

面向命名数据网络的轻量级路由协议

邹博文¹, 嵩天²⁺, 李天龙¹, 杨雅婷²

1. 北京理工大学 计算机学院, 北京 100081

2. 北京理工大学 网络空间安全学院, 北京 100081

+ 通信作者 E-mail: songtian@bit.edu.cn

摘要:命名数据网络(NDN)是一种以信息为中心的新型网络架构方案。为了获取网络全局路由信息,典型NDN路由协议基于数据同步协议实现全局路由更新。然而,这类同步协议工作于应用层,在动态网络环境中,存在网络信息更新代价高、网络动态变化感知能力差等问题。针对这些问题,设计并实现了一种轻量级命名绑定网络层路由协议(NBRP)。该方法将路由节点名称与其发布内容名称进行绑定,使得路由节点可以定向进行路由更新信息的请求,不需额外同步协议即可完成轻量级路由信息交换。为进一步减少路由更新的传输代价,设计了一种可复用的路由信息包命名格式和路由信息增量传输机制。同时,为测试方案的有效性,在Linux内核中实现了相应功能模块并进行测试。实验结果显示,在动态网络环境下,相比于传统NDN链路状态路由协议,提出的方法能够降低72%的路由更新同步代价,且链路状态感知与响应速度提高近一倍。

关键词:命名数据网络(NDN);路由协议;链路状态感知;命名绑定

文献标志码:A **中图分类号:**TP393

Lightweight Routing Protocol for Named Data Networking

ZOU Bowen¹, SONG Tian²⁺, LI Tianlong¹, YANG Yating²

1. School of Computer Science & Technology, Beijing Institute of Technology, Beijing 100081, China

2. School of Cyberspace Science and Technology, Beijing Institute of Technology, Beijing 100081, China

Abstract: Named data networking (NDN) is a novel network architecture that revolves around information-centric principles. In order to acquire global routing information within the network, typical NDN routing protocols rely on data synchronization mechanisms to achieve global routing updates. However, such synchronization protocols operate at the application layer, which introduces challenges in dynamic network environments, such as high costs associated with network information updates and limited ability to perceive network dynamics. To address these issues, this paper designs and implements a lightweight named-binding routing protocol (NBRP). This approach binds routing node names with their published content names, enabling routing nodes to make targeted requests for routing update information without requiring additional synchronization protocols, thus achieving lightweight routing information exchange. To further reduce the transmission costs of routing updates, this paper introduces a reusable routing information packet naming format and a routing information incremental transmission mechanism. Additionally, to validate the effectiveness of the proposed solution, corresponding functional modules are implemented and tested within the Linux kernel. Experimental results demonstrate that, in dynamic network environments, compared with traditional NDN link-state routing protocols, the proposed method can reduce routing update synchronization costs by 72% and in-

基金项目:国家重点研发计划(2019YFB1803200)。

This work was supported by the National Key Research and Development Program of China (2019YFB1803200).

收稿日期:2023-04-03 **修回日期:**2023-05-29

crease link-state awareness and response speed by nearly twofold.

Key words: named data networking (NDN); routing protocol; link-state awareness; name-binding

随着直播视频业务蓬勃发展,网络流量日益增长,传统IP架构面临大规模信息传输负载压力等多种传输问题。以端到端为传输核心方法的IP架构难以应对大规模数据传输需求。当前大规模分发需求日益增加,相较于传统的端点寻址,用户更希望快速获得数据。因此,命名数据网络(named data networking, NDN),作为一种以信息为中心的网络架构,受到网络领域学者的广泛关注^[1]。

然而,在路由协议方面,IP网络广泛应用的开放最短路径优先协议(open shortest path first, OSPF)无法适配NDN网络以数据为中心的网络特性。这是因为在NDN网络环境下,路由语义和路由策略发生了变化,导致OSPF的语义逻辑无法适配。特别是在路由更新方面,网络节点无法通过OSPF的网络领袖选举策略进行选举,因此无法降低路由信息同步消耗。此外,NDN网络具有内容动态生成、支持节点动态接入等网络特性,导致NDN网络中的路由信息的更新更为频繁,传统OSPF并没有针对这些特点进行设计。NDN网络缺乏快速路由信息同步和链路动态感知的路由协议,导致路由信息同步代价大,链路变化感知能力差。

主流NDN路由协议,通过部署应用层的数据同步协议来辅助路由信息的全局同步。数据同步协议的作用是保持网络中的所有路由节点存储的信息状态完全一致。该协议的工作方式如下:首先,发送方在兴趣包中添加哈希字段来表示当前版本信息。接收方通过对比自身和兴趣包中的哈希值,来判断双方信息是否一致。如果不一致,就需要从发送方获取最新版本的信息,最终使得网络中的路由信息保持同步。该方法能够降低泛洪开销,也避免了网络包的循环依赖。

尽管数据同步协议满足了基本的数据同步需求,但在动态网络环境中,基于数据同步协议的链路状态路由协议存在路由更新操作代价大与路由动态感知能力差两个方面的问题,具体分析如下:

(1)路由信息更新同步操作代价大。数据同步协议的目标是域内所有设备路由信息全局同步,具体通过网络泛洪来实现。而泛洪的基本原理是每条网络连接都需要进行数据包的传输。这就使得网络中,两个不负责内容更新的邻居节点,长时间进行非

必要的信息同步流程,造成网络带宽浪费。

图1展示了典型全局路由协议信息同步方案。图示的网络中存在一个生产者 P 和转发路由节点 $R1$ 、 $R2$ 、 $R3$ 和 $R4$ 。如图1所示,橙色箭头表示该同步流程为非必要传输流程,同步请求总次数为14次,冗余同步请求数为11次,即超过78%的路由同步请求为非必要请求。基于数据同步协议的全局信息同步过程需要网络中每两个路由节点进行定期通信。但示例网络中只存在一个动态发布内容的 P 节点,其他节点的路由内容都是从 P 学习得到,即 $R1$ 、 $R2$ 和 $R3$ 之间交换的信息都是过程信息,在 P 没有发生内容变化时不会发生改变,三个路由节点之间的非必要路由同步浪费了网络带宽。全局路由同步方案的同步代价随节点数量和链接数量的增加而几何级增长。

