

一种应用于无线软件定义网络的流表数量约减方法

孔剑锋¹ 刘振海²

¹(盐城工学院教务处 江苏 盐城 224051)

²(盐城工学院信息工程学院 江苏 盐城 224051)

摘要 无线软件定义网络 W-SDN(Wireless Software Defined Network)是软件定义网络 SDN 技术在无线网络环境中的应用。近年来已经在无线传感器网、5G 通信网等网络中获得广泛的应用。在 W-SDN 架构中,无线节点需要根据中央控制器下发的流表,对数据进行转发。通常情况下,无线节点的内存容量和电量较小,所以当网络规模较大或用户较多时,过多的流表数量可能会对无线网络节点带来较大的负担,导致能耗上升,性能下降,寿命缩短等问题。因此,如何尽量约减无线节点中存储的流表数量,对于 SDN 技术在无线网络中的应用具有重要意义。将多协议标签交换网络 MPLS(Multi-Protocol Label Switch)中的标签交换路径 LSP(Label Switch Path)概念引入 W-SDN,提出一种基于 LSP 重用的流表约减方法。该方法采用多级标签堆栈技术和源路由技术,能够有效减少无线节点中存储的流表数量且不会导致数据包头部过长。实验仿真结果证明,相比于传统的流表约减方法,该方法具有更好的约减效果。

关键词 无线网络 软件定义网络 LSP 重用 流表约减 多协议标签交换

中图分类号 TP393

文献标识码 A

DOI:10.3969/j.issn.1000-386x.2019.02.031

A FLOW TABLE REDUCTION FOR W-SDN

Kong Jianfeng¹ Liu Zhenghai²

¹(Teaching Affairs Office, Yancheng Institute of Technology, Yancheng 224051, Jiangsu, China)

²(School of Information Technology, Yancheng Institute of Technology, Yancheng 224051, Jiangsu, China)

Abstract Wireless software definition network(W-SDN) is the application of SDN in wireless network environment. In recent years, it has been widely used in wireless sensor networks, 5G communication networks and other networks. In W-SDN architecture, wireless nodes need to forward data according to the flow table issued by the central controller. Usually, the memory capacity and power of wireless nodes are small, so when the network scale is large or the number of users is large, the excessive number of flow tables may bring a greater burden on wireless network nodes, resulting in increased energy consumption, performance degradation, shortened lifetime and other issues. Therefore, how to reduce the number of flow tables stored in wireless nodes is of great significance for the application of SDN technology in wireless networks. In this paper, we introduced the concept of label switch path(LSP) in multi-protocol label switching(MPLS) into W-SDN, and proposed a flow table reduction based on LSP reuse. The method used multi-level label stacking technology and source routing technology, which could effectively reduce the number of flow tables stored in wireless nodes without causing the header of the packet to be too long. The simulation results show that compared with the traditional flow table reduction method, the proposed method has better reduction effect.

Keywords Wireless network Software defined network LSP reusing Flow table reduction MPLS

0 引言

近年来,无线 SDN(W-SDN)技术被广泛应用于传感器网络^[1-2]、5G 通信网络、M2M 通信等应用场景中,用来实现可控路由、灵活流量工程和集中式网络管理等功能。然而,由于 SDN 技术最初的设计目标为有线局域网场景,因此在将 SDN 技术应用于无线网络环境之前,仍然存在一些问题需要解决。其中,如何有效地减少 W-SDN 中所需的流表数量从而提高网络的性能和效率,是当前最迫切需要解决的问题之一。

由于 W-SDN 继承了 SDN 的框架以及基于流表的转发模式,因此为了实现 W-SDN 中对于流量路由的可控性,W-SDN 的中央控制器必须为每流在路由的每一跳上安装流表。因此,每个无线节点必须以流表的形式存储较多的状态信息,从而导致三个问题:首先,通常情况下,无线节点的存储空间比有线网络中的网络设备小得多,从而只能安装少量流表在每个节点中。因此,网络可容纳的最大流量数量相当有限。其次,当流表数量较多时,流表匹配过程可能消耗过多的能量,从而缩短无线节点的使用寿命。第三,无线节点的处理能力通常较弱,以至于太多的流表可能会带来较大的查表延迟。

