

一种基于图神经网络的 SDN 路由性能预测模型

车向北¹, 康文倩¹, 邓 彬¹, 杨柯涵², 李 剑²

(1. 深圳供电局有限公司, 广东深圳 518000; 2. 北京邮电大学计算机学院, 北京 100876)

摘 要: 软件定义网络作为未来网络架构的发展方向, 通过分离数据平面与控制平面高效设定路由方案. 而在路由方案的优化过程中, 准确预估给定路由方案下的网络性能是其关键. 本文基于图神经网络建模网络中物理链路与路由方案路径的关系, 在给定的路由方案与网络流量下对网络中的各项端到端性能指标(如延迟、抖动)进行准确预估, 以辅助优化路由方案. 本文基于 OMNeT++ 来生成数据并进行实验, 实验结果表明本文提出的模型能够针对延迟抖动等端到端性能指标进行准确预估, 预估平均相对误差不超过 4.1%. 实验也对比了传统最短路径路由算法与基于该预测模型给出的最优路由方案下的端到端性能, 相比传统最短路径路由算法, 平均延迟和平均抖动分别降低了 19.8% 和 33.52%, 最大延迟和最大抖动降低了 36.18% 和 35.45%.

关键词: 软件定义网络; 端到端性能预测; 图神经网络; SDN 路由优化

中图分类号: TP393.0 **文献标识码:** A **文章编号:** 0372-2112 (2021)03-0484-08

电子学报 URL: <http://www.ejournal.org.cn> **DOI:** 10.12263/DZXB.20200120

A Prediction Model of SDN Routing Performance Based on Graph Neural Network

CHE Xiang-bei¹, KANG Wen-qian¹, DENG Bing¹, YANG Ke-han², LI Jian²

(1. Shenzhen Power Supply Bureau Co., Ltd., Shenzhen, Guangdong 518000, China;

2. School of Computer Science, Beijing University of Posts and Telecommunications, Beijing 100876, China)

Abstract: As the development direction of future network architectures, Software Defined Networks can efficiently set routing schemes by separating the data plane and the control plane. In the process of optimizing a routing scheme, it is the key to accurately predict the network performance under a given routing scheme. This paper uses graph neural networks to model the relationship between physical links and routing scheme paths, so that the model can predict various end-to-end performance indicators (such as delay and jitter) in the network under a given routing scheme and network traffic. This paper uses OMNeT++ to generate datasets. The experimental results show that the model proposed in this paper can accurately predict end-to-end performance indicators such as delay and jitter. The average relative error of the estimate does not exceed 4.1%. The experiment also compares the end-to-end performance of the traditional shortest path routing algorithm with the optimal routing scheme based on the prediction model proposed in this paper. Compared to the traditional shortest path routing algorithm, the average delay and average jitter are reduced by 19.8% and 33.52%, and the maximum delay and maximum jitter are reduced by 36.18% and 35.45%.

Key words: software development network (SDN); end-to-end performance prediction; graph neural network; SDN routing optimization

1 引言

随着信息技术的不断发展, 网络规模越发庞大, 传

统的底层网络架构已无法满足需求, 设备配置繁杂, 迭代缓慢, 各种问题层出不穷. 为了解决这些问题, 软件定义网络 (Software Defined Network, SDN) 这一设计理念

得到提出. 在 SDN 范式中, 通常需要高效地在控制平面对网络进行优化, 当上层应用提出网络服务质量目标(如网络延时、抖动等)后, SDN 控制器(Controller)需要迅速对网络进行配置, 并制定出符合要求的路由策略.

这一过程通常分为两步, 首先需要建立模型来对一定场景下的路由策略性能进行预测, 然后, 优化算法需要通过搜索各种配置方案, 以获得满足要求的方案. 整个流程如图 1 所示.

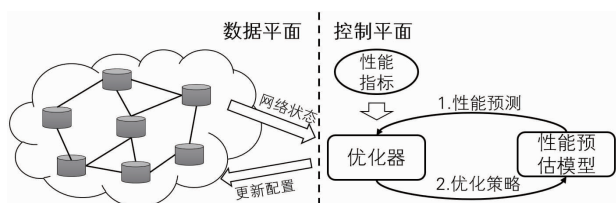


图1 SDN中控制平面优化流程

在整个过程中, 准确预估网络端到端性能是优化路由策略的首要条件. 网络中诸如延迟、抖动、损耗等端到端的性能指标是评估一个路由策略的关键因素. 传统模型在预测求解时往往会对网络进行简化, 假定网络模型具有某些非现实的属性, 例如流量服从泊松分布^[1], 得到的预测结果与真实数据往往存在较大差异. 因此, 如何在 SDN 控制器构建模型对当前场景下的路由策略进行准确的端到端性能评估是一项非常有挑战性且十分重要的工作.

近些年来, 人工智能特别是深度学习技术^[2]得到巨大发展, 神经网络可以对语音、图像以及自然语言等诸多事物进行建模, 并从数据中学习到有意义的语义信息, 进而对新的相关问题进行求解. 人工智能在计算机网络上也有着相当多的应用, 如文献[3]中使用神经网络来对计算机网络进行建模, 并用此类模型对网络模型进行优化^[4], 也有研究将深度强化学习应用在该领域以进行路由策略的优化^[5-7].

