



文章编号:1001-9081(2018)01-0026-05

DOI:10.11772/j.issn.1001-9081.2017071969

软件定义车联网的数据转发策略和路由选择技术

董柏宏, 邓 健, 张定杰, 吴维刚 *

(中山大学 数据科学与计算机学院, 广州 510006)

(*通信作者电子邮箱 wuweig@mail.sysu.edu.cn)

摘要:针对目前车联网(VANET)数据转发效率低的问题,提出了软件定义网络(SDN)的数据转发策略和路由选择技术。首先,采用了软件定义车联网的分层控制结构,由局部控制器和全局控制器组成,实现数据转发和控制分离,可灵活控制数据转发的方向;然后,设计了单条路段的车辆路由机制,该机制预测车辆节点位置并采用贪心策略,实现数据的稳定传输;其次,设计了多个需求间的路段路由机制,该机制采用广度优先搜索(BFS)算法和边集相结合的方式,实现多个需求间路径不相交,缓解带宽瓶颈问题;最后,通过仿真验证,对比无线自组网按需平面距离向量(AODV)路由,所提出的数据转发策略和路由选择算法在数据分组接收率上提高40%以上,平均延迟时间降低60%左右。实验结果表明,软件定义车联网的数据转发策略和路由选择技术能够提高数据转发效率,减少平均收包延时。

关键词:软件定义网络;车联网;数据路由;路径不相交;贪心策略

中图分类号: TP311; TP393 **文献标志码:**A

Technology on data forwarding and routing selection for software defined vehicular Ad Hoc network

DONG Baihong, DENG Jian, ZHANG Dingjie, WU Weigang *

(School of Data and Computer Science, Sun Yat-Sen University, Guangzhou Guangdong 510006, China)

Abstract: Since the data forwarding in Vehicular Ad Hoc Network (VANET) is inefficient, a technology on data forwarding and routing selection based on Software Defined Network (SDN) was proposed. Firstly, a hierarchical architecture of SDN based VANET, which was consist of local controller and global controller, was used to separate data forwarding from transmission control and decide the direction of data forwarding flexibly. Secondly, a vehicular routing mechanism of single road section was designed and data could be transmitted stably by predicting vehicular position and using greedy strategy. Thirdly, to achieve the goal of path disjoint which could avoid the bandwidth bottleneck between multiple demands, a road section routing mechanism was proposed, which combined the Breadth First Search (BFS) and edge set. Finally, compared with Ad Hoc On-demand Distance Vector (AODV) routing, the proposed algorithm could increase data reception rate by 40% and reduce average delay by 60%. The simulation results show that the technology on data forwarding and routing for software defined VANET can effectively improve the data delivery rate, and reduces the average packet delay.

Key words: Software Defined Network (SDN); Vehicular Ad Hoc Network (VANET); data routing; path disjoint; greedy strategy

0 引言

随着现代社会的发展以及汽车迅速普及,汽车已经成为人们生活不可或缺的交通工具,这同时也催生了很多新的应用场景,如车载娱乐应用、车间道路预警安全信息快速通知等^[1]。为了解决车辆之间实现资源共享的问题,车载自组织网络应运而生。车载自组织网络(Vehicular Ad Hoc Network, VANET)^[2]属于移动自组织网络的分支,继承了移动自组织网络的一些特性,比如拥有一定的带宽,可在一定范围内进行数据传输,能够自组织和自管理;除此之外,车载网络也有自己的特性,如汽车的高移动性导致网络拓扑变化频繁,因此需要允许汽车随时加入或者离开网络,并对拓扑的变化作出及

时的处理;另外由于汽车的速度快,容易造成传输链路的断开,导致数据包丢失^[3]。由于车载网的这些特性存在,如何设计出具备良好动态适应性和高可靠性的路由算法是车载网中的一个难题。

目前在车载网络中比较常见的路由协议有最优链路状态路由(Optimized Link State Routing, OLSR)协议、目的节点序列距离矢量(Destination Sequenced Distance Vector, DSDV)路由协议和无线自组网按需平面距离向量(Ad Hoc On-demand Distance Vector, AODV)路由协议等^[4]。在这些协议中,OLSR需要定期广播Hello包来发现邻居节点,在链路断开的时候不能及时响应;DSDV引进Bellman-Ford算法解决路由环的问题,并使用增量更新的方法更新路由表,提高了寻找链路

