

## 基于流量预测的命名数据网络请求端拥塞控制方法

于泽洋, 常成, 李彤

陆军装甲兵学院信息通信系 北京

**【摘要】** 本文旨在为命名数据网络 (Named Data Network, NDN) 设计一种基于调整请求端兴趣包发送窗口的拥塞控制方法。文中首先介绍了目前 NDN 中拥塞控制方法的研究成果, 对请求端拥塞控制和网内节点拥塞控制两种思路的优劣进行了分析, 然后提出了一种基于机器学习流量预测的请求端拥塞控制方案设计, 并在 NDN 网络仿真平台-ndnSIM 上进行了实验验证。实验结果表明, 与传统的请求端拥塞控制相比, 新方法能够提高网络吞吐量、降低丢包率。

**【关键词】** 控制方法; 信息中心网络; 分析

### Requesting side congestion control method for Named Data Network based on flow prediction

Zeyang Yu, Cheng Chang, Tong Li

Armored Force Academy Fengtai District, Beijing

**【Abstract】** This paper aims to design a congestion control method for Named Data Network (NDN) based on adjusting the interest packet sending window of the requesting side. We first introduce researches on congestion control of NDN, analyze and compare two ideas that do congestion control on requesting side or metric nodes in the network. And then we put forward an end congestion control method based on machine learning traffic prediction, and do simulation experiment on the NDN simulation platform - ndnSIM. Experimental result shows that the proposed method can improve network throughput and reduce packet loss rate compared with traditional requesting side congestion control.

**【Keywords】** control method; information center network; analysis

### 1 引言

#### 1.1 命名数据网络

TCP/IP 协议架构的互联网作为众所周知的互联网已经被使用了数十年之久, 已经成为了人类社会中不可或缺的一环。但是随着网络架构日趋复杂、网络应用更加广泛、网络资源膨胀式增长以及人们有了更高的安全性和移动性的要求, 其以寻找内容源存储终端的位置为核心的端到端的传输模式具有很多先天性的不足。相对于获取信息存储的位置, 更加关注信息本身的信息中心网络 (Information-Centric Networking, ICN) 应运而生。在 ICN 中, 整个网络中各个节点与程序在各种信息的请求和应答行为的驱动下运行, 用户或应用程序只直接关注

信息数据本身, 而与该信息块的其他属性 (例如存储位置) 无关<sup>[1]</sup>。

NDN 是 ICN 网络近年发展较为火热的一种典型实现形式。在 NDN 中, 信息 (数据) 不再与主机地址进行绑定, 而是通过名称标识每一个信息。信息的获取由接收端采用兴趣包基于信息名称发起, 路由器基于信息名称路由转发兴趣包, 而后当兴趣包到达拥有与其名称前缀匹配的数据包的节点后, 数据包会按照兴趣包的“来路”原路返回。此外, NDN 路由器增加信息存储模块, 使得信息分布式存放在网络中, 用户可以从距离自己最近的节点获取所需信息。

命名数据网络由于具有网内缓存的特点, 即消

第一作者简介: 于泽洋 (1993-) 男, 山东平度, 硕士, 研究方向: 计算机技术

第二作者简介: 常成 (1986-) 男, 河南郑州, 硕士, 讲师, 研究方向: 军事通信

第三作者简介: 李彤 (1964-) 男, 河南平顶山, 博士后, 教授, 研究方向: 信息与通信工程、装备保障、信息安全

费者请求的数据不仅仅能够从生产者处获取, 也可以从网内缓存获取, 因此尽管拥塞控制在 TCP/IP 网络中是一个发展较为完善的领域, 但是基于确定的端到端的拥塞控制算法在 NDN 中不再适用<sup>[2]</sup>。如何设计一种符合 NDN 网络架构特点的拥塞控制机制, 能够在充分利用带宽的前提下保证网内不会因为队列溢出产生丢包成为该网络架构走向实用阶段的亟待解决的问题。

### 1.2 神经网络流量预测

近年来, 机器学习在各行业领域的应用发展迅速, 特别是在搜索、图像识别、语音语义识别理解和情感分析等方面已成为主流的热门的技术。特别是卷积神经网络在计算机视觉上取得了巨大的成就, 已经能够从高维度的图片数据中提取有用的信息, 进行分类或其他分析工作。尤其是在网络信息处理和大数据领域, 深度学习技术有广泛的研究空间, 蕴含着巨大的应用潜力。