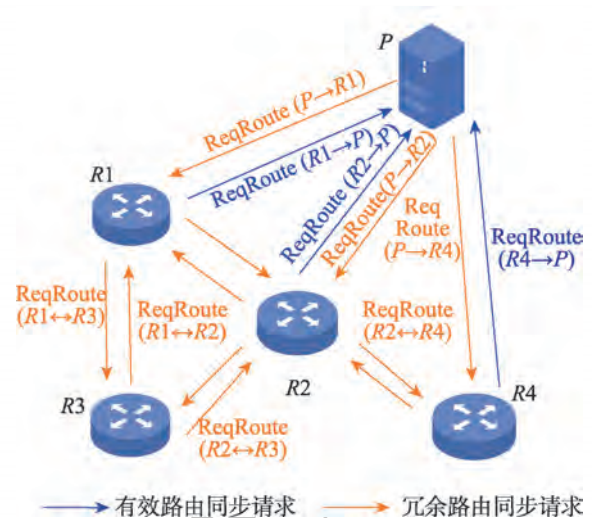


图1 同步协议全局同步方案

Fig.1 Global synchronization process in Sync protocol

(2)网络环境动态变化感知慢。数据同步协议在动态环境下存在同步缓慢问题。在网络环境动态变化的情况下,例如网络拓扑改变、内容前缀更新等情况,数据同步协议的同步代价高。这是因为数据同步协议是基于邻居节点进行探测,需要将探测信息进行全网广播,直至信息收敛后才能保证网络信息同步,影响信息传输性能。

为了解决上述问题,需要设计一种路由同步代价低,且路由动态感知能力良好的NDN网络路由协议。本文设计的路由协议基于绑定信息进行同步,

降低了信息同步的冗余性。网络节点仅关注信息源的同步信息,并不关注非生产节点的过程信息。本文设计了一种轻量级网络层路由协议NBRP(named-binding routing protocol),并在Linux内核中进行实现,该路由协议可独立工作,不依赖其他协议实现路由同步。本文的主要贡献如下:

(1)为了提高信息路由同步效率,提出了一种轻量级的命名绑定机制链路状态路由协议,通过将路由节点与相关内容进行绑定,使得路由信息同步具有导向性,无需泛洪即可定向找到路由更新信息,降低信息同步的操作量;

(2)为了减少路由同步通信代价,提出了一种可复用网络包命名格式和增量机制,利用NDN数据复用与网内缓存特性,避免冗余传输;

(3)为了及时感知网络状态变化,提出一种快速的链路状态感知机制,在路由节点内部实现链路感知,且不需要将感知结果进行网络同步,避免了链路震荡。

另外,为了验证所提路由协议的有效性,在Linux内核中实现了该路由协议并进行了性能测试。实验数据显示,在动态网络环境下,路由更新同步代价降低72%,链路状态感知时间缩短近一倍。

1 相关工作

1.1 NDN基本概念

当前网络将重点放在了内容位置,而不是内容本身,为了解决这一问题,命名数据网络的逻辑概念被提出^[2]。而NDN的内网缓存和数据复用等网络特性^[3],吸引了学者们进行研究。

在NDN网络中存在两种网络包类型^[4],兴趣包和数据包,分别用于请求者获取内容,生产者或中间节点传回所需内容。待定兴趣表(pending interest table, PIT)、转发信息库(forwarding info base, FIB)和内容缓存(content store, CS)组成了NDN的基本架构。当路由节点收到一个兴趣包时,其首先会查找CS,如果CS中缓存了所需内容,则直接将该内容进行回传;而如果CS中不存在该内容,兴趣包会被PIT记录。如果PIT中存在与该兴趣包名称相同的条目,路由节点将该兴趣包的到达接口添加到该条目中,并将该兴趣包丢弃。如果没有,PIT表将会产生一个新条目,记录兴趣包的名称和接口,然后查阅FIB表条目信息,根据转发策略发送该兴趣包。

1.2 NDN路由协议

路由协议在网络架构中占据重要地位,其基本

功能为路径计算和数据传输。在NDN的基本架构被提出后,为了NDN架构的快速部署,NDN开发小组提出了基于NDN版本的OSPF协议(OSPFN)^[5]。该协议支持兴趣包和数据包的收发,但其仍然使用IP地址,无法支持NDN内网缓存等网络特性。在此之后,为了支持NDN网络特性,命名链路状态路由协议(named-data link state routing, NLSR)^[6]被提出,该协议采用了Sync机制进行链路状态广播(link state addresses, LSA)的传输,以此来降低通信负载。然而当时的NLSR采用CCNx内嵌的同步协议,导致其存在高内存负载等问题。因此在新版本的NLSR^[7]中,选择ChronoSync同步协议作为数据同步协议,在单个LSA中广播所有的命名前缀,并增强了安全功能。至于链路状态的感知,NLSR采用特定网络包来定期感知邻居节点的状态,并将这些感知信息广播到网络中。但由于数据同步协议的使用,NLSR依旧存在高负载占用的问题。且定期感知的周期过长,无法动态感知链路状态。

为了解决NLSR存在的路由信息重复传递问题,链路状态内容路由(link-state content routing, LSCR)^[8]被提出。LSCR的作者认为,在进行路由交换的过程中,一个命名前缀被复制多次,造成了带宽的浪费。因此LSCR有选择性地名称前缀同步来降低负载。然而,由于LSCR严格控制前缀的同步,一个LSA中仅能包含一个前缀。在多前缀情况下,LSCR需要发送大量的路由信息包,造成了带宽的浪费。而距离内容路由协议(distance-based content routing, DCR)^[9]通过距离信息来判断信息是否需要发送,在一定程度上解决了重复前缀的问题,但其选择发送策略会导致部分路由信息无法被同步,使得这些信息无法被其他网络节点访问。