收稿日期:2017-08-11;修回日期:2017-09-08。

基金项目:国家自然科学基金资助项目(61379157);广东省科技计划项目(2015B010111001)。

作者简介:董柏宏(1993—),男,广东茂名人,硕士研究生,主要研究方向:车联网; 邓健(1993—),男,广东茂名人,硕士研究生,主要研究方向:车联网、命名数据网络; 张定杰(1992—),男,广东梅州人,硕士,主要研究方向:车联网; 吴维刚(1976—),男,山东泰安人,教授,博士,主要研究方向:车联网、云计算。



的效率,但 DSDV 不能及时发现链路状态的更新,在拓扑变化频繁的车载网中展示出来的性能不够理想; AODV 通过维护一张动态路由表来实现路由,但由于车辆节点的速度变化快,导致需要频繁地去维护次路由表。这些协议在低速的移动自组织网络中能够展现出不错的性能,但是在车辆速度快、拓扑变化明显的车载网中,却不能满足需求。一些实验表明,在典型的高速公路车联网场景中,传统自组织网络协议中 AODV 有着更加好的表现^[5],因此,一些学者就传统自组织网络协议方向提出更多的优化方案。比如基于地理位置计算路由的自适应聚类方案^[6]、针对 AODV 协议提出优化的邻居发现机制^[7]和基于聚类方法增强的车间通信^[8]等方案。这些研究方案都针对将传统自组网协议应用于车联网中在可靠性和路由效率上有一定的优化和提高,但是在拓扑高度动态变化的车联网中,传统分布式自组织网络节点间天然存在协作困难、连接需时长等特点,这使得上述优化的效果提高有限。

软件定义网络(Software Defined Network, SDN)^[9]是近几年兴起的一种新型网络架构,它能够集中控制网络资源,提高带宽的利用率。SDN 和传统网络不同,它将数据转发层和控制层进行分离,上层对数据路由作出决策,之后下发路由给底层,而底层只负责根据下发的路由进行数据的转发。通过这种方式,上层能够统一管理网络资源,对数据路由作出更高效的选择,而下层只需要按照上层下发的路由转发数据,无需寻找和维护路由,从而提高数据传输的效率^[9]。

SDN 能够动态地实现网络资源的弹性分配和控制,在数据传输效率方面有明显的优势,这很符合车载网络的特性和需求。在 SDN 提出来之后,文献[10]从架构和服务的角度展开实验,分析 SDN 应用在 VANET/MANET(Mobile Ad Hoc NETwork)上的优势:可靠性、灵活性和可编程性。文献[11]中采用基站管理车辆节点,基站连接到网络,再与控制器相连,实现车载网络的数据传输;文献[12]中设计了路边单元和基站同时管理车辆节点,路边单元可与基站通信,二者再连接到控制器上,动态管理车辆节点。这两个研究总体上可以看作是通过两种通信方式:车与车(Vehicle-to-Vehicle, V2V)和车与路边单元(Vehicle-to-Infrastructure, V2I)^[13],将车辆组织成一个动态的网络。而在路边单元有效地选择车辆转发节点,使得数据能够在链路稳定的情况下尽快从路头传到路尾,以及提高链路带宽的使用率,是一般的 SDN 未考虑的情况。这是在车联网中应用 SDN 的主要挑战,也是本文的主要工作目标。

因此,本文提出了软件定义车联网的数据转发策略和路由选择技术,本文的主要贡献有:

- 1) 设计了路边单元的车辆路由算法,实现数据稳定高效的传输。
- 2) 设计了全局控制器的路段路由算法,提高空闲链路带宽使用率,缓解带宽瓶颈带来的问题。
- 3) 通过仿真验证了本文设计的机制和算法具有较好的性能。

1 系统模型

在车联网中应用 SDN,首先需要确定软件定义车联网的网络模型,网络模型如图 1 所示,具体模型假设如下:

- 1) 整个车联网分为多个路段,每个路段中间放置一个局部控制器,局部控制器管辖整个路段的车辆,任意两个局部控

制器的管辖范围不存在冲突;车联网中间位置放置全局控制器,管理所有局部控制器。

2) 每个路段有多辆车,车辆进入一个新路段,能自动发现路段中的局部控制器,通过 GPS(Global Positioning System)车载设备,车辆知道自己的位置和速度,并定期向局部控制器汇报信息。车辆与局部控制器的通信只能通过控制信道进行,局部控制器不能帮车辆转发数据。

