基于SDN的胖树数据中心网络的多路径路由算法

农黄武 黄传河 黄晓鹏
(武汉大学计算机学院 武汉 430072)

摘 要 近年来,具有多路径能力的胖树拓扑结构已经被应用在很多数据中心网络(DCNs)中,以提高网络带宽和容错性。但其使用的传统路由协议对多路径路由的支持是非常有限的,并没有充分利用胖树数据中心网络的多余的可用带宽。因此研究了基于SDN的胖树网络的多路径路由。首先提出一个属于线性规划范畴的问题并证明它的NP完全性;然后提出了一个利用软件定义网络架构优点的实用算法,其依赖于一个中心控制器来收集网络状态信息,以作出最优的路由转发决策;最后把算法实现为OpenFlow控制器的一个模块并进行仿真验证。实验结果表明,所提算法无论在提高吞吐量还是减小端到端时延方面都优于传统的基于拓扑感知启发式的多路径算法。

关键词 软件定义网络,多路径路由,负载均衡,OpenFlow控制器
中图法分类号 TP393.2 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2016.6.006

SDN-based Multipath Routing Algorithm for Fat-tree Data Center Networks

NONG Huang-wu HUANG Chuan-he HUANG Xiao-peng
(School of Computer, Wuhan University, Wuhan 430072, China)

Abstract To increase bandwidth and improve fault tolerance, the fat tree topology with multipath capability has been used in many data center networks (DCNs) in recent years. But traditional routing protocols have only limited support for multipath routing, and can not fully utilize the available bandwidth in such networks. This paper studied the SDN-based multipath routing for fat tree networks. Firstly, a linear programming problem was proposed and its NP-completeness was proved. Secondly, a practical solution which takes advantage of the emerging software defined networking paradigm was proposed. Our algorithm relies on a central controller to collect necessary network state information in order to make optimized routing decisions. Finally, the algorithm was implemented as an OpenFlow controller module and was validated by simulation. Experimental result shows that the algorithm outperforms the traditional multipath algorithm based on random assignments both in increasing throughput and reducing end-to-end delay.

Keywords Software defined network(SDN), Multipath routing, Load balance, OpenFlow controller

数据中心网络必须在庞大的服务器群间提供高效的互联,以使数据中心达到理想的经济规模[1]。为了提供充裕的双向带宽,现代数据中心网络通常采用层次型多根网络拓扑,比如胖树[2]。这种层次特性可以使网络便以扩展,同时多根拓扑也因其多路径特性而具有多种路由选择。

在各种层次型多根网络拓扑结构中,胖树因其简单易用性而倍受青睐,近年来已经在很多数据中心设计方案中被采用[3]。图1展示的是一个具有4个层次的四端口交换机胖树结构的网络。在这个网络中,所有交换机具有相同数目的端口(即4个端口),所有链路具有相同大小的带宽。

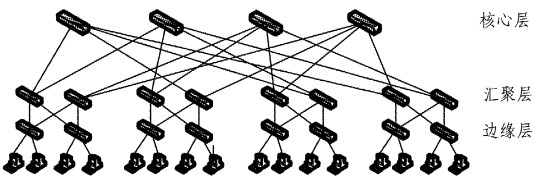

图1 四端口交换机胖树结构

已经得到广泛使用的互联网传统的链路状态和距离向量路由协议不能有效利用胖树拓扑的多路径能力的优点[4]。比如,一些基于IP的路由协议都是仅仅根据数据包的目的地址来计算路由并转发数据包。由此导致的结果就是所有具有相同目的地址的数据包都会经过相同的网络路径。确实也有一些支持多路径的协议,如具有"等价多路径"(ECMP)特性的协议,而这种算法并没有考虑流对带宽的要求,从路径优化的角度来看,这是很有局限的,因为它没有考虑到整个网络的流量状态;最主要的是,这种算法仅仅能从那些具有相同的最小代价的路径中选择出备选的路径,从而导致该算法只能选择数目比较少的路径。我们知道,数据中心网络的路由和互联网路由是有很大的不同的。一般地,互联网路由协议为了减小网络延迟,会更优先考虑选择最短路径。而数据中心网络更少考虑延迟性,因为其节点在地理位置上都是距离很近的本地网络,所以数据中心网络的路由应该先考虑带宽利用率而不是延迟。