这些方法^[8,9]在建模时通常使用比较传统的神经网络结构, 例如全连接结构的神经网络、卷积神经网络(Convolution Neural Network, CNN)以及递归神经网络(Recurrent Neural Network, RNN). 这几种结构的神经网络在图像本文等领域有着不错的性能. 近些年来图神经网络(Graph Neural Network, GNN)的研究受到广泛关注^[10,11], 图神经网络能够对图结构数据进行信息提取, 在诸多领域有着广泛应用, 例如文献[12]中使用 GNN 来分析化学分子的拓扑结构并预测分子性质, 在文献[13]使用 GNN 来建模知识图谱中的关系, 以及在生物学中使用 GNN 对蛋白质结构进行建模^[14].

本文结合图神经网络针对 SDN 网络中控制平面设计了一种模型, 该模型将网络设备的拓扑结构, 路由策

略、网络流量与端到端性能之间的关系进行建模, 从而能够在给定的网络环境下, 对网络中任意设备对之间的平均延迟和数据抖动等性能进行准确预估, 并且模型具有非常好的泛化能力, 当拓扑结构、路由策略以及网络流量发生变化时依旧能够进行准确预测. 这种预测模型能够辅助 SDN 控制器进行路由方案优化, 从而高效的查找到满足平均延迟和抖动要求的路由方案.

2 背景及相关技术

2.1 传统网络架构与 SDN 技术

传统互联网架构主要以 OSI 七层或者 TCP/IP 四层协议模型为主, 各层网络设备之间通过相应的网络协议(交换、路由、标签、安全等协议)来进行数据传递. 大体工作流程都是按照: 邻居建立-信息共享-路径选择三个步骤来实现. 另外, 网络设备之间传递信息采用典型的分布式架构, 设备之间以“接力棒”的形式交互信息, 然后建立数据库信息, 再依据相关路径算法(如 Dijkstra 最短路径算法)传递数据. 各层设备独立计算, 有独立的控制器和转发硬件, 通过协议来进行沟通.

传统网络使用了分布式架构, 这在协议规范不完整的过去促进了互联网的蓬勃发展. 但随着现今通信设备协议等逐步统一完善, 分布式架构已逐渐到达瓶颈, 凸显出诸多问题, 例如传输表信息冗余、流量难以控制、设备无法自定义传输等. 出现这些问题的根本原因在于传统架构中网络设备数据与控制相耦合, 且设备不具有开放性的可编程接口, 从而无法将数据转发与数据控制分离, 导致网络难以采用更高效的集中式管理.

软件定义网络这一理念正是为了解决这一问题被提出^[15], 其核心思想是将网络上的所有信息集中到一个核心控制器(Controller)上, 控制器能够采用集中式的方法直接操纵下层基础设备, 整体地处理网络的信息传输逻辑, 并对应用软件提供可编程接口, 如图 2 所示.

控制平面在收到应用平面的相关指令后, 需要透明地制定满足应用要求的路由策略, 因此在 SDN 的实现过程中, 其中一项关键技术在于如何高效快速地完成这一过程. 这一过程通常分为两步, 首先需要建立模型来对一定场景下的路由策略性能进行预测, 然后, 优化算法需要通过搜索各种配置方案, 以获得满足要求的方案.

总之, 通过对网络进行建模, SDN 能够更大程度发掘网络潜能, 从而在无需修改数据平面的情况下高效管理网络.

2.2 深度学习与图神经网络

神经网络在本质上是一个用来拟合函数的工

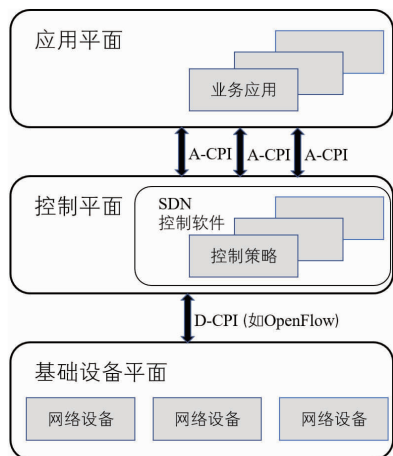


图2 SDN架构图

具,而现实世界中许多事物之间的关系通常可以用一些模型来近似估计.因此通过大量采样获取相应场景下的数据,并用该数据对神经网络进行训练,神经网络便可以对相应问题进行近似建模和预测.不同架构的神经网络(CNN、RNN、GNN)通过在网络架构层面引入不同先验知识,从而更好的对相应领域的问题进行表示学习.

