟链路需要的带宽资源服从[0,50]的均匀分布。链路故障发生过程服从强度为 λ_f 的泊松过程, 链路故障恢复时间服从均值为 1 个时间窗的指数分布。将节点资源权重 α 和链路资源权重 β 都设为 1。对于蚁群算法, 设蚂蚁数目 M 为 50, 迭代次数 N 为 50, 信息素 τ_{ij} 的初始值为 5, 信息素的持久程度 ρ 为 0.8, 最优解对信息素的影响因子 σ 为 100000。

本文主要从虚拟网络故障修复率、虚拟网络成功运行率和工作链路资源利用率 3 个方面对故障备份链路重映射算法和区分用户优先级的故障恢复算法进行验证, 仿真结果如图 2 到图 9 所示。

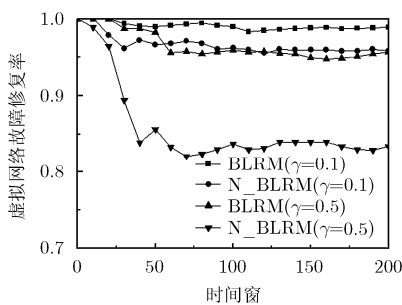


图2 故障备份链路重映射算法对虚拟网络故障修复率的影响

5.2 故障备份链路重映射算法验证实验

虚拟网络故障修复率是指故障发生后能够正常运行的虚拟网络数目与故障发生前正常运行的虚拟网络数目的比值。故障备份链路重映射算法(BLRM)对虚拟网络故障修复率的影响如图 2 所示, 其中 $\gamma = \lambda_f / \lambda_r$ 。当 $\gamma = 0.1$ 时, 未采用故障备份链路重映射算法^[10](N_BLRM)的虚拟网络故障修复率大

概在 95%, 而采用了该算法之后的虚拟网络故障修复率基本上稳定在 98%, 大约提升了 3%, 这主要是因为该条件下发生故障的链路数目较少, 备份链路的修复对虚拟网络可靠性的提升效果体现不明显; 当 $\gamma = 0.5$ 时, 发生故障的链路数目增多, 未采用该算法的虚拟网络故障修复率大约为 83%, 而采用了该算法的虚拟网络故障修复率可以稳定在 95%, 提升幅度达到了 12%左右。

虚拟网络成功运行率指的是成功运行的虚拟网络数目与虚拟网络构建请求数目的比值, 反映了虚拟网络的构建有效性和对链路故障的保护能力, 图 3 反映了故障备份链路重映射算法对虚拟网络成功运行率的影响。当 $\gamma = 0.1$ 时, 由于链路故障的发生概率低, 虚拟网络成功运行率受备份链路故障的影响极小, 主要由虚拟网络的构建有效性决定, 因此采用该算法仿真结果相差不大; 当 $\gamma = 0.5$ 时, 链路故障的增多导致采用故障备份链路重映射算法的虚拟网络成功运行率提升明显。

5.3 区分用户优先级的故障恢复算法验证实验

图 4 和图 5 分别表示 $\gamma = 0.1$ 和 $\gamma = 0.5$ 时采用备份路径构造算法^[9](Res_Backup)、故障链路重映射算法^[17](Res_Remap)和区分用户优先级的故障恢复算法(Priority_Diff)对虚拟网络故障修复率的影响。由于备份路径构造算法在链路故障发生之前就已经为映射成功的虚拟网络设置了备份路径, 因此故障修复率较高, 而故障链路重映射算法只是当故障发生之后才为故障链路寻找迁移路径, 因此故障修复率较低。当 $\gamma = 0.1$ 时, 在区分用户优先级的故障恢复算法下, 网络可靠性级别高的用户虚拟网络故障修复率接近 100%, 而网络可靠性级别低的用户故障修复率基本稳定在 80%; 当 $\gamma = 0.5$ 时网络故障明显增多, 但是网络可靠性级别高的用户仍然保持在 98%左右, 而低级用户稳定在 60%以上。

图 6 和图 7 表示的是 $\gamma = 0.1$ 和 $\gamma = 0.5$ 时虚拟网络成功运行率的变化情况, 虽然采用备份路径构造算法能够获得较高的虚拟网络故障修复率, 但是由于备份路径占用了大量的网络可用资源, 导致拒绝接收的虚拟网络数目增多, 虚拟网络的成功运行率最低, 而采用故障链路重映射算法的虚拟网络成功运行率最高, 区分用户优先级的故障恢复算法高于备份路径构造算法。因此, 区分用户优先级的故障恢复算法能够在保证高级用户的虚拟网络可靠性前提下, 维持虚拟网络具有较高的成功运行率。

