

Research Article

Flexible Wi-Fi Communication among Mobile Robots in Indoor Industrial Environments

Jetmir Haxhibeqiri , Elnaz Alizadeh Jarchlo, Ingrid Moerman , and Jeroen Hoebeke 

Imec, IDLab, Department of Information Technology, Ghent University, iGent Tower, Technologiepark-Zwijnaarde, B-9052 Ghent, Belgium

Correspondence should be addressed to Jetmir Haxhibeqiri; jetmir.haxhibeqiri@ugent.be

Received 21 April 2017; Revised 10 November 2017; Accepted 21 November 2017; Published 16 April 2018

Academic Editor: Alessandro Bazzi

Copyright © 2018 Jetmir Haxhibeqiri et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In order to speed up industrial processes and to improve logistics, mobile robots are getting important in industry. In this paper, we propose a flexible and configurable architecture for the mobile node that is able to operate in different network topology scenarios. The proposed solution is able to operate in presence of network infrastructure, in ad hoc mode only, or to use both possibilities. In case of mixed architecture, mesh capabilities will enable coverage problem detection and overcoming. The solution is based on real requirements from an automated guided vehicle producer. First, we evaluate the overhead introduced by our solution. Since the mobile robot communication relies in broadcast traffic, the broadcast scalability in mesh network is evaluated too. Finally, through experiments on a wireless testbed for a variety of scenarios, we analyze the impact of roaming, mobility and traffic separation, and demonstrate the advantage of our approach in handling coverage problems.

1. Introduction

A never-ending process in industry is the automation process. With increasing automation, industries try to improve efficiency, reduce energy consumption, increase benefits, and working conditions for workers. The increasing number of mobile nodes, such as automated guided vehicles (AGVs) [1], and other field devices increased the need for new ways of communication inside the factory floor, apart from wired communication. Hereby, the wireless technology is seen as an important enabling technology in such cases. By adopting wireless technology, industries enable the device and personnel mobility, reduce cable costs, and connect hard-to-reach areas. This trend of connecting everything to a network inside of factory floor is referred as Industry 4.0 [2].

Mobile robots, such as AGVs, are included in different tasks, from delivering raw materials to production lines, moving materials during the production process, moving finished goods, and up to trailer loading. In order to increase the flexibility of mobile robot usage, recent AGVs are incorporating wireless solution to communicate with each other as well as with the controller.

In industrial environments, deploying wireless communication solutions is challenging. Different radio propagation effects such as reflections, multipaths, shielding, and so on will result in network coverage problems, packet losses, and communication outage for mobile robots. Since factory environments are large areas, multiaccess point (AP) systems will be used to cover the whole area. However, as mobile robots may move around with fast speeds (up to 2 m/s [3–5]), communication paths will change frequently due to handovers from one AP to the other or due to link breakage in case of ad hoc networks. In addition to this, communication between robots and the controller requires low latencies in order to have real-time robot coordination.

Within this challenging environment, robust and reliable wireless communication solutions must be realized. Different problems arise when robots rely only on infrastructure network, like increased latency during handover time, uncovered areas, and so on. In this paper, we will discuss such potential problems. We based our solution on real requirements from an AGV producer. This paper will consider a wide range of scenarios that can be solved with a flexible and modular solution. We propose a mixed

architecture for the mobile node that exploits the possibilities of multiple interface usage.

The main key contributions of the paper are

- (1) the full implementation of the mobile node architecture,
- (2) the evaluation of the architecture in a real testbed with real nodes,
- (3) a solution based on real requirements from producers of industrial AGVs.

This paper is an extended version of the paper published in [6], which provided an initial version of the mixed architecture and preliminary results. In this paper, we provide a more in depth description of the different architectural elements and discusses in detail newly added system functionality for dealing with coverage problems. This is further complemented with novel measurement results.

The outline of the paper is as follows. Section 2 further details the potential problems that arise when solely relying on the presence of a wireless infrastructure network and motivates our decision of adding meshing capabilities. Section 3 discusses related work in this domain, whereas Section 4 presents our resulting node and network architecture. In Section 5, we give a low-level description of the functionalities enabling our communication system architecture for the mobile nodes. In Section 6, we discuss our solution to handle the coverage problems by using mesh capabilities with the proposed mixed architecture. In Section 7, we illustrate our solution overhead in packet latency and throughput. Additionally, we evaluate the broadcast scalability for a group of nodes in mesh network. Further, we illustrate the potential performance issues in infrastructure networks through real-life experiments in a wireless testbed and show how our combined solution can deliver improved performance and flexibility. The achieved latency values of the proposed mixed architecture are benchmarked with the values from infrastructure-only based solution and ad hoc-only based solution. Also, the values are benchmarked with the list of requirements from Section 2. We end Section 7 with results of experiments to assess the behavior of our solution for tackling coverage problems. Finally, Section 8 concludes the work and provides an outlook to future work.

2. Problem Statement

Since wireless communication systems are widely used and deployed even in industrial environments, they can be used also to provide connection for AGVs in the network. However, the challenges that arise when using only infrastructure network in industrial environments are the increased latency during handover time and uncovered areas due to shielding effects.

Different scenarios need to be taken into account. The provided mobile node system architecture needs to function in absence of the infrastructure network. This situation can happen in two cases: where there is no infrastructure network or where it is not allowed to be used due to the interference with other processes that use it.

In scenarios where there is infrastructure in place, the mobile node system architecture needs to handle the possibilities of connectivity problems. Connectivity problems arise due to uncovered areas from APs. The mobile node needs to ensure permanent connectivity to the network and continuous outage time needs to be under certain time duration.

For a network consisting of a multitude of access points (APs), fast movement speeds of AGVs (0–2 m/s [5]) will result in frequent handovers. Such handovers greatly increase the communication latency. For the particular real-life use case we consider here, frequent time-critical broadcast exchanges between mobile robots are required for their distributed coordination, in addition to less time-critical but reliable unicast traffic to and from controllers. More specifically, the latency of broadcast packets has a strict upper bound (20 ms) in order to reach targeted mobile robots in time. The upper bound latency is calculated based on the path accuracy of the mobile robot. In [3], for 100 mm path accuracy, they ask for an overall latency of 50 ms, including also the processing latency. For 20 ms communication latency, the path accuracy will be 40 mm for highest speed of 2 m/s, without taking into account the processing latency. Every handover involves a series of packet exchanges, which consumes valuable time. Hence, frequent handovers may have a detrimental impact on the required performance, as we will show in Section 7. Moreover, robots are not allowed to travel more than two meters without communication. Considering the maximal speed of robots, the maximal continuous outage time should be lower than one second. As many small to medium enterprises, where these communication systems will be deployed, are searching for solutions without high operational costs, the use of unlicensed spectrum has been put forward as another requirement. Finally, as requirements to the mobile robot system may change over time, for example, when scaling up the network, it should support dynamic adaptation of the communication behavior.

