

一体化网络多终端接入智能路由技术

许逸铭, 马礼, 傅颖勋, 李阳, 马东超

引用本文

许逸铭, 马礼, 傅颖勋, 李阳, 马东超. 一体化网络多终端接入智能路由技术[J]. 计算机科学, 2022, 49(12): 332-339.

XU Yi-ming, MA Li, FU Ying-xun, LI Yang, MA Dong-chao. [Intelligent Routing Technology for Multi-terminal Access in Integrated Network](#) [J]. Computer Science, 2022, 49(12): 332-339.

相似文章推荐 (请使用火狐或 IE 浏览器查看文章)

Similar articles recommended (Please use Firefox or IE to view the article)

[基于GAN和中文词汇网的文本摘要技术](#)

GAN and Chinese WordNet Based Text Summarization Technology

计算机科学, 2022, 49(12): 301-304. <https://doi.org/10.11896/jsjcx.210600166>

[基于最小生成树的vSDN故障快速恢复算法](#)

vSDN Fault Recovery Algorithm Based on Minimum Spanning Tree

计算机科学, 2022, 49(11A): 211200034-7. <https://doi.org/10.11896/jsjcx.211200034>

[一种面向SDN的移动网络可靠性评估算法](#)

SDN Oriented Mobile Network Reliability Evaluation Algorithm

计算机科学, 2022, 49(11A): 211000080-8. <https://doi.org/10.11896/jsjcx.211000080>

[基于双重指针网络的车货匹配双重序列决策研究](#)

Study on Dual Sequence Decision-making for Trucks and Cargo Matching Based on Dual Pointer Network

计算机科学, 2022, 49(11A): 210800257-9. <https://doi.org/10.11896/jsjcx.210800257>

[基于后状态强化学习的最优订单接受决策](#)

Optimal Order Acceptance Decision Based on After-state Reinforcement Learning

计算机科学, 2022, 49(11A): 210800261-9. <https://doi.org/10.11896/jsjcx.210800261>

一体化网络多终端接入智能路由技术

许逸铭 马礼 傅颖勋 李阳 马东超

北方工业大学信息学院 北京 100144

(17801088312@163.com)

摘要 针对一体化异构网络中大量终端设备接入网络造成网络流量剧烈波动引发的网络负载均衡问题,提出了一种基于强化学习的智能路由算法 TDANRA。TDANRA 算法通过软件定义网络技术获取细粒度、高精度的网络流量状态参数,根据网络流量状态与链路带宽利用率阈值调整机制自动生成实时的路由策略,指导网络中流量的转发,从而解决网络流量剧烈波动的问题。仿真实验结果表明,TDANRA 算法可以在大量终端设备接入网络的情况下实现网络流量的负载均衡,降低端到端传输时延与数据丢包率。

关键词: 软件定义网络;强化学习;负载均衡;智能路由;链路带宽利用率阈值调整机制

中图法分类号 TP393

Intelligent Routing Technology for Multi-terminal Access in Integrated Network

XU Yi-ming, MA Li, FU Ying-xun, LI Yang and MA Dong-chao

School of Information Science and Technology, North China University of Technology, Beijing 100144, China

Abstract Aiming at the problem of network load balancing caused by the drastic fluctuation of network traffic caused by the access of a large number of terminal devices in the integrated heterogeneous network, an intelligent routing algorithm TDANRA based on reinforcement learning is proposed. Fine-grained and high-precision network traffic status parameters are obtained by software-defined network technology, TDANRA algorithm automatically generates real-time routing policies based on network traffic status and link bandwidth utilization threshold adjustment mechanism to guide the forwarding of network traffic, so as to solve the problem of drastic fluctuation of network traffic. Simulation results show that TDANRA algorithm can realize load balancing of network traffic and reduce end-to-end transmission delay and data packet loss rate when a large number of terminal devices are connected to the network.

Keywords Software defined network, Reinforcement learning, Load balancing, Intelligent routing, Link bandwidth usage threshold adjustment mechanism

1 引言

近年来,网络中的移动终端设备逐年增多,网络流量呈指数形式增长。网络中大量终端设备接入网络,引发了网络流量在时空分布上的剧烈波动,造成网络的负载不均衡问题,导致网络拥塞^[1-4]。传统的路由策略主要是基于最短路径算法来制定,单一地将网络中的流量根据网络节点之间的最短路径进行转发,无法根据网络中流量的时空波动情况来动态地更改路由策略,易造成网络资源利用不充分和网络服务质量难以保障的问题。面对网络环境突变与复杂多样的网络资源分配需求,传统路由策略灵活性和智能化程度不足,无法根据网络状态制定动态的路由策略,难以实现整个网络的流量负载均衡。

基于机器学习的智能网络路由策略可以根据终端接入情况,动态感知网络带宽利用率等网络实时状态,从而实现动态路由,为解决以上问题提供了一种思路。然而,将机器学习技术引入网络管理仍然面临许多困难,其中一个主要原因是现有网络架构的分布式特性^[5],网络中每个节点(如路由器或交换机等)只有网络的部分视图与信息,基于不完整的网络信息来进行机器学习是非常困难的。

软件定义网络(Software Defined Networking, SDN)的发展为机器学习引入网络控制提供了解决方案。SDN 将网络的控制平面和数据平面解耦^[6],提供可编程、灵活、细粒度的流量控制能力,为实现动态、高性能的流量测量奠定了基础^[7]。研究表明,SDN 网络通过精心设计流量规则可以实现准确、及时的流量测量^[8-9]。SDN 在控制平面的控制器中

到稿日期:2021-09-06 返修日期:2022-05-12

基金项目:国家重点研发计划(2018YFB1800302);国家自然科学基金(62001007);北京市自然科学基金(KZ201810009011,4202020,19L2021)

This work was supported by the National Key Research and Development Program of China(2018YFB1800302), National Natural Science Foundation of China(62001007) and Natural Science Foundation of Beijing, China(KZ201810009011,4202020,19L2021).

