傅鲁奇,王昌达. 基于 FFT 的长短周期嵌套故障链路恢复方法[J]. 智能计算机与应用,2026,16(1):89-96. DOI:10.20169/j.issn.2095-2163.24032702

基于 FFT 的长短周期嵌套故障链路恢复方法

傅鲁奇, 王昌达
(江苏大学 计算机科学与通信工程学院, 江苏 镇江 212013)

摘 要: 链路故障作为一类常见的网络问题,会对网络运行的可靠性产生严重影响。然而现有的被动式方法需要较长时延来实现链路故障恢复,而传统的主动式方法则需要消耗大量且昂贵的三态内容寻址存储器(TCAM)。针对这些挑战,本文提出一种基于 FFT 的长短周期嵌套方法(Long and Short Nested Cycles, LSNC),用于链路故障的快速恢复。该方法在外部周期 T 中,利用 DLBS 算法对链路进行备份;每个外部周期 T 分为多个内部周期 t,在内部周期 t 内进行流量监测,对外部周期建立的备份路径进行更新。LSNC 利用快速傅里叶变换(FFT)获取不同区域网络流量的周期,作为外部周期 T 和内部周期 t。本文基于 GEANT 数据集展开实验,证明 LSNC 在平均带宽利用率和备份路径长度两方面优于现有的 BF 和 CR 等较为成熟的方法。

关键词: 链路故障恢复; 网络可靠性; 流量周期; 快速傅里叶变换
中图分类号: TP393.0 **文献标志码**: A **文章编号**: 2095-2163(2026)01-0089-08

Failed link recovery method with Long and Short Nested Cycles based on FFT

FU Luqi, WANG Changda
(School of Computer Science and Communication Engineering, Jiangsu University, Zhenjiang 212013, Jiangsu, China)

Abstract: Link failure, as a common network issue, can significantly impact the reliability of network operation. However, existing passive methods typically entail longer delays for link failure recovery, while traditional active methods require substantial and costly Ternary Content Addressable Memory (TCAM). To address these challenges, this paper proposes a method called Long and Short Nested Cycles (LSNC) based on Fast Fourier Transform (FFT) for rapid link failure recovery. In this method, within an external period T, links are backed up using the DLBS algorithm. Each external period T is divided into multiple internal periods t, during which traffic monitoring takes place to update the backup paths established during the external period. LSNC utilizes FFT to obtain the periods of different region network traffic, serving as the external period T and internal period t. Experimental validation conducted on the GEANT dataset demonstrates that LSNC outperforms existing mature methods such as BF and CR in terms of average bandwidth utilization and backup path length.

Key words: link failure recovery; network reliability; traffic cycles; Fast Fourier Transform

0 引 言

互联网已成为社会基础设施的一个重要组成部分,发挥着支持各种应用进行高质量数据传输的关键作用。然而,实现高质量的数据传输需要可靠的基础网络[1]。据网络服务提供商(ISP)统计,平均有 30%的链路在一年内会发生故障[2]。这使得链路故障成为通信网络中的高发故障。因此,实现快速的链路故障恢复对确保网络可靠性和服务质量至关重要。

软件定义网络(SDN)是当今主流的网络管理框架之一。在 SDN 中,控制平面与数据平面实现解耦,控制平面中的控制器可以收集和分析网络流量数据,然后根据网络功能的需求,向数据平面下发数据包的转发规则,来实现对网络数据流的处理,这一特性极大提高了链路故障恢复的效率。在 SDN 中,修复故障链路的常见方法是流量绕行,即控制平面指示数据平面将要流经故障链路的数据包重定向到非故障链路上,同时保持目的节点不变[3]。流量绕行的工作过程如图 1 所示。图 1 中,红色线为高故障概率的工作链路,绿色线为绕行路径。

基金项目: 国家自然科学基金(62072217)。
作者简介: 傅鲁奇(1998—),男,硕士研究生,主要研究方向:网络安全。
通信作者: 王昌达(1971—),男,教授,主要研究方向:网络安全,网络通信,物联网等。Email:changda@ujs.edu.cn。
收稿日期: 2024-03-27

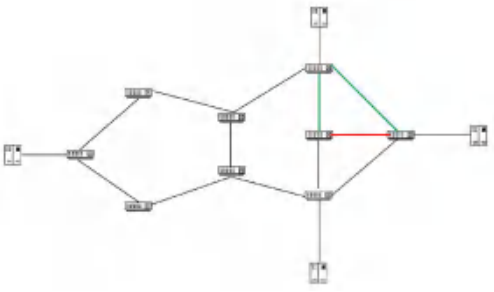


图 1 SDN 网络拓扑

Fig. 1 SDN network topology

在此基础上,链路故障恢复的方法可以分为 2 种:被动式链路故障恢复和主动式链路故障恢复。在被动式方法中,当控制器检测到链路出现故障后,启动对应的重路由算法选择绕行路径。如李书[4]提出一种动态自适应链路故障恢复机制,采用动态重路由方式进行故障恢复,对于控制流和数据流制定不同故障恢复方法,实现链路资源合理分配,避免拥塞。但是根据研究发现,被动式方法在链路故障恢复期间通常会发生较高的数据包传输延迟,一般无法满足商业电信服务对延迟要求应小于 50 ms 的服务质量标准[1,5]。