The above observations and performance requirements lead to a challenging set of functional requirements for our mobile robot system, which we have summarized in Table 1 together with KPIs. All the summarized requirements are based on requirements from an AGV manufacturer company. Based on the above requirements, it is clear that we need to target a design that is capable of connecting either to existing enterprise networks (RQ2) to create its own mesh network (RQ1) or to do both (RQ3).

These requirements have led to a modular and configurable communication system for mobile robots, consisting of two wireless interfaces that can operate either in ad hoc or infrastructure mode and offering the possibility to control in a fine-grained way how traffic is being handled. As such, the system can support a variety of different networking architectures, potentially combining both infrastructure communication and mesh communication and supporting the separation or duplication of different traffic streams according to configuration settings. The design of the system and the supported network architectures are discussed in more detail in Section 4, whereas the advantages of our

TABLE 1: Requirements and KPIs.

Requirements	Description	KPI
RQ1	Function in the absence of fixed wireless infrastructure (network of APs)	
RQ2	Exploit the presence of available fixed infrastructure	
RQ3	Deal with occasional/sudden coverage holes in wireless infrastructure	Relay node role
RQ4	Reliably deliver unicast traffic	98%
RQ5	Timely deliver frequent broadcast traffic	<20 ms
RQ6	Deal with mobility	0–2 m/s
RQ7	Limit continuous outage times	<1 sec
RQ8	Function in unlicensed spectrum	2.4 or 5 GHz
RQ9	Adapt to future needs	Modular architecture

architecture for our particular use case at hand are experimentally evaluated in Section 7 (Subsections 7.4–7.7).

Security issues are outside the scope of this paper. There are plenty of possibilities to tackle security issues, such as dual authentication scheme [7], randomized authentication schemes [8], shared key encryption, and so on. Of course, depending on the complexity of the security method chosen, it may come with a performance penalty.

3. Related Work

Until now, there were several studies on mixed wireless system networks where ad hoc communication is supported by infrastructure. These studies mostly focused on the capacity improvement when an infrastructure network is used next to mesh capabilities [9–11]. More recent ones also include the delay performance and delay-throughput trade-off for such networks [12, 13]. In [9], they prove theoretically that hybrid wireless networks have greater throughput capacity and smaller average packet delay than pure ad hoc networks. In [10], they studied the effect of network dimensions on the capacity of hybrid networks. Apart from throughput, communication delays are important too in industrial applications. In [12], authors propose a multi-channel ad hoc network with infrastructure support that offers a lower average delay compared to ad hoc network or infrastructure only. In [13], authors propose an analytical framework to characterize the communication delay distribution of the network.

Recently, device-to-device communication is being discussed as part of the 5G research. Current research mainly focuses on peer discovery [14, 15], resource allocation [16, 17], and power control [18]. Even though some of the concepts of D2D communication in 5G networks can be applied to use cases currently considering network technologies such as IEEE 802.11, there is still work to be done with respect to dynamic switching between operational modes, IP-based routing in such mixed networks, and multihop relaying support. Apart from technical challenges, pricing is often another problem in case of device-to-device communication using cellular network support [19]. As we described in Section 2, many small to medium size enterprises are looking for communication solutions that do not imply operational costs, like payments for small cellular cell

installation and other network operator costs. As such, this paper considers IEEE 802.11-based infrastructure network with mesh capabilities of the mobile nodes by offering dynamic switching between ad hoc and infrastructure network. The network can operate in unlicensed spectrum, and cheap chipsets are available. Moreover, the whole mobile node architecture is based on open solutions (click modular router [20]) and works on top of COTS chipsets.

In a recently published patent [21], authors describe a solution to increase the number of mobile nodes served by an ad hoc network by introducing infrastructure support. As distributing the network control information through the ad hoc network is bandwidth costly, the authors propose to use the infrastructure network for signaling traffic. In our solution, we go one step further by offering the possibility to separate any traffic type between fixed and ad hoc network, for example, based on broadcast or unicast data traffic.

Apart from hybrid communication possibilities, communication systems of multiple mobile robots form an interesting research domain that is gaining importance in manufacturing in order to improve performance and increase automation. In [22–24], literature reviews regarding mobile robot systems, communication, and heterogeneous network are given together with open research issues and architecture used. In [22], authors highlight localization problems, coverage problems, robust communication needs, and environment hardships in manufacturing environments as important open research issues. In [23], a survey regarding the coordination in multirobot systems is presented, including the communication technologies. The authors highlight the importance of explicit communication, that is, direct message exchanges between robots, to ensure accuracy of the information. In our solution, we offer direct communication between robots through broadcast messages using the ad hoc network.

In [25], authors give a model for integrating three different areas, namely, wireless sensor networks (WSNs), mobile robotics, and teleoperation for use in different fields from medicine to military. Also, they give a survey of the literature regarding the research challenges, including routing and connectivity maintenance on ad hoc mobile robot networks. One of the solutions to maintain connectivity is to give robots a certain fixed role in order to route the traffic [25]. In our solution, we take a different approach,

designing the mobile node to take up multiple roles at the same time (e.g., can be end node and relay node).