图模型对现实世界中的事物有着强大的建模能力,在计算机视觉、分子化学、分子生物学等领域有着广泛应用.图的本质其实是一种事物间的交互关系,这种交互关系可以表现为图上节点能够从其领域节点聚合信息,同时也对邻节点传递信息.本文所提出的SDN路由策略预测模型主要基于GraphSAGE^[16]图神经网络搭建.GraphSAGE模型的核心思想在于学习一个聚合函数,对于图上任意节点的嵌入信息都可以通过该函数将其邻点的特征进行聚合来完成.在每一层信息聚合过程中,每个节点都将上一层的领域及其自身特征进行聚合,来得到该层的特征信息,具体过程如图3所示.

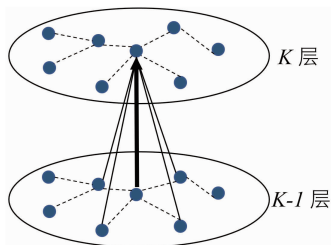


图3 GraphSAGE聚合过程示意图

3 模型与算法

本文设计的模型可以准确预测SDN网络中给定路由策略下的各项网络性能指标.模型输入信息包括:(1)当前给定网络的拓扑结构信息;(2)端到端路由路径组成的路由方案;(3)网络的流量矩阵.模型将输出

当前配置下的相关性能指标,如每条路由线路下的网络延迟,网络抖动等,该指标可以根据业务场景自定义,如图4所示.

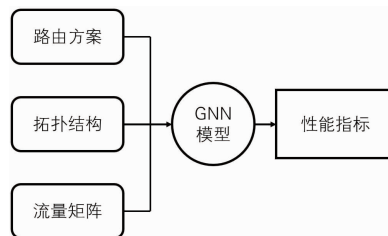


图4 模型的输入输出

3.1 图模型建模

计算机网络可由一系列物理链路 l_k 的集合表示,即网络 $N = \{l_1, l_2, \dots, l_{n_l}\}$.网络中端到端的路径 p_k 由多个连续物理链路的集合表示,即 $p_k = (l_{k(1)}, \dots, l_{k(1_{p,k})})$,其中 $k(i)$ 表示路径 p_k 中的第 i 条链路.路由策略 P 由一系列路径集合表示,即: $P = \{p_k\}, k \in \{1, \dots, n_p\}, n_p$ 表示当前路由策略 P 中路由路径的数量.

传统方法在计算网络性能指标时通常需要做出一一些模型简化的假设,例如在计算路径 p_k 上的延迟 T_{delay} 时对路径上所有链路的延迟进行求和,即:

$$T_{\text{delay}}(p_k) = \sum_i d(l_{k(i)}) \quad (1)$$

其中 $d(l_i)$ 代表链路 l_i 上的延时.这种方法通常建立在忽略网络丢包率以及序列依赖等假设下成立.但在实践中,这些假设中被忽略的细节往往是预测真实网络性能的关键因素.如果使用神经网络等深度学习方法,这些网络内在的物理属性都能够自然而然的被模型学习到,从而无需做出简化假设便能够对预测问题进行精准的建模与求解.

图的本质是事物之间的交互关系,而计算机网络中链路路由路径也存在互相交互的依赖关系.因此,可以将链路路由路径之间的关系建模为二分图模型,其中链路路由路径都表示为图的节点.如图5所示,左边为计算机网络拓扑和路由方案中的路由路径,右边利用图模型将链路节点和路径节点之间的关系进行了描述.整体为二分图模型,图中节点分为链路节点和路径节点两类,路径上的所有链路节点都与该路径节点存在直接相连关系.

通过这种方式对计算机网络中的链路路由路径建立二分图模型,可以使得所得到的图模型在本质上与计算机网络物理结构相分离,同时计算机网络中拓扑信息均通过链路节点与路径节点的交互关系进行内在表示.

接下来本文将借助GraphSAGE算法模型,为每个链路节点 l_i 和路径节点 p_i 引入待学习的嵌入向量 h_{l_i} 与

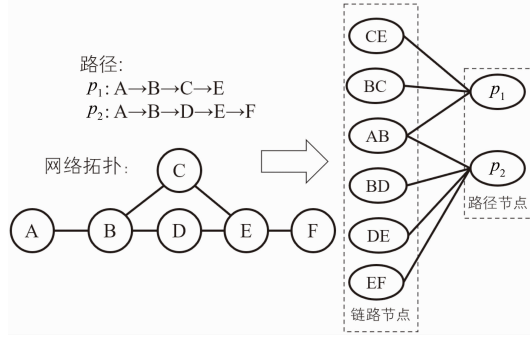


图5 链路-路径节点图建立过程

h_p , 这些嵌入向量应该包含与建模目标相关的语义信息, 如延时、丢包率、链路利用率等. 借助这些嵌入向量, 模型可以对相关路径上端到端的性能指标进行预测.

3.2 模型架构

在上一小节 3.1 中, 通过建立二分图模型, 拓扑结构和路由方案已经被编码到了图结构中, 而流量矩阵中的流量信息可以作为路径节点上的特征信息进行输入.

基于文献[16]中的 GraphSAGE 框架, 在这里提出针对于控制平面预测相关路由策略下的网络性能的算法. 如图 4 所示, 预测算法的目标在于学习到路由方案、拓扑结构、流量矩阵与网络性能之间的关系. 最终完成训练的模型要能在给定的网络拓扑、流量矩阵以及路由方案下对网络性能进行准确预测.