到稿日期:2015-07-20 返修日期:2015-08-20 本文受国家自然科学基金(61373040,61173137),教育部博士点基金(20120141110073)资助。
农黄武(1990-),男,硕士生,主要研究方向为计算机网络、SDN,E-mail:wuhoo4han@foxmail.com;**黄传河**(1963-),男,博士,教授,博士生导师,CCF会员,主要研究方向为计算机网络、移动Ad hoc网络、物联网、分布式并行处理等;**黄晓鹏**(1991-),男,硕士生,主要研究方向为网络仿真和SDN网络。

本文的目标是为数据中心网络充分挖掘和利用胖树拓扑网络结构的优点，因此设计了一个欲达到以下 3 个目标的实用高效的多路径路由算法。首先，算法应该可以提高整个网络的带宽利用率，也就是说，算法可以调节和提高整个网络拓扑的流量分布和吞吐量。其次，算法应可以达到更优的流量的负载均衡，这可以阻止网络因产生热点而导致的拥塞。最后，算法应该有较低的时间复杂度，在处理很大的网络流量时，能做出很快的响应。

本文重点研究基于 SDN 的胖树结构数据中心网络的多路径路由的负载均衡，提出了一个利用新兴的软件定义网络(SDN)架构优点的负载均衡的多路径路由算法[5]。

1 相关研究

在已有的为数据中心网络设计的多路径解决方案中，Al-Fares[6] 提出了两个算法：其中一个使用全局最先匹配，另一个使用模拟退火法。第一个算法仅仅选择所有备选路径中第一个满足流需求的路径，该算法很简单，但其缺点是必须准确获取所有节点对间的所有路径的信息。第二个算法使用概率性搜索方法确定最优路径，该算法执行速度很快，但它有时候收敛得很慢。Heller 提出了 ElasticTree 数据中心网络能量管理方法，其包括两种多路径算法：一个是贪婪装箱算法，另一个是拓扑感知的启发式算法。前者先评估可用的路径，然后选择最左边的具有充足带宽的路径；后者基于胖树的拓扑特性使用一个快速启发式，它依赖于流划分。

现存的每一个多路径算法都有其独特的优点，可以解决数据中心网络的某些问题，但是都有顾此失彼的缺点。全局最先匹配算法，总是选择第一条满足流需求的路径，但是该路径可能不是最优路径，达不到全局最优；模拟退火法，可以找到全局最优路径，但其收敛速度可能很慢，并不能适应响应速度要求很高的网络环境。对于 ElasticTree 数据中心管理方法中提出的贪婪装箱算法，每次数据流到来时都需要重新评估可用路径，开销比较大；基于胖树拓扑特性的启发式算法依赖于流划分，也没有考虑整个网络的流量状态。总之，上述多路径算法都不能根据当前网络的全局负载情况，在保证全局最优的条件下，实现高效、快速的多路径负载均衡转发。

本文描述的算法同样考虑多路径，首先使用基于深度优先查找的思维来计算网络中所有节点对的多路径信息并缓存以提高响应速度；然后在新数据流到来时，使用最差适应法确定备选路径。相比前面的算法，本文使用 SDN 技术提供了一个获取网络级流量负载均衡的快速有效方法。

2 问题描述和分析

2.1 负载均衡多路径路由问题

一个胖树数据中心网络可看作是一个有向图 $G=(H\cup S;E)$，其中，H 是主机集合，S 是交换机集合，E 是交换机之间或交换机与主机之间链路的集合。每一条边 $(v_i,v_j)\in E$ 都有一个非负整数值 $c(v_i,v_j)>0$，表示相应链路的可用带宽。用 $F=\{F_1,F_2,\cdots,F_n\}$ 表示网络中所有流的集合。一条流 F_k $(1\leqslant k\leqslant n)$ 定义为一个三元组 (s_k,d_k,b_k)，$s_k\in H$ 代表源主机，$d_k\in H$ 代表目的主机，b_k 代表所需的带宽。$f_k(v_i,v_j)\in(0,1)$ 代表流 F_k 是否经过链路 (v_i,v_j)。