网络流量的特点包括线性特性、长相关性、自相似和突发性, 并且其长相关性随着时间的变化会表现的十分明显。卷积神经网络在具有上下文依赖性的时序问题上效果不尽人意。虽然理论上其仍可以通过简单的增大卷积核尺寸增大感受视野来解决这一问题, 但是在实际应用中, 计算次数和参数个数会随卷积核尺寸的平方增长, 极大的增加了计算量和模型优化的难度。相比之下, 要想更有效的进行网络流量预测, 更适合使用专门处理时序问题的

循环神经网络。而在众多循环神经网络模型及其变体类型中, RNN 由于不可避免的梯度消失和梯度爆炸问题而实际应用较少。应用较多的是它的两种变体形式: 长短期记忆网络 (Long-Short Term Memory, LSTM) 和门控循环网络 (Gate Recurrent Unit, GRU)<sup>[3]</sup>。前者参与计算的参数更多, 预测较为精确; 后者实际上是前者的一种简化形式, 参与计算的参数较少, 收敛快, 适合处理更庞大的数据集。对于网络流量来说, 由于本身已经是数字类型的数据, 无需进行如汉字的 Word2vec 或独热编码等特殊编码形式, 数据集维度较低, 不会非常庞大, 因此本文选择 LSTM。

## 2 相关研究

### 2.1 多维度特征值的 NDN 流量预测

在总结使用机器学习进行流量预测的相关工作的基础上, 结合 NDN 自身特点, 我们前期研究了一种基于 LSTM 的命名数据网络流量预测方法。使用链路上游接口当前时间段内收到的兴趣包量、链路上游接口当前时间段内排队的数据包量、链路上游接口当前时间段内发送的数据包量和当前时间段内链路上的总流量 (真实流量加上丢包量) 的二维特征值来代替以往只使用链路单位时间总流量作为单一特征值的预测方式。对应的标签为下一时刻链路总流量。经 ndnSIM 仿真实验验证, 与传统的单一特征值的预测方法相比, 该方法拥有更高的准确率。实验结果如图 1 所示。

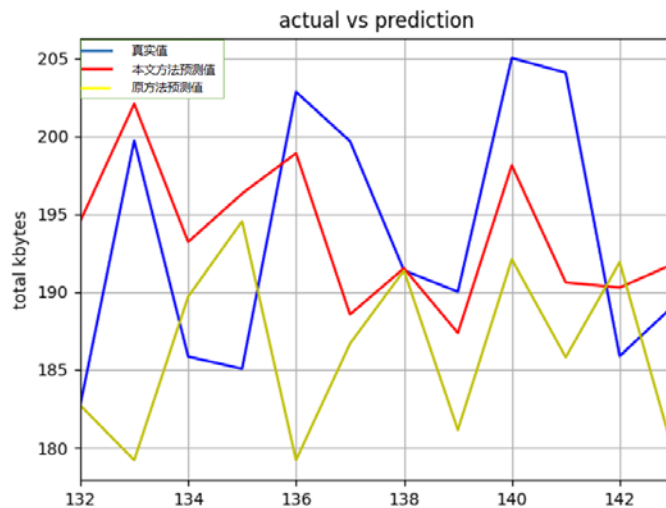


图 1 实验结果示意图

图中横坐标是时间, 以秒为单位。纵坐标是流量, 以千字节为单位。蓝色线表示的是真实的流量值, 黄色线表示的是采用同样的 LSTM 模型但是只根据链路流量数进行预测的结果, 红色线表示的是使用本文实验方法预测的结果。从图中可以看出, 无论是对具体的流量数值还是对流量变化趋势的预测, 该方法均要优于仅使用链路流量的方法。

