Persistent WiFi connectivity during Train journey: An SDN based approach

Arkadeep Sen[1], Krishna M. Sivalingam[2] and Babu Narayanan K J[3]
Dept. of Computer Science and Engineering, Indian Institute of Technology Madras, Chennai, India[1,2]
Centre of Excellence in Wireless Technology (CEWiT), Chennai, India[3]
Email: `arkadeep.sen87@gmail.com`[1], `skrishnam@iitm.ac.in`[2],
`krishna.sivalingam@gmail.com`[2], `babunkj@cewit.org.in`[3]

***Abstract*—With the exponential growth in the number of mobile devices, providing Internet access via WiFi in trains is rapidly becoming a necessity. However, providing persistent Internet access inside a moving train has many challenges. Cellular network is predominantly used for the backhaul connection to the train. However, high deployment cost of a cellular network and presence of coverage holes in the existing cellular networks of telecom companies may dissuade the railway companies to go for such solutions. In this paper, we propose an SDN based architecture to provide Internet connectivity inside trains. The backhaul connection to the train, in the proposed architecture, is provided via WiFi. Deployment of such an architecture is more cost-efficient than that of a cellular network of the same capacity. Moreover, this architecture can also be used to provide connectivity in the coverage holes of the existing networks of the telecom companies. Through simulation, we show that the architecture can provide high throughput while maintaining per packet delay within reasonable limits inside a train.**

I. INTRODUCTION

Providing Internet access through WiFi during a train journey, is becoming one of the most sought after services. However, providing persistent Internet access inside a moving train has many challenges [1], [2], such as frequent handovers, poor bandwidth, etc. Cellular network is predominantly used for providing the backhaul connection to the train. The design of the backbone network can be done using the following two ways: (1) The railway company owns the complete backbone cellular network [3], [4]; or (2) The railway company has agreements with one or more telecom companies to provide the backhaul connectivity via their existing network [5], [6].

However, there are certain aspects, for both the types of approaches, which might dissuade the railway companies to go for such solutions. In the first case, the cost of deployment of the entire cellular network, comprising of the track-side base stations and the core network, might become prohibitively high [7], [8]. Additionally, they will also have to purchase the rights of a spectrum over which the cellular network will operate [7], [8]. For the case of existing cellular networks, coverage holes [9]–[11] exist in most of the rural areas. When the trains pass through these areas, they get disconnected from the network. Moreover, even the areas which get coverage from cellular networks in the rural areas, mostly suffer from poor bandwidth [9]–[11]. So, in both the cases, going for a full-scale deployment of the backbone network for complete coverage may not be always economically viable.

978-3-903176-15-7

In this paper, we propose a WiFi based backhaul connection to the train and a Software Defined Networks (SDN) based architecture which will ensure uninterrupted and continuous connectivity. SDN [12]–[15] decouples the control plane of a switch (router) from its data plane and moves it to a centralized controller. Customized control applications can be deployed on the controller, thus enabling network programmability.

As shown in [7], the deployment of a WiFi network of the same scale and capacity as that of an LTE network is more cost-efficient. Moreover, the additional cost of purchasing the license of a spectrum is not there for a WiFi deployment as WiFi operates in unlicensed spectrum. If the railway company opts for a full-scale deployment of the backbone network then the proposed architecture can be used as it will be more cost-efficient than a cellular network. On the other hand, if the railway company uses existing networks of telecom companies, then the proposed architecture can be used to provide connectivity in the coverage holes of those networks.

II. RELATED WORK

The network architectures for providing Internet connectivity in trains should also have robust mobility management mechanisms to support the frequent handovers efficiently. In this section, we look into some of the mobility management mechanisms in literature.