算法 1 为针对特定网络拓扑、流量矩阵、路由策略的网络性能前向计算过程, 相关参数的训练见 3.3 小节.

算法 1 RPBG (Route Prediction by GraphSAGE)

输入:

计算机网络拓扑 T , 流量矩阵 F , 路由策略 R , 所有链路的路径的初始属性, 非线性函数 σ , 聚合次数 K

训练好的模型参数 $LSTM_w^k, W_p^k, W_l^k, DNN_w$

输出:

性能指标 \hat{y}_p

1. 根据路由策略 R 和计算机网络拓扑 T 建立链路节点 V_l 与路径节点 V_p 二分图
2. 根据链路节点与路径节点自身属性对节点向量 $\{h_v^0, v \in V_l \cup V_p\}$ 进行初始化
3. For $k = 1, \dots, K$ do:
4. For $v \in V_p$ do:
5. $h_{N(v)}^k \leftarrow LSTM_w^k(\{h_u^{k-1}, \forall u \in N(v)\})$
6. $h_v^k \leftarrow \sigma(W_l^k \cdot \text{CONCAT}(h_v^{k-1}, h_{N(v)}^k))$
7. End For
8. For $v \in V_l$ do:
9. $h_{N(v)}^k \leftarrow \text{SUM}(\{h_u^{k-1}, \forall u \in N(v)\})$
10. $h_v^k \leftarrow \sigma(W_l^k \cdot \text{CONCAT}(h_v^{k-1}, h_{N(v)}^k))$
11. End For

$$12. \quad h_v^k \leftarrow \frac{h_v^k}{\|h_v^k\|_2}, \forall v \in V$$

13. End For

$$14. \quad \hat{y}_p \leftarrow DNN_w(h_v^k), \forall v \in V_p$$

第 1 行根据路由策略、网络拓扑建立起路径节点与链路节点之间的图关系, 该过程如图 5 所示. 第 2 行利用节点的属性, 如链路本身的属性 (如带宽等物理属性) 与路径上的属性 (如端到端的流量情况) 对所有节点的初始特征属性初始化.

第 3 行到第 13 行通过 GraphSAGE 中的多层聚合方法来建模路径节点和链路节点之间的相互关系. 具体来说, 第 4 行到第 7 行对图中的路径节点进行信息聚合, 由于每条路径由多条链路依次顺序连接得到, 因此第 5 行按照该序列利用长短记忆神经网络 LSTM^[17] 来提取特征, 第 6 行对结果进行非线性变换以增加其表达能力. 第 8 行到第 11 行对所有链路节点进行信息聚合, 由于每条链路可能参与了多个路径, 因此第 9 行利用加和函数按列对其邻接路径节点特征进行信息聚合. 第 12 行对所有特征向量进行归一化.

最终, 针对于特定的场景, 在 14 行利用前馈神经网络来得到端到端的性能指标 (如路径的延迟时间、抖动等), 并通过预测性能指标与真实值的误差来更新网络中的整个参数.

3.3 模型参数的训练

上述算法在训练过程中需要优化的最终目标包括以下参数:

(1) 第 5 行中各层长短期记忆网络权重 $\{LSTM_w^k\}_{k=1, \dots, K}$.

(2) 第 6 行与第 10 行中各层非线性变化参数 $\{W_l^k, W_p^k\}_{k=1, \dots, K}$.

(3) 第 15 行中将嵌入向量映射为性能指标的神经网络参数 DNN_w .

为了对参数进行训练, 需要获取训练样本. 首先, 搭建与预测任务拓扑规模相当的网络结构, 记录其拓扑结构, 然后随机生成一系列流量矩阵和路由方案, 并进行实验或者仿真以测量任意路径下的端到端网络性能. 反复实验获取到不同网络拓扑、路由方案、流量矩阵组合下的端到端性能数据后, 记录相关数据作为训练数据集 D_{train} .

接下来, 按照 3.2 小节中的算法来对训练集中特定网络拓扑 T 、流量矩阵 F 、路由策略 R 下的性能进行计算, 以计算值 \hat{y}_p 与真实值 y_p 之间的均方误差作为损失函数来对算法中所有参数进行训练, 该损失函数 l_{loss} 表达如式 (2) 所示:

$$l_{\text{loss}} = \frac{1}{|D_{\text{train}}|} \sum_{D_{\text{train}}} (\hat{y}_p - y_p)^2 \quad (2)$$

模型整体参数的训练过程如图 6 所示,在更新时针对训练集中不同设定下的数据进行梯度求导并更新模型参数。

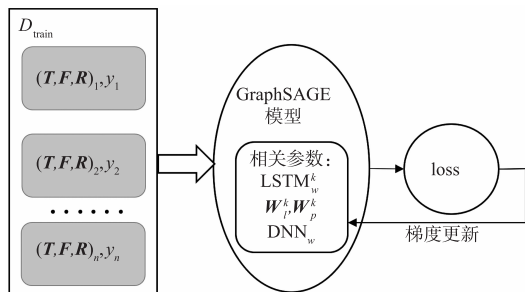


图6 参数训练过程

除了上述所涉及到的各类参数之外,算法无论在训练过程还是预测过程,还涉及到具体的链路节点和路径节点的嵌入向量的计算,这部分向量只是用来进行辅助训练和计算网络性能指标。