## 2.2 NDN 拥塞控制研究

当前国内外学者关于 NDN 网络拥塞控制机制的主要研究成果包括请求端控制拥塞控制机制、逐跳拥塞控制机制、混合控制机制和基于缓存的拥塞控制机制<sup>[4]</sup>。其中, 逐跳控制机制通过中间节点控制兴趣包转发速率来实现拥塞控制, 优点是检测拥塞速度快, 对于未发生拥塞的链路影响小, 但同时也在请求端未减少兴趣包发送窗口的情况下容易造成局部链路长时间处于瘫痪状态的后果, 同时也会增加路由节点的工作负荷, 不利于规模化部署; 混合控制机制在中间节点检测到拥塞后需要通过向下游发送 NACK 包向请求端显式的通知拥塞, 明显地增加了网络的无效负载, 同时也在一定程度上具有请求端拥塞控制的缺陷; 基于缓存的拥塞控制机制需要在拥塞时改变缓存策略, 缓存利用率低, 且同样因没有请求端参与导致对网络整体的拥塞控制效果不明显。相比于上述三类方法, 请求端拥塞控制直接借鉴了传统 TCP/IP 网络架构的拥塞控制机制, 发展较为成熟且易于实现。常见的请求端控制拥塞控制机制包括: 兴趣包控制协议 (Interest Control Protocol, ICP)、内容中心传输控制协议 (Content Centric TCP, CCTCP) 和显式控制协议 (Explicit Control Protocol, ECP)。都是通过调整请求端发送兴趣包发送窗口来调整网络中的数据量<sup>[5]</sup>。前两者通过比较超时重传时间 RTO 和兴趣包与数据包往返时间 RTT 来探测拥塞。其中 RTO 的计算如式 (1) 所示。

$$RTO = RTT_{min} + (RTT_{max} - RTT_{min}) \times \alpha \quad (1)$$

当 RRT 超过 RTO 时, 表示网络因拥塞而导致数据包超时, 此时需要降低兴趣包发送速率缓解网络拥塞, 反之增加兴趣包发送速率。后者在网络中间节点探测拥塞并使用 NACK 包显式的通知请求端。

请求端拥塞控制通过衡量 RTT 来控制兴趣包

发送速率的方式更适合单源单路径的传输模式, 在 NDN 架构多源多径的环境特点下往往并不适用。一方面 RTT 抖动过大, 只能反映网络整体的拥塞状况, 对具体的某条或者某几条路径上的拥塞无法检测; 另一方面由于数据包可能是从不同的两条链路传回来的, 单纯的控制请求端的兴趣包发送速率可能会对其中发生拥塞的那条链路起到控制作用, 但是对于未发生拥塞那条链路则会造成使用率降低, 浪费网络资源。

## 3 方案设计

针对以上问题, 我们依托深度学习流量预测技术, 设计一种不依赖于 RTT 的请求端拥塞控制方式, 能够在兼具请求端拥塞控制方式优点的同时, 使其独立于 RTT 而能够适应 NDN 网络多源多径的特点。

具体方法是, 请求端在发送兴趣包时, 为每个流单独设置一个使用 CUBIC 算法调整大小的发送窗口  $W$ 。当第一次拥塞发生时, 记录当前的窗口大小为  $W_{max}$ 。然后以乘法减少的方式迅速降低至  $W_{min}$  ( $W_{min} = \beta * W_{max}$ ), 而后按照 CUBIC 曲线增大。与传统的 TCP-CUBIC 不同的是不使用超时而是使用数据包回到请求端时携带的信息调整窗口。设 CUBIC 曲线的三次方程式如式 (2) 所示:

$$\begin{cases} W = C(x - K)^3 + W_{max} \\ W_{min} = \beta W_{max} \end{cases} \quad (2)$$

其中  $0 < \beta < 1$ 。由式 (2) 可知,  $K$  为重新收敛到  $W_{max}$  的时间, 而  $\beta$  控制收敛的幅度。将式 (2) 整理得到式 (3):

$$K = \sqrt[3]{\frac{(1-\beta)W_{max}}{C}} \quad (3)$$

由式 (3) 可知,  $C$  控制了收敛的速度。当  $C$  一定时, 有  $\beta$  越大, 相当于三次曲线向左平移, 此时  $K$  越接近零, 即曲线重新收敛到  $W_{max}$  越快, 且到达  $W_{max}$  之后如果此时链路仍未发生拥塞, 收敛到新的  $W_{max}'$  所用的时间也越短; 反之,  $\beta$  越小, 收敛速度越慢, 寻找  $W_{max}'$  也越保守。