In [16], [17] modifications to the cellular networks (LTE and 5G) to facilitate Internet access inside trains are proposed. However, they depend on the default handover mechanism provided by the Mobility Management Entity (MME) of the cellular networks. This may not be an ideal approach for networks with high mobility and frequent handovers. In [18], a seamless handover mechanism for high-speed trains is proposed for LTE based network. To support handover efficiently they have proposed to deploy two antennas (one on the head carriage and the other on the tail carriage). During handover of the head antenna, the tail antenna will be used for the data transmission and once the handover is completed the head antenna will takeover the data transmission. However, as there are only two antennas and if the handover mechanism fails for both the head and tail antennas, then the train will get disconnected. In [19] a similar solution for WiMAX based network is proposed, where two mobile routers are placed at the head and the tail carriages. However, this paper also does not support more than two mobile routers and as a result, has the same shortcomings as that of [18]. In [20], summary of various other types of proposed handover mechanisms is presented. We do not present additional details about these handover mechanisms here.

In our proposed solution, we modify the existing WiFi architecture to suit the highly mobile nature of such networks. Moreover, the handover process is orchestrated by SDN controllers, both inside the train and in the backbone network, so that the packets are delivered correctly during and after the handover.

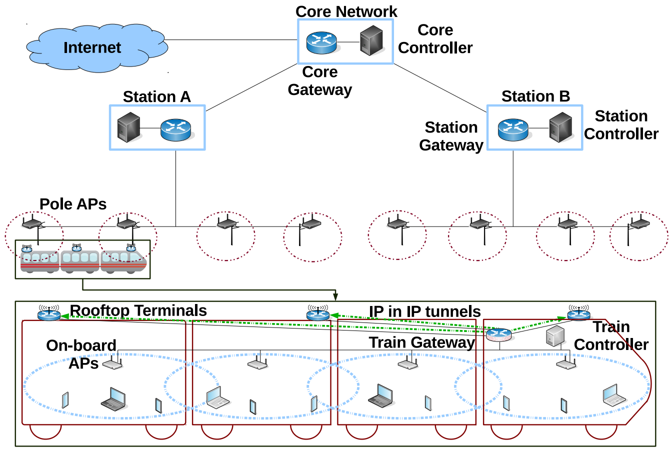


Fig. 1: Overall Network Architecture.

III. SDN BASED WIFI CONNECTIVITY IN TRAIN

This section presents the details of the proposed architecture for SDN based WiFi connectivity in train.

Figure 1 presents the overall network architecture. Each station has a Station Gateway (SG) which connects to the Core Network and the Core Gateway (CG) connects to the Internet. The stations are connected to the Core Network via optical fiber. The Pole APs (PAP) are installed on the track-side poles every 650m and these PAPs also are connected via optical fiber to their corresponding stations. The PAPs and the SGs connect with the corresponding Station Controller (SC) and the SCs and the CG connect with the Core Controller (CC) via OpenFlow protocol. The Rooftop Terminals (RT) are the wireless devices which connect with the PAPs to deliver packets. The ranges of the PAPs and the RTs are taken as 170m (approx.). On a train at least 3 RTs will be placed. They will be placed at equal distance along the width of the train so that there will always be at least one RT connected with a PAP. The trains will have at least 15 coaches. The minimum width of the trains will be 360m (taking average width of each coach as 24m). In each coach of a train at least one on-board AP will be available. The on-board APs, the Train Gateway (TG), the RTs, all connect with the Train Controller (TC) via OpenFlow protocol.

The RTs will be assigned IP addresses which can be used to uniquely identify the train. They will also be configured with the unique train number of the train. Initially, at the origin station, the mapping of the train number to the IP and MAC addresses of the RTs should be done at the corresponding SC as well as the CC.

A. Train Connection Establishment and Information Dissemination Protocol (TCEIDP)

The Train Connection Establishment and Information Dissemination Protocol provides a framework to establish a successful Layer 3 (and above) connection between the RTs and the PAPs as well as provides a framework for the RTs to send their IP and MAC addresses along with the train number to the PAPs. TCEIDP follows a client-server model. The RTs run the client and the PAPs run the server. TCEIDP runs on UDP and both the client and the server use the IP address of the WLAN interface of the devices on which they are running. Whenever the client sends a TCEIDP message it sends to the TCEIDP server port. Similarly, any message from the server to the client is sent to the TCEIDP client port.