一般情况下,解决 W-SDN 流表数量的方法有两种。第一种方式是流量工程^[3-5],通过这种方式流量可以更加均匀地分布,因此每个节点中的流表数量也获得了平均,从而能够避免某个节点存储过多的流表。第二种方式是源路由,它将路由信息存储在分组报头中,而不是无线节点的流表中^[6-7]。因此,使用源路由,无线节点不需要存储流量状态信息,但是需要数据包带有更长的包头,从而导致带宽浪费和可扩展性的降低。针对以上两种方法的不足,在本文中,我们提出了一种称为 LSP(Label Switch Path)重用的新型机制,相比于以上两种方法,它可以通过更短的数据包头部,实现最好的流表约减效果。为了实现这种方法,我们将 MPLS 中的标签交换路径 LSP(Label Switch Path)引入到 W-SDN 中。基于 Openflow 支持的标签弹出(Pop),压入(Push)和交换(Swap)操作,我们在数据包头中嵌入多个标签,并使用生存时间 TTL(Time To Live)控制数据包在某个 LSP 上传输的跳数,从而使得数据包能够在不同的 LSP 之间切换。因此,同一条 LSP 可以具有不同源和目的地的流量重用,单个流量可以使用不同的 LSP 完成端到端的传输。在这种机制下,安装在

每个无线节点中的流表仅与通过节点的 LSP 相关,而与流量的数量无关。因此,可以有效减少每个节点维护的状态信息。

综上所述,本文的贡献主要如下:

(1) 提出了一种 LSP 重用方法来解决无线 SDN 中的流表数量约减问题;给出了该方法的详细设计和实现,并在软件平台上实现了原型系统。

(2) 研究了最优的 LSP 规划问题;建立了最小化流表数量的数学模型;提出了求解数学模型的启发式算法。

(3) 采用仿真实验,评估了不同 W-SDN 流表数量约减方法的性能,并证明了本文提出方法的可行性和有效性。

1 无线软件定义网络中的流表约减

从逻辑上讲,W-SDN 主要由一个中央控制器和一些 SDN 使能的节点组成。通过中央控制器根据标准 SDN 范式安装的流表,每个节点可以实现针对每流的可控性。通过这种机制,每个节点的控制功能集中在中央控制器中,因此控制平面和无线网络的转发平面是分离的,而这种分离机制能够为无线通信网络带来网络的可全局视图、网络可编程性、网络可虚拟化等好处。无线软件定义网络(W-SDN)是 SDN 技术在包含多个无线和有线网络的无线异构网络中的应用,其基本结构如图 1 所示。目前,已经存在多种 W-SDN 架构或项目,如 SoftCell^[8]、SoftRAN、MobileFlow、OpenRadio^[9]等。但是,这些架构均没有关注减少无线节点中流表数量的重要性。为了解决这个问题,本文提出了一种称为 LSP 重用的新机制。

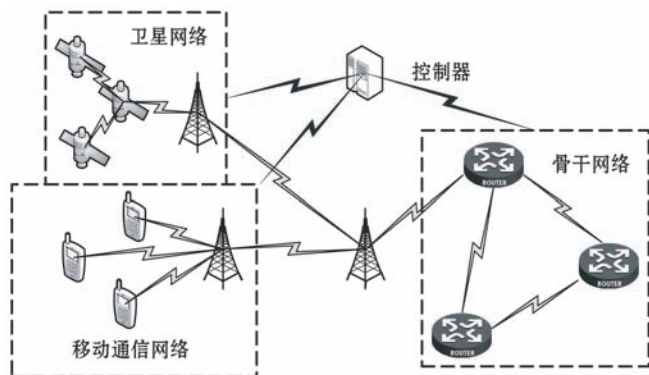
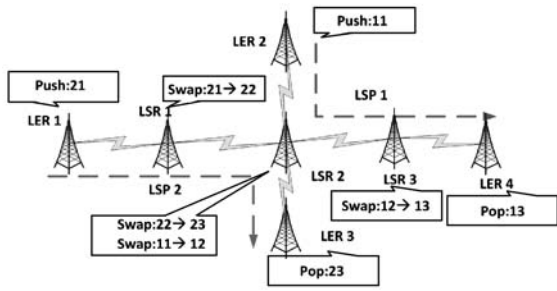