通信作者:马礼(mali@ncut.edu.cn)

提供了开放接口,基于此接口,控制器可获取网络全局视图,实现对网络的集中控制,研究人员可在控制器中利用获取的网络全局信息制定智能路由策略,指导数据平面中的流量的转发,实现网络的智能化管理。由于 SDN 网络下的路由优化问题可以被看成是在给定网络环境状态下的决策问题,强化学习的优势在于解决序列决策优化问题,因此更适合使用机器学习中的强化学习来设计 SDN 网络的智能路由策略。

移动网络的迅速发展产生了大量网络终端接入网络需求,短时间内大量终端设备接入网络,造成网络流量波动与新的负载不均衡问题,本文提出了一种 TDANRA(Terminal Device Access Network Routing Algorithm)算法,并结合 SDN 网络架构与强化学习中的双延迟深度确定性策略梯度(Twin Delayed Deep Deterministic policy gradient algorithm, TD3)算法。利用 SDN 中的控制器收集实时的网络状态信息,TDANRA 算法制定动态的网络路由策略,实现了 SDN 网络中网络流量的负载均衡,降低了端到端传输时延与网络中的丢包率,提高了网络使用效率。本文的主要贡献如下:

(1)提出了基于强化学习的智能路由算法 TDANRA,解决了大量终端设备接入网络造成的网络流量波动与负载不均衡问题。

(2)在 TDANRA 算法中,通过调整算法的奖励函数中的权重系数,实现不同侧重(如低时延、低丢包率等)的网络负载均衡需求。

(3)提出了链路带宽利用率阈值调整机制,动态调整网络中链路的带宽利用率阈值,解决了终端设备接入网络所造成的链路带宽利用率上升问题,降低了网络中的拥塞概率。

2 相关工作与背景

当面对复杂多样的网络资源分配需求时,传统路由机制在灵活性和智能化程度上仍然不足^[10],因此,设计能够适应网络实时变化的路由策略非常重要。现有的解决方案主要分为启发式算法与机器学习算法两种。

2.1 启发式算法路由策略

Naeem 等^[11]提出了一种高效并行路由优化方案(Qos-enabled Efficient Parallel Routing Optimization, SEQOS),该方案考虑 3 种 QoS 要求,即丢包率、延迟和抖动,设计了一种贪婪启发式算法,利用 GPU 计算最优转发路径,可最大限度地利用网络资源。Deng 等^[12]将流动态划分为不同的优先级,提出应用感知 QoS 路由算法(Application-aware QoS Routing Algorithm, AQRA),根据每个物联网应用的 QoS 要求(时延、抖动、丢包率),自适应调整模拟退火算法中代价函数中的权重,计算出一组路由路径,保证了高优先级物联网应用的 QoS 要求,实现了物联网的负载均衡。Tang 等^[13]针对云大数据中心流调度问题,提出了一种动态负载均衡启发式调度方法,将拥塞最严重的链路上占用最大带宽的流动态迁移到拥塞程度最轻的链路上,通过动态均衡数据流来最大化网络吞吐量,保证所有链路负载均衡。然而上述方法基于特定模型场景,算法的适用范围有限。

2.2 机器学习路由策略

Guo 等^[14]针对流量工程(Traffic Engineering, TE)问题,提出了一种强化学习算法 ROAR(Routing Optimization with Reinforcement learning),ROAR 可建立流量需求和分流策略之间的直接关系,快速确定新的满足流量需求的路由方案,实现 SDN 的链路负载均衡。考虑到数据中心网络中多种网络资源(带宽、缓存)共存,Liu 等^[15]提出了基于网络资源重组的路由算法 DRL-R(Deep Reinforcement Learning-based Routing),该算法为流量优化分配网络资源,根据网络状态自适应地执行合理的路由,提高网络吞吐量,DRL-R 中缓存是影响路由决策的一个重要因素。Yao 等^[16]提出基于强化学习 DQN 算法的 NetworkAI 架构,NetworkAI 可自主学习到一个最佳路由策略以最小化网络的时延,然而该架构需要为网络中每一对“源-目的”交换机运行专用的强化学习 agent,因此不适用于解决大规模网络的负载不均衡问题。Sun 等^[17]将牵制控制理论与强化学习相结合,提出了名为 ScaleDRL 的路由方案,其中控制理论帮助深度强化学习减少了通信网络中需要的控制信号。ScaleDRL 利用控制理论的思想选择网络中的链路子集,并将其命名为关键链路,根据 SDN 控制器收集到的流量分布信息,ScaleDRL 使用强化学习算法对关键链路动态调整链路权值,实时调控网络流量的转发路径,降低网络中流量的传输时延。Chen 等^[18]基于强化学习提出了一种智能路由算法,用于解决 SDN 网络在吞吐量与时延方面的流量工程问题,该算法可以向上或者向下优化网络吞吐量,但需要为网络中每一台交换机部署 agent,且受限于计算资源,因此适用范围较小。