而主动式方法则是提前为具有高故障概率的工作链路准备绕行路径。Adrichem 等学者[6]提出一种链路故障快速切换方法,通过提前为工作路径中每个节点到目的节点提供备份路径,实现链路故障恢复。Chu 等学者[7]提出了一种通过预先配置 IP 隧道的方式来解决链路故障问题,避免链路拥塞。虽然传统主动式方法通常可以满足电信服务级延迟要求,但却需要使用昂贵的 TCAM 来维护备份路径,限制了其实用性[8]。因此,Wang 等学者[9]提出一种两阶段的链路故障恢复方法,通过不断监测网络状况调整备份路径。根据网络状态,调整备份路径,减少资源的消耗。但是该方法在调整备份路径过程中,消耗的计算资源过多,调整时间长。

综上所述,为了减少链路故障恢复过程中的数据包传输延迟以及尽量减少 TCAM 的使用量,本文设计了一种基于快速傅里叶变换(Fast Fourier Transform, FFT)的长短周期嵌套方法 LSNC(Long and Short Nested Cycles)。该方法由 2 个长短不同的周期嵌套组成,在外部周期 T 中,当每个 T 开始时,利用 DLBS 算法生成链路的备份路径;而在内部周期 t 中,通过对链路的流量进行监测,当通过链路的流量超过设定的流量阈值时,对于该链路所在的备份路径进行标记。在下一个内部周期 t 开始时,对标记的备份路径进行更新。而 2 个周期参数的计算,则是基于不同时间跨度的流量,利用快速傅里叶变换(FFT)计算不同区域的流量周期,分别作为外部周期参数(T)和内部周期参数(t),其中 $t < T$。LSNC 方法是将备份路径的局部更新融入全局更新当中,拓展了全局更新的时间间隔,有效地解决了传统主动式恢复方法的资源密集性的问题,避免了频繁地更新备份路径。本文使用了公开的 GEANT 数据集验证了 LSNC 方法的可行性与有效性。实验结果表明,LSNC 在平均带宽利用率和备份路径长度两方面都优于 2 种已知的基准方法:CR[6]和 BF[10]。LSNC 在均衡网络负载和高效利用底层网络资源方面具有显著优势。

1 长短嵌套周期参数计算方法

1.1 数据降噪

考虑在真实环境下,长期的流量数据采集过程中可能出现噪声数据[11],进而对实验结果产生负面影响[12]。因此,本文使用滑动窗口和多项式回归结合的方法对数据进行降噪处理。

本文选取一个网络流量集合 D, 利用该集合获取 2 个时间跨度不同的流量时间序列。一个是样本时间长度为 T_X 的流量时间序列 $X = \{x_1, x_2, \cdots, x_n\}$, 另一个是样本时间长度为 T_Y 的流量时间序列 $Y = \{y_1, y_2, \cdots, y_n\}$, 其中 $T_X > T_Y$。

对流量时间序列 X 和 Y 进行降噪处理,首先设置滑动窗口,其中窗口大小为 k。对在该窗口内的流量值利用多项式回归模型进行拟合,用拟合值来代替时间序列中的噪音值,对此可以表示为:

$$g(z) = a_n z^n + a_{n-1} z^{n-1} + \cdots + a_1 z + a_0 \quad (1)$$

其中, $g(z)$ 表示拟合多项式; $a_i (i \in \{0, \cdots, n\})$ 表示多项式系数; z 表示自变量; n 表示阶数。

在完成对整个流量时间序列降噪的基础上,利用标准化公式,对流量时间序列进行标准化处理,消除尺度上的影响,提高频谱性分析过程中的准确性标准化公式为:

$$x_i = \frac{X_i - \mathrm{mean}(X)}{\mathrm{std}(X)} \quad (2)$$

其中,mean(·)表示均值函数;std(·)表示标准差函数。

1.2 流量区域划分

不同的流量、其周期特征也各异,为了提升方法获取的流量周期参数的准确性,本方法利用经过各网络区域的流量特征,对网络区域进行划分。由于数据包的长度和数量可以衡量网络的负载状态[13],因此本文利用数据包长度和数据包数量这 2 个流量

特征作为网络区域划分的依据。

首先，利用网络监测统计通过各链路的数据包长度（fl_i）和数据包的数量（fn_i），将数据用式（2）进行标准化；然后，利用线性公式计算各链路的 link_f；最后，根据 link_f 值对链路进行排序，以链路作为构成网络区域的基本单位，借助于预设的网络区域划分比例，将网络分为两大区域，分别为高负载区域与非高负载区域。研究推得的公式如下：

$$\mathrm{link}\_f_i = \mathrm{fl}_i + \mathrm{fn}_i \tag{3}$$

其中，fl_i 表示网络数据包的长度，fn_i 表示网络数据包的数量。

1.3　频谱分析

根据网络区域类型的划分结果，将 X 和 Y 划分为 4 个流量子时间序列 X_1, X_2, Y_1, Y_2，其中 X_1 和 Y_1 表示处于高负载区域的流量所组成的时间序列，X_2 和 Y_2 表示其他流量组成的时间序列。

对 X_1, X_2, Y_1, Y_2 这 4 个流量子时间序列进行频谱分析，本节以 X_1, Y_1 为例。首先，利用 FFT 对 2 组时间序列进行频谱分析，由此确定 2 组时间序列的主频率；然后，通过主频率去计算时间序列的周期；其次，将获取的 2 个周期 T 和 t 作为备份路径更新时间的参数，其中 T 为外部周期，t 为内部周期；最后，利用自相关系数来检验周期的适用性。同理，可获得的 X_2, Y_2 外部与内部周期。