During recent years, mobile robot communication experienced an evolution in their application as well as the protocols being used. Many works put forward ad hoc or mesh communication as a promising solution for realizing interrobot communication. For instance, [26] illustrates how an infrastructure network can be extended with multihop relaying functionality. In [27, 28], authors propose a model of a cooperative robot's system, where relaying robots (follower robots) will assist in establishing connectivity between the operating center and the robot that does the task (tank robot). Researchers in [29] construct an ad hoc network for communication between robots by classifying each robot as either a search robot or relay robot. This way they increase the area of communication (coverage zone). However, the robot roles are strictly defined based on their position in the network topology. We also consider multihop communication capabilities as one of the key requirements for our communication solution, but we also consider direct ad hoc or mesh communication between all mobile robots. Moreover, we think that there is no need to classify robot roles into relaying ones and main robots (like in [25, 27–29]) as their role can change depending on their position and the network topology, requiring support for multiple roles at the same time. In our solution, one of the goals of using the multihop functionalities is to extend the coverage zone of the APs of the infrastructure network. So far, most research into interrobot communication has focused on pure ad hoc networking. For instance, in [30], a review for routing protocols that can be used in robot networks is given. They show that the AODV routing protocol can be used in scenarios where robots have speeds up to 6 km/h, which is similar to the robot speeds that we consider in this paper. An architecture for mobile nodes using multiple interfaces is presented in [31]. However, they implemented their solution only in a network simulator, whereas our solution has been implemented on real nodes and has been tested in a testbed. Moreover, our proposed solution is capable of combining both mesh and infrastructure communication in a variety of ways and offering flexibility to distribute traffic.

In ad hoc networks, link management and neighbor discovery mechanisms are crucial in the performance of the routing protocol. In [32], link break detection is done within the routing protocol by using hello messages. We use the same approach by sending beacons in the ad hoc network to announce the presence of a node and to detect link breaks in the absence of beacons. In [33], an analytical model for the neighboring mechanism of OLSR is given while in [34], a hybrid asynchronous algorithm for neighbor discovery that leads to 24% shorter time for discovery is presented.

In industrial settings, it is also important to be able to meet the performance and latency requirements as we have indicated in Section 2. In [35], a routing algorithm for mesh networks is presented for use in industrial applications. They use a QoS manager which, after a calibration phase, manages QoS flows based on the requests from stations on specific QoS flow requirements, packet data unit (PDU) size, and destination. The calibration phase makes the solution more

difficult to be deployed in highly dynamic environments. Finally, [36] describes a solution for wireless mesh network infrastructure with extended mechanisms to foster QoS support for industrial applications. Like in [35], they propose a mesh network with a central admission unit to decide for the communication flows requested by different applications. In [36] with their solution they could offer streams with RTT less than 100 ms. The mechanisms are only applied to a mesh case, whereas we believe that a mixed solution such as the one we propose can offer additional benefits, especially when further extended with more advanced QoS mechanisms.

4. Communication System and Network Architecture

In the following subsections, we will describe the designed mobile robot communication system and potential network architectures that can be realized.

4.1. Mobile Robot Communication System Architecture. In Section 2, we motivated our decision to design a communication solution that makes use of two wireless communication interfaces. Each of these interfaces can either operate in ad hoc mode for establishing mesh communication or in infrastructure mode in order to connect to an existing enterprise network. From an application point of view, it should not matter which interface is being used for transmitting packets or how this interface has been configured. Similarly, external components, such as a controller, that want to communicate with a particular mobile robot, should also not be bothered with underlying communication details.

Figure 1 gives an overview of the high-level architecture we designed for the communication system of the mobile robot. We provide an abstraction layer that transparently manages and dynamically configures the underlying network interfaces. Application layer will communicate with single virtual interface using single IP address. This one is regardless of which physical interface will be used for actual communication. Additional logic for routing and traffic management is designed that is able to consider the specifics of the underlying physical interfaces. A basic implementation of the dynamic mobile ad hoc networks (MANETs) on demand (DYMO) routing protocol [37] is being used for unicast mesh routing. Regarding traffic management, the node design foresees a number of traffic classification components that can be dynamically configured. According to their configuration, unicast and broadcast traffic streams can be separated and directed to different interfaces or traffic can be even duplicated for redundancy purpose. Also, a neighbor discovery mechanism in mesh network based on beacon generation is designed.

To tackle coverage problems, a mechanism is needed to detect when a node goes outside of coverage zone of access points. Based on the coverage problem detection mechanism, the node will start looking for other communication possibilities within the network through mesh links it can establish with its neighbors.

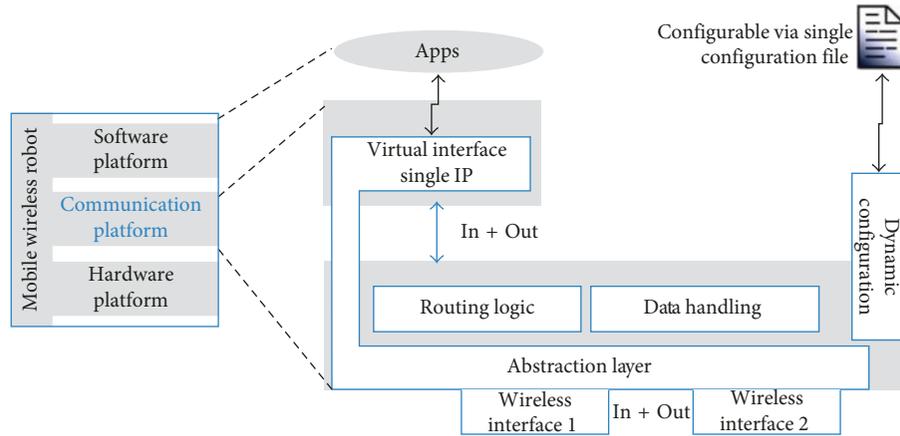


FIGURE 1: High-level mobile robot communication architecture.

4.2. *Network Architecture.* As motivated in Section 2, the system should be able to function in different use case scenarios. In this subsection, we discuss a number of potential network architectures that are derived from the real use cases in industrial environments which are shown in Figure 2. The proposed mobile robot system solution is able to support each of the network architectures by simply reconfiguring a set of parameters.

Many small to medium warehouses, where mobile robot systems are needed, are not willing to make additional investment in network infrastructure. In other factory environments where infrastructure already exists, it might be in use by other production processes. Figure 2(a) shows the first architecture that can be realized in such cases (RQ1). Both wireless interfaces can then operate in ad hoc mode, forming a mesh network with parallel links that operate on different frequencies increasing the capacity. In case wireless infrastructure is present and can be used, a mixed network can be established as shown in Figure 2(b) (RQ2). One of the interfaces is used to connect to the existing network, whereas the other interface is used to form a mesh network. Depending on additional configuration settings, it can be further decided how traffic is distributed over the different interfaces by exploiting different techniques for traffic separation. This is shown in Figure 2(c) for the case of a multimesh configuration, where one of the interfaces is used for unicast traffic and the other interface is used for broadcast traffic. Finally, Figure 2(d) shows how the communication can be configured in order to tackle coverage problems by making use of mesh functionality in the specific area that experiences these coverage problems (RQ3).

