Spiderman Handover for Railway Communications

Digital Object Identifier 10.1109/MITS.2016.2557228
Date of publication: 25 October 2016

1939-1390/16©2016IEEE

Juan-Carlos Maureira
Center for Mathematical Modeling—University of Chile
E-mail: jcm@dim.uchile.cl

Abstract—In this paper, we propose a new mechanism, called the Spiderman Handover, for enabling Wi-Fi technology to support high speed mobility. It introduces two new 802.11 operational modes: the Spiderman mode and the Wireless Switch Access Point mode. We describe its operation, highlighting the **Spiderman Handover** and the **Gratuitous ARP Loop** routing update procedure as key elements for providing seamless horizontal handover for high-speed vehicles such as trains. We discuss and evaluate its operation theoretically and by means of simulation, concluding that the proposed mechanism is able to provide track-side connectivity with virtually no packet loss while keeping the observed latency invariant to the handover frequency.

1. Introduction

In this paper, we present the Spiderman Handover, a mechanism for enabling Wi-Fi technology to support high speed mobility. Our motivation is to show that this kind of technology can be used for providing a simple Wi-Fi based solution for continuous Internet connectivity, not only to low mobility users, but a complete network on-board a moving vehicle such as a high-speed train.

Figure 1 depicts our reference scenario: at the bottom we observe a moving vehicle, in our case a high-speed train following a given trajectory. In this vehicle, an in-motion network composed of multiple devices communicates with external networks through an on-board access terminal. This terminal uses a Wi-Fi based link to exchange Ethernet packets with the track-side infrastructure network. This network is composed of a set of Wi-Fi Access Points (APs) forming a line alongside the vehicle's trajectory. All APs are interconnected through a switched backbone network (OSI layer 2) which transports the traffic to a network gateway, from where it is routed at the IP layer to external networks such as the Internet.

Providing continuous connectivity in this scenario presents two challenges: (i) how to provide lossless horizontal handover for a moving network at high-speed, and (ii) how to build a track-side network composed of hundreds of thousands of access points. In this paper, we mainly address the first challenge, discussing only important aspects of the second one, which will be fully addressed in a future publication.

Now, with respect to the first challenge, we note the standard handover might produce packet losses from both outbound and inbound traffic at the Wi-Fi link in our reference scenario. On one hand, the access terminal buffers the outgoing packets when discovering the next AP in order to perform a handover operation. This buffering might lead to packet losses (tail drops) at output transmission queues. On the other hand, the inbound traffic is misrouted (and lost) on the infrastructure network when the access terminal leaves the former AP and becomes associated to a new one. This condition holds until the layer 2 path for reaching the access terminal (via the new AP) is established by traffic that is outbound for the gateway. While for low mobility the losses suffered by user's traffic might be handled by transmission protocols at higher layers in the network stack, the high frequency of the handover operation worsens the packet losses in high mobility scenarios. In conclusion, in spite of several attempts to improve the handover operation, such as the 802.11f and 802.11k protocols, the standard handover continues to be unsuitable for providing continuous bi-directional connectivity in our reference scenario.

The contribution of this paper is two-fold: on the one hand, it introduces two new Wi-Fi operational modes in order to support seamless handover at OSI layer 2, not only in a low mobility context, but in a high mobility scenario as well. And in the other hand, this paper comprehensively summarizes part of the work developed in [1], providing new analysis and discussions of revisited results. The structure of this paper is organized as follows: in the next section we formulate the problem and review the literature for open issues related to the horizontal handover in a high mobility scenario. Section 3 presents our proposed

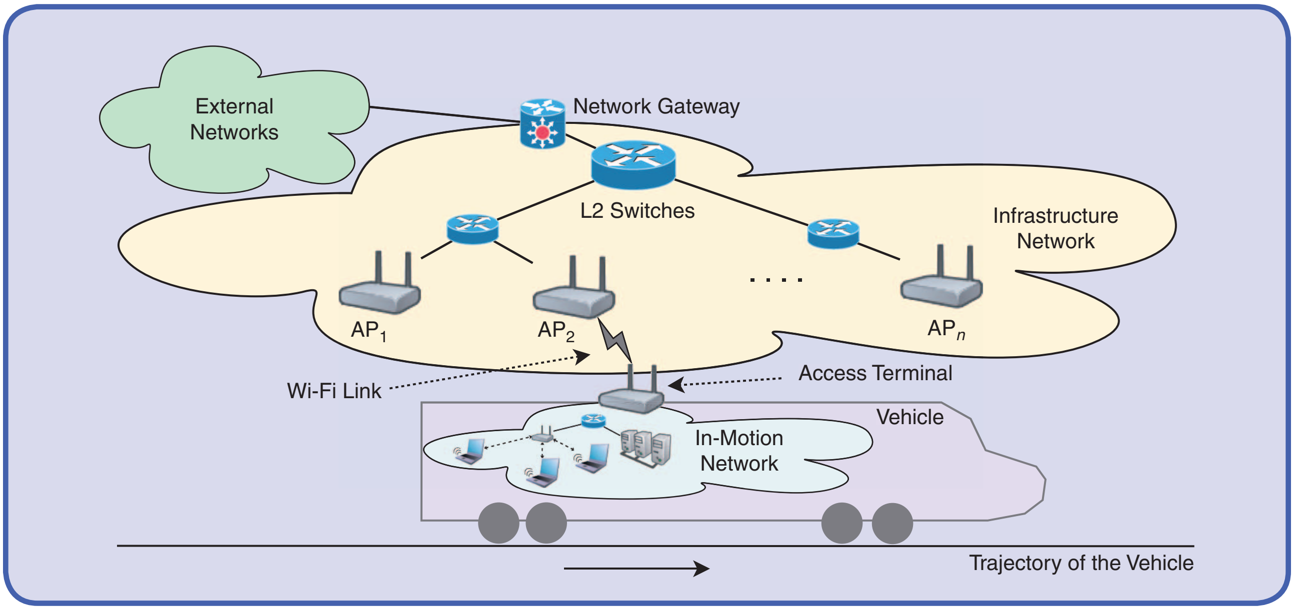

FIG 1 Scenario for communication on trains.

solution, called the Spiderman Handover, describing its architecture and operation. In Section 4, we discuss the conditions required for its successful operation in a high mobility scenario. Next, Section 5 evaluates the proposed handover scheme considering two scenarios, describing the handover time and QoS parameters to asses its performance. Finally, in Section 6, we summarize our results and present our conclusions, discussing the benefits and limitations of our proposed solution, from which we draw the further work of this research.

2. Problem Definition

"On-board Internet" has become an attractive service for train operators not only for providing Internet access to passengers, but also for improving security, monitoring and when possible, signaling and control. Current technologies adopted by train operators for providing such a connectivity have been classified by [2] and [3] into three categories: (i) a blend of Satellite, Cellular and Wi-Fi connections, (ii) the aggregation of several cellular connections, and (iii) an exclusive Vehicle-to-Infrastructure (V2I) connection. The first is the initial solution adopted by train operators and currently in service across many countries in central Europe, but with limited success due to its complexity, high cost and poor quality of service (QoS). The second has been exploited with better acceptance in the UK and the USA. However, poor cellular coverage in the countryside together with the overbooking of shared network capacity tends to yield an unsatisfactory user experience. In contrast, the third seems to better suit the high mobility scenario when studying the solution [4] implemented for the train service between the Shanghai airport and downtown. This solution is capable of delivering a continuous V2I connection up-to 16 Mb/s at speeds of 500 Km/h thanks to a proprietary handover scheme based on a distributed media access controller (MAC) together with an exclusive track-side infrastructure. This example raises the question of whether or not it is possible to implement a similar solution only relying on non-licensed and non-proprietary communication technology such as Wi-Fi. Experimental and theoretical studies ([5]–[7]) have shown that it is possible to use Wi-Fi technology in a high mobility scenario. However, its performance is limited by packet losses experienced during handover. From the physical standpoint, we can reasonably exclude the physical layer as cause of these losses since supporting evidence given in [8] suggests that Wi-Fi communication is not affected by Doppler shift or any other modulation problem when the transmitter is moving at high-speed. Indeed, if a high-power Wi-Fi link is able to communicate the in-flight status of a ballistic munition travelling at 600 m/s over a 70 Km trajectory, we can assume it will work in our reference scenario. However, from the data-link standpoint we can not state the same. The losses observed at the inbound and outbound traffic during the transfer of the data-link layer from one access point to another seem to be the main contributor to the handover losses. The literature identifies two factors as possible causes: (i) the time taken by the overall handover operation, and (ii) the number of handover operations per unit of time (the frequency of handover). On one hand, the first factor is associated by many authors ([9]–[12]) with the time taken by the neighbor discovery or scanning phase, highlighting as causes the co-channel interference of the Wi-Fi bands and the diversity of implementations of the Wi-Fi standards by vendors. While several scanning schemes have been proposed ([13]–[15]), most of them aim at reducing the handover time under the premise that seamless handover does not mean loss-free handover. On the other hand, the second factor is associated to the losses caused when transferring too often the data-link layer from one access point to another. While these losses can be mitigated by using a "Make-before-Break" handover scheme ([16]–[18]), there still exist systematic losses observed in the handover operation. In fact, the use of multiple wireless interfaces only deals with the losses observed in packet flows departing from the access terminal (the outbound traffic), having no consideration for the losses suffered by misrouted packets coming from external networks (the inbound traffic). These latter losses are discussed in [19] and [20] as a layer 3 handover problem, whereas results presented in [21] suggest that the overall layer 3 handover time (including both layer 2 handover and route updating) might be significantly reduced. Nevertheless, the connection break due to the high number of layer 2 handovers per second still causes a sufficiently significant loss of packets when travelling at high-speed. While emerging technologies such as Software-Defined Networks [22] (SDN) and OpenFlow [23] hold promise when dealing with the layer 3 handover problem, they are not yet suitable for high mobility scenarios. For instance, [24] proposes a novel handover scheme that combines an OpenFlow based infrastructure network with an access terminal featuring multiple (virtual) radio interfaces, but it is only applicable to pedestrian-type speeds. The delays introduced by the neighbor discovery and OpenFlow routes reconfiguration are far beyond the tolerance demanded by a high-speed train mobility.