为了分别获取时间序列 X_1 和 Y_1 对应的周期 T 和 t，首先，研究用到的计算公式为：

$$X|K| = \sum_{n=0}^{N-1} x[n] \mathrm{e}^{-\mathrm{i}2\pi\frac{Kn}{N}} \tag{4}$$

$$A = \mathrm{Amp}(F) \tag{5}$$

其中，$X|K|$ 是频域的复数表示；$x[n]$ 表示时域信号；N 表示信号的长度；A 表示一组频率的振幅集合；$\mathrm{Amp}(\cdot)$ 函数用于计算频率对应的振幅；x_i 表示第 i 个流量；K 表示频率。

基于此，利用式（4）、式（5）分别对时间序列 X_1 和 Y_1 进行频谱分析[14]，获取一组频率集合，并通过函数 $\mathrm{Amp}(\cdot)$ 确定各频率的振幅值；然后，研究中还将用到如下公式：

$$\mathrm{argTop}(F)\{f_1, \cdots, f_k\}, f_q\{1, \cdots, [\frac{\pi}{2}]\} \tag{6}$$

其中，$F = \{X|K|_1, X|K|_2, \cdots, X|K|_k\}$ 表示一组频率的集合；$\mathrm{argTop}(\cdot)$ 函数可用于提取排列在降序顺序中的前 k 个频率；f_i 表示第 i 个频率；$q \in \{1, \cdots, k\}$。

这里，利用式（6）对振幅值进行降序排序，获取前 k 个振幅值。然后，提取上述振幅值对应的频率，将频率转化为周期，计算公式为：

$$T = \left\lfloor \frac{D}{f} \right\rfloor \tag{7}$$

其中，T 表示与第 i 个频率相关联的周期；D 表示此时间序列的时间跨度。

2　长短周期嵌套故障链路恢复方法

2.1　备份路径选择算法

假设网络 $G(V,E)$，其中 V 表示网络节点集合，E 表示链路集合。本节设定 B_p 作为链路 p 故障恢复时绕行的备份路径。备份路径由一组有序链路组成，表示为 $B_p = \{l_{o1}, l_{12}, \cdots, l_{ij}, l_{jd}\}$，备份路径的源节点 n_o 与目的节点 n_d 分别为故障链路的两端节点。本文首先利用式（3）计算各链路的 link_ f 值[15]，按照其大小对链路进行降序排序，且按 $a:b:c$ 的比例对链路进行分类。然后按照链路类型为链路赋予权重，其中权值越高表示链路的负载越大，该链路出现拥塞的概率越高[16]。

本文根据链路权重与备份路径的跳数，以及预设的权重系数 α 与 β 共同构建目标函数，目标函数见下式：

$$\mathrm{Min}(\alpha w_{l_i} + \beta t_{l_i}) \tag{8}$$

其中，w_l 表示链路 l 的备份路径的链路权重之和，t_l 表示链路 l 的备份路径的跳数之和。

为避免流量迁移后，造成链路拥塞，需要满足带宽条件，定义公式具体如下：

$$\mathrm{bw}_l + \sum_{p \in C} y_l^p \cdot \mathrm{bw}_l^p \leqslant b \cdot \mathrm{bw}_p \tag{9}$$

其中，bw_l 表示通过链路 l 的原有数据流占用的带宽；y_l^p 表示链路 p 的备份路径通过链路 l；bw_l^p 表示流量迁移将会占用链路的带宽；集合 C 表示通过链路 l 的备份路径的集合；bw_p 表示链路最大带宽；b 表示链路利用率。

由此，本文提出一种基于限定无环 KSP（K-Shortest Paths）算法[17]的备份路径选择算法 DLBS，用于在外部周期 T 中为链路选择备份路径，主要步骤如下。

步骤 1　从图 G 中去除由节点 n_o 与 n_d 构成的链路。

步骤 2　计算连接这 2 个节点的前 K 条不相交的最短路径，放入备份路径 $B = \{B_1, B_2, \cdots, B_K\}$ 中，若数量少于 K 条，则将所有最短路径选出并放入备份路径集合 B 中。

步骤 3　检验集合 B 中的备份路径是否包含环

路,以及带宽是否满足式(9),若不满足从集合中去除。

步骤 4 根据式(8)函数值对路径进行排序,选取 BP_{l_i} 值最小的路径作为该链路的备份路径。在上一周期 T 中,未检测到数据流通过的链路,算法不会为其分配备份路径。

2.2 备份路径更新算法

LSNC 方法在下一个外部周期 T 开始时,对该网络内所有链路的备份路径使用 DLBS 算法进行更新。更新过程包括统计周期 T 内的链路消耗的带宽数据,并重新计算链路权重和备份路径。

在内部周期 t 中,通过 SDN 控制器监测链路的带宽利用率,来了解链路的拥塞情况。在本文中,根据参考 Wang 等学者[1]的研究将带宽利用率阈值设定为 70%。如果链路的利用率超过该阈值,该链路将被视为拥塞,而包含该链路的备份路径将被标记。同时,该链路所属的备份路径也将进行标记。在下一内部周期 t 时,会对标记的备份路径进行更新,即利用 DLBS 算法实现全网备份路径的局部更新。根据张震宵[18]的研究经验数据,当某一链路出现带宽利用率高于 90%时,将立即进行备份路径更新。更新过程,包括统计周期内的链路流量与链路的带宽数据,重新计算链路权重和备份路径。LSNC 方法的流程如图 2 所示。