5. Low-Level Description

The modular Click Router framework [20, 38] was used for the communication system. Click Router is software architecture for building flexible and configurable routers. A router in Click is created by a chain of the packet processing modules. Each individual module implements simple router functions that chained with all other needed modules providing the router’s functionalities. Different basic

functions are already implemented by Click elements such as packet queuing, scheduling, interfacing with network interfaces, and so on.

As one of the requirements (RQ8) was that the system should be adaptable to future needs, Click Router is a good choice. It has a modular structure, and it is easy to support future extensions or the replacement of existing elements with more advanced element versions. We have extended the Click framework with additional features for event handling, flexible configuration, and dynamic interface management in order to fulfill the requirements for our mobile robot communication system. Elements can subscribe for events of interest and will take specific actions when the event is announced by any other element. This facilitates information sharing between Click elements that are not neighbors in the Click chain.

Further, all configurable parameters are specified in a text file. Such parameters include network interface roles, interface classifiers, timing parameters for the neighbor discovery, link breaking and mesh routing, IP address for virtual interface, and so on. This way the system can be configured dynamically, enabling administrators to define the system behavior in a single configuration file without changing the Click chain. Finally, the dynamic interface management makes the Click chain independent of the number of used interfaces.

Figure 3 gives an overview of the different functionalities and building blocks that implement them, as well their interactions. In the following subsections, we will give a short description of the main functionalities.

5.1. *Convergence Layer and Interface Management.* Each mobile robot will have one unique IP address, regardless of the fact that there are two network interfaces. A virtual interface with single IP on top of multiple physical interfaces is created. It enables the application layer to communicate through only one interface irrespective of which underlying physical interface is used for communication. The actual selection of which physical interface should be used can be done based on the interface role and/or traffic type (Section 5.2). The interface role determines whether the interface is

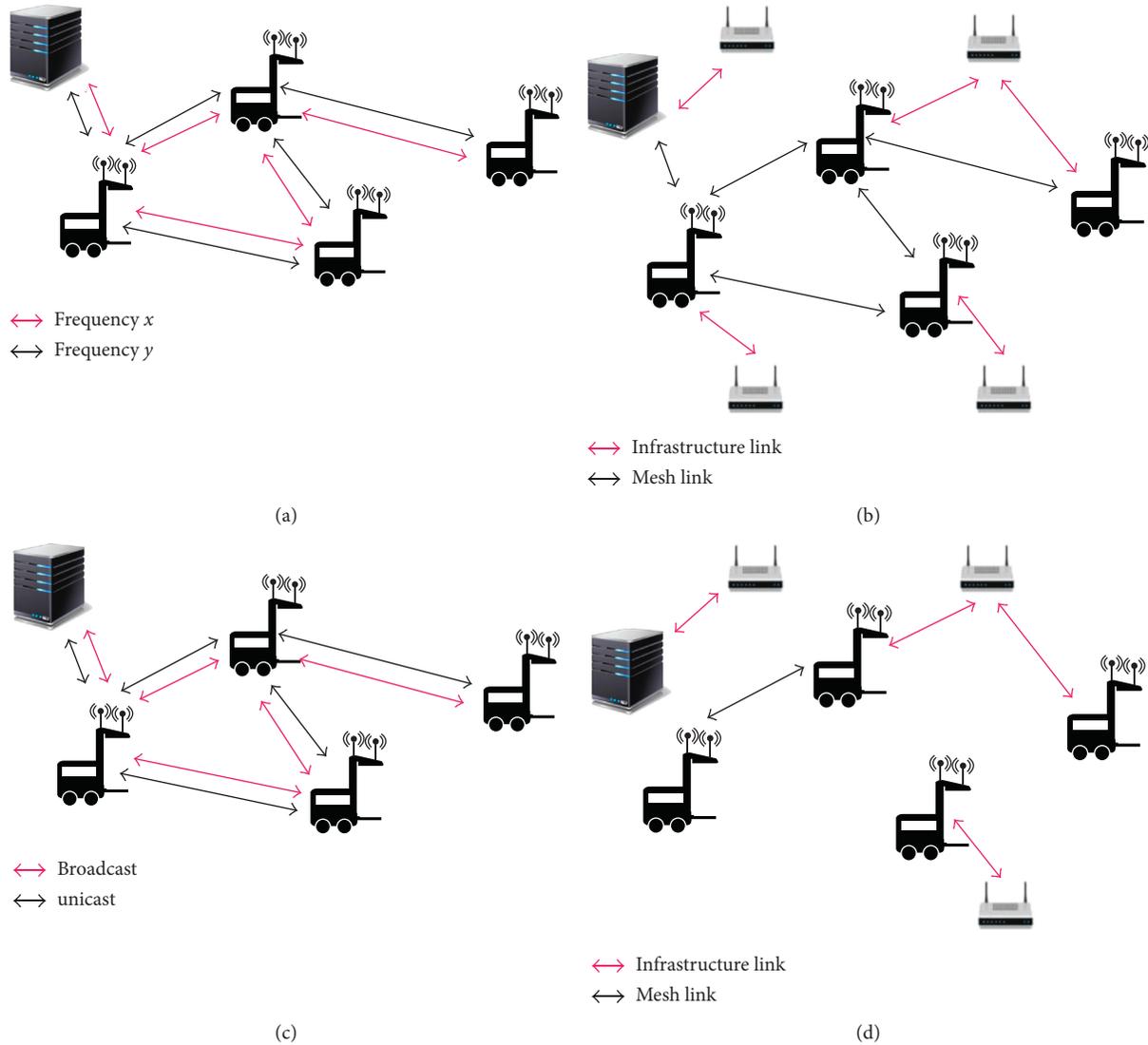


FIGURE 2: Potential network architectures that can be realized by reconfiguring the designed mobile robot communication system. (a) Multiinterface mesh network in the absence of fixed infrastructure or in case fixed infrastructure cannot be used. (b) Mixed mesh (single interface) and infrastructure network. (c) Multiinterface mesh with traffic separation. (d) Infrastructure network with mesh support for handling coverage problems.

used in ad hoc mode or connects to an access point. The traffic type specifies for which traffic (unicast, broadcast, or both) the interface can be used. Once a decision is made on which interface will be used for sending the packet, the packet is tagged with the MAC address of the physical interface. The convergence inspects the tag and takes care of the actual transmission.