In summary, as the layer 3 approaches do not seem to be completely applicable to our reference scenario, we focus our attention on reducing, or even eliminating, packet losses caused by buffer overflows and misrouting of packets during the handover operation at OSI layer 2.

3. Proposed Solution: The Spiderman Handover

The Spiderman Handover is a mechanism for providing horizontal handover to an in-motion network travelling at high-speed. Its objective is to avoid or even completely eliminate packet losses caused by the high frequency of handover. In short, it is an access terminal which uses two wireless NICs to establish a bridged link with the APs along the trajectory in the same way that the superhero Spiderman does when flying among buildings.

Figure 3 depicts the concept. Let be the switch M in motion with two cables of limited length. Cable 1 connects (bridges) the port 1 of M with the port 1 of A at the infrastructure network. Thus, A learns all the MAC addresses behind M through port 1. When M is moved until the cable 1 is extended to its maximum length, we use Cable 2 to bridge port 2 of M with port 1 of B. M is aware of the loop created by adding the second cable by keeping port 2 inactive, but with a link connected. Before removing Cable 1, M executes a routing-update procedure on port 2 to inform B (and all the other bridges) of the new route for reaching M (via B). When the routing-update packet reaches M through port 1 of A, the routing-update procedure is complete and the MAC address tables of all bridges on the path between A and B will know M through B. Then, M "unplugs" the cable from A leaving only the bridged link with B active. This mechanism is then repeated between B and C and so on.

The Spiderman Handover is implemented by two new 802.11 operational modes: **the Spiderman Mode** on the access terminal side, and **the Wireless Switch Access Point Mode**, or WSAP mode for short, on the access point side.

3.1. Spiderman Mode

The Spiderman mode is defined as a **dual wireless NIC bridge with handover capabilities**. The word *bridge* means this mode operates only at layer 2. It uses 802.11 4-address frames to encapsulate Ethernet packets and exchange them with the associated AP. Both wireless radios have their own wireless stack, controlled by the Spiderman Agent (SA) which implements the layer 2 handover logic. All the primitive operations are implemented in the 802.11 management module, and the SA commands the management module of each radio interface to perform scanning, authentication, association and packet forwarding. The SA determines which radio is *Active* or *Passive*. Thus, all packets coming from the MAC Relay Unit are forwarded to the active radio, and all packets coming from either radio are forwarded to the MAC Relay Unit. The MAC Table keeps account of all MAC addresses known to the MAC Relay Unit. Once the passive radio gets its association, the routing-update module uses this MAC Table to send routing-update packets through the passive radio. These packets should be received through the active radio in order to finish the handover operation. Each management module should forward routing-update packets with the highest priority in order to ensure the shortest delay in the updating procedure. The SA acts as an additional bridged port for the MAC Relay Unit, hiding all the internals of how the bridge link is handled. Figure 2(a) depicts the Spiderman Device (SD) architecture, showing the connections followed by packets in solid (black) lines and the relationships among components in dashed (red) lines. The Ethernet device connects the SD with the in-motion network as though it were an additional stacked switch. So the complexity of adding it is simple since the internals of the in-motion network are "transparent" to the SD.

3.2. Wireless Switch Access Point Mode

The Wireless Switch Access Point mode is **a layer 2 AP that considers each wireless association as a bridged**

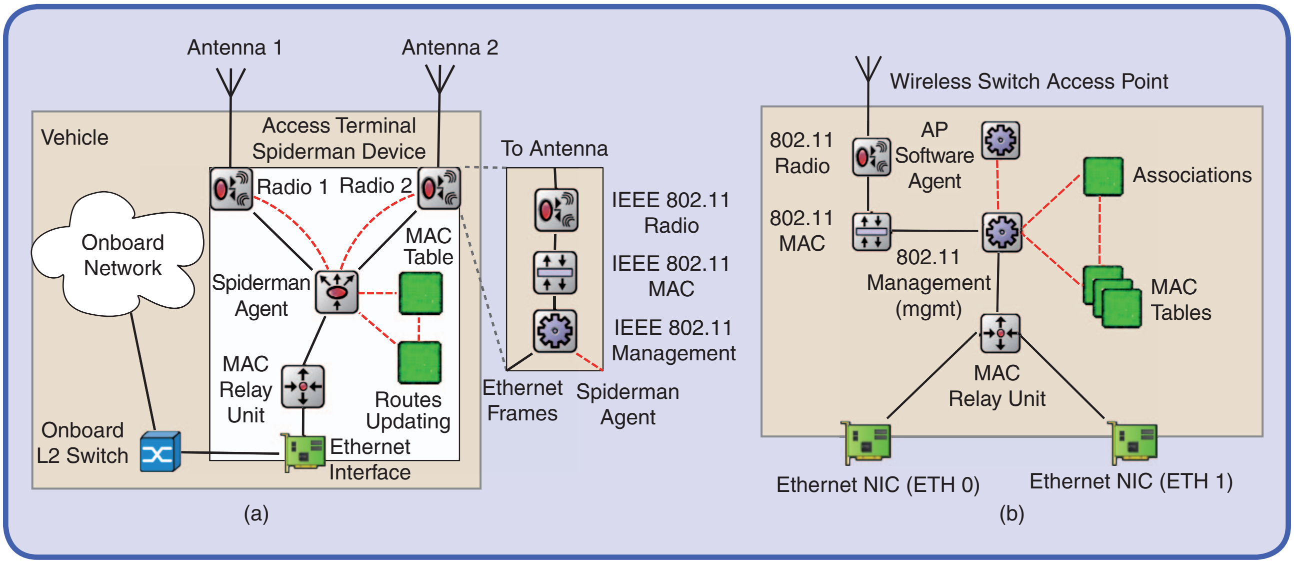

FIG 2 Spiderman Architecture. (a) The Spiderman Device; (b) The Wireless Switch Access Point.

port. It is similar to the *Master* mode of 802.11, which implements the access point functionally. However, it allows the exchange of 802.11 4-address frames with the associated SD, keeping a record of the MAC addresses known to each association in separate MAC Tables. The 802.11 management module acts as a secondary MAC Relay Unit, considering each Spiderman association as a bridged port. So the management module is able to forward the traffic to the correct SD by matching the association table with its corresponding MAC table. The 802.11 4-address frame format allows encapsulation of Ethernet traffic between the infrastructure and in-motion networks in the same way that a Wireless Distribution System (WDS) does. Figure 2(b) shows the internal architecture of a WSAP device. The primary MAC Relay Unit makes this device behave in a way similar to a multi-port bridge device, better known as a classic layer 2 switch. Therefore, the WSAP device can have multiple Ethernet NICs and it can implement any layer 2 routing protocol without loss of generality due to the wireless radio.

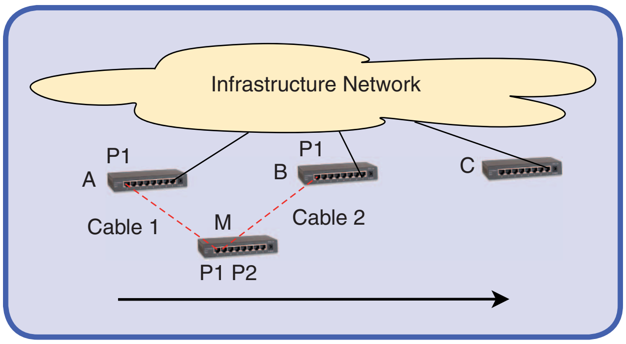

FIG 3 Moving switch example.