最终通过这种方式训练后网络中的各项参数,对于全新的拓扑结构、路由策略以及流量矩阵,在建立图模型后,按照 3.2 节中的算法完成信息聚合后便能够对网络性能进行准确预测。

4 实验

4.1 实验环境

本文提出的模型能够准确预测给定网络拓扑、路由策略和流量矩阵下网络中端到端的性能指标,实验分为两部分,首先验证了这种预测模型的有效性,其次说明了这种预测模型在路由优化中的应用价值。

首先使用 OMNeT++^[18] 网络仿真软件搭建了一系列拓扑结构的网络模型,其中流量矩阵 F 中的元素按照以下表达式(3)设定:

$$F(i, j) = U(0.1, 1) \times \frac{\alpha}{N-1} \times b \quad (3)$$

其中 $M(i, j)$ 表示从源点 i 到终点 j 的流量情况, $U(0.1, 1)$ 表示 0.1 至 1 的均匀分布, α 为控制整体网络中流量强度的参数, N 表示网络节点数量, b 为链路最大带宽,本文实验中设为 10Mbps。

网络结构采用了文献[19]中的 14 节点、20 条链路的 NSF 网络,文献[20]中的 24 节点的 Geant2 网络以及文献[21]中 17 节点、26 条链路的 GBN 网络,NSF 网络结构如图 7 所示。为了简化实验,将这些网络中的链路设置为相同的属性,最大带宽均设为 10Mbps。仅仅改变网络中的流量情况,流量情况最终将反映在路径节点的属性上。

4.2 预测模型的训练和评估

本小节通过实验将要证明以下两点结论:

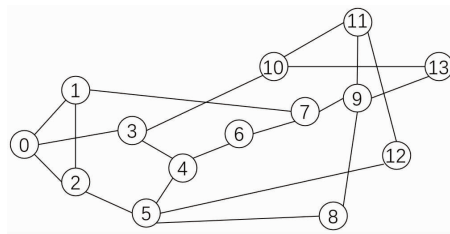


图7 NSF网络示意图

(1) 在特定拓扑结构的网络上完成训练后,本文提出的模型能够在流量矩阵和路由策略发生变化时进行准确预测。

(2) 当在一种拓扑结构上进行训练后,模型也能在不同拓扑上进行泛化预测。

首先利用 OMNeT++ 模拟器对 NSF、Geant2 以及 GBN 网络进行模拟仿真以获取训练和测试数据。具体来说,在每种拓扑结构下,先随机生成 100 种不同的路由方案,在每种路由方案下按照式(3)生成 2000 种不同的流量矩阵,并针对每种拓扑结构、路由方案和流量矩阵的组合,利用 OMNeT++ 模拟器进行仿真生成网络种端到端的网络延迟和网络抖动数据。使用 Geant2 网络中 80 种不同路由方案来训练模型,并将模型应用到其余 20 种路由方案下测试,同时也将此训练好的模型直接用于测试 NSF 以及 GBN 网络。

实验中,路径嵌入向量维度设置为 64,链路嵌入向量设置为 32。使用 tensorflow 进行代码实现,在图神经网络中取 K 为 2,采用 leaky Relu 作为非线性函数。利用路径嵌入向量来获得端到端性能指标的前馈神经网络,采用了两层神经网络,最后一层通过线性映射直接给出预测值。优化器采用 Adam 优化算法来进行训练。

训练过程中,通过网络端到端延迟值的预测值与真实值之间的均方误差来对模型中的 $LSTM_w^k$, W_i^k , W_p^k , DNN_w 等参数进行优化,训练好的模型用来预测三种网络中端到端的延迟值。为了对网络抖动性能进行预测,实验借助了迁移学习的思想,在之前训练好的模型基础上,重新利用抖动数据单独针对 DNN_w 参数进行迁移训练以预测网络抖动性能。

最终在评测时,模型给出路由策略中的所有路径上的端到端延迟值和抖动值,并与真实值进行比较,计算出相对误差。将每种网络结构下所有的相对误差取平均作为评测指标,最终结果如下表 1 所示。

表 1 三种网络对于网络延迟和网络抖动的平均预测相对误差

平均相对误差	Geant2	NSF	GBN
网络延时	0.8%	2.1%	2.7%
网络抖动	1.7%	3.9%	4.1%

从数据中发现,对于抖动的预测结果由于借助了

迁移学习的思想,并没有单独训练一套图神经网络来对路径和链路节点进行嵌入表示,相比于预测网络延时,性能要弱一些.但总的看来,在不同的流量情况下,本文所提出的模型还是能够对网络延迟、网络抖动等指标都能进行较为准确的预测.