Interfaces can be added or removed dynamically, and their role can be changed dynamically. An interface manager monitors the changes on interfaces such as connectivity, whether the interface is up and running, and so on.

5.2. Traffic Classifiers. Different levels of traffic classifications are foreseen in order to properly route the packets through the Click chain. Firstly, packets are classified based on their type: IP or ARP packets. ARP packets will be handled by Click

functions for reply/request of ARP packets. IP packets are further classified into routing control packets (i.e., DYM0 in our case) and other data packets. Routing control packets are handed over to the Click element that implements routing protocol logic. The other packets go through a next level of classification, namely, into broadcast and unicast traffic. Since there is a beaconing system for neighbor discovery, each neighbor beacon packets are filtered out from the broadcast packets and given to neighbor discovery functions. On the contrary, the remaining broadcast traffic and unicast traffic will be delivered to the virtual interface, routed based on the main routing table (unicast) or tagged according to the interface configuration for traffic separation.

5.3. Neighbor Discovery and Link Break Detection. Neighbor detection in the mesh network is done based on keep-alive

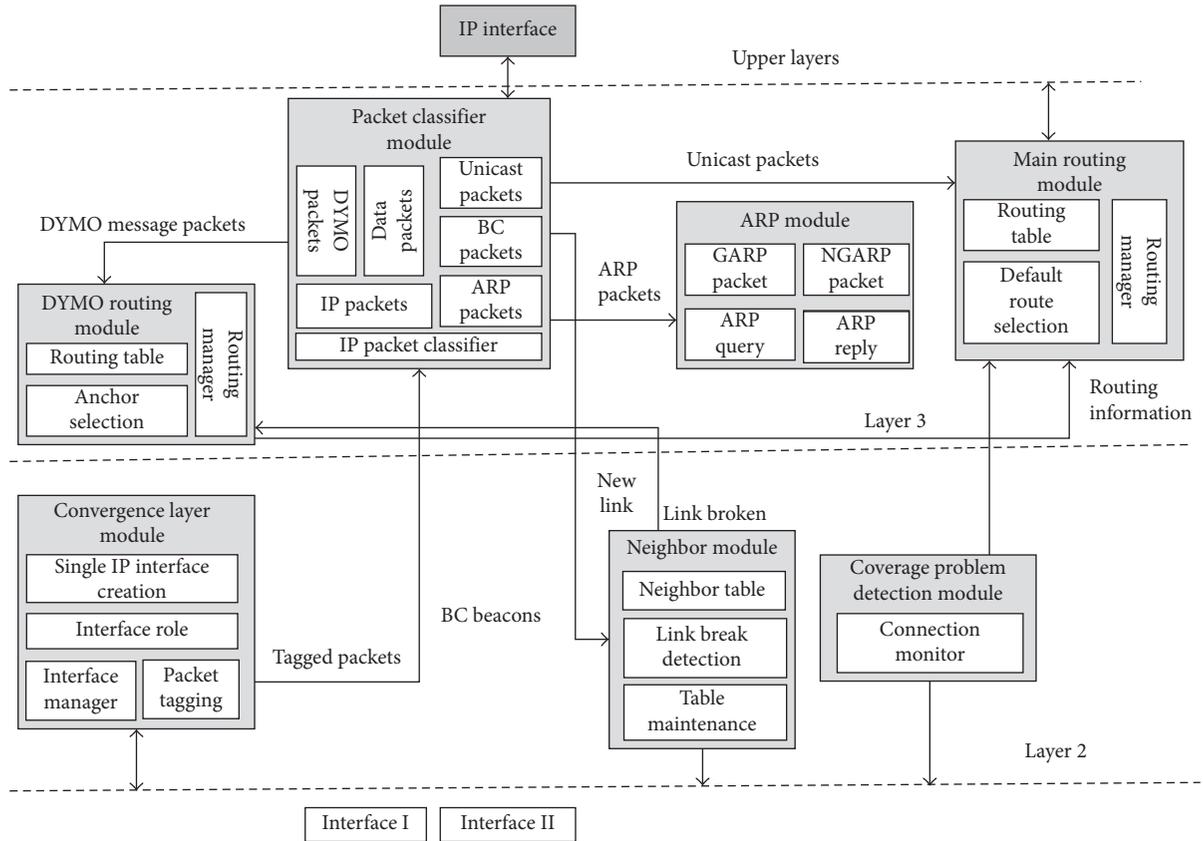


FIGURE 3: Detailed node architecture of the mobile robot communication system, implemented in Click Router.

beacons. Each mobile robot will broadcast beacons in the mesh network every N_{ms} seconds. All other mobile nodes will update their neighbor’s table based on the reception of these beacons. For each entry in the neighbor’s table, a timer is kept. As soon as the timeout is passed and no more beacons are heard from that peer, the entry will be removed from the table. Parameters such as keep-alive beacons interval and neighbor’s table entry timeout are dynamically configurable.

Given the fact that in our use case, the broadcast traffic is used frequently to communicate between mobile robots, real broadcast traffic can be used in addition to beacons for neighbor discovery. Hereby, we enable the suppression of the real beaconing traffic in order to reduce the network load.

The neighbor discovery module will maintain the neighbor table and will use the eventing mechanism to inform routing module about neighbor changes.

5.4. Routing. Each mobile robot can make use of two interfaces: for example, one to connect to an access point and another one to establish a mesh network. In this case, routing is done in two different networks implying the need to have a routing table that incorporates routes from both networks.

For routing in the mesh network, the DYMO routing protocol is used. A DYMO routing table for unicast routing

in the mesh network is maintained. DYMO message packets (Route Request (RREQ), Route Reply (RREP), and Route Error (RERR)) are generated to find and maintain needed routes.

The main routing table will contain the default route towards the access point as well as other routes in the mesh network. Hereby, the main routing table is subordinated to the DYMO routing table, meaning that it will be updated based on the DYMO routing table updates.

Contrary to unicast routing, broadcast routing does not require a routing table. The only thing needed for broadcast packets is the tag that defines over which interface the broadcast packet has to be transmitted. The assignment of this tag is again based on the interface configuration for traffic separation.