Every time an RT successfully completes association with a PAP using IEEE 802.11 MAC management frames, the TCEIDP client, running on the RT, broadcasts a TCEIDP CONNECT message on the WLAN interface, which starts the connection establishment process. The CONNECT message contains the IP address of the RT. When the server, running on the PAP, receives the CONNECT message, it replies with a TCEIDP GATEWAY message to the client using the RT's IP address. The GATEWAY message contains the IP address of the PAP. The receipt of this message completes the connection establishment process at the client side. The client then replies to the server with a TCEIDP ACK message using the PAP's IP address. The ACK message contains the train number and the IP and MAC addresses of the RT's WLAN interface. The ACK message completes the connection establishment process at the server side as well as delivers the train information pertaining to that particular RT to the connected PAP.

B. Tracking the connection status of the train

The connection status of the train has to be tracked both in the backbone network as well as inside the train in order to deliver the packets correctly.

1) Tracking in the backbone network: The initial connection establishment using TCEIDP is done at the origin station before the start of the journey. The information received by the PAPs about the RTs is sent to the SC. The SC, in turn, sends the same to the SG and the CC. The CC after receiving this information sends the IP address of the SG to CG. The CG adds the IP address to the list IP address of the SGs of the connected stations. Subsequently, when the train moves, the RTs connect with other PAPs.

Before the start of the TCEIDP connection establishment process by an RT, the PAP, with which the RT connected, only has the MAC address of the RT. The PAP does not wait for the TCEIDP to complete and requests the corresponding SC to send the complete information about the RT along with the train number. If the SC has the mapping of the train number to the list of IP and MAC addresses of the RTs of the train, then it sends the information to the PAP as well as the SG. It also sets the status of the RT to *CONNECTED*. When the PAP receives the information from the SC, it maps the IP and MAC addresses of the RT to the train number.

When the SC does not have the information about the RT, the PAP gets the information about the RT through TCEIDP.

After this, the PAP sends this information to the SC, which, in turn, sends the information to the SG and the CC. If the SC has received the information about an RT of the train for the first time, then it indicates this to the CC. The CC then forwards all the information it has about all the RTs of the train to the SC. The CC also updates the CG about this and the CG adds the IP address of the corresponding SG to the list IP address of the SGs of the connected stations. Once the SC receives all the information from the CC, it adds the details of the other RTs to the list of IP and MAC addresses of the RTs mapped to the train number. The SC sets the statuses of all the other RTs to *DISCONNECTED*.

When the RTs get disconnected from the PAPs, the PAPs remove them from the list of connected RTs for the train. The PAPs also update this information at the corresponding SC. When the SC receives this information, it updates the connection status of the RT to *DISCONNECTED*. The SC also updates the SG about the disconnection. When the RTs of the train are no longer connected to any of the PAPs controlled by a particular SC, the SC updates this information at the CC. The CC, in turn, informs the CG which removes the IP address of the corresponding SG from its list of IP address of the SGs of the connected stations.

2) Tracking inside the train: Inside the train, the TC keeps track of the connection status of the RTs. After an RT connects with a PAP and completes the connection establishment process using TCEIDP, the RT updates this information at the TC which, in turn, updates this information at the TG. The TG adds the RT to the list of connected RTs.

Whenever an RT gets disconnected from the PAP it was connected to, it updates this information at the TC which, in turn, updates this information at the TG. The TG removes the RT from the list of connected RTs.

C. Packet traversal inside train

Inside the train, there are two levels of network address translation, one at the on-board APs and another at the RTs.

1) Packet traversal from the mobile device to the Rooftop Terminals: For every flow initiated by a mobile device inside the train, the TC assigns a unique port number to the flow. The TC sends the NAT entry corresponding to the flow to all the RTs. The NAT entry has source IP address as the IP address of the current on-board AP connected to the mobile device and the source port number as the unique port number. A packet first goes through a NAT at the current on-board AP and then is forwarded to the TG. The packet will have source IP address as the IP address of that on-board AP and source port number as the unique port number. The TG has the information of the RTs which are currently connected with a PAP. The TG chooses an RT (most recently connected) and sends the packets to that RT through an IP-in-IP tunnel. The RT does another NAT using the NAT entry added by the TC. The outgoing packets will have source IP address as the IP address of that RT. This IP address can be used to uniquely identify the packets originating from the train. The RT then sends the packets to the PAP with which it is connected.