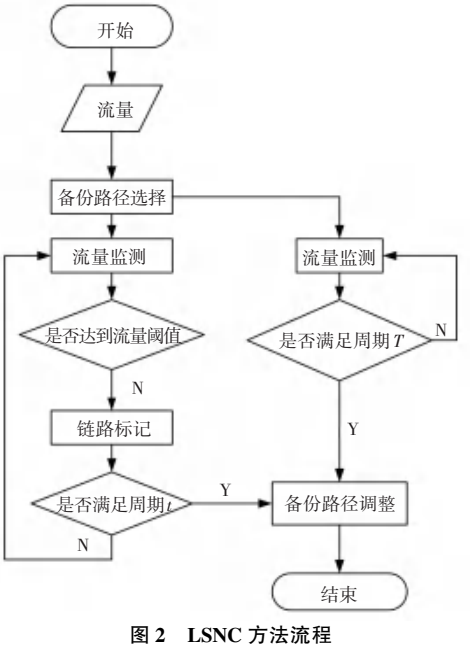


图 2 LSNC 方法流程

Fig. 2 Flowchart of LSNC method

3 实验结果与分析

3.1 实验环境

本文所有实验均在 PC 端上进行,安装有 Windows10 操作系统,内存 12 GB,搭载 Intel i5-7300HQ CPU,1 TB 硬盘与 GTX1050 显卡,开发环境使用 Python 3.6.0。本文从 SDNlib 中获取经典流量数据集 GEANT(http://sndlib.zib.de.),该数据集信息包括源目的节点,以及流量数值信息。该数据集的拓扑网络一共包含 22 个节点以及 36 条边,本文假设 22 个节点均为 SDN 交换机,上述交换机作为源节点与目的节点相互发送消息,同时负责路由。在仿真环境下,为了展现 LSNC 对各条链路发生故障时恢复效果,本文假设每条链路发生故障的概率与链路负载成正比[19],所有链路的限制带宽相同[18]。

本文根据 GEANT 数据集,构建一个加权无向拓扑图,如图 3 所示,展现了节点之间的连接关系与各链路的权重,根据上文所给出的链路类型,为对应链路赋予权重,权重值依次为 1、2、3。

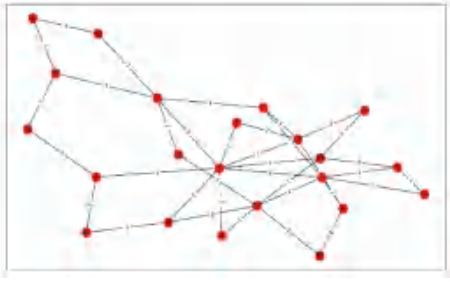


图 3 GEANT 无向加权图

Fig. 3 GEANT undirected weighted graph

3.2 评价指标

(1) 平均带宽利用率。定义为备份路径中所有链路的带宽利用率累积值与备份路径长度之比,其计算公式见如下:

$$\mathrm{bw}_{\mathrm{av}} = \frac{1}{N} \cdot \sum_{1 \leqslant i \leqslant N} \mathrm{bw}_i \tag{10}$$

$$\mathrm{bw}_i = \frac{\mathrm{Link}^i_{\mathrm{fact}}}{\mathrm{Link}^i_{\mathrm{max}}} \tag{11}$$

$$\mathrm{Link}^i_{\mathrm{fact}} = \mathrm{Link}^i_o + \mathrm{Link}^i_b \tag{12}$$

其中,$N \in N^+$ 表示备份路径中的链路数量;i^{th} 表示第 i 条链路,bw_i 表示带宽利用率;$\mathrm{Link}^i_{\mathrm{fact}}$ 表示备份路径启用时实际利用的带宽,包括原始流量 Link^i_o 和转移的流量 Link^i_b;$\mathrm{Link}^i_{\mathrm{max}}$ 表示最大可用带宽。

评估链路的带宽利用率在评估网络负载均衡方面具有重要意义,同时保持带宽利用率在可接受范围内通常有助于实现有效的数据传输。

(2)备份路径长度。定义为所有链路的备份路径长度的总和,其计算公式见如下:

$$\mathrm{len} = \sum_{l_{ij} \in E} \mathrm{len}_{l_{ij}} \tag{13}$$

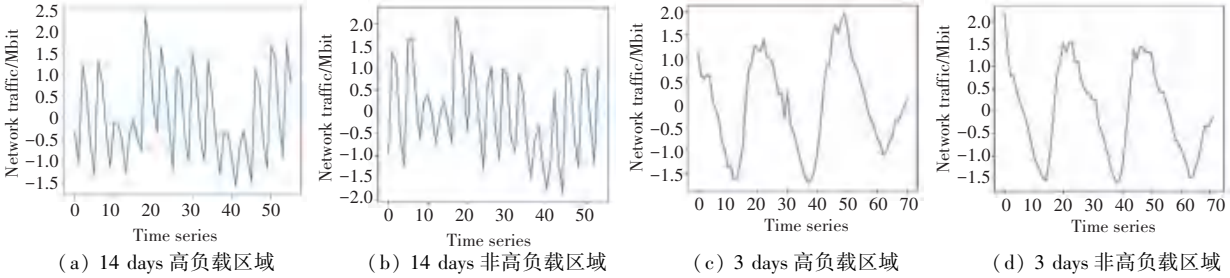
其中，$\mathrm{len}_{l_{ij}}$ 表示链路 l_{ij} 的备份路径长度。