5.5. Coverage Problems Detection. The first step to handle the coverage problems is to detect when they happen. To detect the disconnection from an access point, we use the `wpa_supplicant` [39] control interface. The connection status is probed periodically, and the specific actions are taken when the node is detected to have entered the uncovered zone. After the disconnection detection, the mobile robot will only use the ad hoc interface for communication. Changing from two interfaces to one interface has impact on the address resolution protocol (ARP), which will be handled by issuing Gratuitous ARP (GARP) packets and Non-

Gratuitous ARP (GARP) packets. The AP monitoring interval parameter can be configured directly in the configuration file. For further understanding, our solution for coverage problem handling we refer the reader to Section 6.

5.6. Address Resolution Protocol (ARP) Module. ARP module handles the ARP requests and response from the network. It contains ARP responder table where it saves all the MAC addresses for which it can reply. At the same time, it maintains the ARP reply table by the replies it receives. During the anchor selection process, it issues GARP and NGARP packets.

6. Handling Coverage Problems

As we discussed in Section 4, one of the network topologies that can be supported by our solution is an infrastructure network with mesh support to overcome coverage problems. This section will discuss in detail how we exploit the mesh capabilities in order to tackle those coverage problems.

In order to minimize the problems of “dead zones” where robots will become unreachable, other mobile robots, which do have connectivity to an AP, are used as AP coverage extenders. Each node which has a direct link to an AP and helps other nodes to communicate with the rest of network will be further referred to as an anchor node. Each node that does not have a direct AP connection, but has a connection through an anchor, will be further referred to as anchored node. Each node that does not have a direct AP connection, but has a multihop connection to an anchor, will be referred to as a multihop anchored node. Assuming that the traffic separation configuration mandates unicast traffic to be sent over the infrastructure network, a mechanism that allows the unicast traffic to be transmitted through mesh links to the anchor node has to be incorporated.

The proposed solution considers the problem of link breakage and link breakage healing in mesh networks, as well as the detection of the coverage problems from APs. As soon as one mobile robot detects the disconnection from the infrastructure network, it starts looking for another mobile robot which is able to act as an anchor for the unicast traffic. For this, the anchor node will use its second network interface, whereas its first interface is connected to an AP. In addition, the anchor node should be able to inform the network that it has been selected as anchor for certain mobile nodes. Thereby, it should be able to reroute all unicast traffic destined for the anchored mobile nodes as well as to reply on incoming ARP requests on behalf of anchored nodes, as the other nodes have to use the MAC address of the anchor interface (i.e., interface of the anchor node that is connected to the infrastructure network) to reach the anchored node. Further, existing ARP entries for a newly anchored node must be updated based on which interface is being used by the mobile node.

Finally, also mechanisms to detect unidirectional links between anchor and anchored mobile nodes and vice versa are provided. The combined mechanisms enable multihop routing between anchored and anchor mobile nodes and make sure that the shortest route (in terms of hops) in the mesh network is always used for unicast traffic. The

anchorage process is triggered by the node which is in the uncovered zone and maintained by it.

6.1. Anchor Selection Process. The two Click functional modules that are responsible for facilitating the anchor selection process are the Coverage Problem Detection Module and Neighbor Module.

The Coverage Problem Detection module uses the `wpacli` [39] library functions to probe periodically for the AP connectivity. When the connection towards the AP becomes inactive, a disconnection event will be raised that will trigger the anchor selection process in the DYMO routing module. Simultaneously, the same event will clear the default route towards the AP in the main routing table. The routing module will wait for the anchor selection process to be finished by DYMO routing in order to choose one of the paths towards its neighbor as default route. Conversely, when the node is already anchored the Coverage Problem Detection module will monitor whether the node re-enters a covered zone. In that case it will issue a connection event, which will trigger a process in DYMO routing to release the anchor and to inform the network that the node is directly connected through an AP.

The neighbor module will facilitate the welfare of the communication between the anchor and anchored node during the time of anchorage. Beacon mechanism is used for recognizing neighbors. In case neighbor beacons are not being received for a certain period, called the link breakage timeout, the node will remove the link from its neighbor list. If the link between the anchor and anchored node is broken, then the anchored node will initiate again the anchor selection process.

Two packets that are used to inform the network that a node is anchored or released are the GARP reply and the NGARP reply [40, 41]. Both of these packets are broadcasted by the anchor through the ARP module. GARP and NGARP replies are ARP reply packets that are sent without any request. Both the targeted MAC and IP addresses are set to broadcast addresses. In our case, the sender IP will be the IP of the anchored node while the sender MAC will be the MAC of the interface that provides connectivity to an AP. In case of the GARP, the ARP entry will be updated, while in case of the NGARP, the ARP module will clear the ARP entry match from the table. This way the ARP tables can be updated across the entire network for nodes that are being or no longer being anchored. If there is no entry in the ARP table for a specific destination and no reply is being received upon ARP requests, then the sender is enforced to buffer all the outgoing traffic.

For routing in this mixed network, we use the DYMO routing protocol with an extension for finding a default route through the mesh network in case the mobile node enters an uncovered zone. These mesh routes are stored in the DYMO routing table and trigger updates in the main routing table. As such, when in a covered zone, the main routing will contain a default route entry through an AP, whereas in an uncovered zone, the main routing table will contain a default route through the mesh network.

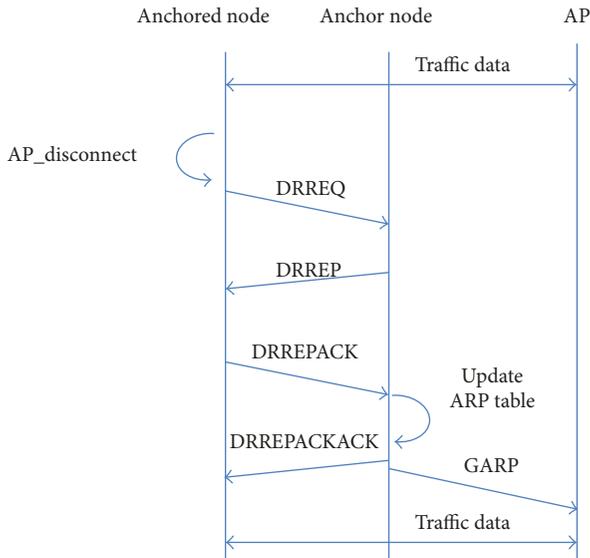


FIGURE 4: Time diagram of packets for anchor selection process.