上述相关工作显示,当前NDN下的链路状态路由协议,在动态网络环境下,存在网络同步代价高、链路状态动态感知能力差等问题,造成了网络带宽的浪费,影响数据稳定传输。

2 协议设计

2.1 NBRP概览

在传统NDN链路状态路由协议中,为了控制路由版本信息,需要部署应用层的同步协议。数据同步协议采用泛洪策略来进行路由信息的广播。尽管其应用了优化措施来降低同步负载,但泛洪操作不可避免地浪费了网络带宽。这是因为泛洪策略需要

遍历每条网络连接来传递路由信息包,即便网络连接两端节点已经获取了该路由信息包^[10]。并且,当感知到链路状态变化时,新的路由信息包需要被广播,造成网络震荡,影响了数据传输。这一问题的部分原因在于,每次发送拓扑更新信息,路由节点都需要重新构建网络拓扑。因此,需要设计一种能够定向更新路由信息的路由协议,使得路由信息的同步可以根据网络需求进行传递,在提高传递效率的同时,不破坏NDN网络结构特性。并且在链路状态感知的过程中,降低拓扑更新时间,避免链路震荡。

本文提出了一种基于命名绑定机制的链路状态路由协议。命名绑定机制,即通过对路由节点设定名称标识,建立与名称前缀之间的对应关系。路由节点的名称标识可以手动或自动生成,且在整个网络域中唯一。尽管本文对路由节点设定了名称标识,但这一命名过程并不等同于IP分配。NBRP将绑定定义为路由节点与其发布内容的对应关系。在绑定关系确立后,路由节点将关注生产者绑定的内容信息,而不是路由更新的过程信息。路由信息的传输不依赖泛洪,而是向生产者节点通过发送请求来获取路由信息。如图2所示,在命名绑定关系的帮助下,R1得知网络中仅存在P一个生产节点,而R2、R3和R4皆为转发路由节点,不存在绑定关系。此时R1将只向P请求路由信息,因为网络中仅有P存在绑定关系。非生产节点之间,例如R1和R2之间,不进行路由信息的同步。网络中的其他节点也仅向P进行路由请求。图中同步请求总次数仅为4次,且无冗余路由请求。在命名绑定关系的帮助下,同步代价仅

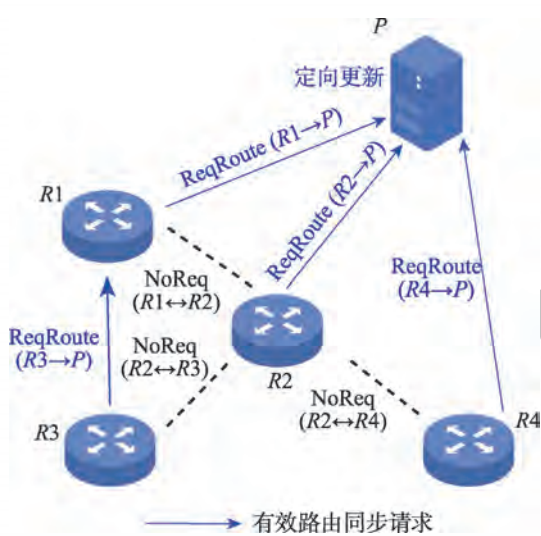


图2 NBRP的定向路由信息请求
Fig.2 Routing information request in NBRP

随网络中生产者数量增加而线性增长。命名绑定关系减少了路由信息同步次数,降低了路由信息更新代价。

本文路由协议的架构如图3所示,除去控制单元负责调度外,本文路由协议还存在几个主要数据结构:网络信息模块,由邻居表(neighbor list, NL)和拓扑表(topology list, TL)组成,用于存储邻居节点信息和计算构建网络拓扑;路由信息模块,由路由信息库(routing info base, RIB)和更新日志(Log)组成,用于记录绑定关系和本地信息更新记录,并进行FIB信息的及时更新。RIB中通过自旋锁双索引哈希表来保存绑定关系。

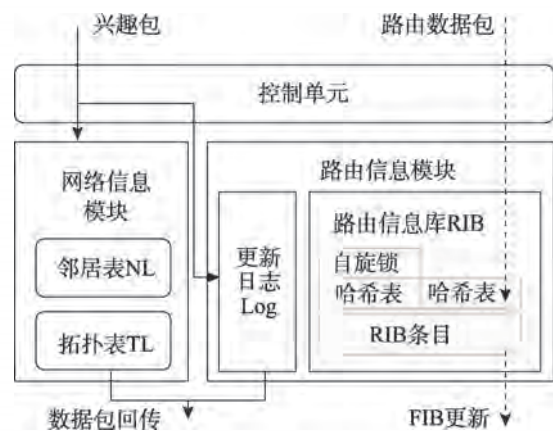


图3 NBRP路由协议架构
Fig.3 NBRP architecture

2.2 路由节点命名绑定与定向更新

NBRP的命名方式为层次化命名。层次化命名便于信息的聚合与管理,且可以通过前缀压缩来节省存储空间。

关于路由节点的命名,每个路由节点可以自动生成命名,或是由网络管理员手动命名,名称格式为“/ <network> / <domain> / <router>”。“<network>”表示路由的网络环境,“<domain>”用于提供可读信息,例如设备信息或是地理位置信息,方便网络管理员进行管理。而“<router>”,如果没有被手动配置,将会设置为MAC的哈希值,来保证全网唯一。

本文将绑定关系定义为路由节点与其发布内容的责任关系。如果一个路由节点发布了某项内容,或是可以在节点本地访问某项内容,那么可以认为该路由节点绑定这一内容,路由节点即与该内容建立绑定关系。在绑定关系确立后,路由节点将注重绑定内容的路由信息,而不是从其他路由节点学习到的路由信息。路由信息的传输,也不再依赖泛洪,而是对特定路由节点进行路由信息请求,减少了访

问对象和访问次数,以此来降低路由同步更新代价。

命名绑定关系需要在路由节点内部进行存储,这对于路由节点而言是额外的空间开销。对于传统链路状态路由协议而言,为了保证版本统一,路由节点需要存储多个版本的链路状态信息,而在此之中存在着许多过程信息。但NBRP的绑定关系表仅存储数据生产者节点的发布内容信息,并不包含其他节点的过程信息。因而在总体存储空间开销上,NBRP与传统链路状态路由协议差别不大。