3.3. System Operation

Figure 4 shows the sequence of basic activities performed by the Spiderman handover. We assume that *Radio 1* is passive and *Radio 2* is active. The process begins when the passive radio has lost the connection with its WSAP. At that time the SA commands the passive radio to start the scanning procedure, which we explain in detail later. The passive radio keeps scanning until it locates the next WSAP with which to connect. Then, authentication and association is carried out. When the passive radio obtains a positive association, the routing-update procedure begins on the passive radio. This update procedure uses the MAC Table to know which addresses require updating. The routing-update packets must be received by the active radio in order to confirm that the path between the two WSAPs involved in the handover operation have been fully updated. When the last update packet has arrived via the active radio, the +5 routing-update module signals the SA that the routing update is complete. The SA notifies Radio 1 to become active and Radio 2 to become passive. At this point, the handover operation is complete. Furthermore, the passive radio holds the link with its WSAP until leaving its coverage area in order to avoid losing any misrouted packets. Then, as soon as Radio 2 loses contact with its WSAP, the SA commands it to start the neighborhood discovery phase, restarting the procedure.

3.4. Scanning Method

In our reference scenario, due to the predictability of the vehicle's trajectory, the track-side WSAP access network can be configured to use a predefined sequence of channels. Thus, the scanning method is able to preferentially scan the channels in sequence according to the currently active channel, which speeds up the neighborhood discovery phase. After

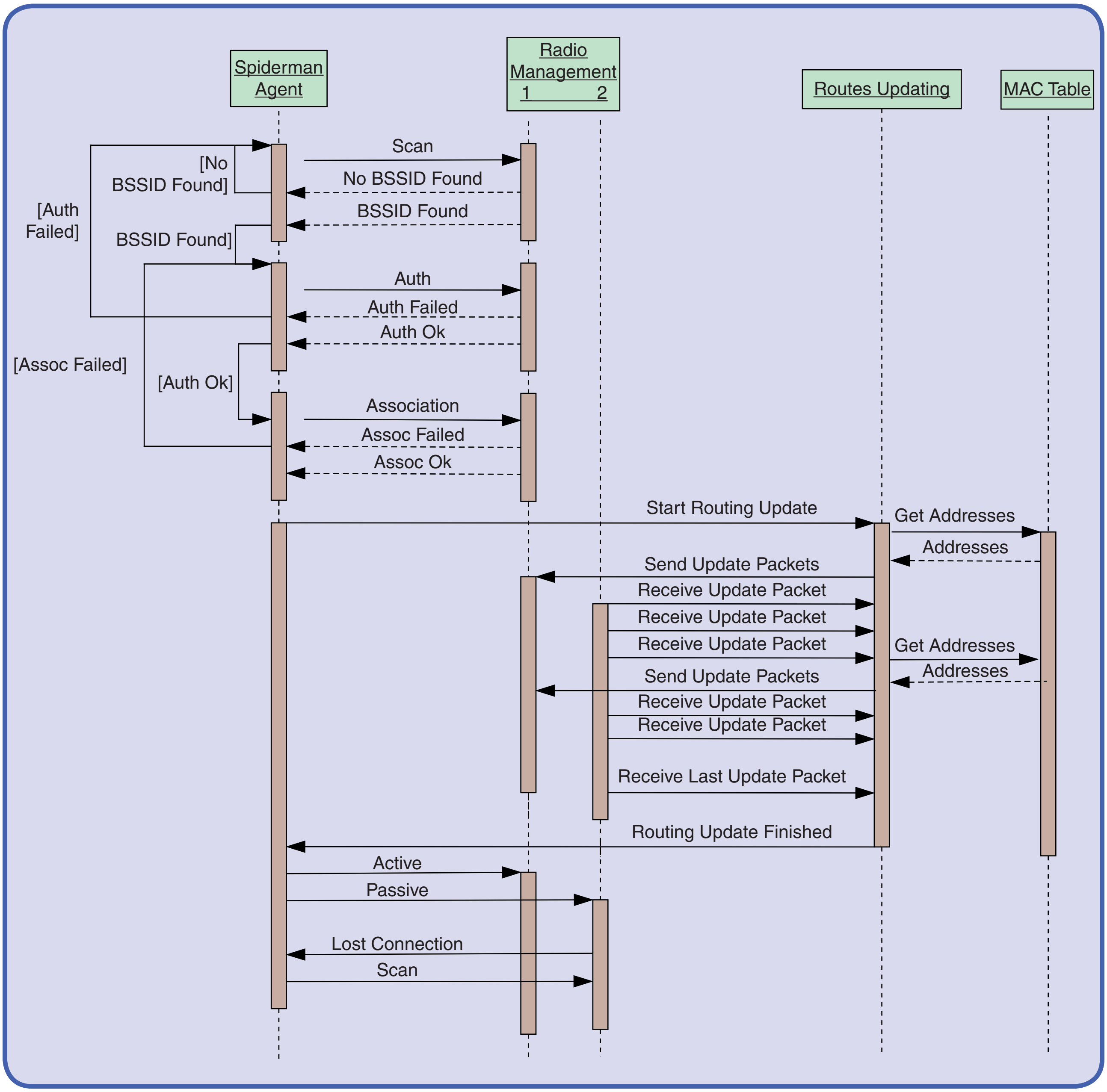

FIG 4 The Spiderman Handover procedure.

a certain number of attempts of this selective probing, the scanning method falls back to full scanning mode to allow the SD to overcome WSAP failures or misconfiguration. After performing a full scan, it begins selective probing again, repeating the loop until the next BSSID is found. By choosing the orthogonal channels of the 2.4 GHz ISM band, co-channel interference can be avoided and the channel can be probed in the minimum time allowed by the IEEE 802.11 standard (*MinChannelTime*). This increases the number of times a channel is scanned which leads to a better probability of quickly finding the next AP. The passive radio follows this procedure when searching for the next WSAP. However, when both radios are disconnected due to WSAP failures or misconfiguration, the SA begins cooperatively scanning all available channels using both radios. Thus, the list of available channels is covered using half the time typically required by a single radio and, when one of them finds the next WSAP, the active/passive roles are restored.

3.5. Routing-Update Procedure

The Address Resolution Protocol (ARP) was introduced in RFC 826 to deal with the translation of addresses between the layer 2 and layer 3 protocols. In short, it provides the layer 2 address (a MAC Address for the Ethernet protocol) of a device given a layer 3 address (an IP Address for the IP protocol) within a layer 2 network segment. Let us consider an example: two devices with IP addresses A and B are connected by the same layer 2 segment. A wants to send a layer 3 dataframe to B. Therefore, A broadcasts an ARP-Request asking for the MAC address of B. This broadcast is received by all the devices in the segment, but only answered by B via an unicast ARP-Response. When A learns the physical address of B, layer 2 communication is possible. All the bridged links along the path between A and B have learned by which link (port) A and B are known. This learning process is performed by a MAC Relay Unit inside each layer 2 switch device, which handles all the bridged links connected to it. A Gratuitous ARP is an ARP frame containing the same source and destination addresses (for both layers). When the Gratuitous ARP is directed to a particular layer 2 address, the frame is an ARP-Response. When the Gratuitous ARP is a broadcast, it is an ARP-Request.

Let us consider two consecutive WSAPs, X and Y. The SD's active radio is exchanging traffic with X and the passive radio has just associated with Y. The in-motion network is known to external networks through the layer 2 route whose destination is the active radio (the active route) and no packets have been forwarded via the passive radio. The Spiderman routing-update procedure, called **Gratuitous ARP Loop (GAL)**, is based on the broadcast of *Gratuitous ARP Requests* through the passive radio. These packets flood the infrastructure network and eventually return to the SD via the active radio, closing the "loop". The procedure starts when the SD's passive radio establishes a positive association. The GAL retrieves all the MAC addresses known to the Spiderman Agent from the MAC Table. Then, it sends a set of Gratuitous ARPs (Request) for each MAC address in bursts to the transmission queue of the passive radio. In summary, our procedure depends on three variables: **BurstSize**, **InterARPDelay** and **InterBurstDelay**, which can be adjusted according to the speed of the vehicle, the WSAP radio coverage area and the overlapping distance among two consecutive WSAPs. Figure 6 depics the procedure.