SDN 网络中备份路径长度是评估采用的备份方法对 TCAM 资源利用情况的关键指标[20]。由于备份路径作为提前备份的转发路由，需要在 SDN 交换机的流表项中存储。因此，备份路径长度直接与所需的流表项数量相关联。在 SDN 网络中较长的备份路径长度表示消耗更多的 TCAM 资源[21]。

3.3　频谱分析结果

公开的 GEANT 数据集每天以 15 min 为间隔采集网络流量样本。在实验中，基于该数据集分别抽样提取了 2 组数据子集。第 1 组数据子集为每间隔 6 h 进行抽样提取，时间跨度为 14days。第 2 组数据子集为每间隔 1 h 进行抽样提取，时间跨度为 3 天。为保证多项式回归的精度以及结合流量时间序列的采样频率，本方法将滑动窗口 k 设置为 8，预设区域比例 5∶5。基于此，对 2 组流量数据进行数据预处理，根据流量流经的网络区域类型，将其分为 4 组流量数据。

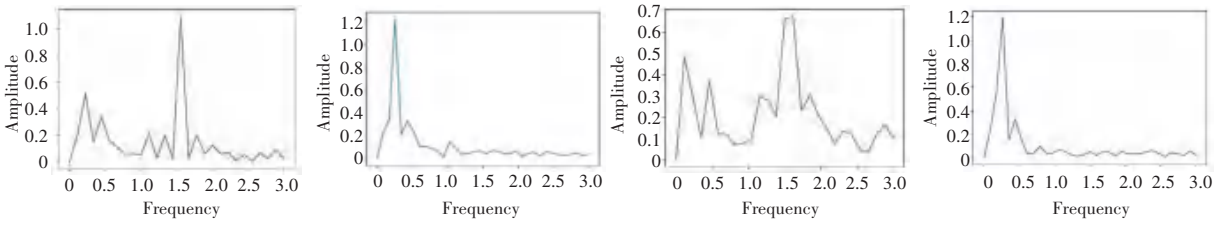
图 4 展示了 4 组网络流量的数值变化，从图 4 中可以发现流量数值的波峰波谷间隔有一定规律性。在图 4 基础上，本文对 4 组数据进行频谱分析，网络流量频增分析结果如图 5 所示。图 5(a)展示 14 days 高负载区域流量序列的频谱分析图，该图揭示了其最大频率。根据式(4)~式(7)，可以计算出其外部周期 T_1 为 22 h。图 5(b)展示 3 days 高负载区域流量序列的谱分析图，可以计算出其内部周期 t_1 为 2 h。图 5(c)、图 5(d)中的数据显示 14 days 和 3 days 非高负载区域流量序列的谱分析图，通过计算就可以得出外部周期 T_2 和内部周期 t_2 为 24 h 和 3 h。



(a) 14 days 高负载区域　(b) 14 days 非高负载区域　(c) 3 days 高负载区域　(d) 3 days 非高负载区域

图 4　网络流量变化

Fig. 4　Network traffic variation



(a) 14 days 高负载区域　(b) 3 days 高负载区域　(c) 14 days 非高负载区域　(d) 3 days 非高负载区域

图 5　网络流量频谱分析

Fig. 5　Spectral analysis of network traffic

通过上述的频谱分析，一个时间序列可以获取多个频率、以及频率对应的周期，因此本实验中为了验证 T_1, T_2, t_1, t_2 作为所分析网络流量数据的周期适用性，使用周期的自相关系数对其进行评估[22]，可用如下公式求值：

$$\rho(k)=\frac{\sum_{t=k+1}^{n}(x(t)-\mu)(x(t-k)-\mu)}{\sum_{t=1}^{n}(x(t)-\mu)^2} \tag{14}$$

其中，n 表示时间序列的长度；$x(t)$ 表示时间 t 的时间序列值；k 表示滞后步长；μ 表示时间序列的均值。

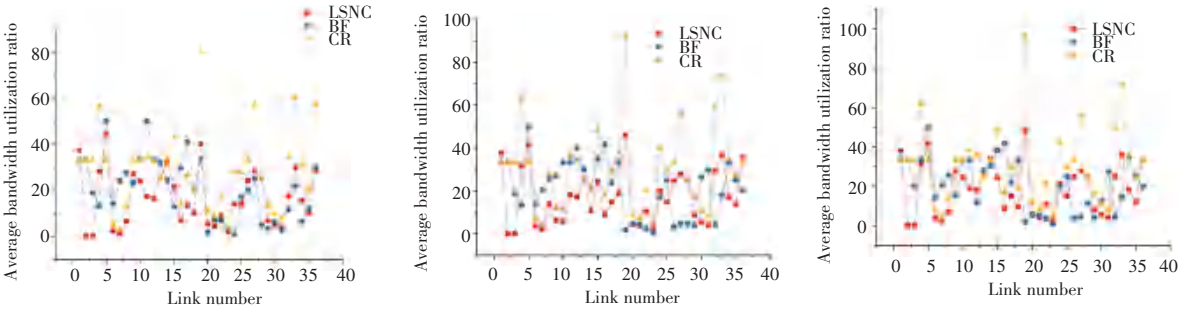
自相关系数显示了时间序列与其滞后值之间的相关性随时间变化的情况。该系数的取值范围在−1 到 1 之间，其中 1 表示完全正相关，−1 表示完全负相关，0 表示没有相关性。通过式(14)的计算，T_1, T_2, t_1, t_2 的自相关系数分别为 0.652 0、0.692 0、0.674 0、0.786 9，支持所选 T_1, T_2, t_1, t_2 之间存在相关性。