2.3 强化学习 TD3 算法

TD3 算法^[19]使用强化学习中的 Actor-Critic 算法架构。Actor 根据当前环境状态输出动作,其本质为策略函数,TD3 算法使用深度神经网络 $\pi_0(s_t)$ 作为策略函数;Critic 基于价值函数 $Q(s_t, a_t)$ 评价 Actor 在状态 s_t 下输出动作 a_t 的优劣,价值函数输出结果被称为 Q 值;智能体通过 Q 值指导 Actor 策略神经网络的更新,TD3 算法中使用神经网络 $Q_\theta(s_t, a_t)$ 来拟合价值函数。

Critic 使用梯度下降法来更新神经网络参数,其损失函数为均方误差损失函数:

$$L(\theta) = \mathbb{E}[(y_t - Q_\theta(s_t, a_t))^2] \quad (1)$$

其中, y_t 为目标 Q 值,目标 Q 值由时间差分(Temporal difference, TD)^[20]法计算而来,时间差分法基于 Bellman 方程,Bellman 方程体现的是当前时间的状态-动作的价值与下一时间状态-动作的价值之间的关系:

$$y_t = r(s_t, a_t) + \gamma \min_{i=1,2} Q_{\theta_i'}(s_{t+1}, a_{t+1}) \quad (2)$$

Actor 神经网络由确定性策略梯度算法(Deterministic Policy Gradient Algorithms, DPG)^[21]来更新网络参数:

$$\nabla_\theta J(\theta) = \mathbb{E}_{s \sim p_\pi} [\nabla_a Q_\theta(s, a) |_{a=\pi_\theta(s)} \nabla_\theta \pi_\theta(s)] \quad (3)$$

在基于价值的强化学习算法中,最常见的问题是过度估计 Q 值所引发的高偏差与高方差问题。TD3 算法使用两个独立的 Critic 神经网络 Q_{θ_1} 和 Q_{θ_2} 来计算 Q 值,取两个神经网络的目标 Q 值中的最小值来更新 Critic 神经网络参数,以

$$\text{s. t. } 0 < u_i(e_{i,j}) < 100\%, \forall e_{i,j} \in E \quad (10)$$

$$0 \leq l_i(p_{i,j}) \leq 100\%, \forall p_{i,j} \in sw_i \quad (11)$$

式(7)~式(9)为本文的优化目标,即在保证网络中流量顺利转发的前提下,合理分配网络中的流量转发路由路径,最小化网络中的链路带宽利用率,使网络可承载更多网络流量,降低端到端时延以及交换机的丢包率;式(10)限制每条链路的利用率不超过100%;式(11)限制交换机端口的丢包率上限为100%,下限为0。

4 算法架构

对于大量终端设备接入网络这一场景,综合考察链路带宽利用率、时延以及丢包率等网络性能指标后,利用强化学习、软件定义网络等技术,以期改进网络性能。本文针对大量终端设备接入网络这一问题提出了一种链路带宽利用率阈值调整机制,基于强化学习 TD3 算法提出 TDANRA 算法。本节介绍 TDANRA 算法的相关定义以及算法流程。

4.1 算法相关定义

强化学习(Reinforcement Learning, RL)是受到生物能够有效适应环境的启发,以试错机制让智能体(Agent)与环境(Environment)进行交互,其最终目标是学习到一个最优策略来最大限度地提高累积奖励^[21]。具体来说,在每一个时间间隔 Δt ,强化学习智能体观测到当前环境状态 s ,根据策略 $\pi: S \rightarrow A$ 输出动作 a ,在环境中执行动作 a 后,环境状态由 s 变为 s' ,同时反馈给智能体并奖励 r ,智能体根据奖励 r 来优化更新策略 π ,以最大化累积奖励的期望 $R_t = \sum_{i=t}^T \gamma^{i-t} r(s_i, a_i)$,其中 $\gamma \in [0, 1]$ 是折扣因子,决定短期奖励与长期奖励的优先重要性。

根据本文所要解决的问题,强化学习中的状态、动作与奖励的具体定义如下。

(1)状态:控制器在 t 时刻进行一次网络状态测量,获得当前网络中各链路的带宽利用率、交换机之间当前路由路径的时延以及交换机各端口的丢包率,这些数据是对当前网络状态的描述,也是 TD3 算法的输入参数,网络状态可表示为:

$$\begin{aligned} s_t = & [u_i(e_{1,2}), u_i(e_{1,3}), \dots, u_i(e_{i,j}) \\ & d_i(path_{1,2}), d_i(path_{2,3}), \dots, d_i(path_{i,j}) \\ & l_i(sw_1), l_i(sw_2), \dots, l_i(sw_i) \\ & v_i(e_{1,2}), v_i(e_{2,3}), \dots, v_i(e_{i,j})] \end{aligned} \quad (12)$$

其中, $u_i(e_{i,j})$ 为 t 时刻链路 $e_{i,j}$ 的带宽利用率, $d_i(path_{i,j})$ 表示 t 时刻交换机 sw_i 与交换机 sw_j 之间路由路径的时延, $l_i(sw_i)$ 为 t 时刻交换机 sw_i 的丢包率, $v_i(e_{i,j})$ 为 t 时刻交换机 sw_i 链路带宽利用率阈值。