Illustrating the GAL operation, let us assume that the MAC Table has 48 registered MAC addresses and the BurstSize is equal to 10. The passive radio obtains positive association with a new WSAP. The SA commands the routing-update module to start the GAL procedure. The GAL gets the first 10 MAC addresses from the MAC Table and broadcasts a Gratuitous ARP packet for each one at regular intervals of InterARPDelay seconds via the passive radio. When ARP packets are received via the active radio, they are registered as having arrived in the MAC Table. After finishing the first burst of 10 MACs, GAL waits for InterBurstDelay seconds before starting the second burst of 10 unreceived MACs, and it continues until all MACs in the MAC Table are registered as received via the active radio, concluding the GAL process. As the GAL time corresponds to the routing-update procedure time, we use it to represent the handover time for SD.

3.6. Infrastructure Network

Building a large track-side network may be challenging in terms of capital investment and operational costs. For this reason, these types of networks should be designed by finding a suitable trade-off between desirable properties such as resilience, deployment costs, and QoS parameters. While a deep analysis of this trade-off is the subject of a future publication, we briefly describe our proposal for designing a linear access network with a resilient backbone topology capable of supporting effective layer 2 communication for thousands of nodes.

Let us consider a set of WSAPs located alongside the track at regular intervals. They form a *linear access network* if each pair of consecutive WSAPs are directly connected. A *gateway node* is any node that is connected to a backbone node, which are side-by-side at the same site. The *backbone topology* consists of all links connecting backbone nodes. In order to find the location of backbone nodes, we define two recursive construction rules for a set of n access nodes: (i) choose the first, the middle and the last node. Put a backbone node at those positions and place a link connecting the middle one with the first and the last. And (ii) divide the set into 3,5 or 7 subsets containing the same number of contiguous nodes. Choose the central access node of each subset and place a backbone node there. From the backbone node of the central subset, place a link to the backbone nodes of all other subsets. Apply either these two rules recursively to each set of access nodes bracketed by two consecutive backbone nodes as long as the number of access nodes between two successive backbone nodes is greater than some minimum value, let us say D. Finally, select a central backbone node and apply a **Spanning Tree protocol** (STP) in order to span the resulting chordal topology [25] into a tree topology, which is reconfigured by the STP when a node or link fails. Using a combinatorial analysis, we have shown that the probability of having network partitions is low and it decreases in steps of D [1, Section 7.3], and the distribution of the number of hops to the STP root node is regular enough as to exhibit uniform QoS parameters. Figure 5 depicts an example of a backbone topology for 134 access nodes. The STP distance to the root node g_{67} is shown below each gateway node.

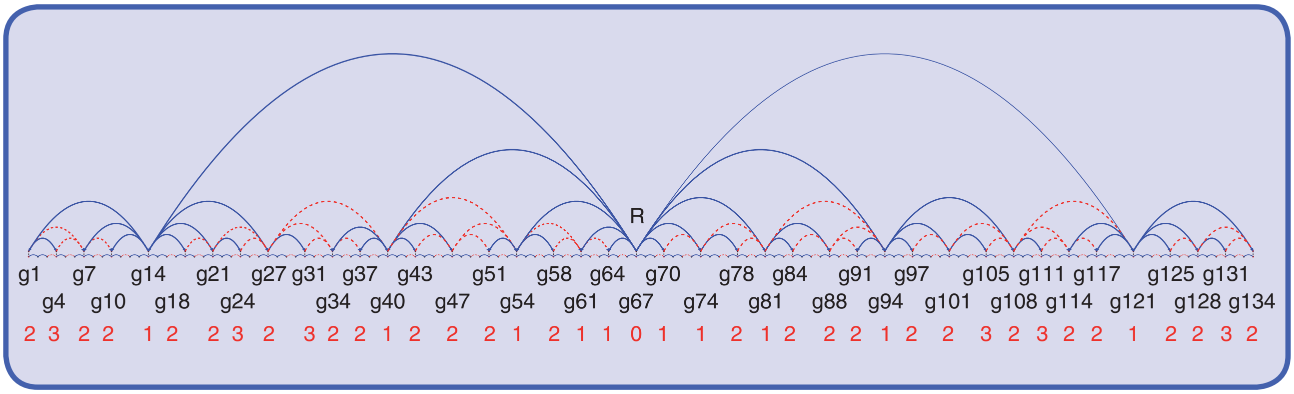

FIG 5 Backbone topology for access network of 134 nodes with STP distances to the root node R.

4. Discussion

We focus the discussion on the operation of the Gratuitous ARP Loop (GAL) algorithm since it is the key to the Spiderman Handover for achieving a seamless operation. We discuss the limits of GAL parameters (*InterARPDelay*, *InterBurstDelay* and *BurstSize*) required to operate within the high mobility scenarios shown in Figure 1.

First, let us consider two consecutive WSAPs with an overlapping distance of d_o meters between their coverage areas, a vehicle speed of v m/s, and a negligible Authentication, Authorization and Accounting (AAA) delay. The effective handover time is given by Equation 1.

$$t_h = \frac{d_o}{v} - t_{\text{discovery}} \tag{1}$$

Thus, when considering $v = 350$ km/h, $d_0 = 80$ meters, and $\max(t_{\text{discovery}}) = 500$ ms, the minimum theoretical handover time is approximately 322 ms. Ideally, the SD should finish the updating procedure before t_h in order to ensure no packet losses due to misrouting. However, this lower bound time might be larger if the next WSAP is detected before the 500 ms (a shorter $t_{\text{discovery}}$). The **maximum theoretical handover time** is attained when $t_{\text{discovery}} \to 0$, in which case it is approximately 822 ms for the chosen values of v and d_0.

Second, as the *InterARPDelay* aims at avoiding buffer overflows and collisions when transmitting Gratuitous ARP packets within a single burst, we note that two situations may occur:

1) The WSAP begins re-transmission of the first broadcast before the passive radio begins transmission of the second one: in this case, the radio might find the medium "busy". Therefore, it will perform an exponential back-off, delaying transmission of the second broadcast frame.
2) When the transmission takes place at the limit of the WSAP coverage area, the passive radio might not detect the first broadcast re-transmission due to a low SNR. Therefore, it will begin transmission of the second one, producing a collision between both.

Hence, the GAL should wait a reasonable time before sending the next Gratuitous ARP packet in order to reduce the risk of delaying it (back-off) or losing it (collision). We measured this risk by using a testbed consisting of two 802.11b WSAPs 150 meters apart and connected by a backhaul dummy network without background traffic. We placed a SD in the middle between the two WSAPs and we performed the GAL for 250 MAC addresses in bursts of 50 addresses. We considered values of *InterARPDelay* from 0 to 5 ms in steps of 1 ms. From the results we drew two conclusions: (i) when sending the Gratuitous ARPs one after the other (*InterARPDelay* = 0 ms), we evidenced re-transmission due to both causes stated above, which will eventually lead to longer delays in propagating the new route to the SD; and (ii) the minimum mean time taken by a burst of 50 MAC addresses for completing the loop without exhibiting back-offs and collisions is ≈ 150 ms. Consequently, an *InterARPDelay* of 3 ms can be used to obtain the same result while avoiding the buffering of Gratuitous ARPs in the priority queue of the passive radio without considering background traffic. However, this value should be set higher when transmitting bursts in real-world traffic conditions.

Third, note that as soon as the first burst is sent and propagated into the infrastructure network, traffic starts arriving at the WSAP with which the passive radio is associated. Thus, if there is insufficient time between two consecutive bursts, the traffic might be buffered and eventually tail dropped. Hence, some extra time is required after transmitting a burst. For this purpose, an *InterBurstDelay* helps to avoid such excessive queuing of the users' dataframes due to the transmission of a burst. Finding an appropriate value for this delay, we note that the time required by the GAL to complete the updating procedure (t_{gal}) should be equal to or less than the minimum handover allowed. In other words: $t_{\text{gal}} \leq \min(t_h)$. Let be d_{IArp} the *InterARPDelay*, d_{IBurst} the *InterBurstDelay*, B_s the burst size and $(1 + a)T_s$ the number

