Software Defined Internet of Vehicles: Essentials and A Case Study

Jiacheng Chen

Shanghai Jiao Tong Univ.

Oct. 7
Outline

- background and motivation
- software defined Internet of vehicles
- case study —— dynamic connection management
- future works and further discussions
Google Map

Navigation features:
• realtime traffic
• realtime road incident (e.g., accident, construction)
• traffic prediction

other navigation apps in China
technologies in Google Map

information gathering:
- anonymous GPS tracking
- crowdsourcing from users’ report (Waze, bought by Google in 2013 with more than $1b)
- from local departments of transportation (sensors using infrared or radar)

information processing:
- data mining algorithms (e.g., traffic pattern recognition, prediction)
- computer vision & image recognition
future intelligent transportation systems

• specialization for the vehicular scenarios
  • connected cars with various equipments (e.g., sensors, cameras, on-board diagnostics)
  • platforms/OS for driver-friendly interactions

• building blocks
  • communications, especially Internet access
  • storage and computing

future ITS fits in with the Internet of vehicles (IoV) paradigm
### from VANET to IoV

<table>
<thead>
<tr>
<th>VANET:</th>
<th>IoV:</th>
</tr>
</thead>
<tbody>
<tr>
<td>• safety-related services</td>
<td>• Internet services</td>
</tr>
<tr>
<td>• communications</td>
<td>• plus computing</td>
</tr>
<tr>
<td>• mostly distributed</td>
<td>• better centralized</td>
</tr>
<tr>
<td>• inherits from MANET</td>
<td>• inherits from IoT</td>
</tr>
</tbody>
</table>

IoV is the evolution form of VANET. It is more powerful, yet more challenging to implement.
communications in IoV

• major requirements:
  • flexibility
  • efficiency
  • scalability
  • service differentiation

• DSRC:
  • standardized vehicular communications protocol
  • IEEE 802.11p for PHY and MAC
    • CSMA/CA protocol
  • IEEE 1609.4 for multi-channel management
    • service advertisement
integrating SDN paradigm with IoV

• software defined networking (SDN)
  • centralized controllers manage data forwarding

• if SDN controllers take control of resource management in IoV…
  • flexibility
  • resource utilization efficiency
  • can serve large number of vehicles
  • service-oriented resource allocation
  • easy to implement NFV
  • universal and lightweight on-board hardware
  • NF updating is transparent to hardware
  • etc.
SD-IoV reference scenario

logical components:
- vehicles
- RSUs
- controllers
- NF servers
- app servers

interactions:
- vehicle—RSU (V2R)
- vehicle—vehicle (V2V)
- RSU—fog
- RSU—cloud
SD-IoV architecture

cloud system
- app servers

classification system (wired data plane, control plane, and application plane):
- controllers
- network app servers
- SDN APIs
- wired backhaul

client system (wireless data plane):
- RSUs
- vehicles
SD-IoV control process

• basic unit is time slot
• each slot contains two interval
  • 1st interval —— control messaging
    ① vehicles report status to RSU in sequence
      • location, request, etc.
    ② RSU contacts controller
    ③ controller feedbacks decision
    ④ RSU distributes decision to vehicles
  • 2nd interval —— data transmission

• IEEE 1609.4 control process
  • CCH and SCH intervals
case study — dynamic connection management

• general objective:
  • decide how data is forwarded among vehicular nodes

• features:
  • efficient resource utilization
  • full exploration of transmission opportunities
  • flexible topologies
  • service-oriented
  • multiple concurrent request support
overlay vehicular network

- in network virtualization...
  - underlay physical network
  - overlay virtual network

- similarly, in vehicular network...
  - underlay network comprises vehicles and RSUs
  - an OVN is formed by a subset of underlay nodes
  - dedicated resources are allocated to the OVN
OVN creation