3) 全局控制器定期和局部控制器通过控制信道进行通信,局部控制器之间不能直接通信,但可以通过全局控制器间接通信。

4) 系统使用 IEEE 802.11p 指定的控制信道发送协议控制信息,使用 IEEE 802.11p 指定的六条服务信道中一条进行数据传输。

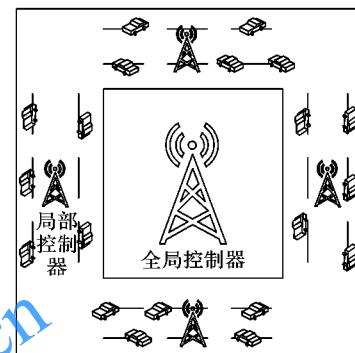


图 1 软件定义车联网的网络模型
Fig. 1 Network model of software defined VANET

2 软件定义车联网的路由选择算法

软件定义车联网路由选择算法分两部分:局部控制器车辆路由算法和全局控制器路段路由算法,这两种算法分别在局部控制器和全局控制器中运行,局部控制器车辆路由算法运行的结果作为全局控制路段路由算法的输入。

2.1 协议概述

每一个路段中行走的车辆会周期性发送 Hello 包给对应的局部控制器,局部控制器收到 Hello 包之后,记录车辆的位置和速度信息。而在车辆刚进入一个新路段时,局部控制器会添加该车辆的信息;另一方面,当车辆走出一个路段时,局部控制器会删除这些车辆信息。局部控制器根据车辆提供的位置和速度信息,预测该车辆节点在下一次路由周期所在的位置,并结合车辆节点前后周期的位置选择车辆转发节点。

局部控制器选择完转发节点之后,向全局控制器报告路况,全局控制器构建动态的拓扑图,当需求产生时,全局控制器根据该拓扑图选择路段路由。

2.2 局部控制器车辆路由算法

2.2.1 算法描述

局部控制器根据车辆信息选择有效的转发节点,其运行的车辆路由算法主要分为 3 个步骤:预测车辆位置、选择首尾转发节点、选择中间转发节点。

- 1) 预测车辆位置。

通过预测车辆的位置,找到路由计算周期前后均在通信范围内的转发节点,有利于数据的稳定传输。因为路由计算周期(cycle)很短,所以认为在这个周期内,车辆的速度变化不会太大,因此可以大体预测到车辆在下一个路由计算周期



的位置。局部控制器根据车辆节点当前的位置(`loc_now`)和速度(`speed`)，预测车辆的大致位置(`loc_next`)，具体做法如下：

```
loc_next = loc_now + speed * cycle;
```

2) 选择首尾转发节点。

根据车辆的当前位置和预测的下一个路由计算周期位置，选择首尾转发节点。选择第一个转发节点的依据是车辆路由计算周期前后位置均在路头通信距离的算术平方根范围之内，基于贪心策略的选取，选择预测位置距离路首最远的车辆作为第一个转发节点；选择最后一个转发节点的方法是车辆路由计算周期前后面为均在路尾通信距离的算术平方根范围之内，且选择预测位置距离路尾最远的车辆作为最后一个转发节点。

选择路段首尾转发节点的距离定为通信距离的算术平方根的原因在于可以保证路段间的通信连通，也就是说上一个路段的最后一个转发节点发送的数据，下一个路段的第一个转发节点可以保证能在有效的通信范围之内收到数据；而选择离路首和路尾最远的转发节点的原因是这样可以尽可能在该路段中选择数目最少的转发节点，降低数据传输的延时。

3) 选择中间转发节点。

中间节点的选取依赖于上一个转发节点，该节点的预测位置到上一个转发节点的当前位置距离不能超过车辆的通信距离。基于贪心策略，选择预测位置距离上一个转发节点当前位置最远的车辆作为中间的转发节点。以此类推，将刚选出的中间转发节点作为上一个转发节点，继续选出其他的中间节点，直到选出的节点当前位置距离最后一个转发节点的预测位置少于通信距离为止。将预测位置考虑到选择转发节点的方法中，可以在很大程度上保证在一个路由计算周期之内，链路不会断开，数据可以实现稳定传输。

至此，该路段完整的转发链路已经选择出来，具体的伪代码如下所示：

```
Input: NodeSet
// 初始化
Initialize: CycleSet, PathSet, Head, Tail
function FindForwardNode( Node Set, CycleSet )
    if Node Set is empty
        return
    end if
    //CycleSet 为预测集
    CycleSet = compute Node Set location after Cycle
    //根据距离路头距离排序
    Sort NodeSet and CycleSet by Distance ( CycleSet, Road. begin )
    for i in CycleSet
        //选出第一个转发节点
        if CycleSet[ i ]. X < roadBegin + 200
            Head = NodeSet[ i ]
        end if
        //选出最后一个转发节点
        if RoadEnd - 200 < cycleSet[ i ]. X < RoadEnd
            Tail = NodeSet[ i ]
        end if
    end for
    PathSet. add( Head and Tail )
    //贪心策略选出剩下的转发节点
    for j to k in CycleSet
```

```
Head = max( CycleSet[ j..k ]. X - Head, X < 400 )
```

```
PathSet. add( Head )
```

```
end for
```

```
end function
```