负载均衡多路径路由问题可以表示成一个属于线性规划的问题，其目的是最小化网络中所有链路的最大负载。负载均衡多路径路由问题可以形式化地表示如下。

算法应该满足如下限制：

$$\sum_{k=1}^{n} f_k(v_i,v_j)d_k\leqslant \max\leqslant c(v_i,v_j),\ \forall (v_i,v_j)\in E \tag{1}$$

$$\sum_{v_i\in H\cup S} f_k(v_i,v)=\sum_{v_j\in H\cup S} f_k(v,v_j),\ \forall k\in\{1,\cdots,n\},\ \forall v\in H\cup S\backslash\{s_k,d_k\} \tag{2}$$

$$\sum_{v_j\in H\cup S} f_k(s_k,v_j)=\sum_{v_i\in H\cup S} f_k(v_i,d_k)=1,\ \forall k\in\{1,2,\cdots,n\} \tag{3}$$

式(1)定义了最大负载，同时也表明了链路容量的限制，即对链路需求的总带宽和不超过链路的可用带宽。式(2)表明流守恒限制，即在一个中间节点，任何流的数目都不会改变。式(3)表示流需求满足限制，即对于任何流，从源节点出去的流量和进入目的节点的流量都是相同的。

2.2 问题的 NP 完全性

本小节证明负载均衡多路径路由问题在胖树拓扑中是一个 NP 完全问题。

证明：证明结果是从一个 NP 完全的划分问题中推导出来的[7]。划分问题决定一个整数集合 $X=\{X_1,X_2,\cdots,X_n\}$ 是否可以划分为两个子集 X_s 和 $X\backslash X_s$，使得 X_s 中所有元素总和等于 $X\backslash X_s$ 中所有元素总和，如式(4)所示。

$$\exists X_s\subseteq X,\ \sum_{x_i\in X_s} x_i=\frac{1}{2}\sum_{x_i\in X} x_i \tag{4}$$

为了说明问题，本文使用集合 X 划分问题的一个实例，然后构造一个胖树网络中一个负载均衡多路径路由问题的实例。主要思路是构造一组两个主机之间的数据流，每个数据流所需带宽(b_k 如上文定义)分别对应于 X 集合中的一个整数，在这两个主机间为这一组流留有两条不重合的路径。若有一个整数集的划分，则把流分配到相应的路径上，以达到最优负载均衡；反之亦然。

图 2 展示的是一个 4 端口交换机胖树结构的一个部分，主机 1 和主机 3 之间有 ABD 和 ACD 两条不重合路径。

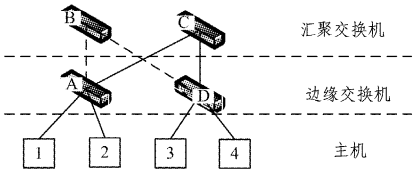

图 2 4 端口交换机胖树拓扑的一个部分

详细的推导过程如下：

步骤 1 建立一个具有 K 个端口交换机的胖树拓扑，每一条链路的带宽容量设置为无限大，以避免违反链路带宽的容量限制。

步骤 2 选择两个在同一个部分但不同边缘交换机的主机，并标识为 h_1 和 h_2，然后把所有的流都指派到它们之间转发。

步骤 3 构造 n 条流，每一条对应于集合 X 的一个整数，对于每条流 F_i，有 $s_i=h_1$，$d_i=h_2$，$b_i=x_i$，其中 $x_i\in X$。

步骤 4 构造额外的 $K/2-2$ 条流，对于每条流 F_k 则设置源节点 $s_k=h_1$，目的节点 $d_k=h_2$，数据流所需带宽 $b_k=\sum_{x_i\in X} x_i/2$。

假如对于一个集合 X 的划分问题有解 X_s，对于一个 K 端口胖树中的主机 h_1 和 h_2，它们之间有 $K/2$ 条互不重合的路径，

每一条经过一个汇聚交换机。在上述的第 4 步推导过程中，构造了 $K/2-2$ 条流，每一条流的带宽需求是 $\sum_{x_i \in X_s} x_i = \frac{1}{2} \sum_{x_i \in X} x_i$。为了整个网络的负载平衡，每一条流都会被分派到不同的路径中，从而只剩下两条路径。