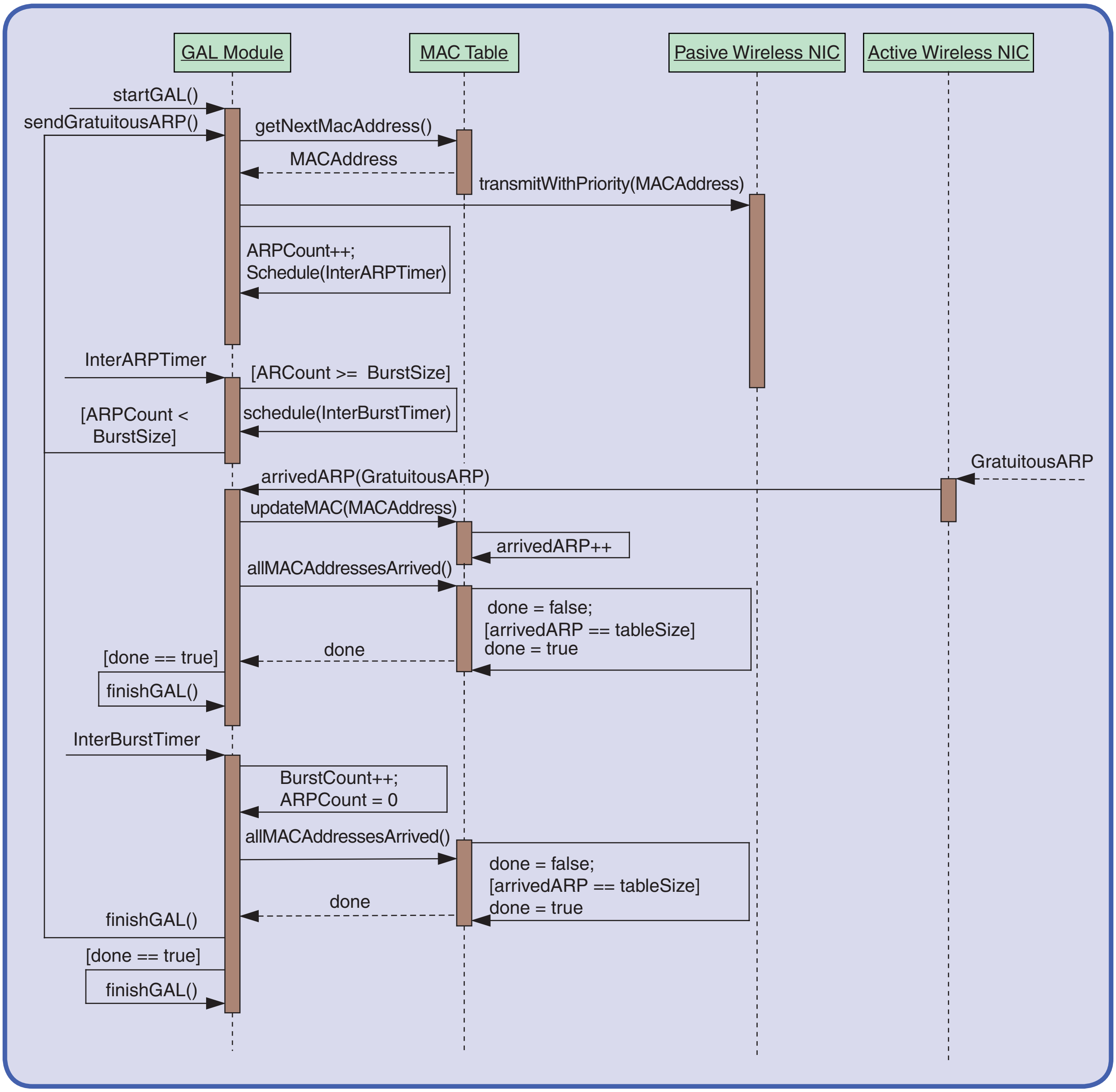

FIG 6 The Gratuitous ARP Loop procedure.

of MAC addresses registered in the MAC Table considering a fraction α of re-transmissions. The maximum theoretical GAL time is given by Equation 2.

$$\max(t_{\text{gal}}) = d_{\text{IArp}} \cdot (1+\alpha)\,T_s\left(\frac{B_s-1}{B_s}\right) + d_{\text{IBurst}}\left(\frac{(1+\alpha)\,T_s - B_s}{B_s}\right) \tag{2}$$

Figure 7(a) shows the GAL time (t_{gal}) for different percentages of re-transmitted MAC addresses when assuming several *InterBurstDelay* values. Therefore, when assuming $\min(t_h) = 0.8$ seconds (≈ 220 km/h with $d_o = 80$ meters), we observe that an *InterBurstDelay* of 0.02 seconds allows us to re-transmit each of 50 MAC addresses once (a total of 100 transmissions). In summary, we set the *InterBurstDelay* according to the minimum theoretical handover time allowed for the imposed mobility conditions (maximum speed and overlapped distance of WSAP coverage areas) and the maximum percentage of MAC addresses that might be re-transmitted during the route-updating procedure.

Finally, the *BurstSize* helps to reduce the excessive queuing when transmitting either user dataframes or Gratuitous ARPs with the pasive radio. The basic rule for adjusting it comes from the idea that longer bursts mean shorter GAL times (and shorter bursts mean longer GAL times). Thus, for a low traffic load, a longer burst is reasonable since there is no risk of producing excessive queueing of user dataframes, which lowers the GAL time we can achieve. However, for a high traffic load, a shorter burst is required to avoid buffer overflows for both Gratuitous ARPs and user dataframes. Figure 7(b) depicts the GAL time for different *BurstSize* values for the parameters shown within the plot. For a minimum handover time of 0.8 seconds and a *BurstSize* of 10, the GAL process is able to re-transmit each of 50 MAC addresses in the worst case that the first burst is completely lost. However, to maintain the possibility of re-transmitting all addresses during a shorter minimum handover time, let us say 0.72 seconds, the *BurstSize* must be increased to 30. We evaluated different values for the GAL parameters by means of experimentation. The setup was similar to the one previously explained, but now considering background traffic. We measured the passive radio transmission queue when performing a GAL of 250 MAC addresses, finding that with an *InterARPDelay* of 7 ms, an *InterBurstDelay* of 20 ms and a *BurstSize* of 10, we did not evidence a significant increase in the transmission queue length. Figure 7(c) depicts the maximum theoretical GAL time according to these values for a range of MAC addresses. The long dashed horizontal lines (in blue) show the minimum handover time allowed for a given speed, which are indicated beside the plot. Note when assuming a vehicle speed of 100 m/s (360 km/h) and a 200% ARP re-transmission rate (each Gratuitous ARP is transmitted 3 times), the GAL can not operate for more than 10 MAC addresses. For 70 m/s (252 km/h), not more than 20 MAC addresses can be handled. And for 60 m/s (216 km/h), the GAL can experience a 100% re-transmission rate for 50 MAC addresses without finishing early.

4.1. Scalability

When thinking of a simplistic network design, where on-board stations communicate (in layer 2) with an IP gateway located at the infrastructure network, the proposed handover mechanism may present scalability issues. However, these issues can easily be mitigated by applying well-known design principles such as segmenting the network into VLANs. For IPv4, either *Natting* or *IP Mascarading* on a per-VLAN basis can be used to hide station addresses behind a single IP address, which implies the use of a unique MAC address for traversing the infrastructure network. On the other hand, IPv6 can be handled by tracking the MAC address of a next-hop IP address assigned to a router on-board the vehicle. In either case, the GAL only needs to update a single MAC address in order to exchange traffic with stations within on-board VLANs. In practice, either by using IP Masquerading or static routing, the scalability of the handover of *n* MAC addresses can be transformed into a single MAC address handover, solving any scalability issue related to the number of hosts and the speed of the vehicle.