2.2.2 举例说明

为了方便理解，举例说明。为了方便讲解，将路段长设置为1000 m，且在X轴上，车辆通信范围在400 m，因此路段首尾保护区间为200 m，车辆路由周期前后位置如图2所示，表1中是车辆路由周期前后的坐标。

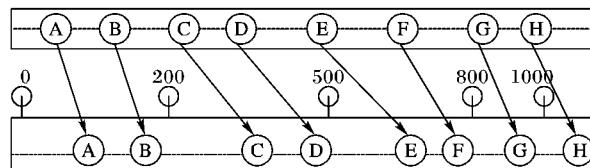


图2 车辆节点周期前后位置变化

Fig. 2 Vehicle location change before and after routing cycle

表1 各车的当前位置和预测位置

Tab. 1 Current location and predicted location of every vehicle

车辆	当前位置/m	预测位置/m	车辆	当前位置/m	预测位置/m
A	50	100	E	520	650
B	110	160	F	700	780
C	280	320	G	820	920
D	340	460	H	940	1050

选取路段首尾转发节点。A从50 m变化到100 m，B从110 m变化到160 m，A和B两辆车前后位置都是在距离路口200 m以内，也就是都在0~200 m的范围内，由于B离路口更远，因此选取B作为第一辆转发节点；选择最后一个转发节点，G前后位置都是距离路尾200 m，即都在800~1000 m，H目前虽然处于这个路段，但按照目前速度的预测，在下个路由计算周期H将会离开本路段，考虑到链路的稳定性，不会选择H作为转发节点，而是选择G作为最后一跳转发节点。

选择中间的节点。第一个转发节点已经确定为B，位置从110 m到160 m，接下来第二个转发节点，C从280 m变化到320 m，D从340 m变化到460 m，因此C和D下一个路由计算周期位置距离B目前的位置分别为210 m和350 m，均不超过400 m，都可以保证在路由周期内车辆的移动不会影响到数据的传输，而E从520 m到650 m，距离差为130 m，超过400 m，在E之后的车辆比如F、G、H当然都不需要被考虑，因此基于贪心策略，选择距离差最远的D作为第二跳转发节点。重复这个步骤获取到整个链路的转发节点B→D→E→G。

2.3 全局控制器路段路由算法

2.3.1 算法描述

局部控制器在选好路段中的转发节点之后，向全局控制器汇报路况信息，当全局控制器接收到需求时，根据路段路由算法产生数据传输要经过的路段。路段路由主要的步骤分为构建动态拓扑图和搜索路径。

1) 构建动态拓扑图。

由于局部控制器管理的是路段中两个方向的车流，因此在向全局控制器汇报的时候，需要将这两个方向是否选择出完整传输路径的信息告诉全局控制器。为了方便表示这两个方向的车流，全局控制器中构建的拓扑图用有向图表示，有向图的边表示局部控制器选择出来的链路，有向图的节点表示



路口,如果路口 a 到路口 b 存在完整链路,则路口 a 到路口 b 可达;否则不可达。

2) 搜索路径。

在有向图构建出来之后,当第一次接收到源节点的路径请求时,全局控制器使用广度优先搜索算法搜索源节点到目的节点的路径,并将这个路径放入边集中,表示该链路已经被占用。接下来收到其他需求的时候,使用最短路径算法的同时要考虑该段链路是否已经被占用:如果被占用,则不考虑该段链路,转而考虑其他链路,倘若不存在一条从源节点到目的节点的路径,则该需求暂时不被受理。

全局控制器的路段路由算法伪代码如下所示:

```
Input : Node, Graph, DisableNode, PathSet
//初始化
Initialize: TargetPath
function FindRoute( Graph, DisableNode )
    if BeginNode = EndNode
        return
    end if
    PathSet. add( BeginNode )
    while PathSet is not empty and TargetPath is empty
        tmpPath = PathSet. Pop()
        //依次找出每条可达链路
        for i in Node
            if ( ! tmpPath. find( Node[ i ] ) ||
                ! find( tmpPath. EndNode ) )
                tmpPath. pushback( Node[ i ] )
                PathSet. add( tmpPath )
            end if
        end for
        //找出若干条稳定链路
        for j in PathSet
            if ( PathSet[ i ] is whole path )
                TargetPath = PathSet[ i ]
            end if
        end for
    end while
    return TargetPath
end function
```