由此, 我们使用 LSTM 预测的流量值作为设置乘法递减因子  $\beta$  的依据。对于端到端的连接来说, 中间链路拥塞情况不同, 要使整个连接不发生拥塞, 则需要各条链路中当前最拥堵的来调整兴趣包发送窗口。取当前数据包出队列数占队列数  $Q_{cur}$  占该节

点缓存的出队列总数  $Q_{max}$  比例最大的链路的预测值进行分析:

a. 如果  $Q_{cur}/Q_{max} \leq a (0 < a < 0.9)$ , 则代表当前连接畅通, 此时如果预测到下一时刻流量大于当前时刻, 则保持  $\beta$  值不变, 如果预测到下一时刻流量值小于当前时刻, 则令  $\beta$  增大, 即令:

$$\beta' = \min\{\beta * TotalFlow/PreFlow, b\} (\beta < b \leq 0.9)$$

b. 如果  $a < Q_{cur}/Q_{max} < 0.9$ , 表示当前链路当前传输延迟较高但仍未发生拥塞, 此时若预测到下一时刻流量增加则通过减小  $\beta$ , 放慢窗口向  $W_{max}$  收敛的速度, 同时也降低当窗口超过  $W_{max}$  时寻找新的  $W_{max}'$  的速度, 即令:

$$\beta' = \beta * TotalFlow/PreFlow$$

c. 如果预测到下一时刻流量减少则同样保持  $\beta$  不变; 如果  $Q_{cur}/Q_{max} \geq 0.9$ , 此时表明链路拥塞, 请求端兴趣包发送窗口将降低至  $W_{min}$  并按照三次曲线开始增长。其中引入参数  $b$  是防止在链路畅通情况下流量骤减可能导致  $\beta \geq 1$  而失去流控效果。由于丢包造成的兴趣包重传对传输时延和网络负载均将造成不良影响, 要在尽可能增大网络吞吐量的同时减少丢包数, 因此判断拥塞状态的链路比值设置为 0.9 而不是 1。

## 4 实验验证

### 4.1 实验环境

实验数据采集需要的网络拓扑使用 ubuntu20.04 操作系统下的 ndnSIM 进行仿真。机器学习网络模型采用 Pytorch 架构构建的双层 LSTM 模型并采用上述网络拓扑采集到的流量数据预先进行训练。将训练好的 Pytorch-LSTM 模型读取到 ndnSIM 的 C++ 文件中进行预测。而后网络仿真拓扑中的请求端所安装的客户端软件按照文件中预测结果计算并调整参数  $\beta$ , 以此来实现 python 与 C++ 的协同工作。

命名数据网络实用拥塞控制方案 (A Practical Congestion Control Scheme for Named Data Networking, PCON) 是一种基于 AQM 码(通过测量数据包排队时间)检测拥塞, 然后通过明确标记某些数据包向消费者发出信号, 而后消费者据此实现许多经典的基于损耗的 TCP 算法, 如 Taheo、Reno、New Reno、HTCP、HSTCP、BIC 和 CUBIC, 是目前较为常用和成熟的 NDN 请求端拥塞控制机制<sup>[7]</sup>。本文

将对 PCON (CUBIC) 来验证请求端流量预测窗口控制方法的可行性。

### 4.2 实验设计

实验网络拓扑如图 2 所示。其中 C1 为内容请求者, 根据实验需求维护不同的兴趣包发送速率。M1、M2、M3、M4 模拟多径的网络中间节点。P1 为拥有 C1 所请求的名称前缀的内容源。数据包大小为固定的 1024 字节, 中间节点最大输出队列长度为 30 个数据包。链路传输速率如图中标识所示, 传播延迟均为 10 毫秒。

按照时间间隔为 1 秒、共 3000 次的方式采集机器学习的训练集。采集训练集时节点 C1 运行 PCON (CUBIC) 拥塞控制程序进行固定  $\beta$  大小的窗口调整, 初始参数  $\beta=0.8$ ,  $C=0.4$ 。在实验测试本文方法时, 节点 C1 每收到两次机器学习模型预测结果(即每 2 秒)调整 1 次参数  $\beta$ , 每组参数测试时长为 10 分钟。