(2)动作:在网络路由决策问题中,强化学习的动作直接或者间接决定了网络中流量的转发。本文中,强化学习动作设定为网络中每条链路权重, $a_t = [\omega(e_{1,2}), \omega(e_{1,3}), \dots, \omega(e_{i,j})]$ 。在算法开始训练前,网络中每条链路的权重初始值均置为1。

(3)奖励:强化学习中的智能体根据奖励来评估动作的优劣,进而改进策略。根据本文研究的问题以及优化目标,将

链路带宽利用率、时延、丢包率作为计算奖励的依据。奖励计算公式如下:

$$\begin{aligned} r = & \alpha \sum_{i=1}^I \sum_{j=1}^J [v(e_{i,j}) - u(e_{i,j})] - \beta \sum_{i=1}^I \sum_{j=1}^J d(path_{i,j}) - \\ & \gamma \sum_{i=1}^I \sum_{j=1}^J l(p_{i,j}), 0 \leq v(e_{i,j}) \leq 80\% \end{aligned} \quad (13)$$

其中, $v(e_{i,j})$ 代表链路 $e_{i,j}$ 带宽利用率的阈值,初始值设为80%,在5.2节中将通过实验验证链路带宽利用率阈值初始值的设置。

当网络中链路的带宽利用率超过阈值亦或者端到端时延或丢包率过大时,奖励值就会下降。因强化学习的目的为最大化累计奖励,当奖励值下降时,TDANRA 算法会调整路由策略,减少流经负载过高链路的流量,降低链路带宽利用率、端到端时延与交换机丢包率,实现网络负载均衡。仿真实验中,时延的数值与链路带宽利用率、交换机丢包率的数值相差较大,式(13)中时延的权重较大,因此在 TDANRA 算法根据奖励更新神经网络参数时,算法对时延敏感。由于链路带宽利用率与丢包率的权重占比小,更新后的 TDANRA 算法会导致网络低时延、高链路带宽利用率、高丢包率的结果。式(13)中 α, β, γ 为权重系数,可通过权重系数实现调整链路带宽利用率、时延、丢包率在奖励中的权重比,TDANRA 算法根据奖励更新神经网络后可实现整体网络的低链路带宽利用率、低时延与低丢包率。

4.2 链路带宽利用率阈值调整机制

针对网络中大量终端设备接入引发的流量剧烈波动,本文设计了一种链路带宽利用率阈值调整机制,以提高算法收敛速度,加速实现终端设备接入的网络区域流量负载均衡。

利用 SDN 控制器的集中控制优势,当终端设备接入、断开网络时,控制器可快速获取终端设备的信息,如终端设备所连接的交换机 sw_i 信息、 sw_i 的邻接链路集 $E(sw_i)$ 等。当控制器监听到终端设备接入或断开网络时,TDANRA 算法会减小或提高邻接链路集 $E(sw_i)$ 的带宽利用率阈值。

终端设备接入交换机 sw_i 后,与网络中其他的终端设备进行数据交互,流经链路集 $E(sw_i)$ 的网络流量会增多,即 $E(sw_i)$ 的带宽利用率升高,当 agent 根据式(13)计算奖励 r 时,奖励 r 就会因链路带宽利用率升高而减少。因强化学习的最终目标是最大限度地提高累积奖励,在 agent 根据奖励 r 更新策略后,输出动作 a ,即网络中链路的权重值,邻接链路 $E(sw_i)$ 的权重会减小,间接减少邻接链路集 $E(sw_i)$ 的转发流量,降低 $E(sw_i)$ 的带宽利用率,从而提高策略更新后的奖励计算值 r ,实现网络终端设备接入的网络区域流量的负载均衡。

当终端设备与交换机 sw_i 断开后,链路集 $E(sw_i)$ 的网络流量会相对减少,即 $E(sw_i)$ 的带宽利用率降低,这时相对提高 $E(sw_i)$ 的带宽利用率阈值就会增大奖励值 r 。基于同样的思路,agent 在更新策略后,邻接链路 $E(sw_i)$ 的权重会上升,增加邻接链路集 $E(sw_i)$ 转发流量,使链路的带宽利用率维持在稳定状态。

经过多次交互,agent 学习到一个策略:当终端设备接入网络时,就降低邻接链路集 $E(sw_i)$ 的权重值,以减少流经的

链路集 $E(sw_i)$ 的流量,避免链路拥塞;反之,当终端设备断开连接时,提高 $E(sw_i)$ 的权重值,使链路带宽利用率保持稳定,实现网络的负载均衡。根据实验结果,将阈值 $v(e_{i,j})$ 的单次降低值设置为 0.5%。相对地,当终端设备与交换机断开连接时,邻接链路 $E(sw_i)$ 阈值需上调 0.5%。

4.3 DN 中的 TDANRA 算法

根据以上定义,TDANRA 算法的训练过程概括如下:TDANRA 算法从 SDN 控制器获取网络状态参数 s ,TDANRA 算法中的 Actor 神经网络以网络状态参数 s 作为输入,输出动作 a ,即网络中各条链路的权重值;控制器在获取链路权重之后使用 Floyd 算法来计算网络中各交换机之间的加权最短路径,以此指导网络中流量的转发;路由路径更新后,通过下一次网络测量得到新的网络状态 s' ,计算奖励 r ;Actor 神经网络根据式(3)更新网络参数,Critic 神经网络根据式(1)和式(2)进行参数更新,目标 Actor 神经网络与目标 Critic 神经网络参数通过延迟策略更新技术进行更新。