3.4　链路故障恢复方法对比实验结果

为了验证 LSNC 方法的有效性，本小节对比分析了另外 2 种链路故障恢复方法：CR[6] 与 BF[10]。图 6(a)~(c)中的数据是 3 种比较方法的平均带宽利用率，纵轴表示平均带宽利用率，横轴表示链路序号。顶点为方形的折线代表 LSNC 方法，顶点为圆

形的折线代表 BF 方法,顶点为三角形的折线代表 CR 方法。

为了展现不同链路故障恢复方法对各链路进行故障恢复时的表现,研究让各条链路依次单独发生故障。此外,根据 Zhang 等学者[13]的研究结果和本文在实验中使用的数据集的经验,本实验将带宽设置为 500 Mbps。同时,在实验中,通过调节链路分类的比例参数来模拟不同的网络状态场景,图 6(a)对应的场景是网络处于空闲状态,参数之比为 6∶2∶2;图 6(b)对应的场景是网络处于低负载状态,参数之比为 2∶6∶2;图 6(c)对应的场景是网络处于高负载状态,参数之比为 2∶2∶6。



(a) 空闲状态 (b) 低负载状态 (c) 高负载状态

图 6 不同状态下备份路径平均带宽利用率

Fig. 6 Average bandwidth utilization of backup paths under different states

图 6(a)的数据显示,CR 方法有 1 条链路的备份路径平均带宽利用率超过 70%,有 5 条链路的备份路径平均带宽利用率达到或接近 60%,BF 方法则有 2 条,而 LSNC 方法没有。这表明 CR 方法在完成对故障链路的数据迁移后,容易造成链路拥塞。同时,研究还统计了 3 个方法对于所有链路的备份路径平均带宽的均值,发现 LSNC 方法比 BF 方法降低了 2.6%,比 CR 方法降低了 11.1%。

图 6(b)的数据展示,在 CR 方法中有 6 条链路的备份路径的平均带宽利用率达到或者接近 60%,BF 方法和 LSNC 方法各有 1 条接近 60%,这是由于此时模拟的网络状态开始忙碌,LSNC 方法难以搜寻到合适的空闲备份路径,转而寻找转发跳数少、但带宽利用率低的备份路径。该状态下,统计所有链路的备份路径平均带宽的均值进行比较,发现 LSNC 方法比 BF 方法降低了 3.1%,比 CR 方法降低了 12.2%。

图 6(c)与图 6(b)中的数据趋势接近,但在部分链路上有较大差别,如链路 4。在图 6(c)中可以发现 CR 方法中平均带宽利用率超过或达到 60%依旧为 6 条,BF 方法和 LSNC 方法均为 1 条。在该状态下,比较所有链路的备份路径平均带宽的均值,可以发现 LSNC 方法比 BF 方法降低了 1.3%,比 CR 方法降低了 11.6%。

表 1 给出了 3 种恢复方法在完成故障链路的流量迁移之后,其备份路径中的链路带宽利用率超过 70%的链路数量,可以看出 LSNC 方法在 5 个时间段中,其带宽利用率超过 70%的链路数量始终低于 BF 方法和 CR 方法,证明其在均衡网络负载方面的优势。

表 1 高负载链路数量

Table 1 Number of overload links

Method	LSNC	BF	CR
Time 1	25	95	159
Time 2	22	101	154
Time 3	24	96	156
Time 4	22	101	151
Time 5	24	99	155

图 7 给出了在低负载状态下,与图 6 不同的时间段的平均带宽利用率,用以验证 LSNC 方法在不同时间段下的适用性。从图 7 中的结果数据可以看出,CR 方法有链路的备份路径其平均带宽明显高于 70%,有 2 条链路接近 60%,BF 方法也有 2 条接近 60%,而 LSCN 方法则没有出现此类情况。LSNC 方法所有链路的平均带宽利用率比 BF 方法低 3.64%,比 CR 方法低 9.78%。

由于负载更高的链路,通过该链路的流量数据也越多,因此相较于其他链路,该链路出现故障可能性更大。本文中为了验证 LSNC 方法在不同类型链路上的适用性,在图 7 数据的基础上挑选了高负载链路进行实验,比较上述 3 种方法对高负载链路进行故障恢复的效果。高故障概率链路的备份路径平均带宽利用率如图 8 所示。LSNC 方法和 BF 方法的高负载链路的备份路径平均带宽利用率均不超过

50%，而 CR 方法则有 1 条达到了 85%，达到了拥塞水平，3 个方法中的 7 条高概率链路其平均带宽利用率分别为 LSNC 方法的 36.6%、BF 方法的 31.9%、CR 方法的 51.5%。