步骤 5　对于第 3 步构造的 n 条流，如果 $x_i \in X_s$，则相应的流 F_i 被分派到剩余两条路径中的第一条，否则将会被分派到第二条路径。这样就可以找到一个解决方案，对于 n 条数据流，据其所需带宽为 b_k，分派到不相交的转发路径，并且每条路径承载的转发任务是全局均衡的，即 $\sum_{x_i \in X_s} x_i = \frac{1}{2} \sum_{x_i \in X} x_i$，并达到最优的负载平衡。

假设负载平衡多路径路由问题存在最优解使得最大负载为 $\sum_{x_i \in X_s} x_i = \frac{1}{2} \sum_{x_i \in X} x_i$，那么第 4 步中构造的 $K/2-2$ 条流将被分派到不同的路径中，而第 3 步中构造的两条流必须共享剩余的两条路径，据此本文可以构造两个相应的子集 X_s 和 $X \backslash X_s$。这样就找到了划分问题的一个解。

3　基于 SDN 的算法

本节将提出一个基于 SDN 的负载平衡多路径路由算法。SDN 可以使用 OpenFlow 实现：在网络中有一个中心 OpenFlow 控制器和 OpenFlow 交换机通过 OpenFlow 协议进行通信。控制器可以收集网络状态，并指导所有数据流在网络中进行转发。本文的主要思想是利用中央控制器收集每一条链路的流量负载信息，以便所提算法可以做出全局最优的路由决策。当有一条流进入系统时，第一个接收到该流的交换将通知控制器。控制器将计算所有可以转发该流的路径，比较路径的瓶颈链路的负载，然后选择一条最小负载的路径，同时对后续到来的数据流使用上节方法进行转发。转发路径确定之后，控制器将通过路径途经的所有交换机并更新它们的流表，以便该流的后续数据包可以经过相同的路径进行转发。

3.1　设计思路

为了优化数据中心网络的带宽利用率，需要对网络的所有可用资源有一个全局的了解[8]。为此，本文把负载均衡多路径路由方案放在能够与网络中交换机通过 OpenFlow 协议通信的中央控制器中实现。在 OpenFlow 数据流有 10 个域组成的字段唯一定义，包括源目的 IP 地址、端口号等等。控制器通过查询、插入、删除、修改 OpenFlow 交换机流表的表项来控制流在系统中的转发。更重要的是，中央控制器可以收集整个网络交换机的带宽和流信息，以作出全局最优的路由决策，然后通过修改路径途经的交换机的流表来指导流在网络中转发。

3.2　算法的具体实现代码

算法　getForwardPath

```
输入:源主机 IP,目的主机 IP
输出:转发路径及其瓶颈链路带宽
Begin
  if(S 和 D 同属一个边缘交换机)
    return path(path(S-D),infinity);
  else
    paths= getAllpath(S,D);
    for(int i=0;i<paths.length;i++);
      bottleneck[i]=paths[i].getMinBandwidth();
    end for
    for(int j=0;j<bottleneck.length;j++)
      minBandwidth = minBandwidth<bottleneck[i]? bottleneck
      [i]:minBandwidth;
      minpath=paths[i];
    end for
    return resultPath(Path,minBandwidth)
  end if
End
```

3.3　算法的时间复杂度

在一个 k 端口交换机的胖树拓扑中，有 $k^3/4$ 个主机和 $5k^2/4$ 个交换机，在任意的两个主机之间，最多有 k 条不同的路径，对于每一条路径，算法为了找出瓶颈链路最多检查 8 条链路(路径的最大长度为 8)。所以路由进程的时间复杂度是 $O(k)$。换言之，对于一个具有 n 个节点的网络，本文提出的多路径路由算法的时间复杂度为 $O(n^{1/3})$。

4　仿真实验结果

为了验证本文的设计，把负载均衡多路径路由算法实现成 OpenFlow 控制器的一个模块，并在 Mininet 中进行实验验证。

本文把算法实现为 Floodlight 控制器的一个模块。该模块有两个功能：监控链路状态和为新到来的流计算转发路径。

Floodlight 控制器模块周期性地调度监控事件，相应的控制器向网络中每个交换机发送 OFStatisticsRequest 命令，每隔特定 T 秒，每个交换机将会返回 OFStatisticsReply 信息。控制器处理返回的信息，其中包括抽取的交换机 ID、端口号、总发送字节数及返回信息的接收时间。对每个端口的流速率的评估可以根据这些信息计算出来。信息占用的空间是很小的：OFStatisticsRequest 信息只有 8B，而 OFStatisticsReply 信息只有 104B。端口信息统计请求和回复的代价是很低的。在一个 Mininet 的 4 端口或 8 端口($k=4$ 或 8)交换机组成的的胖树拓扑中，发现如果使 T 更小(比如从 6s 到 2s)，CPU 的利用率仅仅提高大约 2%～5%。