当网络中终端设备接入时,通过链路带宽利用率阈值调整机制来改变奖励 r 的计算方式,间接指导 TDANRA 算法中神经网络的更新,降低因终端设备接入网络而引发的拥塞概率,实现网络中流量的负载均衡。网络中某一条网络链路的带宽利用率大于阈值 80%时,中断当前训练轮次,计算本轮训练的累计奖励值。奖励值与网络中链路带宽利用率、端到端时延、丢包率相关,通过每一训练轮次的累计奖励值,可判断 TDANRA 算法的收敛情况,以及网络负载均衡程度。TDANRA 算法的整体流程如算法 1 所示。

算法 1 TDANRA 算法

输入:网络状态信息

输出:网络中各链路的权重值

1. 初始化 critic 神经网络 Q_{θ_1} , Q_{θ_2} , actor 神经网络 π_{θ_0} ;
2. 初始化 critic, actor 神经网络对应的目标神经网络 $\theta_1' \leftarrow \theta_1$, $\theta_2' \leftarrow \theta_2$, $\theta' \leftarrow \theta$;
3. 初始化经验缓存区 B;
4. for episode=1 to M do;
5. for t=1 to T do;
6. if 终端设备接入交换机 sw_i ;
7. $v(e_{i,j}) = v(e_{i,j}) - 0.5\%$, $e_{i,j} \in E(sw_i)$
8. if 终端设备与交换机 sw_i 断开;
9. $v(e_{i,j}) = v(e_{i,j}) + 0.5\%$, $e_{i,j} \in E(sw_i)$
10. 收集网络状态信息 s_t ;
11. 计算动作并添加噪声 $a_t = \pi_{\theta_0}(s_t) + \theta$
12. 使用 Floyd 算法计算路由策略;
13. 依据新路由策略指导网络中流量转发;
14. 收集新网络状态信息 s_{t+1} , 计算奖励 r ;
15. 在 B 中保存 (s_t, a_t, r_t, s_{t+1}) ;
16. 在 B 中随机选取大小为 N 的训练数据集 (s_j, a_j, r_j, s_{j+1}) ;
17. $a_{j+1} = \pi_{\theta'}(s_{j+1}) + \epsilon$, $\epsilon \sim \text{clip}(\mathcal{N}(0, \sigma), -c, c)$
18. $y = r_j + \gamma \min_{i=1,2} Q_{\theta_i'}(s_{j+1}, a_{j+1})$
19. 更新参数 $\theta_i = \min_{\theta_i} N^{-1} \sum (y - Q_{\theta_i}(s_j, a_j))^2$
20. if t mod d then;
21. 通过 deterministic policy gradient 算法更新参数 θ ;

$$22. \quad \nabla_{\theta} J(\theta) = N^{-1} \nabla_a Q_{\theta}(s, \pi_{\theta}(s)) \nabla_{\theta} \pi_{\theta}(s);$$

23. 更新目标神经网络参数:

$$24. \quad \theta_i' \leftarrow \tau \theta_i + (1-\tau) \theta_i', \theta' \leftarrow \tau \theta + (1-\tau) \theta'$$

25. end if

26. if $\exists u(e_{i,j}) > 80\%$;

27. break;

28. end if

29. end for

30. end for

5 实验与分析

5.1 实验环境

实验采用的计算机 CPU 为 AMD Ryzen 7 4800 H, 内存为 16 GB, 操作系统为 Ubuntu 18.04, 实验语言平台环境为 Python 2.7.17。TDANRA 算法基于 PyTorch 实现。本实验利用 Mininet 网络仿真平台搭建 SDN 网络环境, 采用开源的 Ryu 控制器作为整个实验网络的 SDN 控制器。使用 Iperf 流生成工具模拟网络大规模流量, ping 命令模拟网络小规模流量。

仿真实验的网络拓扑结构如图 2 所示, 包含 14 个节点与 21 条链路, 每条链路的带宽设为不同数值。Mininet 中可通过命令行向网络中添加主机, 实验过程中每隔 5 s, 新增 10 个终端设备, 每个终端设备接入网络中不同 SDN 交换机, 利用 Mininet 这一功能来模拟终端设备接入网络。随机断开终端设备与网络的连接, 模拟终端设备断开网络。

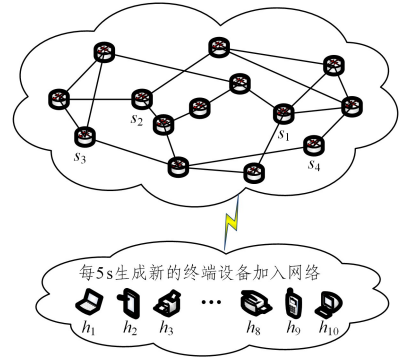


图 2 仿真实验网络拓扑

Fig. 2 Simulation experiment network topology

5.2 实验结果分析

为验证 TDANRA 算法的性能, 本文通过仿真实验与其他算法进行对比, 性能评价指标为: 交换机吞吐量、链路带宽利用率、端到端时延以及丢包率。对比算法包括: 1) 开放最短路径优先算法 (Open Shortest Path First, OSPF); 2) TD3 算法, TD3 算法不使用链路带宽利用率阈值调整机制, 其神经网络结构与 TDANRA 算法相同。