2.3.2 举例说明

在一个 3×3 的网格地图中,局部控制器选择链路之后向全局控制器汇报路况信息,如图 3 所示。假设在这个地图中,所有局部控制器管理的两个方向车流都能够选择出完整的链路,也就是相邻路口间的链路都是连通的。

在某个时刻产生了一个需求 1,源节点 A 在路口 1 的位置上,目的节点 B 在路口 16 的位置上,全局控制器在这个拓扑图中使用广度优先搜索算法计算出了从 A 到 B 的最短路径,经过了 $1 \rightarrow 2 \rightarrow 6 \rightarrow 7 \rightarrow 11 \rightarrow 12 \rightarrow 16$,如图 4 中实线箭头所示,将这段链路加入到边集中,表示已被使用。

过了一段时间需求 2 产生,需求 2 的源节点 C 在路口 5 的位置,目的节点 D 在路口 8 的位置,在使用广度优先搜索算法的时候,发现路口 6 到路口 7 这段链路已经被占用,因此避开这个路段,转而使用路口 6 到路口 10 这段链路,因此需求 2 经过的路段为 $5 \rightarrow 6 \rightarrow 10 \rightarrow 11 \rightarrow 7 \rightarrow 8$,如图 4 中虚线箭头表示的链路所示。这两个需求可以同时使用路口 7 与路口 11 之间的链路,因为这两个需求使用的相反方向的车流作为传输路径。

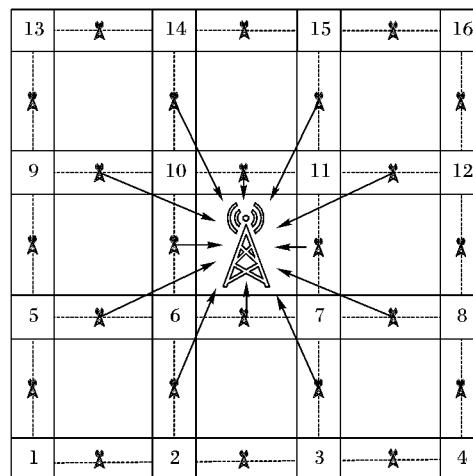


图 3 局部控制器向全局控制器汇报路况

Fig. 3 Sending road condition information from local controller to global controller

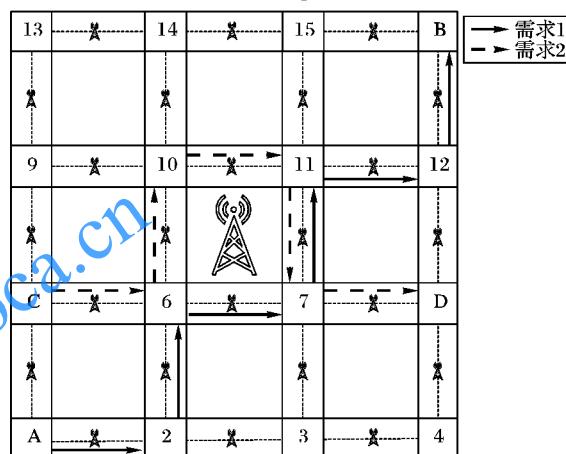


图 4 多个需求的路段路由

Fig. 4 Road routing for multiple requirements

3 仿真结果及分析

本次仿真实验在 NS3 (Network Simulation-3) 上模拟,利用 SUMO (Simulator of Urban MObility) 交通仿真模拟软件生成道路信息和车辆节点运行轨迹作为 NS3 的输入文件。它不但摈弃了当前主流网络仿真软件的缺点,还包括了很多新的特性,包括全新的 C++ 内核、更易拓展的模拟结构、能更加准确地模拟实际无线通信环境、网络接口和设备接口与实际网络具有高度一致性等^[14]。NS3 系统网络仿真模型如图 4 所示,每个路段长为 1000 m,路段中间放置局部控制器,地图中心点放置全局控制器,控制信道传输距离为 2000 m,服务信道传输距离为 400 m,总共生成 1000 辆车,发车间隔为 5 s。

为了方便获取数据,本次仿真产生两个需求:源节点 1 在 $(0,0)$,目的节点 1 在 $(3000,3000)$;源节点 2 在 $(0,1000)$,目的节点 2 在 $(3000,1000)$ 。源节点持续向目的节点发包,发包速率为 1024 B/s。对比实验采用经典的移动自组织网络协议 AODV 算法,与本文提出的 SDVN (Software Defined Vehicular Networking) 算法在不同的最大车速 (10 m/s , 15 m/s , 20 m/s , 25 m/s , 30 m/s) 下进行仿真。5 种车速的选择恰好覆盖了车联网拓扑较稀疏到较密集的场景:车速太低,车辆之间的平均距离容易超过最大通信距离,导致车联网节点之间不连通的情况频繁出现;车速过高的情况下,会出现严重