图 8 和图 9 分别表示 $\gamma = 0.1$ 和 $\gamma = 0.5$ 时工作链路资源利用率的变化情况, 由于采用备份路径构造算法占用了很大一部分网络资源作为备份路径,

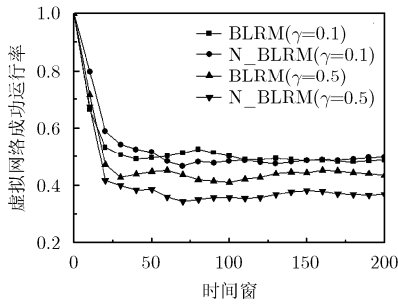


图3 故障备份链路重映射算法对虚拟网络成功运行率的影响

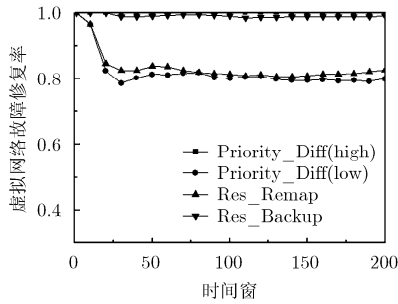


图4 $\gamma=0.1$ 时虚拟网络故障修复率

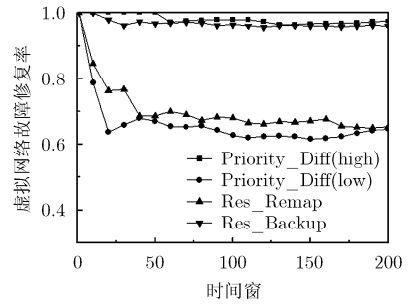


图5 $\gamma=0.5$ 时虚拟网络故障修复率

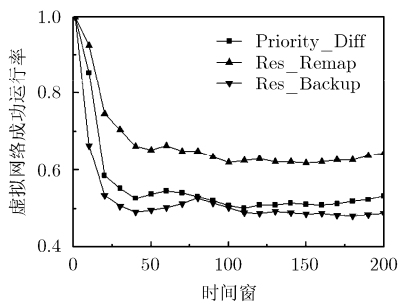


图6 $\gamma=0.1$ 时虚拟网络成功运行率

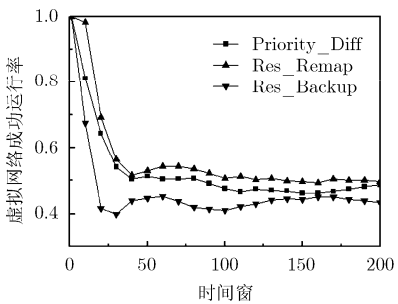


图7 $\gamma=0.5$ 时虚拟网络成功运行率

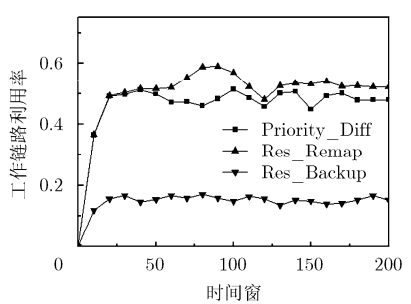


图8 $\gamma=0.1$ 时工作链路资源利用率

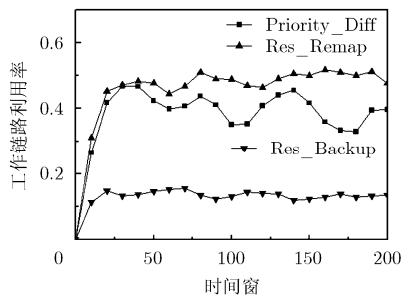


图9 $\gamma=0.5$ 时工作链路资源利用率

导致工作链路资源利用率远远低于其它算法；区分用户优先级的故障恢复算法稍逊于故障链路重映射算法，能够保证物理网络具有较高的资源利用率。

6 结论

以 Openflow 网络为代表的虚拟化网络能够满足不同用户的多样化业务需求，支持多种路由协议，保护用户信息的安全，推动传统互联网架构向下一代架构体系演进。本文以网络中最易发生的单链路故障为前提，基于不同用户的网络可靠性需求不同，设计了区分用户优先级的故障恢复算法；为了增强备份链路的可用性，提出了故障备份链路重映射算法。最后从虚拟网络故障修复率、虚拟网络成功运行率、工作链路资源利用率方面证明了两种算法的优越性。