The time flow of the packets between an anchor and an anchored node during the anchorage process is given in Figure 4. When the AP disconnection event is raised, the mobile node will broadcast a Default Route Request (DRREQ) packet for anchor discovery. We use a four-way handshake to select the anchor node, avoiding the possibility of selecting two anchors at the same time and creating confusion in the network. After issuing the DRREQ, other nodes that receive the DRREQ and that have a direct default route towards an AP in their main routing table will send back a Default Route Reply (DRREP) packet. The mobile node might receive multiple DRREPs from different nodes but will choose the first received DRREP (which is typically the one with the lowest number of hops) and ignore the others. In reply to the DRREP, the mobile node will send a unicast Default Route Reply Acknowledgement (DRREPACK) packet to the node that issued the DRREP. Upon reception of the DRREPACK, the anchor node will issue a GARP in the infrastructure network. This way the rest of network will be informed that the IP of anchored node is now reachable through the anchor, enabling the anchor to intercept all packets destined for the anchored node. At the same time, the anchor updates the ARP response table in the ARP module with the IP address of the anchored node to be able to respond to future ARP requests from the network for the anchored node IP. Finally, the anchor will inform the anchored node with the Default Route Reply Acknowledgement of Acknowledgement (DRREPACKACK) packet. After receiving the DRREPACKACK from the anchor node, bidirectional communication is possible between the anchored node and any other node in the network.

In order to ensure that the anchor is selected correctly and the network is informed on time, there are timers for each of the packets issued by the mobile nodes during the anchorage process. If any of the packets is lost, the timer expires and the mobile node has to restart the anchorage process.

6.2. Handling Unidirectional Link Problem between Anchor and Anchored Node. The link between the anchored node and its anchor should always be bidirectional as it will be used for bidirectional traffic. However, the beaconing process to detect neighboring nodes does not ensure that links are bidirectional. It might happen that the anchored node is hearing beacons from the anchor but not the other way around, resulting in a unidirectional link. In order to eliminate such unidirectional links, Route Error (RERR) packets in the mesh network are used to inform about the existence of the unidirectional links.

The time flow of the packets for the case of a unidirectional link from the anchor to the anchored node is given in Figure 5(a). In this case, the link break event is raised at the anchor because of the absence of beacon packets from the anchored node. The anchor will broadcast a RERR with default IP address 0.0.0.0 in its payload, indicating that the anchored node should no longer use the default route through the anchored node. At the same time, it will inform the network with an NGARP broadcast packet that the anchored node is no longer reachable on its MAC address. The IP address of the anchored node will be removed from the ARP responder table in the ARP module in the anchor as well. As the link in the direction from the anchor to the anchored node is active, the anchored node will receive the RERR packet. It will remove its default route towards the anchor from its main routing table, which in turn will raise the “remove default route” event. This event will trigger the start of a new anchor selection process in the DYMO routing module.

The time flow of the packets for the case of a unidirectional link from the anchored node to the anchor is given in Figure 5(b). In this case, the link break event is raised at the anchored node. The only difference from previous case is that the RERR packet is generated anchored node.

It is obvious that in case when the link is broken in both directions, the link break is detected at both sides, and the anchorage data will be cleared at both sides.

This way we ensure that at any time, the link between the anchored and the anchor node is bidirectional. Moreover, the use of the NGARP packet will ensure that any time the anchor loses its connection towards the anchored node (either as a result of the presence of a unidirectional link or of a total link break), the network is informed and should wait until the anchored node is reanchored again on another node. This will minimize the number of lost packets due to the wrong or outdated ARP and routing table information in the network.

6.3. Multihop Anchor Selection Process. If the node density is high, then most probably the anchored node will find its way towards an AP in just one hop. However, this is not necessarily true in case the node density is relatively low or the uncovered regions are large. In those cases, a multihop connection and packet forwarding mechanism is needed between the anchored node and the anchor. On top of the normal forwarding procedure already offered by the default DYMO routing protocol, each node needs to take certain actions when forwarding special packets for handling the

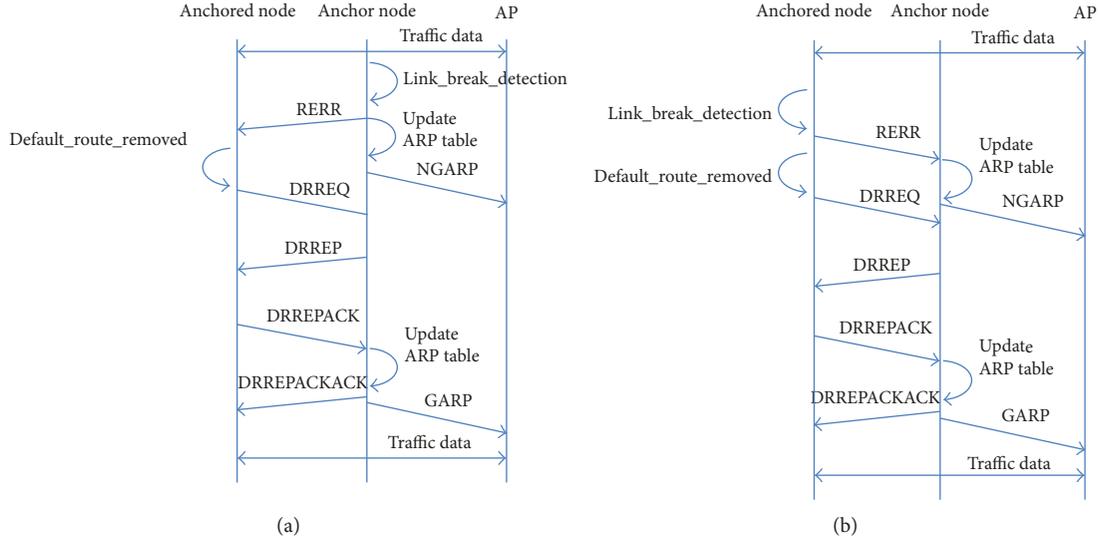


FIGURE 5: Unidirectional link detection. (a) From anchor to anchored node. (b) From anchored node to anchor.

network coverage problems like DRREQ, DRREP, DRREPACK, or DRREPACKACK. The time flow of the packets in case of a multihop anchor selection process is given in Figure 6.