另外,所有结果中采用的预测模型都是 NSF 拓扑下训练好的模型,其中 GBN 与 Geant2 的拓扑数据只是参与测试,并未用于训练.从实验数据中可以发现,本文所提出的基于图神经网络模型能够相当高效的对网络中的物理链路与路由策略中的路径之间的关系进行建模.而一旦知道了链路和路径之间的关系,物理网络本身的拓扑结构在计算时是可以忽略的,这也是本文提出的模型尽管只在 NSF 拓扑下进行训练,但却可以泛化在 NSF 与 GBD 拓扑结构下进行准确预测的原因.

4.3 预测模型在路由策略优化中的应用

上一小节通过实验证明了本文提出的模型能够对于不同拓扑与路由策略下的网络性能进行准确预估,这一小节将应用此模型来对路由策略进行优化,并与传统基于最短路径路由算法进行性能对比.采用上述实验中在 Geant2 网络上训练的延迟与抖动预估模型,在 NSF 网络拓扑上进行性能对比.对比的性能指标包括以下两方面:(1)所有路径上的平均端到端延迟和抖动;(2)所有路径上的最大延迟和抖动.

针对特定网络拓扑和流量情况,本文提出的优化算法首先随机生成 200 种不同的路由方案,然后利用模型预测相应的优化指标,选择性能最好的路由方案与传统基于 dijkstra 最短路径的路由算法进行对比.在实验中对比了不同流量强度下的路由性能,即改变式(3)中的 α 值来控制流量强度.在不同的 α 值下,首先随机生成了 100 种不同的流量矩阵,根据此流量矩阵作为输入,并依据这两种路由算法获得的相应最佳路由方案.在 OMNeT++ 上进行仿真,针对这两种路由方案,分别对相关性能参数进行仿真测试,记录网络中端到端的所有性能指标,并计算所有路径上的平均端到端延迟和抖动与最大延迟和抖动.根据这些结果,画出流量强度系数 α 与相关性能结果的曲线图,如图 8 至图 11 所示.

实验结果表明,在流量强度较低时($\alpha < 9$),这两种算法性能差异并不大.但当流量强度较高时,传统基于 dijkstra 最短路径的路由算法会产生流量拥堵等问题,而根据本文提出的预测模型得到的路由方案可以有效的避免此类问题,从实验数据也可以发现,随着流量强度不断增大,本文路由算法给出的方案相比传统算法性能提高的幅度越来越大.在流量强度系数达到 15 时,该模型相比基准模型对于平均延迟和