图 3 记录了网络拓扑中交换机的吞吐量。实验中各终端设备数据流发送速率为固定的 100 Mbps, 选取网络拓扑中的 4 个交换机 s_1, s_2, s_3, s_4 , 记录在终端设备接入网络造成网络波动的情况下各交换机的吞吐量。

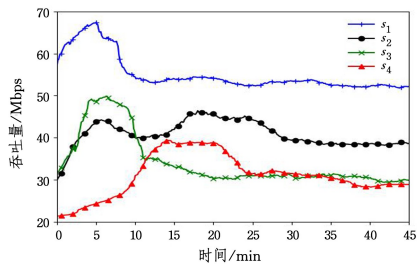


图 3 交换机吞吐量
Fig. 3 Switch throughput

图 3 中,实验前期各交换机的吞吐量因终端设备接入显示出明显的波动;实验中后期,TDANRA 算法经过一段时间训练后,通过学习到的经验更新网络中各链路的权重值,以改变网络中各交换机间路由路径,引导网络中流量向负载率较低的链路流动,使网络中交换机的吞吐量维持在一个相对稳定的水平。由图 3 可以看出,实验中后期 4 个交换机的吞吐量趋于平稳,说明 TDANRA 算法可解决网络流量波动问题,实现网络中流量负载均衡。

图 4 给出了 TDANRA 算法、TD3 算法在训练过程中累积奖励 reward 的增长情况。实验的训练周期 (episode) 设为 300。实验过程中,终端设备接入网络所引发的流量波动造成训练中断,使得累积奖励值曲线波动较大。图中红色曲线为 TDANRA 算法的累积奖励值曲线,通过对比可得出结论:与 TD3 算法相比,本文提出的 TDANRA 算法收敛速度更快,负载均衡效果更好,TDANRA 算法的链路带宽利用率阈值调整机制有利于网络在大量终端设备接入的情况下更加高效地实现网络流量的负载均衡。

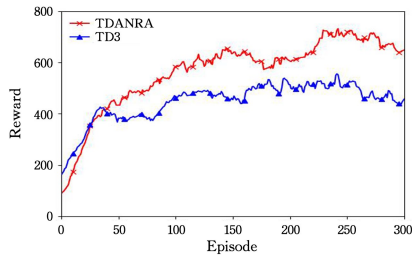


图 4 累积奖励值对比(电子版为彩图)
Fig. 4 Cumulative reward value comparison

图 5 给出了终端设备的数据流传输速率增长情况下各算法在端到端时延性能方面的对比结果。

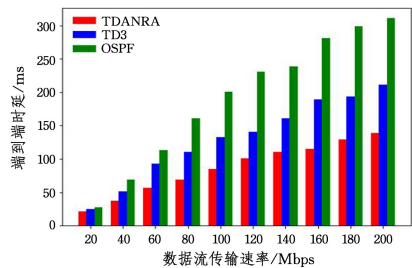


图 5 端到端时延
Fig. 5 End-to-End delay

由图 5 可以看出,在数据流传输速率增长的情况下,

TDANRA 算法可满足网络的低时延需求,且在终端设备接入网络的情况下,TDANRA 算法因链路带宽利用率阈值调整机制的作用,其时延相比 TD3 算法减少了 12%;OSPF 算法将网络中数据流单一地通过最短路径转发,因路由路径的传输能力限制,端到端时延随着数据流发送速率增长而快速增加。

本文在实验过程中搜集各交换机丢包率,计算网络的平均丢包率。图 6 给出了不同数据流传输速率下各算法的平均丢包率,在低数据流传输速率的情况下,3 种算法的平均丢包率都较低,随着数据流传输速率的增加,因单一路径的带宽限制以及网络中 SDN 交换机的端口流量转发能力限制,OSPF 算法易引发网络中流量拥塞,进而导致交换机端口丢包率升高。由图 6 可知,OSPF 算法平均丢包率增长幅度最大。TDANRA 算法与 TD3 算法根据实时网络状态动态更改网络链路权重,为网络中流量计算出带宽利用率低的路由路径,进而转发网络中的流量,降低网络中流量拥塞的发生概率与各交换机的丢包率。因 TDANRA 算法中的链路带宽利用率阈值调整机制,TDANRA 算法对网络中终端设备接入更为敏感,它会有针对性地调整终端设备接入链路的带宽利用率阈值,降低流经终端设备接入链路的流量,避免流量拥塞的发生。因此在终端设备接入网络的情况下,TDANRA 算法的平均丢包率低于 TD3 算法的平均丢包率。

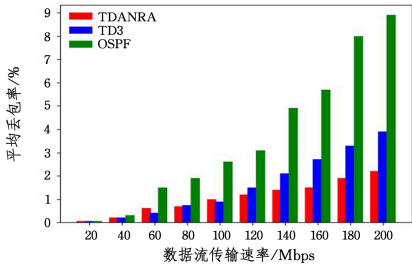


图 6 平均丢包率
Fig. 6 Average packet loss rate

图 7 利用箱型图展示 3 种算法在不同数据流传输速率下的链路带宽利用率,箱体中的点代表链路带宽利用率的均值,箱线图表示数据的分布情况,箱线图的上、下横线表示链路带宽利用率的最大值以及最小值。由图 7(a)可知,因网络中的流量由单一路径转发,OSPF 算法的链路带宽利用率分布极不均匀,且 OSPF 算法的链路带宽利用率上升幅度明显高于其他算法;由图 7(b)和图 7(c)可以看出,相比 OSPF 算法,TDANRA 算法与 TD3 算法可根据网络实时状态动态更新网络中各交换机间的路由,将各链路的带宽利用率维持在一个稳定的水平,因此两种算法的链路带宽利用率分布均匀,链路带宽利用率增长幅度较小。其中 TDANRA 算法性能表现更好,在链路带宽利用率阈值调整机制的指导下,TDANRA 算法减少终端接入链路集的权值,降低流经终端接入链路集的流量,从而维持终端接入链路集的带宽利用率的稳定。由图 7 可知,在终端设备接入的情况下,TDANRA 算法的链路带宽利用率相比 TD3 算法减少了 7.3%。