本文选择 Mininet2.1.0 作为仿真器，使用 4 端口和 8 端口($k=4$，8 和 16)交换机胖树拓扑作为研究对象，使用一个 Python 脚本执行自动测试过程，包括交换机初始化、连接 Floodlight 控制器、主机初始化、流量自动产生。使用 DITG 产生 UDP 流量，每个主机都有一个 DITG 服务端和客户端。基于流量负载，Python 脚本将调用 DITG 组件选择一个随机的时间从一个 DITG 服务端发送 UDP 流到一个 DITG 客户端。

本文实验胖树拓扑中，交换机端口数 $k=4$，8 和 16，一个胖树包含 $k^3/4$ 个主机和 $5k^2/4$ 个交换机。如果使用 $k=4$，那么就有 16 个主机和 20 个交换机。设置所有的链路的带宽为 100Mbps，使用 DITG 产生 UDP 流，源主机和目的主机都是随机地选举，每个流的数据包产生速率服从于均一分布，其中最小为 8Mbps，最大为 12Mbps，平均为 10Mbps。每个流的持续时间为 30s。通过控制服从于指数分布的流达到时间，来改变网络的输入流量负载。在本实验中，逐步增加注入网络流量的速率，变化范围是网络链路总带宽的 1%～90%。16 个主机，每个 100Mbps，则总带宽有 1.6G。对于每次输入量进行 15 次左右的实验，每次持续 1000s。

(下转第 76 页)

刘忠伟，章毓晋. 综合利用颜色和纹理特征的图像检索[J]. 通信学报，1999，20(5)：37-41

[5] Haralick R M，Shanmugam K，Dinstein I. Texture Features for Image Classification[J]. IEEE Transactions on Systemse，Man and Cybernetics，1973，3(6)：610-621

[6] Gao Cheng-cheng，Hui Xiao-wei. GLCM-Based Texture Feature Extraction [J]. Computer Systems & Applications，2010，19(6)：195-198(in Chinese)
高程程，惠晓威. 基于灰度共生矩阵的纹理特征提取[J]. 计算机系统应用，2010，19(6)：195-198

[7] Equitz W，Niblack W. Retrieving Images from a Database Using Texture-algorithms from the QBIS System：Technical Report RJ：9805[R]. 1994

[8] Wang Zhi-zhi. Remote Sensing Object Classification Algorithm Based on the Fusion of Texture Features and Spectral Features [D]. Xi'an：Xidian University，2010(in Chinese)
王知莹. 基于纹理及光谱信息融合的遥感图像分类方法研究[D]. 西安：西安电子科技大学，2010

[9] Jiao Peng-peng，Guo Yi-zheng，Liu Li-juan，et al. Implementation of Gray Level Co-occurrence Matrix Texture Feature Extraction Using Matlab [J]. Computer Technology and Development，2012，22(11)：169-171，175(in Chinese)
焦蓬蓬，郭依正，刘丽娟，等. 灰度共生矩阵纹理特征提取的Matlab实现[J]. 计算机技术与发展，2012，22(11)：169-171，175

[10] Yuan Li-hong，Fu Li，Yang Yong，et al. Analysis of Texture Feature Extracted by Gray Level Co-occurrence Matrix [J]. Journal of Computer Applications，2009，29(4)：1018-1021(in Chinese)
苑丽红，付丽，杨勇，等. 灰度共生矩阵提取纹理特征的实验结果分析[J]. 计算机应用，2009，29(4)：1018-1021

(上接第 34 页)