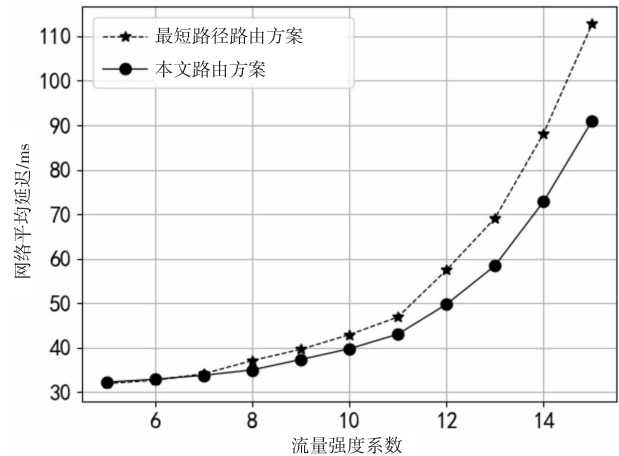


图8 端到端路径平均延迟

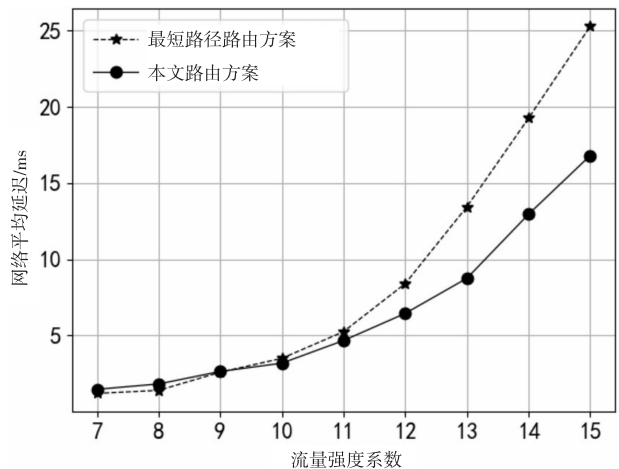


图9 端到端路径平均抖动

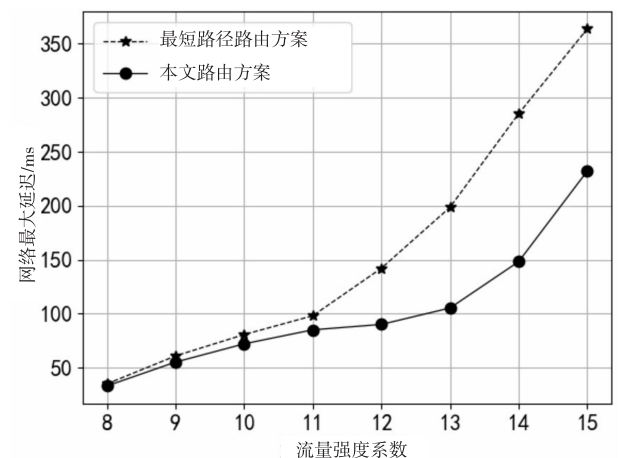


图10 端到端路径最大延迟

平均抖动分别降低了 19.8% 和 33.52%，对于最大延迟和最大抖动降低了 36.18% 和 35.45%。这其中的原因是本文提出的预估模型在不同的流量强度下的数据上进行了训练,从而能够有效捕捉到流量强度对于性能的影响.

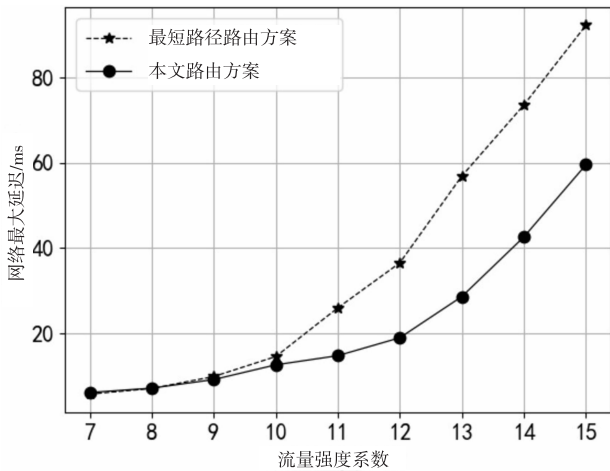


图11 端到端路径最大抖动

5 结论与展望

SDN 技术将控制平面与数据平面相分离,为网络优化带来无限可能,是未来网络技术的热点方向.然而实现 SDN 技术的一大难点在于控制平面如何高效地建立路由策略,准确预估当前路由策略下的网络性能是建立路由策略的基础.传统方法将网络模型进行简化,忽略了网络中许多关键信息,通常无法很好的预测网络性能指标.

本文提出了一种基于图神经网络的计算机网络性能预估模型,能够捕获到网络拓扑中物理链路与路由策略路径之间的复杂关系,从而在给定的拓扑结构、路由策略和流量矩阵下能够快速准确的预测出相应延迟、抖动等性能指标,以便后续高效地建立路由策略.而当前主流路由优化算法通常基于深度优先搜索来完成,效率较低.另外,拓扑结构与网络性能之间的映射也是不可微的,难以用梯度方法优化求解.在未来,如果在本文提出的预测模型基础上,结合诸如进化策略^[22]或者遗传算法^[23]等零阶优化算法来对网络路由方案进行优化,SDN 控制器对路由策略的优化过程一定能有更高效的解决方案.

参考文献

- [1] Ciucu F, Schmitt J B. Perspectives on network calculus: no free lunch, but still good value [J]. *Acm Special Interest Group on Data Communication*, 2012, 42(4): 311–322.
- [2] Meng D, Sun L. Somew trends of deep learning research [J]. *Chinese Journal of Electronics*, 2019, 28(6): 1087–1090.
- [3] Wang M, Cui Y, Wang X, et al. Machine learning for networking: workflow, advances and opportunities [J]. *IEEE Network*, 2018, 32(2): 92–99.
- [4] Mao B, Fadlullah Z M, Tang F, et al. Routing or computing? the paradigm shift towards intelligent computer network packet transmission based on deep learning [J]. *IEEE Transactions on Computers*, 2017, 66(11): 1946–1960.
- [5] Chen X, Guo J, Zhu Z, et al. Deep-RMSA: A deep-reinforcement-learning routing, modulation and spectrum assignment agent for elastic optical networks [A]. *2018 Optical Fiber Communications Conference and Exposition (OFC)* [C]. San Diego: IEEE, 2018. 1–3.
- [6] 张德干, 葛辉, 刘晓欢, 等. 一种基于 Q-Learning 策略的自适应移动物联网路由新算法 [J]. *电子学报*, 2018, 46(10): 2325–2332.
Gan Z D, Hui G E, Huan L X, et al. A kind of new routing algorithm with adaptivity for mobile IOT based on Q-learning [J]. *Acta Electronica Sinica*, 2018, 46(10): 2325–2332. (in Chinese)
- [7] Xu Z, Tang J, Meng J, et al. Experience-driven networking: a deep reinforcement learning based approach [A]. *International Conference on Computer Communications* [C]. Honolulu: IEEE, 2018. 1871–1879.
- [8] Mestres A, Alarcón E, Ji Y, et al. Understanding the modeling of computer network delays using neural networks [A]. *Proceedings of the 2018 Workshop on Big Data Analytics and Machine Learning for Data Communication Networks* [C]. Budapest Hungary: ACM, 2018. 46–52.
- [9] Xiao Shihan, He Dongdong, Gong Zhibo. Deep-Q: traffic-driven QoS inference using deep generative network [A]. *Proceedings of the 2018 Workshop on Network Meets AI & ML* [C]. Budapest Hungary: ACM, 2018. 67–73.
- [10] Zhou J, Cui G, Zhang Z, et al. Graph Neural Networks: A Review of Methods and Applications [EB/OL]. <https://arxiv.org/pdf/1812.08434.pdf>, 2018.
- [11] Xiao L, Lu R. A fully complex-valued gradient neural network for rapidly computing complex-valued linear matrix equations [J]. *Chinese Journal of Electronics*, 2017, 26(6): 1194–1197.
- [12] Gilmer J, Schoenholz S S, Riley P, et al. Neural message passing for quantum chemistry [A]. *International Conference on Machine Learning* [C]. Sydney: ICML, 2017. 1263–1272.
- [13] Hamaguchi T, Oiwa H, Shimbo M, et al. Knowledge transfer for out-of-knowledge-base entities: a graph neural network approach [A]. *International Joint Conference on Artificial Intelligence* [C]. Melbourne: IJCAI, 2017. 1802–1808.
- [14] A Fout, J Byrd, B Shariat, A Ben-Hur. Protein interface prediction using graph convolutional networks [A]. *Advances in Neural Information Processing Systems* [C]. Long Beach: NIPS, 2017. 6530–6539.
- [15] 胡涛, 张建辉, 邬江, 等. SDN 中基于分布式决策的控制