参考文献

- [1] Chowdhury N M M K and Boutaba R. A survey of network virtualization[J]. *Computer Networks*, 2010, 54(5): 862-876.
- [2] Carapinca J and Jimenez J. Network virtualization — a view from the bottom[C]. 1st ACM Workshop on Virtualized Infrastructure Systems and Architectures Proceedings, Barcelona, Spain, Aug. 17-21, 2009: 73-80.
- [3] Bless R and Werle C. Network virtualization from a signaling perspective[C]. IEEE ICC Workshops Proceedings, Dresden, Germany, June 14-18, 2009: 1-6.
- [4] Sherwood R, Chan M, Gibb G, et al. Carving research slices out of your production network with openflow[J]. *ACM SIGCOMM Computer Communication Review*, 2010, 40(1): 129-130.
- [5] Chowdhury N, Rahman M, and Boutaba R. Virtual network embedding with coordinated node and link mapping[C]. IEEE INFOCOM Proceedings, Rio de Janeiro, Brazil, Apr. 19-25, 2009: 783-791.
- [6] 吕博, 杨帆, 王振凯, 等. 一种基于区域划分的虚拟网映射新算法[J]. 电子与信息学报, 2011, 33(10): 2347-2352.
Lü Bo, Yang Fan, Wang Zhen-kai, et al. A novel virtual network mapping algorithm based on regionalization[J]. *Journal of Electronics & Information Technology*, 2011, 33(10): 2347-2352.
- [7] 李文, 吴春明, 陈健, 等. 物理节点可重复映射的虚拟网映射算法[J]. 电子与信息学报, 2011, 33(4): 908-914.

- Li Wen, Wu Chun-ming, Chen Jian, *et al.* Virtual network mapping algorithm with repeatable mapping over substrate nodes[J]. *Journal of Electronics & Information Technology*, 2011, 33(4): 908-914.
- [8] Butt N, Chowdhury M, and Boutaba R. Topology-awareness and re-optimization mechanism for virtual network embedding[C]. Proceedings of 9th International Networking Conference, Chennai, India, 2010: 27-39.
- [9] Raihan M, Issam A, and Boutaba R. Survivable virtual network embedding[C]. Proceedings of the 9th International Networking Conference, Chennai, India, 2010: 40-52.
- [10] Yeow W L, Wsestphal C, and Kozat U C. Designing and embedding reliable virtual infrastructures[J]. *ACM SIGCOMM Computer Communication Review*, 2011, 41(2): 57-64.
- [11] Chowdhury M, Raihan M, and Boutaba R. ViNEYard: virtual network embedding algorithms with coordinated node and link mapping[J]. *IEEE/ACM Transactions on Networking*, 2012, 20(1): 206-219.
- [12] Houidi I, Louati W, and Zeghlache D. A distributed virtual network mapping algorithm[C]. IEEE International Conference on Communications, France, 2008: 5634-5640.
- [13] Fajjari I, Saadi N A, Pujolle G, *et al.* VNE-AC: virtual network embedding algorithm based on ant colony metaheuristic[C]. IEEE International Conference on Communications, Kyoto, 2011: 1-6.
- [14] Iannaccone G, Chuah C, Mortier R, *et al.* Analysis of link failures in an IP backbone[C]. Proceedings of ACM SIGCOMM Internet Measurement Workshop 2002, Marseille, France, 2002: 237-242.
- [15] Markopoulou A, Iannaccone G, and Bhattacharyya S. Characterization of failures in an IP backbone[C]. Proceedings of INFOCOM 2004, Hong Kong, China, 2004: 2307-2317.
- [16] Zegura E, Calvert K, and Bhattacharjee S. How to model an Internetwork[C]. IEEE INFOCOM Proceedings, San Francisco, Mar. 24-28, 1996: 594-602.
- [17] 齐宁, 汪斌强, 王志明. 可重构服务承载网容错构建算法研究[J]. 电子与信息学报, 2012, 34(2): 468-473.
- Qi Ning, Wang Bin-qiang, and Wang Zhi-ming. Research on reconfigurable service carrying network resilient construction algorithms[J]. *Journal of Electronics & Information Technology*, 2012, 34(2): 468-473.
- 蔡进科: 男, 1988 年生, 硕士生, 研究方向为数据中心网络、网络虚拟化.
- 顾华玺: 男, 1977 年生, 博士, 副教授, 研究方向为网络技术、片上网络以及光互连等.
- 卢 冀: 男, 1981 年生, 博士后, 研究方向为云计算、网络虚拟化、网络服务.