在路由节点对绑定关系访问和更新的过程中,可能会由于访问冲突造成以下问题:(1)查询效率低。例如当路由节点需要获取网络中某一节点R1可提供的内容,或是在路径替换时,查询可提供“/a1”前缀的路由节点集合时。如果没有快速定位功能,每次查询都要对绑定关系进行一次全体遍历操作,造成不必要的时间开销。为了提高查询速度,需要设计高效的查询结构,避免对绑定关系的全局遍历。(2)并发插入访问冲突。当路由节点内部同时出现绑定关系访问和绑定关系修改操作时,操作的先后顺序会影响数据的准确性。在访问过程中进行修改,容易导致“脏读”的出现,影响路由信息的同步。为了避免绑定关系的访问受到影响,就需要避免多个线程同时访问或修改同一条目上的绑定关系表。

对于绑定信息的快速定位,可以通过高效数据结构进行解决。NBRP对于绑定关系的存储采用双索引哈希表结构,即将路由节点名称和内容名称作为索引项。图4为NBRP的命名绑定关系存储结构示意图。查询某节点的所有可绑定信息,例如查询1查询R5的绑定信息,定位查询的复杂度为 $O(1)$,而如果想要定位其中的“/b”内容,遍历的时间复杂度为 $O(k)$,其中 k 为节点R5的绑定信息量。

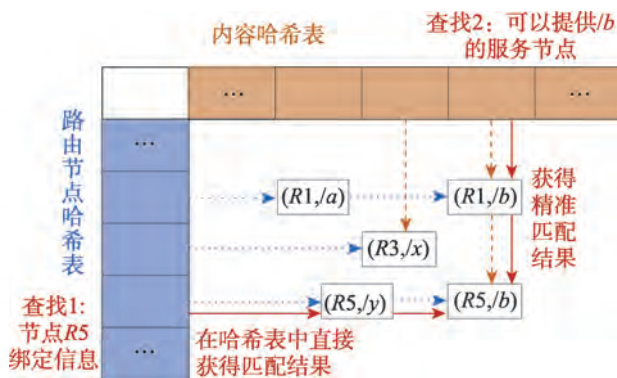


图4 命名绑定关系存储结构

Fig.4 Storage structure of named-binding relationship

而对于并发插入访问的并发问题,可以通过添加全局锁解决。但单纯对绑定关系表进行加锁处理,这种加锁方式粒度过粗,在进行并发操作时容易达到访问瓶颈。因此,NBRP采用粒度更小的自旋锁,并施加于哈希表的索引项中。这一操作,使得对不同路由节点的并发访问修改不会发生阻塞。在避免并发冲突的情况下,保证了访问效率。

2.3 路由包设计

对于网络包的命名格式,本文主要设计了三种网络包,即探测信息包(Info packet)、邻居节点信息包(neighbor link-state address, NLSA)和路由信息包(routing link-state address, RLSA),分别用于提供自身路由信息、建立网络拓扑和交换路由信息。三种网络包名称结构如下:

探测信息包:也可称作Hello包或Info包,用于通告路由节点自身的相关信息。本文通过研究相关工作注意到,大部分Info包通信存在限定条件,即路由节点已知网络中其他路由节点的名称。也因此这些Info包是指向性的,包名称中包含双方路由节点的名称。这种设计提高了通信稳定性,但问题是在实际场景中,当一个路由节点第一次接入到网络时,该路由节点无法得知其他路由节点的命名。反过来,其他路由节点也无法向新路由节点发送Info包,因为缺少该路由节点的名称信息。因此,本文设计了一种单向Info包,在名称中只包含了发送者的名称,例如“/NBRP/Info/<sender>”。Info包只有兴趣包,没有数据包。在NDN中,兴趣包会存储在PIT表中,直至超时或是返回相应数据包才进行相应重传操作。而NBRP中Info包不存在数据包,因此在路由节点存在多个发送接口时,Info兴趣包的大规模发送容易造成PIT表无端占用,影响其他常规兴趣包的注册。此外,Info包在没有设定传播范围的情况下,易造成网络泛洪。

为了避免无数据包带来的PIT表项占用问题,NBRP将Info兴趣包的生存周期设置为最小值,且规定Info包不会触发NBRP的超时重传机制。此外,Info兴趣包的发送模式为广播,最大传播范围为一跳,以此来避免网络泛洪。

邻居节点信息包(NLSA):为了避免网络传输发生循环问题,路由节点需要建立全局网络拓扑,这就需要NLSA。NLSA的包名称格式,仅包含接收方,即“/NBRP/NLSA/<receiver>”。

路由信息包(RLSA):RLSA的包名称格式为“/NBRP/RLSA/<receiver>/<token>”,相比于NLSA多

一个凭证(token)字段,该字段由接收方生成并给出,表示本次RLSA兴趣包请求的信息边界,初始默认值为0。NBRP通过token字段和时隙机制来保证路由信息的时效性,因此无需版本控制字段。

NLSA和RLSA都是定向访问数据包,即每个节点都要向数据生产者发送相应兴趣包,而数据生产者需要针对这些兴趣包一一回应数据包。这就导致终端服务器的访问压力过大,为了缓解这一问题,NBRP利用了NDN缓存规则。在NDN中,一份数据包可以满足多个同名兴趣包。而NLSA和RLSA仅包含接收方信息,因此可以通过PIT表项汇聚功能来汇聚兴趣包,并通过中间节点CS缓存来提供数据包,降低终端节点的访问压力。

2.4 通信流程

通信流程分为三个部分,即建立邻居关系、构建网络拓扑和交换路由信息。

2.4.1 邻居关系建立

当路由节点启动并接入到一个未知网络后,其将对本机网卡进行定期的Info兴趣包广播,以此来向网络中的其他路由节点通告自身存在。Info兴趣包仅传播一跳,被其他路由节点接收处理。接收到Info兴趣包的路由节点将检查兴趣包的名称字段,决定是将该Info兴趣包丢弃,还是与发送方建立邻居关系。如果不丢弃Info兴趣包,接收方将发送方信息添加到NL中用于后续通信。之所以将Info包设计为单向通信,是因为新接入网络中的路由节点无法得知其他路由节点的名称,因此只能在包名称中添加自身节点名称。在完成一次Info包的循环后,路由节点会建立邻居表。邻居表中的条目设有生存周期,以防止因路由节点崩溃离线、网络连接失效等原因造成的邻居表中节点信息过时。