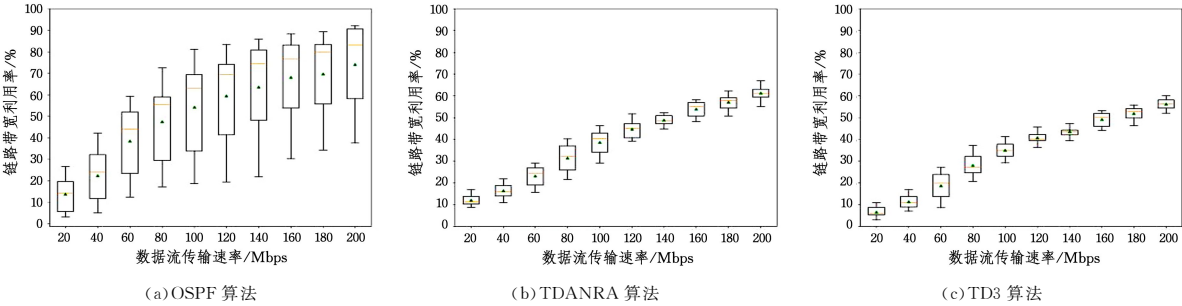


图 7 不同算法平均链路带宽利用率对比

Fig. 7 Comparison of average link bandwidth utilization of different algorithms

本文设计的链路带宽利用率阈值调整机制中,有一个重要因素就是链路带宽利用率初始阈值。通过实验探讨不同的初始阈值对 TDANRA 算法的影响,实验中数据流传输速率为 200 Mbps,算法训练周期皆为 300。图 8 给出不同链路带宽利用率初始阈值在链路带宽利用率、端到端时延、平均丢包率这 3 个评价指标上的表现。图 8(a)通过箱型图展示了不同链路带宽利用率初始阈值对链路带宽利用率的影响。当初始阈值为 80% 时,网络中各链路带宽利用率分布均匀,且其

链路带宽利用率均值与其他链路带宽利用率初始阈值的均值相比为最低。图 8(b)展示了不同链路带宽利用率初始阈值对端到端时延这一指标的影响。当初始阈值为 80% 时,网络中的时延为 129 ms,相比其他初始阈值,其时延最低。由图 8(c)可知,当链路带宽利用率初始阈值为 80% 时,网络中的平均丢包率最低,为 2.4%。通过实验数据可以得出,当链路带宽利用率初始阈值为 80% 时,TDANRA 算法在链路带宽利用率、端到端时延、平均丢包率这 3 个指标上表现最好。

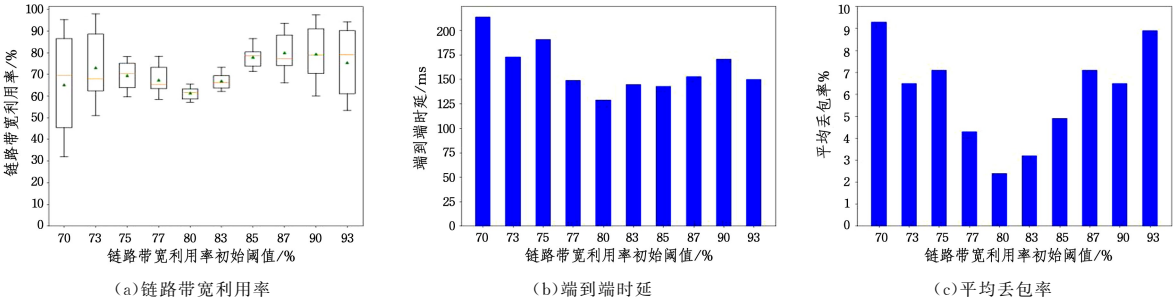


图 8 不同链路带宽利用率初始阈值对比

Fig. 8 Comparison of different initial thresholds of link bandwidth utilization

结束语 本文为解决网络中终端设备接入网络所造成的负载不均衡问题,提出了一种结合强化学习与软件定义网络技术的 TDANRA 算法,通过软件定义网络技术获取网络状态参数,借助强化学习的神经网络生成实时的动态流量路由策略,并且设计一种链路带宽利用率阈值调整机制加快 TDANRA 算法的收敛速度。通过仿真实验证明,对于终端设备接入网络所引发的负载不均衡问题,TDANRA 算法的性能表现比其他对比算法更好,可有效降低网络中的端到端时延和丢包率,使网络中各链路的带宽利用率保持均衡,减少网络拥塞的发生。

本文研究了终端设备接入网络所带来的负载不均衡问题,并给出了一种解决方案。对于网络设备动态接入网络造成拓扑变化这一情况,本文尚未涉及。在未来的工作中,我们计划研究网络拓扑动态变化时负载不均衡问题的解决方案,并通过实验验证其可行性。

参 考 文 献

[1] FOOLADIVANDA D,ROSENBERG C. Joint Resource Allocation and User Association for Heterogeneous Wireless Cellular Networks[J]. IEEE Transactions on Wireless Communications, 2013,12(1):248-257.