器负载均衡机制[J]. 电子学报, 2018, 46(10): 2316 – 2324.

Tao Hu, Jian-Hui Zhang, Jiang Wu, et al. Controller load balancing mechanism based on distributed policy in SDN [J]. Acta Electronica Sinica, 2018, 46(10): 2316 – 2324. (in Chinese)

- [16] Hamilton W L, Ying Z, Leskovec J, et al. Inductive representation learning on large graphs[A]. Advances in neural information processing systems. [C]. Long Beach: NIPS, 2017. 1024 – 1034.
- [17] Staudemeyer R C, Morris E R. Understanding LSTM—a Tutorial into Long Short-term Memory Recurrent Neural Networks[EB/OL]. <https://arxiv.org/pdf/1909.09586.pdf>, 2019.
- [18] András Varga, Rudolf Hornig. An overview of the OM-NeT + + simulation environment[A]. Proceedings of the 1st International Conference on Simulation Tools and Techniques for Communications, Networks and Systems & Workshops [C]. Marseille France: ICST, 2008. 1 – 10.
- [19] Hei X, Zhang J, Bensaou B, et al. Wavelength converter placement in least-load-routing based optical networks using genetic algorithms [J]. Journal of Optical Networking, 2004, 3(5): 363 – 378.
- [20] Barreto F, Wille E C, Nacamura L, et al. Fast emergency paths schema to overcome transient link failures in OSPF routing [J]. International Journal of Computer Networks & Communications, 2012, 4(2): 17 – 34.
- [21] Pedro J, Santos J, Pires J, et al. Performance evaluation of integrated OTN/DWDM networks with single-stage multiplexing of optical channel data units [A]. International Conference on Transparent Optical Networks [C]. Stockholm: IEEE, 2011. 1 – 4.
- [22] Qin A K, Huang V L, Suganthan P N, et al. Differential evolution algorithm with strategy adaptation for global numerical a Optimization [J]. IEEE Transactions on Evolutionary Computation, 2009, 13(2): 398 – 417.
- [23] 王福才, 周鲁苹. 混合精英策略的元胞多目标遗传算法及其应用[J]. 电子学报. 2016, (3): 709 – 717
Fucai Wang, Lupin Zhou. Cellular multi-objective genetic algorithm based on hybrid elite and application [J]. Acta Electronica Sinica, 2016(3): 709 – 717. (in Chinese)

作者简介



车向北 男, 1984 年 08 月出生, 陕西宝鸡人, 深圳供电局有限公司高级工程师. 主要从事电力监控系统网络安全工作.
E-mail: chexiangbei@163.com



康文倩 女, 1988 年 04 月出生, 江苏徐州人, 深圳供电局有限公司工程师. 主要从事电力监控系统网络安全工作.
E-mail: wenqiankang@163.com



邓彬 男, 1989 年 4 月出生, 湖北黄石人, 深圳供电局有限公司系统运行部工程师. 主要从事调度自动化系统、智能调度技术等方向研究.
E-mail: 15219495096@163.com



杨柯涵 男, 1996 年 4 月出生, 陕西汉中, 北京邮电大学计算机学院硕士研究生, 主要研究领域为深度学习, 计算机网络.
E-mail: kehanyang@bupt.edu.cn



李剑 男, 1976 年 12 月出生, 陕西西安人, 北京邮电大学计算机学院教授、博士生导师. 主要从事密码学、网络空间安全、量子信息、计算机应用、人工智能、软件定义网络等方向研究.
E-mail: lijian@bupt.edu.cn