2) Packet traversal from the Rooftop Terminals to the mobile device: The RT applies NAT on any incoming packet from the PAPs by consulting its NAT entries. The destination address of the packet is now changed to the IP address of the current on-board AP with which the mobile device is connected. The packet is then forwarded to the TG. The TG then forwards the packet to the current on-board AP. The on-board AP, after receiving the packet, applies NAT and changes the destination IP address of the packet to the IP address of the mobile device and the destination port number, from the unique port number to the original port number. After the NAT, the packet is forwarded to the mobile device.

D. Packet traversal in the backbone network

In the backbone network, there are three levels of network address translation. One at the PAPs, another one at the SGs and the final one at the CG.

1) Packet traversal from the Pole AP to the Internet: For every flow initiated from a train, the CC assigns a unique source port number across all the PAPs for the flow. The CC then sends this information to the SC which, in turn, sends it to all the PAPs connected to the RTs of the train. The PAP then applies NAT and changes the source IP address of the packet to the IP address of the PAP and the source port number to the unique port number. The PAP stores a modified NAT entry which uses the train number in place of the source IP address. The PAP then forwards the packet towards the station with which it is connected. The SG applies NAT on the packet and changes the source IP address of the packet to its IP address. The SG also creates a similar modified NAT entry. After the operation is completed, the packet is forwarded to the Core Network. The CG similarly applies NAT and changes the source IP address of the packet to its IP address. Similar modified NAT entry is also created at the CG. The packet is forwarded to the Internet after the NAT.

2) Packet traversal from the Internet to the Pole AP: The incoming packet from the Internet is received by the CG. As the CC updates the CG about the stations, which are currently connected with the train (at most two), the CG selects a station (most recently connected) and sends the packet to the SG of the station. The destination IP address of the packet is changed to the IP address of the SG after NAT. The SG has a list of connected RTs of the train and their corresponding PAPs. The SG chooses one of the PAPs (most recently connected) from the list. The destination IP address of the packet is changed to the IP address of that PAP and the packet is forwarded to it. The PAP has a list of the RTs of the train which are connected to the PAP. Again an RT is chosen (most recently connected). The destination address of the packet is changed to the IP address of the chosen RT and the destination port number is changed from the unique port number to the original port number. The packet is then forwarded to the RT.

E. Mobility Management in the backbone network

For proper mobility management, along with the tracking of the connection status of the train, corresponding NAT entries of the train should be sent to the newly connected PAP. If the PAP is connected to the same SC as the previous PAP, then the SC will send all the NAT entries, pertaining to the train, to the PAP. If the PAP is connected to a different SC as the previous PAP, then this is the first time the SC will be getting the information about the train. Thus it will not have any NAT entries pertaining to the train. The CC will send the NAT entries to the SC (along with other information as mentioned in Section III-B1). The SC will then forward all the NAT entries to the PAP and also to the corresponding SG.

As a result, the newly connected PAP will be able to correctly apply NAT to the packets originating from the train and forward them to the SG which, in turn, will forward them to the CG. Conversely, if packets destined for the train arrive from the Internet first to the CG, it will choose this station (most recently connected) and forward the packets to this SG. Since, all the NAT entries are already added to the SG it will be able to correctly apply NAT and forward the packets to this PAP (most recently connected). Similarly, as all the NAT entries are already added, the PAP will also be able to apply NAT correctly and forward the packet to the connected RT.

IV. SIMULATION-BASED PERFORMANCE STUDY

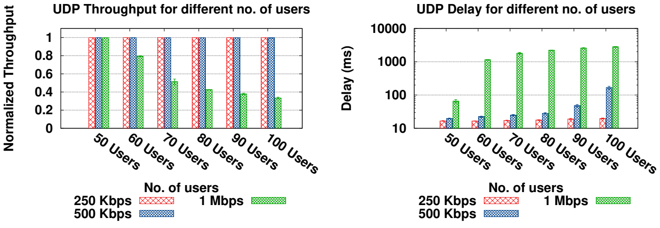
The proposed SDN based architecture for WiFi connectivity in train is implemented in ns3 network simulator [21] (version 3.27). 4-way handshake authentication process is used between the PAPs and the RTs to reduce the time taken for connection setup between them. The mobile devices, inside the train, remain stationary with respect to the train. TCP and UDP applications are setup on the mobile devices and the remote hosts, which are connected with the CG. The remote hosts send TCP and UDP traffic to the mobile devices at varying rates. The parameters used for running the simulations are summarized in Table I. The experiments are run for a total of 120 simulation seconds.