2.4.2 网络拓扑构建

网络拓扑建立对链路状态路由十分关键。如果路由节点无法建立网络拓扑,那么网络包转发将只能根据贪心算法进行计算,而贪心算法只能保证局部最优,无法保证全局最优,难以确保网络传输稳定性。网络拓扑的建立流程如下:首先,路由节点遍历邻居表。如果邻居表为空,路由节点将跳过拓扑建立阶段并等待下一个拓扑建立循环。如果非空,路由节点将向邻居表中的每个节点发送NLSA兴趣包。在收到相应数据包后,路由节点提取并发现新的网络节点,直至数据包中没有新节点信息。此时路由节点根据预先设置的转发策略,进行拓扑建立

和传输路径计算。

2.4.3 路由信息高效交换

为了避免泛洪问题,NBRP利用了NDN的缓存机制和增量策略来降低通信负载。

实现高效路由信息交换,需要设置高效的路由信息的传输格式和存储结构。在先前工作中,路由数据包的信息包含所有的路由条目。部分优化策略会提供摘要信息以降低负载。但即便如此,数据包的大小也随着过程信息的增加而增加。在NBRP设计中,数据包的路由信息应仅包含路由节点的绑定信息,而不包括从其他路由节点学来的路由信息。实现这一点的前提条件是确定路由节点与发布内容之间的关系,即命名绑定关系。同时为了支持这项功能,本文建立了一个更新日志表Log,用于记录内容的更新操作,其表项的三元组格式为(Prefix, Time, Operation)。

在Log中,NBRP采用时隙设计模式。时隙机制用于控制路由数据包的大小,并为路由信息的获取提供按需访问功能。信息槽默认值为RLSA周期的一半。在传输过程中,路由节点根据本机时间和RLSA兴趣包的token,来决定数据包中包含的路由信息。Token值由路由节点自身生成,为本机时间序列值。使用token后,无需保证设备之间的时间同步。路由节点可以根据自身需求获取其他节点的路由信息,因此不需要版本控制功能,避免了路由节点下线未清除缓存时出现的版本错误问题。

关于存储格式,本文建立了路由信息表(RIB)来存储路由信息,并通过RIB对FIB更新。与NDN转发平台的RIB不同,NBRP中的RIB属于路由协议层级,被路由协议控制。此外,RIB代替了链路状态数据库来提供数据存储功能。

在确定了传输格式和存储格式后,路由信息交换流程如下:(1)路由节点向TL中记录的所有路由节点发送RLSA兴趣包。(2)接收方收到RLSA兴趣包后,检查token值和本机时间序列来判断信息边界。接收方将此信息边界作为新token值填入到数据包中,然后从Log读取操作信息,填充数据包。

路由信息和token的更新机制如图5所示:(1)路由节点R2向路由节点R1发送RLSA兴趣包,token值为 a_1 。路由节点R1在此期间并未更新内容,因此回传空数据包,表示路由节点R1尚未对绑定数据进行任何修改。(2)当一个更新操作执行并被添加到Log后,路由节点R1的token值从 a_1 变为 a_2 。(3)路

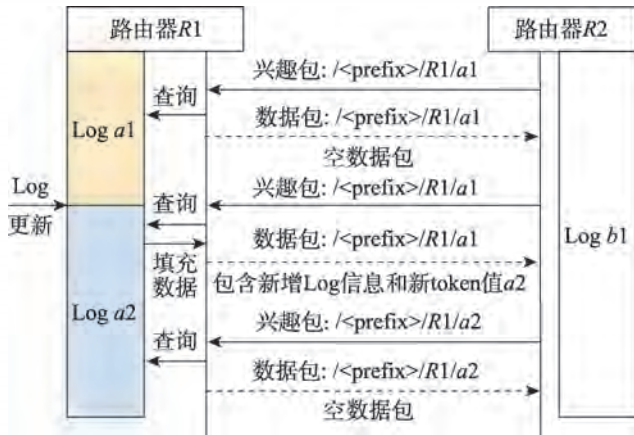


图5 路由信息交换流程

Fig.5 Routing information exchange process

由节点 R2 在下次 RLSA 兴趣包发送后,接收到了包含新 token 值和 Log 更新内容的 RLSA 数据包,此时路由节点 R2 将根据数据内容对 RIB 进行更新,然后更新 FIB 完成路由信息交换流程。

可以看出,本文路由协议不要求全局路由信息的同步,也不关注路由节点是否传输从其他路由节点学来的信息,这些因素确保高效网络信息传输。

2.5 链路状态快速感知

在绑定关系的帮助下,在网络拓扑建立阶段,路由节点就可以完成部分路由信息更新。已有路由算法,通过 Info 包的超时重传来判断链路状态,然后将感知信息进行全网广播,保持全局路由信息同步。

但这种方法导致路由节点频繁进行网络拓扑重计算,造成网络震荡,增加路由节点的网络拓扑收敛时间,影响了路由信息传输同步的表现。

为了解决以上问题,本文设计了一种非全局信息同步的链路状态感知机制。该机制需要 PIT 表进行协作,以下为具体感知流程:(1)超时侦测。当在 PIT 表中某一兴趣包条目发生特定次数的重传超时后,PIT 表将会向路由协议发送超时通知。(2)故障感知。根据绑定关系,路由协议可以判断出哪个或哪些路由节点拥有当前超时请求的内容,也可以得知超时兴趣包被转发到哪个路由节点。路由协议据此判断出目标节点的链路状态不稳定,无法继续提供服务。(3)路径重定向。路由协议从 FIB 中移除目标节点绑定的所有内容信息,直到下一次路由信息更新循环。