### 4.3 实验结果

丢包率和网络吞吐量是衡量网络性能的重要参数。网络节点丢包将会造成大量的重传, 影响请求端获取需要的信息的时间。在 TCP/IP 网络中, 网络吞吐量指节点单位时间从网络中获取数据包的最大速率, 同样也是影响请求端节点获取网内信息快慢的重要因素。本文选用丢包率和单位时间网络吞吐量作为评价标准。参数  $a$  和  $b$  取不同值时的结果如图 3 所示。

由实验结果示意图与实验结果数据分析可知, 随着参数  $b$  的增加, 丢包率逐渐增加, 这是由于请求端判断链路状态为拥塞时瓶颈节点处数据包队列数占队列最大数的比例越大, 因此, 请求端对拥塞的反应越滞后。也因此, 当  $b$  距离 1 较远时请求端降低兴趣包发送窗口时的窗口越大, 向网络中单位时间内注入的兴趣包越多, 返回的数据包也越多, 而随着  $b$  进一步增加, 丢包现象愈发严重, 所以网络吞吐量先增加后减少。

整理实验数据, 当参数  $a$  与  $b$  有如下取值时, 本文方法的网络吞吐量大于 PCON 方法, 而丢包率小于它, 见表 1。

由此, 本文方法相比于传统方法, 在网络吞吐量和丢包率这两个评价标准上的表现可以达到更加优异的效果。

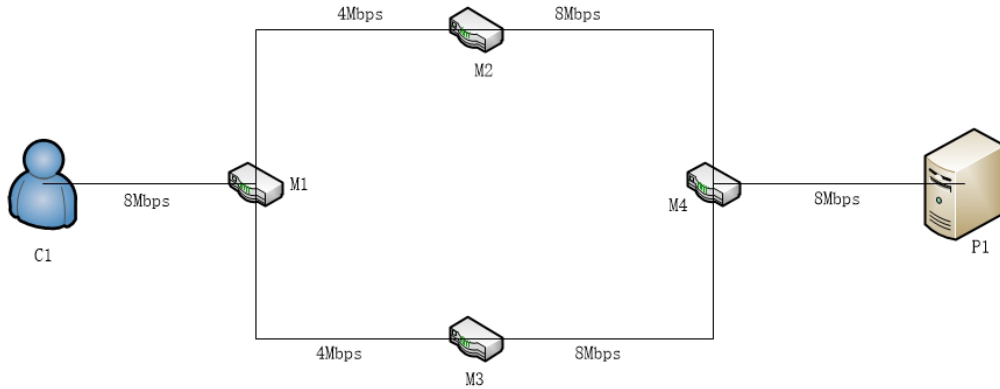


图 2 实验网络拓扑图

实验结果示意图

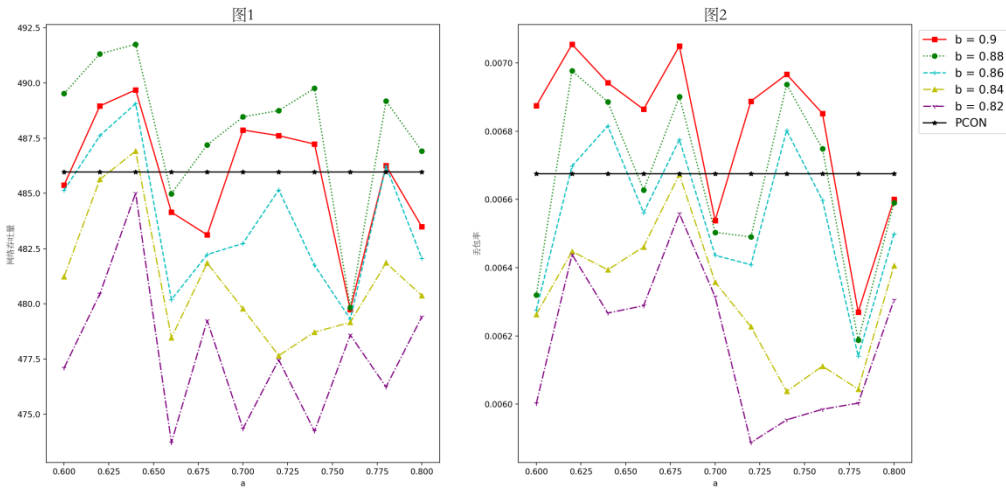


图 3 实验结果示意图

表 1 参数取值

|   |      |      |      |      |      |      |      |      |      |      |
|---|------|------|------|------|------|------|------|------|------|------|
| a | 0.78 | 0.6  | 0.7  | 0.72 | 0.78 | 0.6  | 0.72 | 0.78 | 0.62 | 0.64 |
| b | 0.82 | 0.84 | 0.84 | 0.84 | 0.84 | 0.86 | 0.86 | 0.86 | 0.88 | 0.9  |

### 5 结束语

针对 NDN 网络多源多径的特点导致的拥塞控制中请求端节点无法再以 RTT 作为拥塞信号的问题, 本文通过对 NDN 现有的拥塞控制算法和改进的 NDN 专用的流量预测方法的分析, 提出了一种基于端口流量预测的请求端 NDN 拥塞控制方法。该方法考虑并设计了一种利用流量预测结果作为拥塞信号并据此进行请求端节点兴趣包发送窗口调整的机制, 避免时延敏感, 保证了吞吐量和丢包率。在下一步研究中, 将分析在实际的网络场景中, 请求端节点如何获取网络中各个节点的端口流量预测

值, 以及如何依据链路中所有节点端口的预测值进行兴趣包发送窗口调整, 使本文的拥塞控制方法能够应用于实际网络的拥塞控制。

### 参考文献

- [1] 雷凯. 信息中心网络与命名数据网络[M]. 北京: 北京大学出版社, 2015: 46.
- [2] Sichen Song, Lixia Zhang. Exploring Rate-Based Congestion Control in NDN[C]. Paris, France: ICN'21, September 22 - 24, 2021: 141-143.