Parameter	**Value**
IEEE 802.11n band	2.4 GHz inside train, 5 GHz outside train
No. of Stations	2
No. of PAPs	6
Distance between each PAP	650m
No. of coaches	15
Total Train length	360m (24m per coach, including gaps)
No. of RTs	3
Range of RTs and Pole APs	170m
No. of on-board APs	15 (same as no. of coaches)
Speed of the train	28 m/s (100.8 km/h)
No. of users	varies from 50 to 100
Traffic for each mobile device	Varies from 250 Kbps to 1 Mbps for both TCP and UDP applications

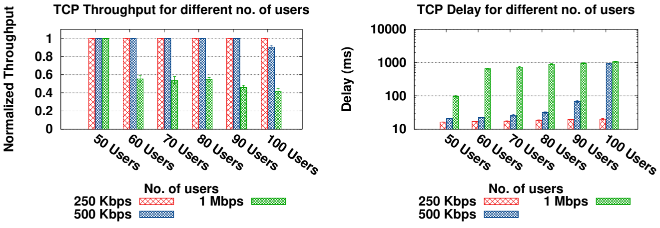
TABLE I: Simulation parameters

A. Performance Evaluation Results

Figures 2a and 2c present the average normalized UDP and TCP throughput experienced by the mobile devices respectively. When the remote hosts send traffic at rates of 250 Kbps and 500 Kbps, the average throughput experienced by the mobile devices is almost 250 Kbps and 500 Kbps respectively. For all the case 1 Mbps traffic rate, maximum throughput is achieved for 50 users. However, the average throughput gradually decreases with the increase in the number of mobile devices. Due to a combination of the train speed and high traffic rate, packets get dropped at the PAPs and the RTs. Moreover, the newer arriving packets at these devices get queued up. These packets may also get dropped due to the queues being full. As a result, the average UDP throughput decreases. Though, TCP retransmits all the dropped packets, however the retransmission increases the completion time of the client applications. So, the average TCP throughput decreases.



(a) Normalized UDP Throughput. (b) UDP Delay.



(c) Normalized TCP Throughput. (d) TCP Delay.

Fig. 2: Performance plots for TCP and UDP applications.

Figures 2b and 2d present the per packet average UDP and TCP delay experienced by the mobile devices respectively. The average delay, in general, increases as the number of mobile devices increases. As the number of users increases, so also does the number packets in the system (more buffering). Additionally, as the traffic rate is increased from 250 Kbps to 1 Mbps, the average delay increases. This also results from the increase in the number of packets (thus more queuing). Due to retransmission (TCP retransmission and retransmission at the IEEE 802.11 MAC layer of the PAPs and RTs) of dropped packets, the number of packets in the system goes up even further. As a result the packets spend more time in the queues.

V. CONCLUSION

In this paper, we have proposed an SDN based architecture for persistent WiFi connectivity during train journey, which uses WiFi for the backhaul connection to the train. This architecture can be used either for a full-scale deployment or for providing connectivity in coverage holes of existing cellular networks. The performance study shows that the architecture is able to provide high throughput while maintaining per packet delay within reasonable limits.

ACKNOWLEDGMENTS

The authors would like to acknowledge Prof. Bhaskar Ramamurthi, IIT Madras, for introducing us to this problem and for providing his valuable insight and expertise in the solution design. Part of the work was supported by an Institute Research & Development Awards (IRDA) grant from IIT Madras (2017-2020).

REFERENCES

[1] D. T. Fokum and V. S. Frost, “A Survey on Methods for Broadband Internet Access on Trains,” *IEEE Communications Surveys & Tutorials*, vol. 12, no. 2, pp. 171–185, 2010.