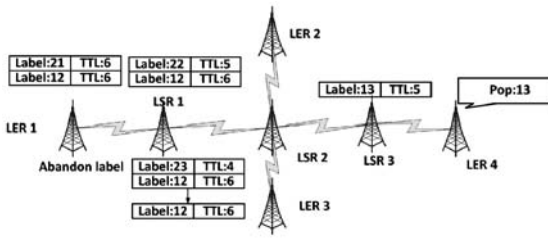
图 1 W-SDN 的体系结构

1.1 示例说明

如图 2 所示,我们首先用一个例子来说明 LSP 重用的工作机制。



(a) 无线网络拓扑以及两条 LSP



(b) LSP 重用的工作过程

图 2 LSP 重用机制的示例网络

假设一个无线网络如图 2(a) 所示。有两个 LSP 是 LSP1: LER2 → LSR2 → LSR3 → LER4 和 LSP2: LER1 → LSR1 → LSR2 → LER3。我们假设有一个新的流从 LER1 传输到 LER4。但是,没有 LSP 连接 LER1 和 LER4,因此需要使用 LDP、RSVP 等建立新的 LSP。因此,随着流量的增加,LSP 的数量将会扩大。LSP 的数量可能会达到 n^2 ,其中 n 是标签边缘路由器(LER)的数量。我们认为,通过 LSP 重用技术可以有效降低网络中需要的 LSP,因为 LSP 可以被不同目的地的流共享。如图 2(b) 所示,在启用 Openflow 的 LER1 中,两个 MPLS 标签被堆叠到数据包标头中。最外面的标签沿着 LSP1 发送分组,而第二层标签沿着 LSP2 发送分组。我们将阈值设置为 4。基于 Openflow 提供的数据包编辑能力,最外层标签的 TTL 设置为 6,内层标签的 TTL 设置为 6。当数据包从 LER1 传输到 LSR2 时,沿着 LSP2 交换机的路由器最外面的标签如图 2(b) 所示,当数据包到达 LSR2 时,最外层标签的 TTL 减少到 4,LSR2 脱去最外层标签,并露出内层标签。因此,LSR2 检查内层标签并通过 LSP1 传输数据包。然后沿着 LSP1 的路由器交换内部标签,直到数据包达到 LER4。从以上例子可以看到,在传统情况下,需要安装 4 个新的流表项来完成从 LER1 到 LER4 的传输。但是,使用 LSP 重用技术,可以使用 LSP2 和 LSP1 的片段来完成整个端到端传输,因此不需要安装新的流表。

1.2 LSP 重用

标签交换路径(LSP)指的是在 MPLS(Multi-Protocol Label Switch)网络中,由包头中 MPLS 标签标识的一条已建立的逻辑隧道。由于 Openflow 协议完全支持 MPLS 操作,如标签压入,弹出和交换等,因此我们

可以将 MPLS 和 LSP 直接部署于 W-SDN,而无需对软件或硬件进行任何修改。使用 LSP 传输数据类似于源路由,但每个无线节点维护的状态信息(也就是流表数量)不低于使用分布式路由协议的状态信息,因为它必须建立 n^2 条 LSP 来实现整个网络的互联,其中 n 是网络中标签边缘路由器(LER)的数量。我们对传统的 MPLS 转发模式进行了修改,并通过在数据包报头中嵌套多个标签并利用 TTL 弹出最外层标签实现了一种 LSP 重用机制。LSP 重用机制主要需要执行三个功能:

- 1) LSP 规划 LSP 规划过程由中央控制器执行,而非采用传统的 LDP 或 RSVP 协议。LSP 规划过程的目标是用尽可能少的 LSP 覆盖整个网络。
- 2) 路由编辑 路由编辑功能主要在中央控制器和网络边缘节点上执行。路由编辑的主要任务是将 MPLS 标签压入数据包报头并设置标签的 TTL 值。根据用户需求和网络状况,路由计算由中央控制器或分布式路由算法完成,并通过 Openflow 协议以流表的形式发送并安装到流量流入的网络边缘设备。之后,包头编辑过程在该设备处完成,如图 3 所示。