- one OVN is created for one service

- topology rules for OVN...
  - efficiency, feasibility vs. opportunity
  - permitted topologies:
    - single-hop unicast (sender — receiver)
    - single-hop multicast (sender — receivers)
    - dual-hop unicast (sender — helper — receiver)

file transmission  alert  info sharing
constraints for a feasible OVN creation

• unlike NV, nodes cannot be shared
  • single-radio
• a node only plays one role in OVN
  • half-duplex
• topology rules should be satisfied
• sender should be able to serve the receiver
  • service provision capability
    • sensors equipped, cache, etc.
• OVNs should not interfere with each other
  • overlapping OVNs operate on different channels
evaluating OVN

- multi-criteria utility design
  - service completion contribution modifiers
    - priority, transmission rate, completion ratio, urgency
  - node occupation penalty
  - OVN utility
    - channel spatial reuse modifier

- OVNC objective:
  - sum of utilities of all OVN
Subgraph Utility = \( I(\text{ReceiverIsNotAllTimeout}) \times \left( \sum_{\text{Receiver}} \text{ServiceCompletionContribution} - \text{NodeOccupancyPenalty} + 2 \right) \times \left( 1 - \text{Normalize}(\text{MaxSenderReceiverDistance}) \right)^{I(\text{IsV2V}) \times \lambda} \)

NodeOccupancyPenalty = \( \sum_{\text{sender, helper}} \text{ServiceCompletionContribution} \)

ServiceCompletionContribution_{\text{Node}} = I(\text{HasRequest}) \times (\text{Priority})^\alpha \times \min \left\{ \left( \frac{\text{CurrentDataSize} + \text{TransmittedData} \times I(\text{IsReceiver})}{\text{TotalDataSize}} \right), 1 \right\}^\beta \times \left( I(\text{IsNotTimeout}) \times \frac{\text{CurrentTimeSlot} - \text{StartTimeSlot} + 1}{\text{TotalTimeoutSlot}} \right)^\gamma \)

\( \text{SCC} \in [0, 1], \text{utility} \geq 0 \)
\( \alpha, \beta, \gamma, \lambda > 0 \)
graph-based representation

• prerequisites
  • connectivity graph
  • interference graph
  • request-provision bipartite graph

• OVNC result representation
  • service graph

• matrix representation
solutions for OVNC

• specifically-designed genetic algorithm
  • graph-based representation
  • initial population generation
  • crossover
  • mutation
  • fitness evaluation

• greedy heuristic algorithm
  • sort according to SCC
  • select and create OVN
  • iterate
evaluation

total transmitted data with different traffic flow levels

total number of finished requests with different traffic flow levels

<table>
<thead>
<tr>
<th>TFL</th>
<th>traffic density (per lane per mile)</th>
<th>speed (km/h)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1-12</td>
<td>110</td>
</tr>
<tr>
<td>2</td>
<td>12-30</td>
<td>90</td>
</tr>
<tr>
<td>3</td>
<td>30-67</td>
<td>70</td>
</tr>
</tbody>
</table>
future works

• enable more intelligence
  • learning methods for resource management
  • prediction based on historical data

• upgrade more NFs
  • mobility management/handoff/authentication
  • heterogeneous resource management
extending SDN to wireless — in reality

• at&t’s ambition
  • virtualize 75% of its carrier network by 2020
  • still, everything happens beyond the BSs
  • enough, since UEs only communicate with BSs

• SDN for peer-to-peer/ad-hoc communications
  • D2D, M2M, V2V, opportunistic networks…
  • main obstacles for implementation:
    • a reliable control channel and a controller
    • realtime status report
    • wireless extension of OpenFlow
    • new hardware

vehicular network is the most promising one to embrace SDN
extending SDN to wireless — for research

• for CSer:
  • focus on SDN itself, such as:
    • northbound/southbound API design
    • controller & OpenFlow-enabled hardware (e.g., AP) design
    • flow table design

• for EEer:
  • focus on applications based on SDN, such as:
    • resource management in above scenarios
    • network control and management (NF)
    • data offloading in heterogeneous network scenario
list of references on SDVN

thanks! & questions?