堵车,不利于实验的进行。

1) 数据分组接收率。

图 5 是数据分组接收率随车辆最大速度变化的结果,从结果中可以看出,当车辆速度变大的时候,SDVN 和 AODV 算法的数据分组接收率都呈现出下降的趋势,但 AODV 算法下降的趋势更加明显,而 SDVN 则在较大的车辆速度下,仍能保持较高的数据分组接收率。

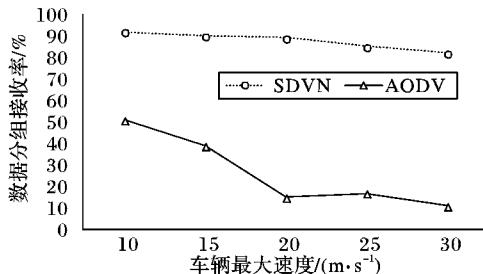


图 5 数据分组接收率随车辆最大速度变化

Fig. 5 Data reception rate vs maximum vehicle speed

2) 平均延迟时间。

图 6 为平均延迟时间随车辆速度变化的结果。从结果中可以看出,AODV 算法在车辆速度小的时候,端到端的延迟比 SDVN 算法高很多。这是因为车辆速度小的时候,车辆密集,AODV 算法需要花费大量的时间去查找源节点到目的节点的路径;而本文算法计算路由的周期是一定的,在链路上存在足够多位置相对均匀的车辆,就可以选择跳数基本相同的转发节点,因此延迟会相对稳定。

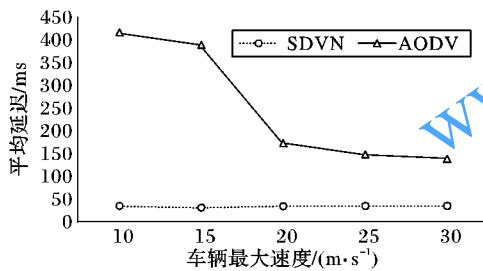


图 6 平均延迟时间随车辆最大速度变化

Fig. 6 Average delay vs maximum vehicle speed

3) 路由维护开销。

图 7 为平均路由维护开销随车辆速度变化结果。车速比较慢的时候,SDVN 算法开销要比 AODV 小一些,当车辆速度快的时候,二者的开销比较接近。这是因为 AODV 是采用广播的方式探测链路的,在当车辆速度慢的时候,车辆分布比较密集,源节点发出的 RREQ(Route REQuest)不断被广播,导致开销越来越大;而 SDVN 是定期计算链路,与车辆密度大小关系不太,因此开销相对稳定。

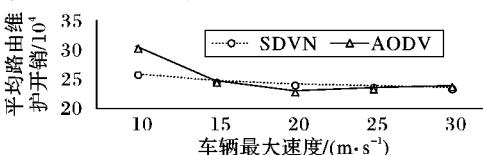


图 7 平均路由维护开销随车辆最大速度变化

Fig. 7 Average cost of routing updating vs maximum vehicle speed

4) 平均转发次数。

图 8 为平均转发次数随车辆速度变化结果。从图中看出 SDVN 路由算法平均转发次数在 22 次左右,而 AODV 路由算法在 16 次。这是因为在 AODV 在链路建立的阶段,会从源节

点到目的节点的多条链路中选择最短跳数的链路进行传输数据;而在 SDVN 算法中,当某部分链路被其他需求所使用时,需要放弃使用该条链路,选择其他未被使用而跳数更多的链路,因此 SDVN 算法平均转发次数略大于 AODV 算法平均转发次数。