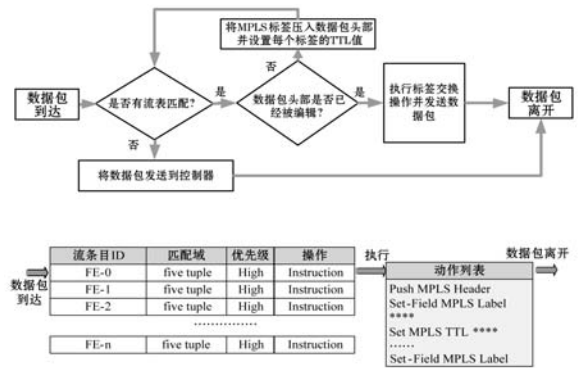


图 3 数据包头部中的标签编辑流程与流表结构

- 3) 分组转发 该过程主要在中继节点或基站进行。但是,与传统的 MPLS 报文转发不同,我们设置 TTL 阈值来弹出 MPLS 标签。正常情况下,路由器按照最外层标签转发数据包,当标签的 TTL 等于阈值时,路由器将弹出最外面的标签。继续按照下一层标签转发数据包,直到数据包到达目的地。处理流程如图 4 所示。

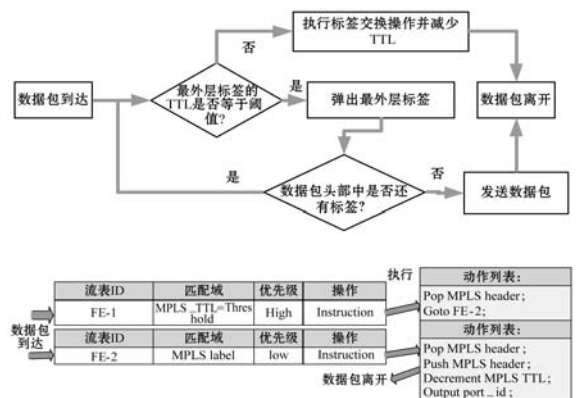


图 4 数据包转发过程中的标签和 TTL 处理流程与流表结构

这一转发逻辑可能需要对传统的网络设备进行更改,或引入 Openflow 的支持来实现。然而,通过最终的实验仿真结果,我们认为这些开销与成本与 LSP 重用方法带来的改进相比是值得的。

2 LSP 规划问题优化模型

为了最大限度地减少网络中需要的流表数量,在使用 LSP 重用机制时,应该对底层 LSP 进行最优规划。我们使用有向图 $G(V, E)$ 来表示网络,其中 V 是顶点集合, E 是链路集合。二进制变量 e_{ij}^p 用于表示链路 (i, j) 和 LSP p 之间的关系。当 LSP p 包含链路 (i, j) 时,等于1,否则等于0。另外,我们用 P 来表示 LSP 的集合。

LSP 规划的目标是,我们希望最小化网络中的 LSP 数量并使得每条 LSP 尽可能短,从而最小化网络中所需的流表数量。其中,网络中使用的 LSP 的总长度可以表示为 $\sum_p \sum_{(i,j)} e_{ij}^p$ 。然后,给定一定数量 LSP 数量的集合 P ,我们可以得到最优的 LSP 规划数学模型如下:

$$\min \sum_p \sum_{(i,j)} e_{ij}^p \quad (1)$$

目标函数表示优化模型希望最小化 LSP 集合中所有 LSP 的总长度,以便在给定的 LSP 数量下,网络中所需的流表数量能够最小。

$$\sum_p e_{ij}^p \geq 1 \quad \forall (i, j) \in E \quad (2)$$

式(2)保证网络的每个链路至少被一个 LSP 覆盖。

$$\sum_i e_{ij}^p - \sum_k e_{jk}^p = \begin{cases} -1 & j \text{ is the source node of LSP} \\ 0 & j \text{ is the intermediate node of LSP} \\ 1 & j \text{ is the end node of LSP} \end{cases} \quad (3)$$

$\forall p \in S, \forall (i, j) \in E, \forall (j, k) \in E$

式(3)确保每个 LSP 都是连续的。

$$\sum_i e_{ij}^p < 2 \quad \forall p \in S \quad (4)$$

$$\sum_{(i,j)} e_{ij}^p \leq 1 \quad \forall p \in S \quad (5)$$

$$\sum_j e_{ij}^p \leq 2 \quad \forall p \in S \quad (6)$$

式(4) - 式(6)保证了 LSP 无环路。为了找到包含最少 LSP 的集合 P ,我们需要让 $|P|$ 从0增长到 $|E|$,并针对每个给定的 P 求解上述数学模型。但是,由于数学模型中的变量是 $|P| \times |N|$ 维的,因此在大规模网络场景中,计算复杂度会较高,然而 LSP 规划过程一般需要快速完成,以防止网络发生链路故障或路由变化等时业务中断,因此需要提高 LSP 规划算法的效率。