这一链路状态感知流程在路由节点内部进行,不需要进行全网广播。图 6 为链路状态感知的工作流程。在该拓扑中, P2 和 R2 之间的链路发生故障。传统 NDN 路由协议下的 R1,会因为无法及时进行链路状态感知^[11],认为 P2 仍然可以正常进行内容服务,使得 C2 的内容请求在下一轮路由信息同步前,无法正确发送到 P3,导致 C2 的内容请求失败。而本文方法与 PIT 表相结合,在 R1 兴趣包出现特定次数发送超时后,PIT 表将通告 R1 的路由协议。路由协议通过查询 RIB,判断出该兴趣包当前被发送给

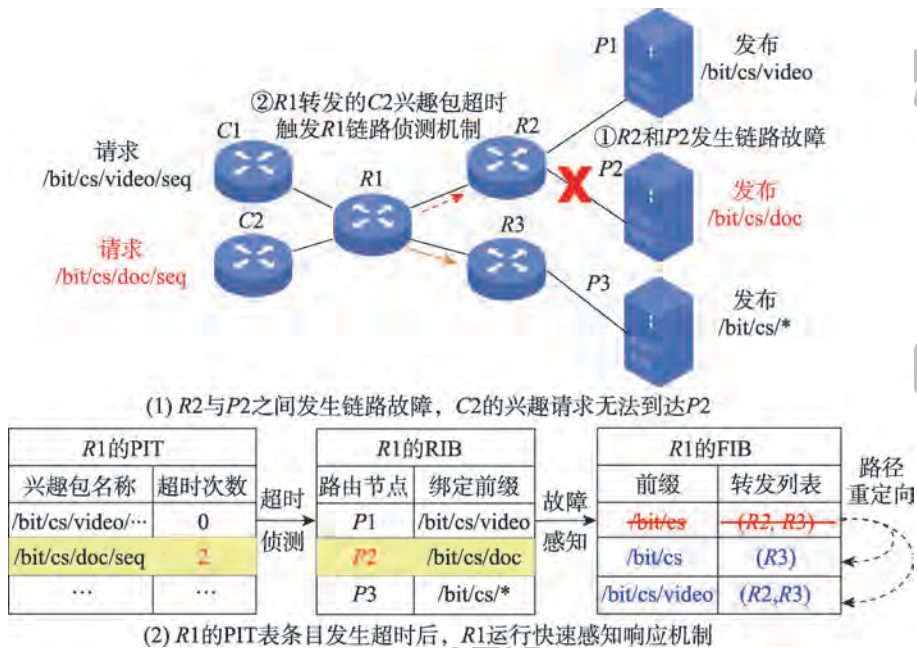


图6 链路状态感知机制

Fig.6 Link-state awareness mechanism

$P2$ 。因此路由协议判断 $P2$ 暂时不可访问,因此更新 FIB 信息,将 $P2$ 的绑定内容进行路径重计算或删除,来确保其他对 $P2$ 绑定内容的兴趣包可以被转移到其他服务节点。该流程持续到下一次路由信息更新,且不需要将该感知结果同步到网络中。

3 实验与结果分析

3.1 实验设置

本文方案的动态网络传输支持性验证指标如下:

(1)路由同步通信代价:路由条目更新时,网络中路由信息包数量。网络中路由同步包数量越少,证明路由协议的同步代价越低。

(2)网络链路感知能力:链路故障时消费者吞吐量变化。在网络链路发生变化后,路径切换速度越快,吞吐量恢复速度越快,证明动态感知能力越强。

(3)移动性支持能力:节点重接入网络后,最大访问恢复延迟。最大访问恢复时间越短,即路由节点重接入网络后可以更快地被其他节点访问,证明移动性支持能力越强。

为了评估本文路由协议在真实环境下的表现,本文在 Linux 内核 4.14 版本下进行了路由架构的实现,并修改了内核的包处理机制来支持 ICN 的包转发功能。然后将该内核模块安装到树莓派 4B 中,配置为 32 位操作系统和 4 GB 内存。此外,为了满足实验需求,部分树莓派安装了外置 USB-A 以太网卡来支持更多的网络连接。同时,为了实现更加真实的网络环境,本文使用了 Linux 下流量控制模块(traffic control),来设置链路中存在一个触发概率为 10% 的 5 ms 延迟震荡^[12]。

为了生成流量,本文在 5 个树莓派上部署了生产者和消费者,并通过路由节点进行连接,实验拓扑如图 6(1)所示。每个生产者最初设定发布 10 000 个内容信息^[13],这些内容信息来自热门网站数据^[14],每个内容信息包含 10、100、1 000 个 1 KB 的内容块。但经历多次内存溢出后,本文将生产者内容发布数量设置为 1 000。内存溢出问题的原因为,尽管树莓派内存为 4 GB,但其可供生产者发布内容的底层内存仅为 718 MB。消费者根据齐夫分布来生成兴趣包获取内容,齐夫分布偏移量值为 0.8,该值与网络中的流量样本相似,可以用于模拟真实的网络请求。

为了衡量路由同步通信代价,在内容更新的情况下网络路由同步包总量这一指标上,本文与 NLSR 和 LSCR 进行对比,NLSR 采用数据同步协议进行同

步,而 LSCR 使用锚点路由机制进行同步;为了衡量路由协议的动态感知能力,在网络拓扑变化时消费者吞吐量这一指标上,本文方法与 NLSR 进行对比。NLSR 定期发送 Info 兴趣包感知链路状态,并将感知结果同步到整个网络中;为了比较路由协议的移动性支持,在最大可访问时间这一指标上,本文与采用短期拓扑存储的 NLSR 和采用 IP 架构进行移动处理的传统链路路由协议 TLSR(traditional link-state routing)进行对比。此外在 3.3 节,本文方法与采用 Sync 机制的 NLSR,使用锚点路由机制的 LSCR,以及采用传统 IP 策略的 TLSR 从动态网络环境指标上进行了对比,并从结果上证明了本文协议在动态环境下的可行性。