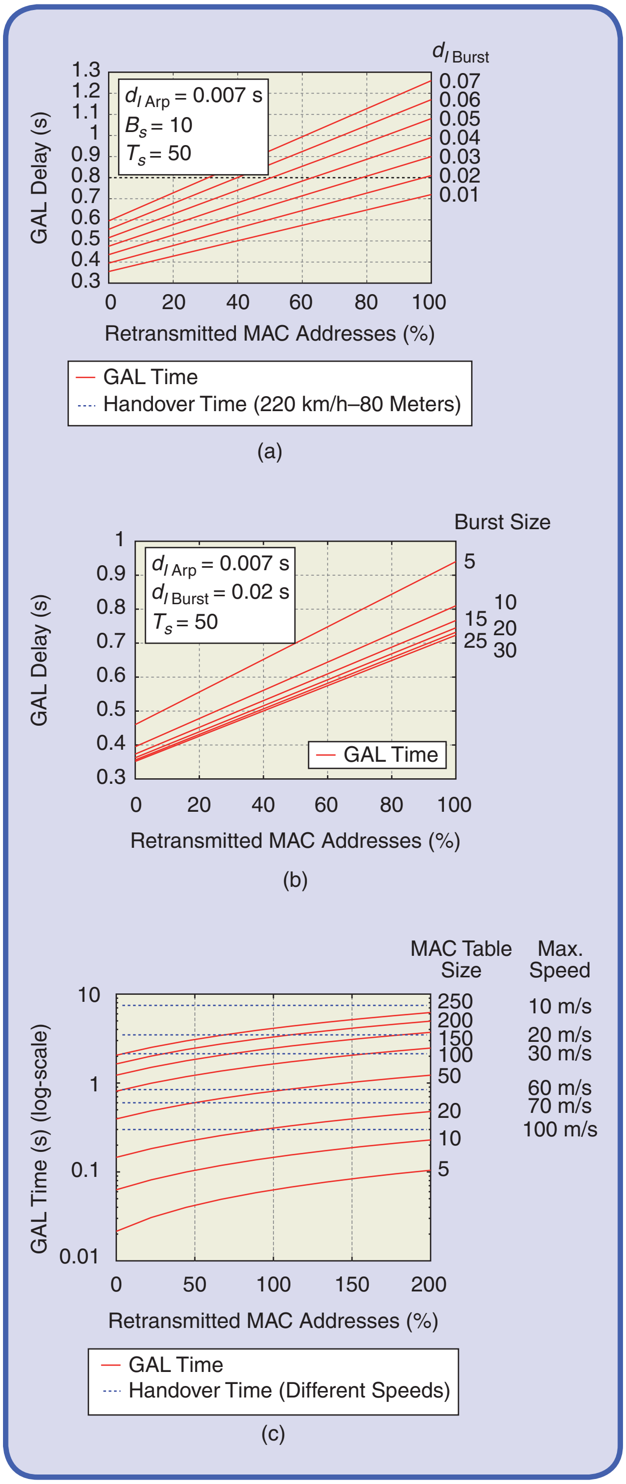

FIG 7 The theoretical Gratuitous ARP time. (a) Considering re-transmissions; (b) Different BurstSize; (c) *InterARPDelay* = 7 ms, *InterBurstDelay* = 20 ms and *BurstSize* = 10.

5. Evaluation

This evaluation compares the Spiderman handover with the Standard 802.11 handover in a simulated high mobility scenario. This scenario considers a small railway of

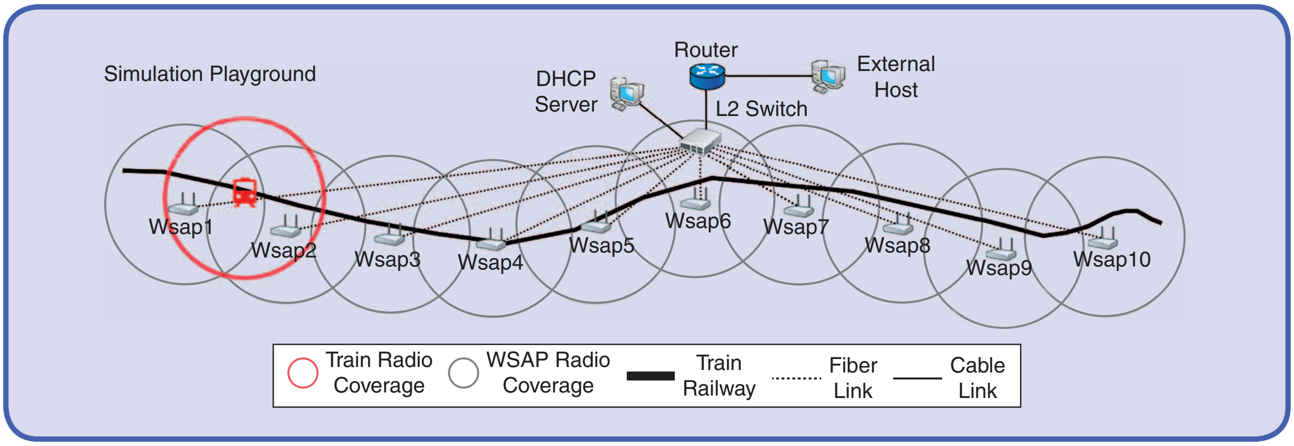

FIG 8 The scenario simulated for the GAL evaluation.

2 km long and two trains identified by the labels **One-Radio Train** and the **Spiderman Train**. The railway is covered by 10 WSAPs and connected to a layer 2 switch by fiber links. This switch is connected to a layer 3 Router (network gateway), providing access to an external host, representing an external network. Each WSAP follows a predetermined sequence of channels (1,6,11) according to the requirements of the Spiderman Device (see Section 3.4). The **One-Radio Train** is equipped with a normal 802.11b access terminal performing Natting of all the hosts inside the in-motion network, and the **Spiderman Train** uses a Spiderman access terminal for bridging the in-motion network with the infrastructure network. We implemented an optimized scanning method for the normal access terminal in order to make the comparison fair in terms of the neighbor discovery times. This optimized scanning method scans the same sequence of channels as the SD. The scanning timers *MinChannelTime* and *MaxChannelTime* are configured to 1 ms and 10 ms respectively for both the Standard and the Spiderman Train, and the WSAP lost detection is set at 10 beacons to avoid false handovers due to saturation of the Wi-Fi link. We evaluated each train for speeds from 10 m/s up to 100 m/s for 1500 seconds of simulation with 10 repetitions per train. The GAL parameters used for this evaluation are: *BurstSize* = 10, *InterARPDelay* = 7 ms and *InterBurstDelay* = 20 ms, corresponding to the values discussed in the previous section. The coverage of each WSAP is 230 meters wide, with an 80-meter overlap between each pair of consecutive WSAPs. Figure 8 depicts this scenario.

We used the *OMNeT++* 4.1 discrete event simulator for performing this evaluation. The described scenario was modelled using the *INET Framework* version 20100323 branched in march 2010 with further modifications to allow for the measurement of the reception power on IEEE 802.11b devices, logging of some radio operations and some improvements to the IEEE 802.11 management stack. Train mobility was modelled according to [26] for a fixed train speed. The radio interference model is based on an "additive-noise-signal" evaluation among all the frames "on the air". The radio propagation model used was the Free *Space Path-loss* with path loss coefficient $\alpha = 3.2$ in order to obtain a sensitivity threshold of –86 dBm on the border of a coverage area. Thermal noise was set to –110 dBm and the radio transmission power was 100 mW.

5.1. Spiderman v/s Standard (Optimized) Handover

We compared the Spiderman Handover with the Standard handover for fixed number of hosts at several speeds. As mentioned earlier, we introduced some optimization to the standard scanning method in order to speed up its operation and make our comparison fair to both handover schemes. We use ideas coming from the literature, specially from neighbor graphs [27] and the tuning of timers [15]. We assumed a constant bit rate (CBR) of background traffic for this evaluation. It is generated by a modified ICMP Ping application (echo) which generates bidirectional traffic between the internal hosts (inside the in-motion network) and the external host. We consider 50 internal hosts, each one transmitting a "ping" at intervals between 0.15 and 0.25 seconds (distributed uniformly at random). Each internal host uses a packet size of 1024 bytes to saturate the wireless up-link. In normal (static) conditions, the up-link has an average of 1 packet in the transmission queue, therefore, the wireless radio always has a packet to transmit. Figure 9 shows the handover time for the Spiderman handover (Spiderman Train) and the Standard Optimized handover (One-Radio Train).

From Figure 9, we note that the Spiderman handover takes longer than the Standard Optimized handover for the same number of hosts. This result is explained by the fact that the optimized neighborhood discovery performs worst when listening for 3 channels in the WSAP channel sequence for *ProbeDelay+MaxChannelTime* seconds before finding the next WSAP. This worst case performance (0.15 seconds) is less than the minimum time taken by the GAL (0.395 seconds) to propagate the routing information for 50 hosts. Nevertheless, recall that the single-radio train is performing natting of all the hosts inside the in-motion network, hence, it only needs to make handovers for one station (the access terminal). In contrast, the Spiderman Handover is individually transferring the data-link layer for 50 hosts.

5.2. GAL Time/Handover Time

We studied the GAL time at the minimum representative speed for a high mobility scenario, let us say 60 m/s. Fixing the speed guarantees the same number of handover operations for each host inside the in-motion network. We examined scenarios involving from 1 to 250 hosts under two traffic profiles: **without background traffic and with background traffic**. Throughout the entire simulation time, all hosts are sending ICMP pings to the external host at regular intervals of 1 second (± 0.01s of random jitter) while one host exchanges a TCP stream with the external host. This TCP stream generates a constant saturation and overflows at the SD Wi-Fi uplink. Simulation results were examined for each scenario, focusing on basic GAL statistics such as the minimum, maximum and average number of Gratuitous ARPs sent and the percentage of re-transmitted MAC addresses.