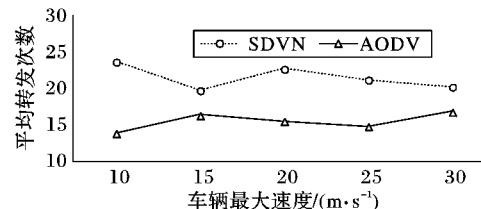


图 8 平均转发次数随车辆最大速度变化

Fig. 8 Average forward times vs maximum vehicle speed

4 结语

由于车载网的拓扑变化频繁、车辆速度快等特性,现有的数据路由算法和机制不能很好地满足车载网的需求,而软件定义网络能够动态分配网络资源,对数据路由动态作出决策,在车载网络中有良好的性能。本文的主要研究内容是如何在软件定义车载网络中,提高数据传输率、可靠性和链路带宽利用率。首先,本文设计了基于位置和速度信息的车辆路由算法,该算法能够保证在一段时间内链路的稳定,实现数据稳定传输;其次,本文设计了路段路由算法,能够充分利用空闲链路的带宽,尽量避免多个需求同时使用一段链路,缓解带宽瓶颈问题;最后,本文通过 NS-3 进行实验仿真,实验结果表明,本文提出的路由算法比传统的自组织网络路由算法有更优的性能。

软件定义网络技术能够很好地解决车载网络中的问题,将软件定义网络更好地应用到其中是车载网络一个重要的研究方向。下一个研究目标是如何利用软件定义网络技术在车联网中按链路稳定性实现带宽分配。

参考文献 (References)

- [1] 马静. 车联网的关键技术及其应用研究 [J]. 江苏科技信息, 2016(24): 50–52. (MA J. Key technology and application research of the Internet of vehicles [J]. Jiangsu Science & Technology Information, 2016(24): 50–52.)
- [2] TOOR Y, MUHLETHALER P, LAOUTI A, et al. Vehicle Ad Hoc networks: applications and related technical issues [J]. IEEE Communications Surveys and Tutorials, 2008, 10(3): 74–88.
- [3] KAUR M. Vehicular Ad Hoc networks [J]. Journal of Global Research in Computer Science, 2012, 3(3): 61–64.
- [4] 章学宇. 车载自组织网络路由协议研究 [D]. 长沙: 湖南大学, 2010: 18–34. (ZHANG X Y. The research on routing protocols in vehicular Ad Hoc networks [D]. Changsha: Hunan University, 2010: 18–34.)
- [5] JAAP S, BECHLER M, WOLF L. Evaluation of routing protocols for vehicular Ad Hoc networks in typical road traffic scenarios [C]// Proceedings of the 11th EUNICE Open European Summer School on Networked Applications. Madrid: [s. n.], 2005: 584–602.
- [6] GOONEWARNE R T, ALI F H, STIPIDIS E. Robust mobility adaptive clustering scheme with support for geographic routing for vehicular Ad Hoc networks [J]. IET Intelligent Transport Systems, 2009, 3(2): 148–158.
- [7] KRCO S, DUPCINOV M. Improved neighbor detection algorithm for AODV routing protocol [J]. IEEE Communications Letters, 2003, 7(12): 584–586.

(下转第 49 页)



- [4] NATU M, SETHI A S. Active probing approach for fault localization in computer networks [C]// Proceedings of the 2006 4th IEEE/IFIP Workshop on End-to-End Monitoring Techniques and Services. Piscataway, NJ: IEEE, 2006: 25 – 33.
- [5] LGORZATA STEINDER M, SETHI A S. A survey of fault localization techniques in computer networks [J]. Science of Computer Programming, 2004, 53(2): 165 – 194.
- [6] KATKER S, PATEROK M. Fault isolation and event correlation for integrated fault management [C]// Proceedings of the 5th IFIP/IEEE International Symposium on Integrated Network Management. Berlin: Springer, 1997: 583 – 596.
- [7] CHENG L, QIU X, MENG L, et al. Efficient active probing for fault diagnosis in large scale and noisy networks [C]// Proceedings of the 29th IEEE International Conference on Computer Communications. Washington, DC: IEEE Computer Society, 2010: 1 – 9.
- [8] PATIL B M, PATHAK V K. Survey of probe set and probe station selection algorithms for fault detection and localization in computer networks [J]. IEEE Transactions on Networks and Communications, 2015, 3(4): 57.
- [9] 孟洛明, 黄婷, 成璐, 等. 支持多故障定位的探测站点部署方法 [J]. 北京邮电大学学报, 2009, 32(5): 1 – 5. (MENG L M, HUANG T, CHENG L, et al. Probe station placement for multiple faults localization [J]. Journal of Beijing University of Posts and Telecommunications, 2009, 32(5): 1 – 5.)
- [10] HUKERIKAR S, DINIZ P C, LUCAS R F, et al. Opportunistic application-level fault detection through adaptive redundant multithreading [C]// Proceedings of the 2014 International Conference on High Performance Computing & Simulation. Piscataway, NJ: IEEE, 2014: 243 – 250.
- [11] GARDNER R D, HARLE D A. Network fault detection: a simplified approach to alarm correlation [C]// Proceedings of the 16th IEEE Global Telecommunications Conference. Washington, DC: IEEE Computer Society, 1997: 44 – 51.
- [12] SCHROEDER B, GIBSON G. A large-scale study of failures in high-performance computing systems [J]. IEEE Transactions on Dependable and Secure Computing, 2010, 7(4): 337 – 350.
- [13] JAKOBSON G, WEISSMAN M. Real-time telecommunication network management: extending event correlation with temporal constraints [C]// Proceedings of the Fourth International Symposium on Integrated Network Management IV. London: Chapman & Hall, 1995: 290 – 301.
- [14] LEMARINIER P, BOUTEILLER A, KRAWEZIK G, et al. Coordinated checkpoint versus message log for fault tolerant MPI [J]. International Journal of High Performance Computing and Networking, 2004, 2(2/3/4): 146 – 155.
- [15] SCHROEDER B, GIBSON G A. Understanding failures in petascale computers [C]// Proceedings of the 6th Scientific Discovery through Advanced Computing Conference. Bristol: IOP Publishing Ltd, 2007: 2022 – 2032.
- [16] 武林平, 孟丹, 梁毅, 等. LUNF——基于节点失效特征的机群作业调度策略 [J]. 计算机研究与发展, 2005, 42(6): 1000 – 1005. (WU L P, MENG D, LIANG Y, et al. LUNF—a cluster job schedule strategy using characterization of nodes' failure [J]. Journal of Computer Research and Development, 2005, 42(6): 1000 – 1005.)
- [17] BAILTY D, HARRIS T, SAPHIR W, et al. The NAS parallel benchmarks 2.0: NAS-95-020 [R]. Washington: NASA Ames Research Center, 1995: 12.