3.2 实验结果

3.2.1 路由同步通信代价

表 1 展示了在路由前缀信息更新时,本文方法和 NLSR、LSCR 的路由同步代价,数据的单位为网络中路由信息包的总量,通过 AT&T 核心网络进行模拟实验。在网络拓扑无变化的情况下,生产者新发布了一条内容信息,统计网络中路由同步需要发送的路由信息包的数量。同步包数量越少,证明路由同步代价越低。可以看出,采用了数据同步协议进行同步的 NLSR,发送的同步包数量最多,这是因为 NLSR 中每两个相邻节点之间都需要进行路由同步,造成带宽浪费;而 LSCR,采用锚点策略,路由节点可以选择性地发送路由同步包来降低同步代价,但其选择策略严重依赖网络中的数据重复度,因此性能并非最佳;而本文方法仅向生产者节点进行数据请求,且请求的数据包可以被其他节点复用,因而网络包数量最低。可以看出,本文方法在路由同步包数量上,相比使用数据同步协议的 NLSR,减少了 72%。

表 1 路由协议同步通信代价对比

| 方法 | 单次路由更新包数量 | 单次路由更新包数量峰值 | 单次路由更新包数量最小值 |
|------|-----------|-------------|--------------|
| NLSR | 215 | 220 | 211 |
| LSCR | 97 | 138 | 77 |
| 本文方法 | 62 | 71 | 41 |

3.2.2 链路状态感知和路径切换性能

图 7 展示了本文方法和 NLSR 在链路故障发生时,消费者吞吐量的变化情况,网络拓扑如图 6(1)所示。在网络稳定 30 s 后,切断 $R2$ 和 $P2$ 之间的连接,评估 $C2$ 的吞吐量变化。可以看出,在断开网络连

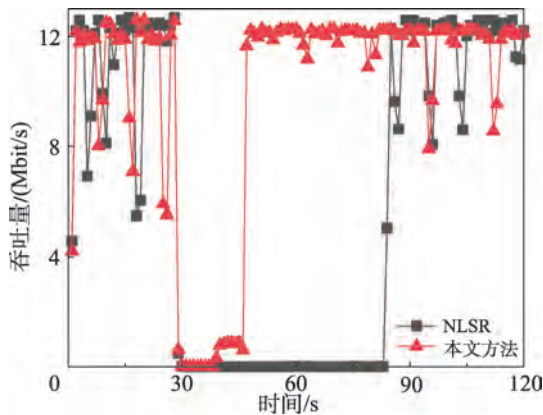


图7 链路故障时的消费者吞吐量变化

Fig.7 Consumer throughput during link failure

接,经过两次PIT兴趣包超时后,中继路由节点R1判断生产者P2不可达,将C2的兴趣包转发到P3中。而NLSR,一方面其感知周期过长,另一方面需要通过同步协议将感知信息广播到网络中,导致路由更新不畅,直到下一次路由更新,才完成了路径替换。结果表明,本文方法可以快速感知网络故障,并据此更新转发请求路径,并且不需要全局同步,避免网络震荡。通过查阅资料,NLSR的感知周期为60s,最小链路动态感知时间为3次Info包RTT和两次发送间隔之和,即14s。而本文方法的最大链路动态感知时间为两次PIT表中兴趣包超时,即8s。链路动态感知时间,相比于NLSR的感知时间,增快了仅一倍。

图7中本文方法在切换路线后,出现了小幅度的吞吐量提升。根据程序分析,这是因为发送程序的滑动窗口设置略小,导致其未能切换路径后瞬间恢复到正常吞吐量。

3.2.3 移动支持效果

为了更好地体现移动性支持性能,本文定义了最大可访问时间概念,即将某路由节点断开网络连接后,重接入网络时可被其他路由节点进行绑定信息访问的最大时间。最大时间值越小,证明路由协议的移动性支持越好。之所以选取最大时间作为指标,是为了避免路由节点重接入网络中时,网络正在进行路由信息更新,导致路由节点在短时间内完成了网络拓扑建立与路由信息更新,无法体现移动性支持指标。

在此实验中,本文方法与传统常规链路状态协议(TLSR)和NLSR进行性能对比。从图8可以看到,在离线15s重接入后,NLSR与本文方法相差不多。而随着重接入时间的增长,NLSR的最大可访问时间显著增加。这是因为NLSR并不会长时间记录

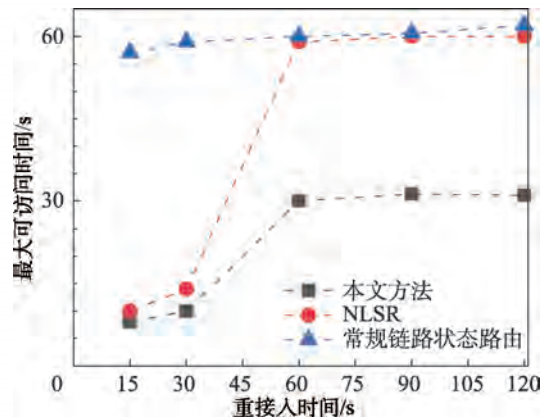


图8 移动性支持比较

Fig.8 Comparison of mobility support performance

路由节点与其负责的路由信息,导致该路由节点由于过时被删除,当路由由节点重接入后,需要重新进行一次路由同步流程方可正常通信。而本文方法,由于绑定关系的支持,即便在FIB中删除了离线节点的绑定内容,在路由节点重接入后,只需要在拓扑更新阶段,即可判断出该路由节点是否接入网络,并对其绑定内容进行相应访问。而TLSR,在设计逻辑上并未考虑移动支持,因而效果最差。

3.3 比较与讨论

本文方法与其他链路状态路由协议,基于文章给出的相关数据,在表2中进行了对比:NLSR在链路感知方面采用了同步协议。通过定期发送询问兴趣包来降低链路感知时间,但增加了同步数据量,且路由条目更新的操作量较大;LSCR采用了锚点路由机制^[15],降低了同步数据量,但由于单LSA只携带单个信息,在一定程度上增加了网络包数量,且没有设计链路感知机制;TLSR,以OSPFN为例,既无法提供移动性支持,也无法降低路由同步消耗,仅用于NDN路由协议过渡使用^[16]。