出于这个原因,我们提出了一个启发式算法,解决 LSP 的最优规划问题。算法的基本思想是,在每次迭代中,规划一个无环 LSP,使其可以覆盖尽可能多的未覆盖的链路,直到所有链路都被 LSP 覆盖。该算法的伪代码如下所示:

算法1 LSP 规则

Input: overlay, topology;

Output: P ;

```

1:  boolean stop = false; boolean loop = false
2:  while stop! = true do
3:    for source = 0 to  $|V|$  do
4:      while loop! = true do
5:        loop = true;
6:        for j = 0 to  $|V|$  do
7:          for k = 0 to  $|V|$  do
8:            if node k and j are connected and the route from
              source to k doesn't contain node j then
9:              if cover(source, k) + cover(k, j)
                > cover(source, j) then
10:                 route(source, j) =
                  route(source, k) + route(k, j)
11:              end if;
12:            end if;
13:          end for;
14:        end for;
15:        if there is any route(source, j) is changed then loop
                  = false;
16:        end if;
17:      end while;
18:    end for;
19:    find the optimal route and update the overlay topology with
      the optimal route;
20: if all links are covered then stop = true
21: else if stop = false;
22: end if;
23: end while;
```

在算法中,我们使用两种拓扑:覆盖拓扑和连接拓扑。前者表示链路是否被 LSP 覆盖,后者表示网络中每个节点的连通性。在算法的过程中,覆盖拓扑将不断更新,但连接拓扑将保持不变。从第4行到第17行的循环中,我们使用动态规划从图中找出从源到任何终点(LER)的最佳路径,该路径能够覆盖最多的未覆盖链路。然后,通过从第3行到第18行的循环,我们计算每个点的最佳路径,并找到覆盖当前图中大多数链路的最佳路径。每次我们计算出最优路径后,我们根据最佳路径删除覆盖拓扑中选定的链路,如第19行所示。当所有链路被 LSP 覆盖时,算法停止(第20 - 21行)。

3 实验仿真

在本节中,我们使用仿真实验来评估所提出方法的性能。不失一般性,我们在仿真中使用 $n \times n$ 矩阵网络拓扑,如图 5 所示。

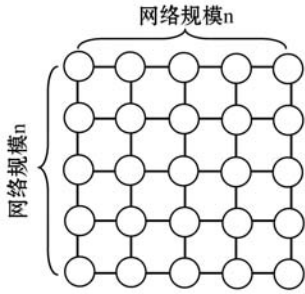


图 5 仿真实验拓扑图

我们选择的比较方法是多点对点隧道合并方式 (MP2P)^[10-12],这是最常用的转发表约简方法;以及 Splano 等提出的 MPLS 网络中最新的转发表约减方式—非对称合并隧道方法 (AMT)^[13-14]。图 6 给出了不同路由数目的情况下,网络所需的最小流表数量。

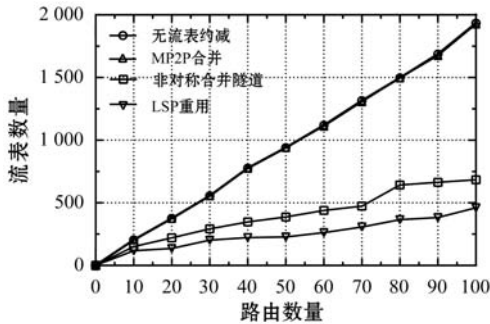


图 6 不同路由数量下的流表数量

根据图 6 所示,流表数量随路由数量的增加而增加,但是无论路由数量多少,采用 LSP 重用方法总能达到最小的流表数量。同时,随着路由数量的增加,LSP 重用相比于其他方式的优势逐渐扩大。当路由数量为 100 时,与 MP2P 合并相比,LSP 重用可以将流表数量减少 71.9%,与 AMT 方法相比,可将流表数量减少 58.3%。之后,我们评估不同网络规模下不同方法的性能。

路由数量设置为 100,不同网络规模下的流表数量如图 7 所示。从图 7 中可以看出,随着网络规模的增加,所有三种方法的流表数量都会增加。同时,LSP 重用方法在任何网络规模下始终能够达到最小的流表数量。因此可以证明,本文提出的方法具有良好的可扩展性,可以应用于大规模无线网络。除了减少网络中流表数量之外,我们还希望流表数量的分配尽可能均匀,以避免在某些特定节点处安装太多流表,从而导致性能或能耗瓶颈。