This work is partially supported by the National Key Research and Development Program of China (2016YFB0200502).

GAO Jian, born in 1992, M. S. candidate. His research interests include parallel computing, runtime system.

YU Kang, born in 1987, Ph. D., assistant engineer. His research interests include parallel computing.

QING Peng, born in 1979, M. S., senior engineer. His research interests include parallel compilation, runtime system.

WEI Hongmei, born in 1968, Ph. D., senior engineer. Her research interests include parallel computing, parallel compilation.

(上接第30页)

- [8] ASWATHY M C, TRIPTI C. A cluster based enhancement to AODV for inter-vehicular communication in VANET [J]. International Journal of Grid Computing & Applications, 2012, 3(3): 41 – 50.
- [9] THOMAS N D, GRAY K. SDN: Software Defined Networks [M]. Sebastopol: O'Reilly Media, 2014, 8(7): 31 – 38.
- [10] KU I, LU Y, GERLA M, et al. Towards software-defined VANET: architecture and services [C]// Proceedings of the 2014 13th Annual Mediterranean Ad Hoc Networking Workshop. Piscataway, NJ: IEEE, 2014: 103 – 110.
- [11] SHIN M K, NAM K H, KIM H J. Software-Defined Networking (SDN): a reference architecture and open APIs [C]// ICTC 2012: Proceedings of the 2012 International Conference on ICT Convergence. Piscataway, NJ: IEEE, 2012: 360 – 361.
- [12] LI H, DONG M, OTA K. Control plane optimization in software-defined vehicular Ad Hoc networks [J]. IEEE Transactions on Vehicular Technology, 2016, 65(10): 7895 – 7904.
- [13] TRUONG N B, LEE G M, GHAMRI-DOUDANE Y. Software defined networking-based vehicular Ad Hoc network with fog computing [C]// Proceedings of the 2015 IFIP/IEEE International Symposium on Integrated Network Management. Piscataway, NJ: IEEE, 2015: 1202 – 1207.

- [14] 崔翠梅, 杨德智, 姜程鑫. 基于 NS3 的移动认知网络仿真系统 [J]. 通信技术, 2016, 49(11): 1509 – 1513. (CUI C M, YANG D Z, JIANG C X. NS3-based Simulation System for Mobile Cognitive Radio Networks [J]. Communications Technology, 2016, 49(11): 1509 – 1513.)

This work is partially supported by the National Natural Science Foundation of China (6137915), the Science and Technology Planning Project of Guangdong Province (2015B010111001).

DONG Baihong, born in 1993, M. S. candidate. His research interests include vehicular Ad Hoc network.

DENG Jian, born in 1993, M. S. candidate. His research interests include vehicular Ad Hoc network, named data network.

ZHANG Dingjie, born in 1992, M. S. His research interests include vehicular Ad Hoc network.

WU Weigang, born in 1976, Ph. D., professor. His research interest include vehicular Ad Hoc network, cloud computing.