- [3] 张敏.Pytorch 深度学习实战[M].北京:电子工业出版社,2021:228.
- [4] 刘玉年.命名数据网络混合式拥塞控制技术研究[D].北京:北京理工大学,2021.
- [5] 黄茹辉.命名数据网络拥塞控制机制研究[D].江苏:江苏大学,2018.
- [6] 自治宏,郑浩楠.基于 TCP CUBIC 机制的协助传输数据方案[J].信息与电脑,2022(10):23-25.
- [7] Klaus Schneider,Cheng Yi,Beichuan Zhang,Lixia Zhang.A Practical Congestion Control Scheme for Named Data Networking[C].Kyoto,Japan:ACM-ICN'16,September 26-28,2016:21-30
- [8] Rozhnova N, Fdida S. An extended hop-by-hop interest shaping mechanism for content-centric networking [C]. 2014 IEEE global communications conference. [S.l.]: IEEE, 2014: 1-7.
- [9] Ben Basat R, Ramanathan S, Li Y, et al. Pint: Probabilistic in-band network telemetry[C]. Proceedings of the Annual conference of the ACM Special Interest Group on Data Communication on the applications, technologies, architectures, and protocols for computer communication. [S.l.: s.n.],2020: 662-680.
- [10] Li Y, Miao R, Liu H H, et al. Hpcc: High precision congestion control[M]. Proceedings of the ACM Special Interest Group on Data Communication. [S.l.: s.n.], 2019: 44-58.

**收稿日期:** 2022年8月18日

**出刊日期:** 2022年10月13日

**引用本文:** 于泽洋, 常成, 李彤, 基于流量预测的命名数据网络请求端拥塞控制方法[J]. 国际计算机科学进展, 2022, 2(3): 13-18.

DOI: 10.12208/j. aics.20220054

**检索信息:** RCCSE 权威核心学术期刊数据库、中国知网 (CNKI Scholar)、万方数据 (WANFANG DATA)、Google Scholar 等数据库收录期刊

**版权声明:** ©2022 作者与开放获取期刊研究中心(OAJRC)所有。本文章按照知识共享署名许可条款发表。<http://creativecommons.org/licenses/by/4.0/>



**OPEN ACCESS**