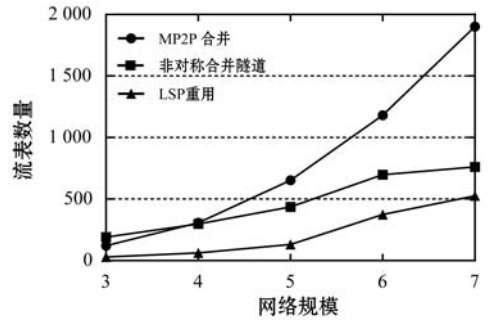
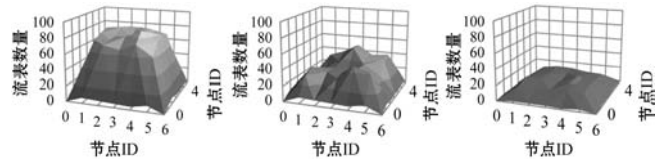


图 7 不同网络规模下的流表数量

图 8 显示了采用三种不同方法时的流表数量分配情况,从图 8 的结果可以看出,LSP 重用方法不仅可以实现最小的流表数量,而且可以实现比其他方法更加均匀的流表分布。



(a) MP2P 合并方法 (b) 非对称合并隧道方法 (c) LSP 重用

图 8 采用不同方法时的流表数量分布

此外,当使用 LSP 重用时,数据包头的长度等于仅在最坏情况下等于使用源路由方法时的长度。所以 LSP 重用导致的带宽浪费比源路由造成的带宽浪费要小得多。在不同最大包长的情况下,两种方法造成的带宽浪费如图 9 所示。

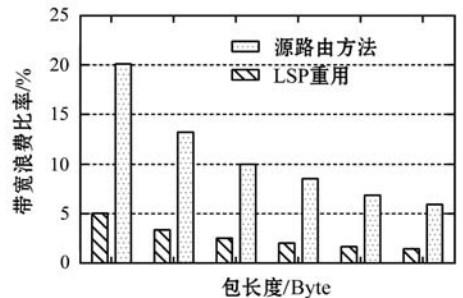


图 9 不同包长下的带宽浪费比率

那么可以得出这样的结论:LSP 重用可以实现类似于源路由的路由控制能力,而只需要更短的头部长度。

4 结 语

本文研究了 W-SDN 中的流表数量约减问题,提出了一种基于 MPLS 网络中 LSP 概念和 Openflow 网络编程能力的 LSP 重用方法。通过在数据包头部中嵌套多个 MPLS 标签并设置这些标签的 TTL 值,我们可以控制数据包在不同的 LSP 之间切换,以便利用不同 LSP 的片段来完成端到端的传输。因此,以流表形式存储的状态信息可以显著减少。通过仿真和实验,证明本