[2] WANG G,ZHONG Y,HAN T, et al. On the Effect of Spatio-Temporal Fluctuation of Traffic in Wireless Networks[C] // IEEE Global Communications Conference, GLOBECOM 2018. Abu Dhabi, United Arab Emirates: IEEE, 2018: 1-7.

[3] TEFERA G,SHE K,CHEN M, et al. Congestion-aware adaptive decentralised computation offloading and caching for multi-access edge computing networks [J]. IET Communications, 2020,14(19):3410-3419.

[4] YU S,CHEN X,ZHOU Z, et al. When Deep Re-inforcement Learning Meets Federated Learning: Intelligent Multitimescale Resource Management for Multiaccess Edge Computing in 5G Ultradense Network [J]. IEEE Internet of Things Journal, 2021,8(4):2238-2251.

[5] ZORZI M,ZANELLA A,TESTOLIN A, et al. COBANETS: A new paradigm for cognitive communications systems[C] // 2016 International Conference on Computing, Networking and Communications, ICNC 2016. Kauai, HI, USA: IEEE Computer Society, 2016: 1-7.

[6] MCKEOWN N,ANDERSON T E,BALAKRISHNAN H, et al. OpenFlow: enabling innovation in campus networks [J]. Com-

- puter Communication Review,2008,38(2):69-74.
- [7] WANG X,DENG Q,REN J,et al. The Joint Optimization of Online Traffic Matrix Measurement and Traffic Engineering For Software-Defined Networks[J]. IEEE/ACM Transactions on Networking,2020,28(1):234-247.
- [8] MOSHREF M,YU M,GOVINDAN R,et al. DREAM:dynamic resource allocation for software-defined measurement[C]//Proceedings of the 2014 ACM Conference on SIGCOMM(SIGCOMM '14). Association for Computing Machinery,2014:419-430.
- [9] MALBOUBI M,WANG L,CHUAH C N,et al. Intelligent SDN based traffic(de) Aggregation and Measurement Paradigm(iSTAMP)[C]//IEEE Conference on Computer Communications. 2014:934-942.
- [10] AKYILDIZ I F,LEE A,WANG P,et al. Research challenges for traffic engineering in software defined networks[J]. IEEE Network,2016,30(3):52-58.
- [11] NAEEM F,TARIQ M,POOR H V. SDN-Enabled Energy-Efficient Routing Optimization Framework for Industrial Internet of Things[J]. IEEE Transactions on Industrial Informatics,2021,17(8):5660-5667.
- [12] DENG G,WANG K. An Application-aware QoS Routing Algorithm for SDN-based IoT Networking[C]//2018 IEEE Symposium on Computers and Communications, ISCC 2018. Natal, Brazil;IEEE,2018:186-191.
- [13] TANG F,YANG L T,TANG C,et al. A Dynamical and Load-Balanced Flow Scheduling Approach for Big Data Centers in Clouds[J]. IEEE Transactions on Cloud Computing,2018,6(4):915-928.
- [14] GUO Y,WANG W,ZHANG H,et al. Traffic Engineering in hybrid Software Defined Network via Reinforcement Learning[J]. Journal of Network and Computer Applications,2021,189:103116.
- [15] LIU W X,CAI J,CHEN Q C,et al. DRL-R:Deep reinforcement learning approach for intelligent routing in software-defined data-center networks[J]. Journal of Network and Computer Applications,2021,177:102865.
- [16] YAO H,MAI T,XU X,et al. NetworkAI: An Intelligent Network Architecture for Self-Learning Control Strategies in Software Defined Networks[J]. IEEE Internet of Things Journal,2018,5(6):4319-4327.
- [17] SUN P,GUO Z,LAN J,et al. ScaleDRL: A Scalable Deep Reinforcement Learning Approach for Traffic Engineering in SDN with Pinning Control [J]. Computer Networks, 2021, 190: 107891.
- [18] CHEN Y R,REZAPOUR A,TZENG W G,et al. RL-Routing: An SDN Routing Algorithm Based on Deep Reinforcement Learning[J]. IEEE Transactions on Network Science and Engineering,2020,7(4):3185-3199.
- [19] FUJIMOTO S,HOOFF H V,MEGER D. Addressing Function Approximation Error in Actor-Critic Methods[C]//Proceedings of the 35th International Conference on Machine Learning,ICML 2018. 2018:1582-1591.
- [20] SUTTON R S. Learning to Predict by the Methods of Temporal Differences[J]. Machine Learning,1988,3:9-44.
- [21] SILVER D,LEVER G,HEESS N,et al. Deterministic Policy Gradient Algorithms[C]//Proceedings of the 31th International Conference on Machine Learning, ICML 2014. 2014, 32: 387-395.
- [22] ZHANG J,YE M,GUO Z,et al. CFR-RL: Traffic Engineering With Reinforcement Learning in SDN[J]. IEEE Journal on Selected Areas in Communications,2020,38(10):2249-2259.



XU Yi-ming, born in 1996, postgraduate. His main research interests include software defined network and reinforcement learning.



MA Li, born in 1968, Ph.D, professor, is a distinguished member of China Computer Federation. His main research interests include advanced computing technology, distributed computing, multi-agent system and IoT system.

(责任编辑:何杨)