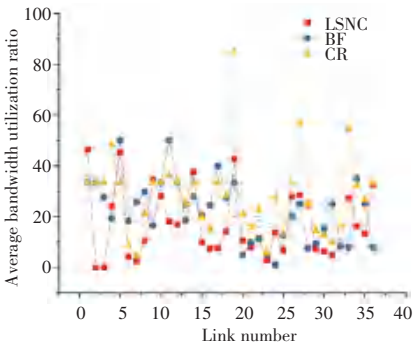


图 7　备份路径平均带宽利用率

Fig. 7　Average bandwidth utilization of backup paths

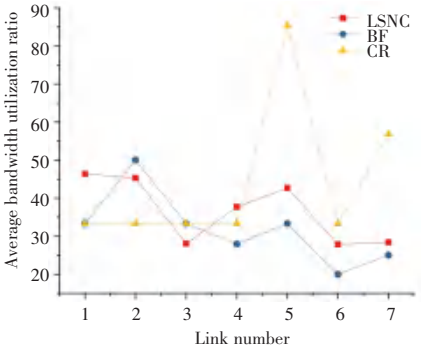


图 8　高故障概率链路的备份路径平均带宽利用率

Fig. 8　Average bandwidth utilization of backup paths for high-failure probability links

上述结果证明在对比 CR 方法时，LSNC 方法在均衡流量迁移之后的网络负载方面更具优势。当高负载链路发生故障时，LSNC 方法对比 BF 方法可知，BF 方法能更好地均衡该场景下的网络负载，但是 LSNC 方法能充分考虑链路特点，在恢复非高负载链路故障方面性能更佳。

图 9 中的数据显示了在 GEANT 数据集中，3 种链路故障恢复方法的备份路径长度实验结果，纵轴表示备份路径长度，横轴表示时间。其中，图 9(a)对应网络处于空闲状态下，图 9(b)对应网络处于低负载状态下，图 9(c)对应网络处于高负载状态下。表 2 给出在 3 种状态下，3 种恢复方法在 6 个时间段的平均备份路径长度。

图 9(a)中的数据显示，在 6 个时间段内，LSNC 方法的备份路径长度均低于 BF 方法和 CR 方法，这是由于 LSNC 方法对于空闲链路不配置备份路径，且结合表 2 可以发现，LSNC 方法的 6 个时间段的平均备份路径长度 102.8，而 BF 方法为 110.3，CR 方法为 106.0。由于 CR 方法的备份路径选取依靠网络拓扑，当网络拓扑没有改变时，CR 方法的备份路径长度不会改变。在图 9(b)中，随着网络逐渐繁忙，LSNC 方法需要绕行更长的备份路径，来避免将拥塞链路作为备份路径，因此 LSNC 有 2 个时间段是高于 CR 方法的，但平均备份路径长度依旧低于 BF 方法，其中 LSNC 方法的平均备份路径长度为 105.8，依旧低于 CR 方法的 106.0 和 BF 方法的 109.3。

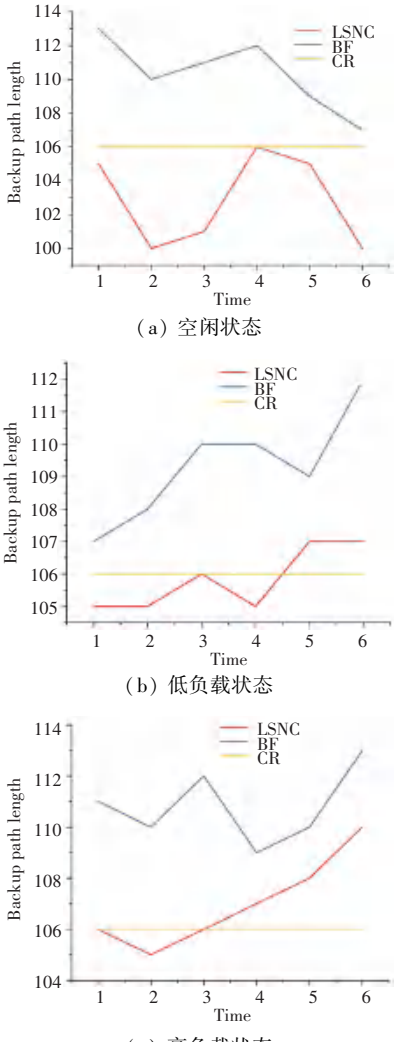


图 9　不同状态下备份路径长度

Fig. 9　Length of backup paths under different states

表 2　3 种状态下平均备份路径长度

Table 2　Average length of backup paths under three different states

States	LSNC	BF	CR
Overloaded	102.8	110.3	106
Normal	105.8	109.3	106
Idle	107.0	110.8	106

在图 9(c)中，网络处于高负载状态，此时 LSNC

方法有 3 个时间段的备份路径长度高于 CR 方法，其中 LSNC 方法的备份路径平均长度为 107.0，BF 方法的备份路径长度为 110.8，CR 方法为 106.0。可以发现随着网络忙碌，LSNC 方法与 BF 方法备份路径的长度开始接近，这是由于 BF 方法基于数据流提供备份，当数据流的数量增加时，BF 方法性能优势将逐步显现。