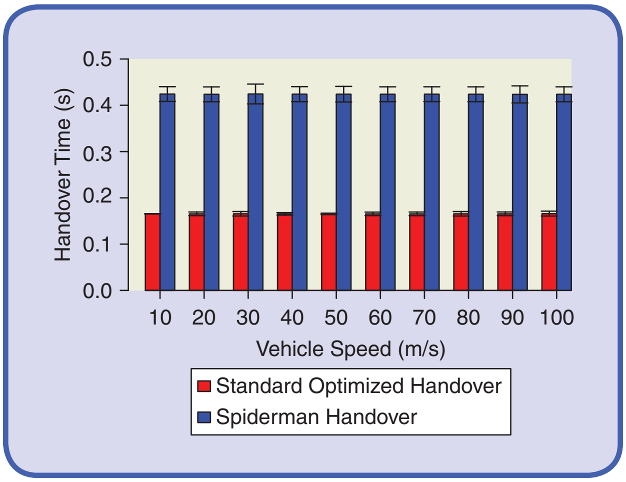

FIG 9 Handover time of 50 MAC addresses at various speeds.

From the results shown in Figure 10 we conclude that the GAL time increases with the traffic load, which is reasonable when considering the aggregation of end user traffic on the SD wireless interface. The scenario with background traffic has a GAL time which is on average 10% higher than that for the scenario without traffic. The standard deviation is also larger for the scenario with traffic. Examining the minimum values observed for the GAL times, we notice they agree with the expected minimum value, except for the 1, 200 and 250 host cases. For the latter two cases, the observed GAL time approaches 1.357 seconds, while the maximum handover time allowed for 60 m/s is 1.33 seconds, suggesting that the GAL has ended prematurely. Indeed, while the WSAPs are approximately 150 meters apart, the distance traveled is not necessarily a straight line between consecutive WSAPs (for example between *wsap9* and *wsap10*). The single host case deserves special attention since the observed GAL time for a single MAC address is not properly represented by the theoretical GAL time (see equations for t_{gal} in Section 4). In practice, the observed GAL time is not affected by the GAL delays (*InterARPDelay* and *InterBurstDelay*) when there are fewer MAC addresses than the burst size. This fact causes the Spiderman Handover to perform similar to the Stantard Optimized Handover in terms of the handover time. Finally, in terms of the observed percentages of re-transmitted MAC addresses, we found the maximum GAL time agrees with the expected maximum theoretical GAL time shown in Figure 10(c).

5.3 Packet Losses

We used the same ICMP Ping application and configuration as was used in the previous evaluation for contrasting

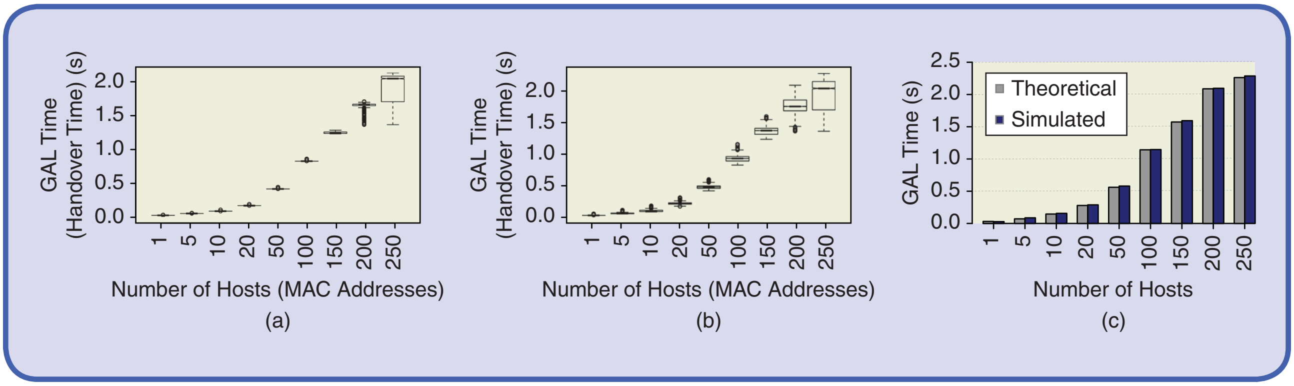

FIG 10 GAL delay for different numbers of hosts. (a) without background traffic; (b) with background traffic; (c) Theoretical and simulated values.

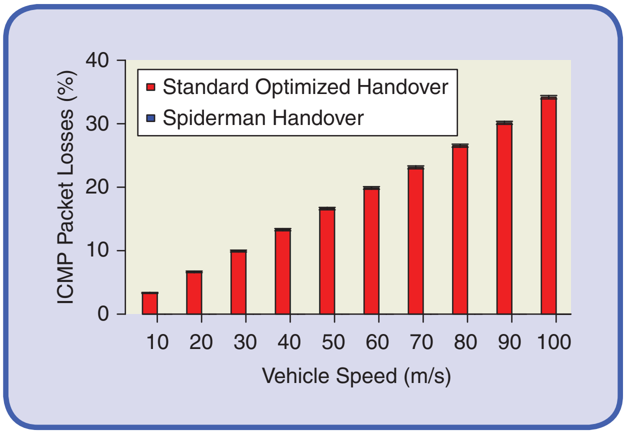

FIG 11 ICMP Packet losses for 50 MAC addresses at different speeds.

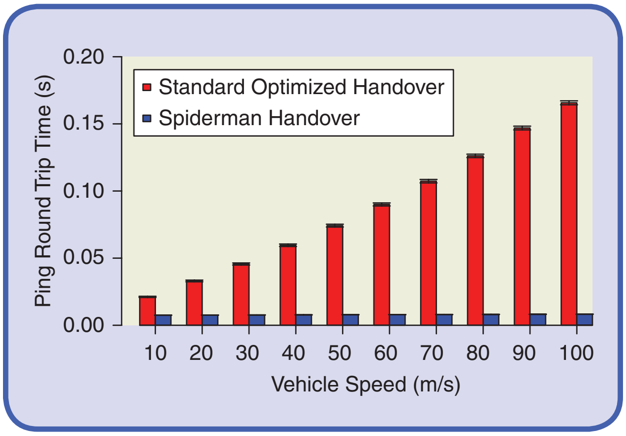

FIG 12 ICMP Ping Round Trip Time (RTT) for 50 MAC addresses at different speeds.

the packet losses between the Spiderman train and the OneRadio-Train. We measured the percentage of packet osses by inspecting the sequence numbers of ICMP packets during 10 replicates of the simulation time (1500 seconds) at different speeds ranging from 10 to 100 m/s for 50 hosts on-board the train.

Results showed that the *Spiderman handover* exhibits no apparent packet losses. In fact, the observed losses for the Spiderman handover are so small ($\approx 0.02\%$) that they are insignificant in comparison to the losses observed for the standard handover approach (from 5% to 40% according to vehicle speed). Nevertheless, this result corresponds to the traffic load generating an average buffer length of 1 packet at the passive radio. Under heavier traffic conditions, buffer overflows might be unavoidable for both cases.

5.4 Round-Trip Time

Once again, we modified the ICMP application to measure the Round Trip Time (RTT) from the in-motion network (50 hosts) to the external host. The traffic conditions were similar to the previous section, saturating the access terminal Wi-Fi uplink up-to an average of 1 packet in the transmission queue.

Results suggest that the RTT increases with the vehicle speed when using the Standard Optimized handover, while it remains invariant to vehicle speed when using the Spiderman handover. This result is explained by the fact that the Standard Optimized handover buffers all the incoming dataframes when discovering the neighborhood, which leads to a buffer overflow of the active radio's transmission queue. This way, disregarding losses, the last queued dataframe suffers the additional delay of dequeuing the whole buffer plus the time when the buffer was active. Thus, when combining this extra delay with the frequency of handover, we notice that the higher the frequency, the larger the average RTT. Nevertheless, this result does not mean that all hosts will always experience a higher delay when increasing the speed. It only means the host may experience a larger delay when the access terminal recovers the data-link layer from the next WSAP. Therefore, its average RTT will increase when performing a larger number of handovers per unit of time.

6. Conclusions

In this paper we have presented the Spiderman Handover, an extension of the Wi-Fi standard to enable V2I communications for high mobility scenarios. Focusing on an in-motion network on-board a high-speed train, we have described the scheme and presented theoretical and empirical analyses of the proposed handover operation. The results of these analyses were used to design two simulated scenarios for testing our initial hypothesis that it is possible to use Wi-Fi technology to provide V2I communications in a high mobility scenario.

From the reported results we draw the following conclusions: (i) the proposed handover scheme is able to cope with packet losses caused by the handover operation for inbound and outbound traffic at the Wi-Fi V2I link. (ii) observed delays and losses are invariant to the handover frequency and vehicle speed. (iii) the impact of the proposed infrastructure network on packet delay and losses is negligible. (iv) scalability issues can be solved by using IP Masquerading or static routing. And (v) Under heavy traffic conditions, performance suffer significant degradation when more than one train is associated to the same WSAP.