[2] S. Xu, G. Zhu, B. Ai, and Z. Zhong, “A survey on high-speed railway communications: A radio resource management perspective,” *Computer Communications*, vol. 86, pp. 12–28, 2016.

[3] “GSM-R,” http://gsmr-info.com/, March 2017.

[4] A. Sniady and J. Soler, “LTE for Railways: Impact on Performance of ETCS Railway Signaling,” *IEEE Vehicular Technology Magazine*, vol. 9, no. 2, pp. 69–77, June 2014.

[5] “Ericsson manages Wi-Fi connectivity for high-speed trains in Germany,” http://hugin.info/1061/R/2043907/763157.pdf, September 2016.

[6] “How does WiFi work on trains?” https://www.virgintrains.co.uk/our-blog/the-experience/2016/06/wifi-on-trains, June 2016.

[7] V. Nikolikj and T. Janevski, “A Cost Modeling of High-capacity LTE-advanced and IEEE 802.11ac based Heterogeneous Networks, Deployed in the 700MHz, 2.6GHz and 5GHz Bands,” *Procedia Computer Science*, vol. 40, pp. 49–56, 12 2014.

[8] M. Mahloo, P. Monti, J. Chen, and L. Wosinska, “Cost modeling of backhaul for mobile networks,” in *Proc. of IEEE ICC*, June 2014, pp. 397–402.

[9] FCC, “2018 Broadband Progress Report,” https://www.fcc.gov/reports-research/reports/broadband-progress-reports/2018-broadband-deployment-report, September 2018.

[10] TRAI, “The Indian Telecom Services Performance Indicators April - June, 2017,” http://trai.gov.in/sites/default/files/Performance_Indicator_Reports_28092017.pdf, January 2018.

[11] OpenSignal, “The State of LTE (February 2018),” https://opensignal.com/reports/2018/02/state-of-lte, September 2018.

[12] N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. Shenker, and J. Turner, “OpenFlow: enabling innovation in campus networks,” *ACM SIGCOMM Computer Communication Review*, vol. 38, no. 2, pp. 69–74, 2008.

[13] B. Nunes, M. Mendonca, X.-N. Nguyen, K. Obraczka, and T. Turletti, “A Survey of Software-Defined Networking: Past, Present, and Future of Programmable Networks,” *IEEE Communications Surveys & Tutorials*, vol. 16, no. 3, pp. 1617–1634, 2014.

[14] M. Kobayashi, S. Seetharaman, G. Parulkar, G. Appenzeller, J. Little, J. van Reijendam, P. Weissmann, and N. McKeown, “Maturing of OpenFlow and Software-defined Networking through deployments,” *Elsevier Computer Networks*, vol. 61, pp. 151–175, 2014.

[15] Open Networking Foundation, “Openflow specifications,” Dec. 2014.

[16] E. Yaacoub, R. Atat, A. Alsharoa, and M.-S. Alouini, “Mobile relays for enhanced broadband connectivity in high speed train systems,” *Physical Communication*, vol. 12, pp. 105–115, 2014.

[17] S. Banerjee, M. Hempel, and H. Sharif, “Upwarc: User plane wireless adaptable reliable communication, a new architecture for high speed train passenger internet services,” in *Proc. of WiMob*. IEEE, 2017, pp. 183–189.

[18] C. Lee, M. Chuang, M. C. Chen, and Y. S. Sun, “Seamless handover for high-speed trains using femtocell-based multiple egress network interfaces,” *IEEE Transactions on Wireless Communications*, vol. 13, no. 12, pp. 6619–6628, Dec 2014.

[19] C.-W. Lee, M. C. Chen, and Y. S. Sun, “A novel network mobility management scheme supporting seamless handover for high-speed trains,” *Computer Communications*, vol. 37, pp. 53–63, 2014.

[20] Y. Zhou and B. Ai, “Handover schemes and algorithms of high-speed mobile environment: A survey,” *Computer Communications*, vol. 47, pp. 1–15, 2014.

[21] G. F. Riley and T. R. Henderson, “The ns-3 network simulator,” in *Modeling and tools for network simulation*. Springer, 2010, pp. 15–34.