考虑到备份路径长度与 SDN 交换机的流表项资源消耗成正相关，结合备份路径长度实验结果可以证明 LSNC 方法能够减少流表项资源的消耗。同时，上文的备份路径平均带宽利用率的实验，证明了 LSNC 方法在减少流表项资源消耗的前提下，通过避免将高负载链路作为备份路径，实现更好的网络负载优化效果。

4 结束语

在 SDN 网络中，链路故障对网络可靠性构成重大威胁，因此通过快速恢复方案解决该问题显得尤为重要。本文提出了 LSNC 方法用以实现链路故障快速恢复。LSNC 方法包括 2 个相互嵌套周期，外部周期阶段 T 开始时，利用 DLBS 算法确定全局链路的备份路径。在内部周期 t 中，对链路流量进行监测，对包含超过流量阈值的链路的备份路径进行标记，在下一个内部周期 t 开始时，对被标记的备份路径进行更新。利用经过网络区域的流量特征划分网络区域，通过 FFT 获取不同区域流量周期，作为长短周期的参数。实验结果表明，LSNC 方法在避免链路拥塞方面的性能优于 CR 和 BF 方法，在空闲状态和低负载状态下，LSNC 方法在备份路径长度方面始终优于 CR 和 BF 方法。

参考文献

[1] WANG Shishuang, XU Hongli, HUANG Liusheng, et al. Fast recovery for single link failure with segment routing in SDNs [C]//Proceedings of 2019 IEEE 21st International Conference on High Performance Computing and Communications. Piscataway, NJ:IEEE, 2019:2013-2018.

[2] TURNER D, LEVCHENKO K, SNOEREN A C, et al. California fault lines: Understanding the causes and impact of network failures[J]. ACM SigCOMM Computer Communication Review, 2010,40(4):315-326.

[3] MOHAN P M, TRUONG-HUU T, GURUSAMY M. Fault tolerance in TCAM-limited software defined networks [J]. Computer Networks, 2017,116:47-62.

[4] 李书. 基于拥塞避免的 SDN 链路故障恢复机制研究[D]. 重庆:重庆邮电大学,2017.

[5] SHARMA S, STAESSENS D, COLLE D, et al. Openflow: Meeting carrier-grade recovery requirements [J]. Computer Communications, 2013,36(6):656-665.

[6] ADRICHEM N L M V, ASTEN B J V, KUIPERS F A. Fast recovery in software-defined networks[C]//Proceedings of 2014 Third European Workshop on Software Defined Networks. Piscataway, NJ:IEEE, 2014:61-66.

[7] CHU C Y, XI Kang, LUO Min, et al. Congestion-aware single link failure recovery in hybrid SDN networks[C]//Proceedings of 2015 IEEE Conference on Computer Communications (INFOCOM). Piscataway, NJ:IEEE, 2015:1086-1094.

[8] ALI J, LEE G M, ROH B H, et al. Software defined networking approaches for link failure recovery: A survey[J]. Sustainability, 2020,12(10):4255.

[9] WANG Likun, YAO Lin, XU Zichuan, et al. CFR: A cooperative link failure recovery scheme in software-defined networks[J]. International Journal of Communication Systems, 2018, 31(10): e3560.

[10] 王金鑫. 软件定义网络中可靠性问题的研究[D]. 成都:电子科技大学,2020.

[11] 邹柏贤,姚志强. 一种网络流量平稳化方法[J]. 通信学报, 2004,25(8):14-23.

[12] 李乐. 时空序列数据预处理方法研究[D]. 北京:中国科学院大学(中国科学院遥感与数字地球研究所),2017.

[13] ZHANG Shujuan, WANG Ying, HE Qichao, et al. Backup-resource based failure recovery approach in SDN data plane[C]// Proceedings of Network Operations & Management Symposium. Piscataway, NJ:IEEE,2016:1-6.

[14] WU Haixu, HU Tengge, LIU Yong, et al. Timesnet: Temporal 2D-variation modeling for general time series analysis[J]. arXiv preprint arXiv,2210.02186, 2022.

[15] 李深昊. SDN 中基于流量分类的路由优化技术研究与实现[D]. 北京:北京邮电大学,2021.

[16] ZHANG Shujuan, WANG Ying, HE Qichao, et al. Backup-resource based failure recovery approach in SDN data plane[C]// Proceedings of 2016 18th Asia-Pacific Network Operations and Management Symposium (APNOMS). Piscataway, NJ: IEEE, 2016: 1-6.

[17] 周子韬. 面向复杂网络拓扑的路径规划算法研究[D]. 南京:南京理工大学,2020.

[18] 张震霄. 具有可靠性的 k 最短路径及其优化问题的研究[D]. 上海:华东理工大学,2020.

[19] LIU Xinhua, YANG Xiang. Disturbance based dynamic load balancing routing mechanism under SDN[C]//Proceedings of 2016 5th International Conference on Computer Science and Network Technology (ICCSNT). Piscataway, NJ:IEEE,2016:651-656.

[20] 刘翔元,基于网络流量分析的网络设备类型识别关键技术研究[D]. 南京:南京邮电大学,2019.

[21] 张方芳. 基于 OpenFlow 流表规则优化与管理的研究[D]. 成都:电子科技大学,2019.

[22] YANG Xuwei, XU Hongli, HUANG Liucheng, et al. Joint virtual switch deployment and routing for load balancing in SDNs[J]. IEEE Journal on Selected Areas in Communications, 2018, 36(3):397-410.