Based on these conclusions, we claim that the Spiderman Handover is able to provide continuous V2I Wi-Fi communication for vehicles along a defined trajectory, not only at low speed, but at high speed as well. However, there are still open questions such as how to cost-effectively deploy and operate the infrastructure network, how failures impact QoS parameters over time and how packet losses are affected when multiple SDs attempting to share the same WSAP. These questions will be addressed in future publications, in particular, we intend to examine the following: (i) the design of the infrastructure network ruled by the an optimal combination of the number of ports for a backbone node (degree) and the total length of links to be used for connecting them. (ii) the resilience of the resulting network in terms of design and operational parameters, and (iii) the use of multiple access terminals placed throughout the length of a train (at both ends for example) as a way to avoid two SDs associating to the same WSAP and hence reducing packet losses caused by saturation of the Wi-Fi channel. In terms of the limitations of using Wi-Fi for V2I communications, there are the complexities of operating and maintaining a large infrastructure network composed of hundreds or thousands of nodes. However, this limitation is offset by the benefit of having an exclusive communication network that has the potential for being tailored to a variety of new services ranging from providing free public Internet access in massive transportation systems to new ITS applications for enabling telecommuters to work remotely.

Finally, we must emphasize that neither the Spiderman Handover nor any other seamless handover scheme can solve the problem of buffer overflows caused by excessive traffic load. Recall that an access terminal aggregates all on-board traffic and its efficiency not only depends on the performance of the handover operation, but also depends on other factors such as network capacity, electromagnetic interference and anything else that might affect the performance of Wi-Fi communication.

7. Acknowledgements

This research was fully funded by CONICYT-INRIA grant number 07 – *DOCINRIA* – 05. Special thanks to Jean-Claude BERMOND and Olivier DALLE for their guidance and support through the development of this research. Also to Andrew Hart and Jorge Amaya for their comments, suggestions and discussions to improve this paper.

References

[1] J.-C. Maureira. (Jan. 2011). Internet on rails, Ph.D. thesis, Université Nice Sophia Antipolis [Online]. Available: https://tel.archives-ouvertes.fr/tel-00594951

[2] D. T. Fokum and V. S. Frost, "A survey on methods for broadband internet access on trains," *IEEE Commun. Survey Tutorial*, vol. 12, no. 2, pp. 171–185, Apr. 2010.

[3] E. Masson, M. Berbineau, and S. Lefebvre, "Broadband internet access on board high speed trains, A technological survey," in *Proc. 8th Int. Workshop Communication Technologies for Vehicles, Nets4Cars/Nets4Trains/Nets4Aircraft*, Sousse, Tunisia, 2015, pp. 165–176.

[4] M. Emmelmann, "System design and proof-of-concept implementation of seamless handover support for communication-based train control," in *Intelligent Transportation Systems Series. Vehicular Networking: Automotive Applications and Beyond*. New York: Wiley-Blackwell, Jun. 2010.

[5] M. Dynamics. (2005-2007). High speed mobility test: Mesh dynamics. Univ. Wüurzburg., Inst. Comput. Sci., Tech. Rep. [Online]. Available: http://www.meshdynamics.com/documents/MD_mobility_report.pdf

[6] M. Hempel, H. Sharif, T. Zhou, and P. Mahasukhon, "A wireless test bed for mobile 802.11 and beyond," in *Proc. 2006 Int. Conf. Communications and Mobile Computing*, New York, 2006, p. 1003. [Online]. Available: http://portal.acm.org/citation.cfm?doid=1143549.1143749

[7] H. Sharif, M. Hempel, and P. Mahasukhon, "Performance of IEEE 802.11b in mobile railroad environments," in *Proc. IEEE 62nd Vehicular Technology Conf.*, 2005, pp. 2527–2531.

[8] W. P. D'Amico and M. H. Lauss, "Wireless local area network flight demonstration for high Doppler conditions," *Johns Hopkins APL Tech. Dig. (Appl. Phys. Lab.)*, vol. 25, no. 4, pp. 335–342, 2004.

[9] A. Mishra, M. Shin, and W. Arbaugh, "An empirical analysis of the IEEE 802.11 MAC layer handoff process," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 33, no. 2, p. 93, Apr. 2003.

[10] D. Murray, M. Dixon, and T. Koziniec, "Scanning delays in 802.11 networks," in *Proc. IEEE Int. Conf. Next Generation Mobile Applications, Services and Technologies*, 2007, pp. 255–260.

[11] R. Pries and K. Heck. (2004, Sep.). Performance comparison of handover mechanisms in wireless LAN networks. Univ. Wüurzburg. Inst. Comput. Sci., Sydney, Australia, Tech. Rep. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.148.5025\&rep=rep1\&type=pdf

[12] R. Corvaja, A. Zanella, D. M., A. Tontoli, and P. Zennaro, "Experimental performance of the handover procedure in a WiFi network," in *Proc. 7th Int. Symp. Wireless Personal Multimedia Communications*, 2004, pp. 12–15.

[13] Y. Chen, M. Chuang, and C. Chen, "Deucescan: Deuce-based fast handoff scheme in IEEE 802.11 wireless networks," *IEEE Trans. Veh. Technol.*, vol. 57, no. 2, pp. 1126–1141, Mar. 2008.

[14] M. Emmelmann, S. Wiethölter, and H. Lim, "Influence of network load on the performance of opportunistic scanning," in *Proc. IEEE 34th Conf. Local Computer Networks*, 2009, vol. 20, pp. 601–608.

[15] S. Rizvi, M.-A. Khan, and A. Riasat, "Active scanning: A better approach to reduce handover time at MAC layer for wireless networks," in *Proc. IEEE 2nd Int. Conf. Computer, Control and Communication*, 2009, pp. 1–4.

[16] V. Brik and S. Mishra, and A. Banerjee, "Eliminating handoff latencies in 802.11 wlans using multiple radios: applications, experience, and evaluation," in *Proc. 5th ACM SIGCOMM Conf. Internet Measurement*, Berkeley, CA, 2005, pp. 27–27.

[17] K. Ramachandran, S. Rangarajan, and J. Lin, "Make-before-break mac layer handoff in 802.11 wireless networks," in *Proc. IEEE Int. Conf. Communications*, 2006, vol. 10, pp. 4818–4823.

[18] A. Adya, P. Bahl, J. Padhye, A. Wolman, and L. Zhou, "A multi-radio unification protocol for IEEE 802.11 wireless networks," in *Proc. 1st Int. Conf. Broadband Networks*, 2004, pp. 344–354.

[19] V. Vassiliou and Z. Zinonos, "An analysis of the handover latency components in mobile IPv6," *J. Internet. Eng.*, vol. 3, no. 1, pp. 230–240, Jan. 2010.

[20] J. Xie, I. Howitt, and I. Shibeika, "IEEE 802.11-based mobile IP fast handoff latency analysis," in *Proc. IEEE Int. Conf. Communications*, 2007, pp. 6055–6060.

[21] S. Fowler and S. Zeadally, "Fast handover over micro-MPLS-based wireless networks," in *Proc. IEEE 11th Symp. Computers and Communications*, 2006, pp. 181–186.

[22] Open Networking Foundation. (2012). Software-defined networking: The new norm for networks. ONF White Paper, pp. 1–12.

[23] N. Mckeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. Shenker, J. Turner, and S. Louis, "OpenFlow: Enabling innovation in campus networks," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 38, no. 2, pp. 69–74, 2008.

[24] P. Dely, A. Kassler, L. Chow, N. Bambos, N. Bayer, H. Einsiedler, C. Peylo, D. Mellado, and M. Sanchez, "A software-defined networking approach for handover management with real-time video in WLANs," *J. Modern Transport.*, vol. 21, no. 1, pp. 58–65, 2013.

[25] J.-C. Bermond, S. Choplin, and S. Pérennes, "Hierarchical ring network design," *Theory Comput. Syst.*, vol. 36, no. 6, pp. 663–682, Oct. 2003.

[26] J.-C. Maureira, P. Uribe, O. Dalle, T. Asahi, and J. Amaya, "Component based approach using omnet++ for train communication modeling," in *Proc. IEEE 9th Int. Conf. Intelligent Transport Systems Telecommunication*, 2009, pp. 441–446.

[27] M. Shin, A. Mishra, and W. Arbaugh, "Improving the latency of 802.11 hand-offs using neighbor graphs," in *Proc. 2nd Int. Conf. Mobile Systems, Applications, and Services*, 2004, pp. 70–83.