本文将对负载均衡多路径路由算法(本文提出，标记为LBMP)和 ElasticTree 中的拓扑感知启发式算法(标记为TAH)进行比较，两个算法均基于胖树的拓扑特性使用一个快速启发式，但 TAH 依赖于流划分。在这 15 次左右的实验中，LBMP 在网络吞吐量和网络延迟方面优于 TAH，如图 3 和图 4 所示。在相同的输入负载下，本文的算法在胖树结构变大时端到端延迟只有微小的上升，这是因为胖树路径的平均长度变大，而吞吐量保持基本一致，这是因为本文算法可以均衡网络的流量负载。但是对于 TAH，结果并不是这样的：在更高的输入负载下，开始形成拥塞从而导致网络吞吐量大幅度降低，特别当网络更大时，该现象更加明显。

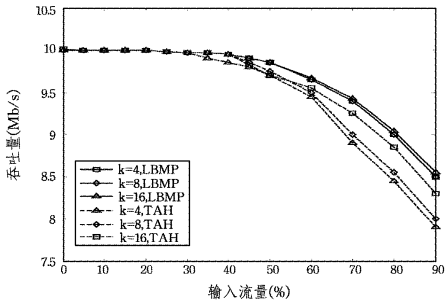

图 3　吞吐量比较

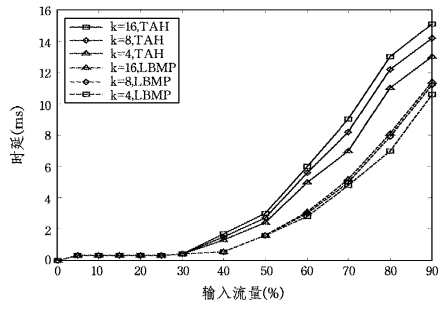

图 4　时延比较

结束语　具有多路径能力的胖树拓扑因其充裕带宽和高容错性，在数据中心网络中已经成为一种很受欢迎的网络拓扑。本文提出了基于 SDN 的负载均衡多路径路由算法，它可以在胖树拓扑中更好地分发流量负载和充分利用有效带宽。首先提出一个属于线性规划范畴的问题，并通过整数划分问题推导证明其 NP 完全性。进而提出了一个基于 SDN 的数据中心网络的多路径路由算法，它是一个实用性算法。该算法利用中央控制器去收集整个网络的状态信息，进而为每一个新进入网络的数据流做出最优的负载均衡的路由决策。最后在 Mininet 仿真平台中验证了算法的正确性和有效性。实验结果表明，本文的算法在吞吐量和端到端延迟方面的性能均优于拓扑感知启发式多路径算法。

未来将计划做更多有效性验证实验，特别是比较 SDN 仿真平台和真实 SDN 网络，以及和其他多路径算法的区别；同时使用真实流量负载进行实验。

参 考 文 献

[1] Fei Y. Introduction to the development of data center network architecture[J]. Network Security Technology & Application，2014，22(6)：23-28

[2] Qiao L，Yin X H，Zhuo D I，et al. Research on SDN Network Architecture for Electric Power Big Data Platform[J]. Electric Power Information & Communication Technology，2015，12(6)：1-6

[3] Sun Y，Cheng J，Shi K. Data Center Network Architecture[J]. Zte Communications，2013，11(5)：5-9

[4] Zahid F，Gran E G，Bogdanski B，et al. A Weighted Fat-Tree Routing Algorithm for Efficient Load-Balancing in Infini Band Enterprise Clusters[C]// 2015 23rd Euromicro International Conference on Parallel，Distributed and Network-Based Processing(PDP). IEEE，2015：35-42

[5] Chemeritskiy E，Smelansky R. On QoS management in SDN by multipath routing[C]// 2014 First International Science and Technology Conference (Modern Networking Technologies) (MoNeTeC). IEEE，2014：1-6

[6] Al-Fares M，Radhakrishnan S，Raghavan B，et al. Hedera：dynamic flow scheduling for data center networks[J]. Proceedings of the 7th USENIX Symposium on Networked Systems Design and Implementation，2010，19(1)：1-15

[7] Garey M R，Johnson D S. Computers and Intractability：A Guide to the Theory of NPCompleteness[M]. New York，NY，USA：W. H. Freeman & Co，1990

[8] Ji P N，Qian D，Kanonakis K. Design and Evaluation of a Flexible-Bandwidth OFDM-Based Intra-Data Center Interconnect [J]. IEEE Journal of Selected Topics in Quantum Electronics，2013，19(2)：33-39