表2 动态网络指标对比

| 方法 | 链路最大动态感知时间/s | 非同步链路动态感知支持 | 移动性支持 | 单次路由更新代价(网络包数量) |
|------|--------------|-------------|-------|-----------------|
| NLSR | 14~60 | 不支持 | 短期 | 215 |
| LSCR | 60 | 不支持 | 不支持 | 97 |
| TLSR | 60 | 不支持 | 不支持 | — |
| NBRP | 8 | 支持 | 长期 | 62 |

本文方法在动态网络环境下,有着稳定快速的路由感知机制和移动性支持能力,最小链路感知时间为两次兴趣包传输超时时间。且与其他链路状态路由协议相比,本文路由协议同步代价较低。这是因为,本文方法依靠绑定机制确定了信息与路由节

点的关系。在绑定关系的帮助下,路由节点可以在避免网络循环的情况下,进行数据包复用和本地链路感知,避免了路由信息局部最优化的出现。尽管本文方法需要额外建立绑定关系表,但在命名绑定关系的帮助下,路由节点无需存储其他节点的过程信息,仅存储数据生产者的相关命名绑定信息。因此,除去必要的链路状态信息用于构建网络拓扑外,本文方法在空间上的占用是可以忽略的。

4 结束语

为了在移动场景下提供稳定路由协议支持,本文提出了一种轻量级的高效NDN路由协议,并在内核层面实现并运行。该方法通过命名绑定建立了路由节点与发布内容之间的绑定关系。同时,为了降低网络通信代价,NBRP利用NDN网内缓存机制,设计了一种可重用网络包格式和增量机制。为了提高动态网络环境的传输稳定性,NBRP利用绑定关系,提出一种链路快速感知机制,在链路故障中快速恢复传输能力。总之,NBRP在内核层面提供了一个可行的NDN路由协议,并在不依赖应用层数据同步协议的情况下满足路由节点同步需求。

参考文献:

- [1] ZHANG L, AFANASYEV A, BURKE J, et al. Named data networking[J]. ACM SIGCOMM Computer Communication Review, 2014, 44(3): 66-73.
- [2] JACOBSON V, SMETTERS D K, THORNTON J D, et al. Networking named content[C]//Proceedings of the 2009 ACM Conference on Emerging Networking Experiments and Technology, Rome, Dec 1-4, 2009. New York: ACM, 2009: 1-12.
- [3] ZHANG L, ESTRIN D, BURKE J, et al. Named data networking (NDN) project: NDN-0001[R]. Xerox Palo Alto Research Center-PARC, 2010: 158.
- [4] TEAM N F D. NFD developer's guide: NDN-0021[R]. 2018.
- [5] WANG L, HOQUE A, YI C, et al. OSPFN: an OSPF based routing protocol for named data networking: NDN-0003[R]. 2012.
- [6] HOQUE A K M M, AMIN S O, ALYYAN A, et al. NLSR: named-data link state routing protocol[C]//Proceedings of the 3rd ACM SIGCOMM Workshop on Information-Centric Networking. New York: ACM, 2013: 15-20.
- [7] WANG L, LEHMAN V, HOQUE A K M M, et al. A secure link state routing protocol for NDN[J]. IEEE Access, 2018, 6: 10470-10482.
- [8] HEMMATI E, GARCIA-LUNA-ACEVES J J. A new approach to name-based link-state routing for information-centric networks[C]//Proceedings of the 2nd ACM Conference on Information-Centric Networking. New York: ACM, 2015: 29-38.
- [9] GARCIA-LUNA-ACEVES J J, MARTINEZ-CASTILLO J E, MENCHACA-MENDEZ R. Routing to multi-instantiated destinations: principles, practice, and applications[J]. IEEE Transactions on Mobile Computing, 2017, 17(7): 1696-1709.
- [10] YANG N, CHEN K, LIU Y. Towards efficient NDN framework for connected vehicle applications[J]. IEEE Access, 2020, 8: 60850-60866.
- [11] LIANG T, SHI J, WANG Y, et al. On the prefix granularity problem in NDN adaptive forwarding[J]. IEEE/ACM Transactions on Networking, 2021, 29(6): 2820-2833.
- [12] AMADEO M, RUGGERI G, CAMPOLO C, et al. Caching popular and fresh IoT contents at the edge via named data networking[C]//Proceedings of the 2020 IEEE Conference on Computer Communications, Toronto, Jul 6-9, 2020. Piscataway: IEEE, 2020: 610-615.
- [13] ZHANG Y, AN X, YUAN M, et al. Concurrent multipath routing optimization in named data networks[J]. IEEE Internet of Things Journal, 2019, 7(2): 1451-1463.
- [14] ALEXA. Top sites[EB/OL]. [2023-01-23]. <https://www.alexa.com/topsites>.
- [15] KARIM F A, AMAN A H M, HASSAN R, et al. Named data networking: a survey on routing strategies[J]. IEEE Access, 2022, 10: 90254-90270.
- [16] LI Z, XU Y, ZHANG B, et al. Packet forwarding in named data networking requirements and survey of solutions[J]. IEEE Communications Surveys & Tutorials, 2018, 21(2): 1950-1987.



邹博文(1997—),男,硕士研究生,主要研究方向为信息中心网络、路由协议。

ZOU Bowen, born in 1997, M.S. candidate. His research interests include information-centric network and routing protocol.



嵩天(1980—),男,博士,教授,CCF高级会员,主要研究方向为网络体系结构、网络安全。

SONG Tian, born in 1980, Ph.D., professor, CCF senior member. His research interests include network architecture and network security.



李天龙(1997—),男,博士研究生,主要研究方向为新一代互联网体系结构、信息中心网络。

LI Tianlong, born in 1997, Ph.D. candidate. His research interests include new generation Internet architecture and information-centric network.



杨雅婷(1994—),女,博士,助理教授,主要研究方向为新一代互联网体系结构、信息中心网络。

YANG Yating, born in 1994, Ph.D., assistant professor. Her research interests include new generation Internet architecture and information-centric network.