文提出的方法能够以很小的流量开销实现最佳的流表约减效果。

参 考 文 献

- [1] 卫岚宁, 林海, 王磊. 基于改进近邻传播算法的无线传感网分簇与节能[J]. 计算机应用与软件, 2018, 35(5): 196-201.
- [2] 王军, 刘文, 程勇. 基于 TEEN 协议和 BP 神经网络的 WSN 数据融合模型[J]. 计算机应用研究, 2017, 34(8): 2486-2489.
- [3] 付应辉, 刘必果, 束永安. 基于 SDN 的胖树数据中心网络多路径负载均衡算法研究[J]. 计算机应用与软件, 2017, 34(9):147-152.
- [4] 杨思锦, 庄雷, 胡颖, 等. 一种动态自调节的 SDN 控制器负载均衡算法[J]. 计算机应用与软件, 2016, 33(12): 71-74, 117.
- [5] 刘必果, 束永安. 一种基于三层结构模型的 SDN 负载均衡方案[J]. 计算机应用与软件, 2017, 34(7):85-90, 124.
- [6] Liu S, Liu J. Delay-aware multipath source routing protocol to providing QoS support for wireless ad hoc networks[C]//IEEE International Conference on Communication Technology. IEEE, 2010:1340-1343.
- [7] Huang L, Shen Q, Shao W. A source routing based link protection method for link failure in SDN[C]//IEEE International Conference on Computer and Communications. IEEE, 2017:2588-2594.
- [8] Gudipati A, Perry D, Li L E, et al. SoftRAN: software defined radio access network[C]//ACM SIGCOMM Workshop on Hot Topics in Software Defined NETWORKING. ACM, 2013:25-30.
- [9] Pentikousis K, Wang Y, Hu W. Mobileflow: Toward software-defined mobile networks[J]. IEEE Communications Magazine, 2013, 51(7):44-53.
- [10] Rondini E, Hailes S, Li L. Load sharing and bandwidth control in mobile P2P wireless sensor networks[C]//Sixth IEEE International Conference on Pervasive Computing and Communications. IEEE Computer Society, 2008:468-473.
- [11] Bhatnagar S, Ganguly S, Nath B. Creating multipoint-to-point LSPs for traffic engineering[J]. IEEE Communications Magazine, 2003, 43(1):95-100.
- [12] Solano F, Fabregat R, Marzo J L. On optimal computation of MPLS label binding for multipoint-to-point connections[J]. Communications IEEE Transactions on, 2008, 56(7): 1056-1059.
- [13] Solano F, Stidsen T, Fabregat R, et al. Label space reduction in MPLS networks: How much can a single stacked label do? [J]. IEEE/ACM Transactions on Networking, 2008, 16(6):1308-1320.
- [14] Solano F, Fabregat R, Marzo J L. Full label space reduction in MPLS networks: asymmetric merged tunneling[J]. IEEE Communications Letters, 2005, 9(11):1021-1023.
- (上接第 130 页)
- [4] Cruz C, Vanlande R, Nicolle C. ACTIVE3D: Semantic and 3D databases for civil engineering projects[C]//International Conference on Information and Knowledge Engineering. Ike'04, June 21-24, 2004, Las Vegas, Nevada, Usa. DBLP, 2004:56-61.
- [5] Jorgensen K A, Skauge J, Christiansson P, et al. Use of IFC model servers modelling collaboration possibilities in practice [R/OL]. Technical report by Aalborg University and Aarhus School of Architecture, 2008. http://it.civil.aau.dk/it/reports/2008_Ifc_model_server.pdf.
- [6] Faraj I, Alshawi M, Aouad G, et al. An industry foundation classes Web-based collaborative construction computer environment: WISPER[J]. Automation in Construction, 2001, 10(1):79-99.
- [7] Ma L, Sacks R, Ma L, et al. A cloud-based BIM platform for information collaboration[C]//International Symposium on Automation and Robotics in Construction. 2016.
- [8] Jeong S, Byun J, Kim D, et al. A data management infrastructure for bridge monitoring[C]//Proceedings of SPIE—The International Society for Optical Engineering, 2015, 9435.
- [9] Das M, Cheng J C, Kumar S S. Social BIMCloud: a distributed cloud-based BIM platform for object-based lifecycle information exchange[J]. Visualization in Engineering, 2015, 3(1):1-20.
- [10] Ismail A, Nahar A, Scherer R. Application of graph databases and graph theory concepts for advanced analysing of BIM models based on IFC standard [C]//International Workshop on Intelligent Computing in Engineering. 2017.
- [11] Kim H, Soibelman L, Grobler F. Factor selection for delay analysis using Knowledge Discovery in Databases[J]. Automation in Construction, 2008, 17(5):550-560.
- [12] Carrillo P, Harding J, Choudhary A. Knowledge discovery from post-project reviews[J]. Construction Management & Economics, 2011, 29(7):713-723.
- [13] Wang L, Leite F. Knowledge discovery of spatial conflict resolution philosophies in BIM-enabled MEP design coordination using data mining techniques: a proof-of-concept[C]//Proceedings of the ASCE International Workshop on Computing in Civil Engineering, 2013:419-426.
- [14] BuildingSmart. IFC TOOLS Project[EB/OL]. [2017-08-07]. <http://www.ifctoolsproject.com/>.
- [15] 陈远, 康虹, 张静雅. 基于 IFC 标准的 BIM 模型编程语言解析方法研究[J]. 土木建筑工程信息技术, 2017(3): 85-89.
- [16] 陈远, 逯瑶. 基于 IFC 标准的 BIM 模型空间结构组成与程序解析[J]. 计算机应用与软件, 2018, 35(4):162-167, 194.