When a node forwards the DRREPACK, it has to update its ARP response table with the source IP address of the DRREPACK packet and the MAC of its AP interface. This is due to the fact that in case the forwarding node reconnects to an AP, it has to directly notify the network about the shorter path to the multihop anchored node by sending a GARP containing the IP of that node. In addition, the forwarding node will broadcast a RERR packet with the IP address of the multihop anchored node in its payload on the mesh network. When the old anchor receives such a RERR packet, it will remove the multihop anchored node from its anchored node table.

Every time the mesh link between the anchored node and the anchor expires due to the absence of traffic, the anchorage process has to start all over. Otherwise, if traffic is going on, the link will not expire as the traffic is used for updating the mesh link expiration timer.

7. Performance Analysis

It is clear that the proposed communication system enables several networking topologies. Combined with the flexibility on how to distribute the traffic, it is interesting to investigate how this flexibility can be exploited in order to deal with the other requirements that are specific for our targeted use case (RQ4-7). For this, we conducted a set of experiments on the w-iLab.t wireless testbed [42], which are now discussed in the following subsections. We tested the Click packet processing overhead, broadcast scalability, and the different network topologies, which we already showed in Figure 2.

7.1. Tools for Running Experiments. Hostapd [43] and wpa_supplicant [39] are used as user space daemon to realize

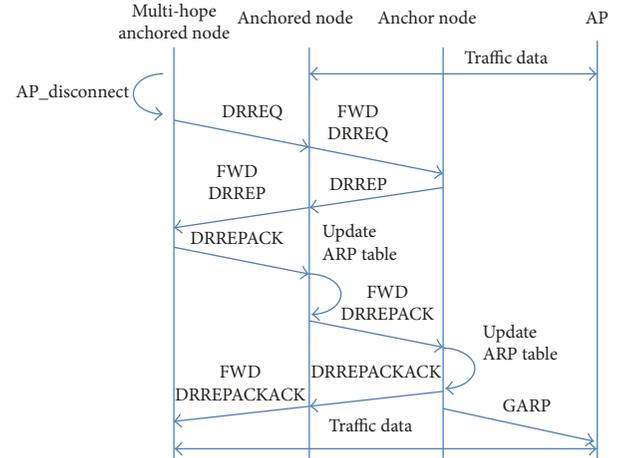


FIGURE 6: Multihop anchor selection process.

access point and client functionality, respectively. The mobile robots consist of embedded PC nodes which are running Linux and our Click Router implementation presented in Section 5. The access points are static embedded PC nodes running Linux. The Wi-Fi cards of all devices have Atheros AR93 chips.

To perform proper diagnostics and performance analysis, we have created on each node a database with three tables: an EVENT table, TOPOLOGY table, and ROUTES table. The EVENT table captures all internal Click Router events. The TOPOLOGY table contains link information, consisting of the time, neighbor MAC address, and event type (“add new link” or “link is broken”). This enables us to derive at each point in time the topology of the mesh network. Finally, the ROUTES table collects all routing information, including time, destination IP, and next hop IP.

7.2. Click Packet Processing Overhead. As already mentioned, all networking modules have been developed in Click

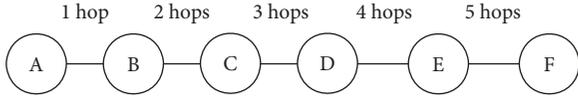


FIGURE 7: Setup to assess the impact of Click packet processing overhead.

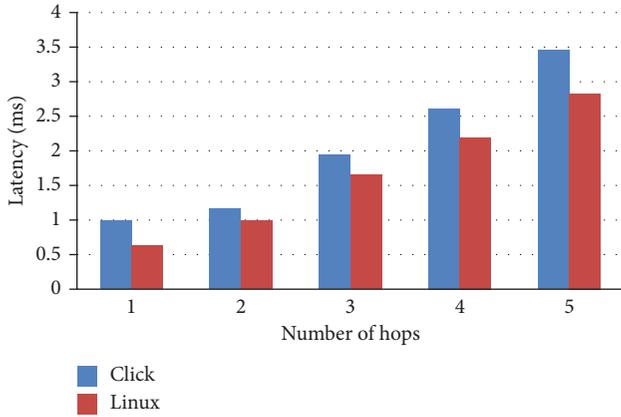


FIGURE 8: Latency versus numbers of hops.

Router, running in user level. Consequently, Click Router packet processing introduces additional overhead compared to kernel-level packet processing. Therefore, it is important to assess the introduced performance penalty. To this end, we measure the latency and UDP throughput for both cases using the setup shown in Figure 7. We use wired connections in order to isolate Click packet processing and avoid performance losses introduced by wireless communication links.

The keep-alive beacon interval is set to 50 ms and packets are sent every 100 ms. We measure the latency for different hops, running every experiment for 50 seconds, and averaging the latency over the measurement time. In Figure 8, the latency for the different number of hops is given. Due to the packet processing in Click Router, the packet latency is about 20% higher compared to normal Linux stack processing, but sufficiently low to be able to meet our latency requirements. The only exception is for one-hop connection, where the latency is approximately 48% higher.

Regarding the UDP throughput, we did measurements with data rates up to 100 Mbps, not noticing any losses. The performance started to decrease when the data rate became higher than 500 Mbps, observing packet losses from 1% up to 40% for the highest sending rate of 1 Gbps over a one-hop direct link. So considering the typical data rates of wireless cards, it is clear that the packet processing overhead in Click is not a bottleneck as it can handle rates up to 500 Mbps without losses.

7.3. Broadcast Traffic Scalability. Our use case heavily relies on broadcast communication between mobile robots. One of the interesting things to analyze is the broadcast scalability, for example, how frequently broadcast data packets can be transmitted within a group of nearby mobile nodes without

having high packet losses. By quantifying the packet loss ratio of directly broadcasted and rebroadcasted packets, we can assess which percentage of traffic was received through direct links and which was received through a multihop path.

We use two different packets with payload size of 50 and 20 bytes, respectively, and two different group sizes, with 8 and 12 nodes, respectively. From each node, we send 10000 broadcast packets at the same frequency.

If the number of nodes in the group is N and the broadcast frequency is f_{BC} , then we will have $N * f_{BC}$ generated and $(N-1) * f_{BC}$ received directly broadcasted packets per time, while we will have $N * (N-1) * f_{BC}$ generated and $N * (N-1) * 2 * f_{BC}$ received rebroadcasted packets per time. At each node, we classify the received packets based on source IP and source MAC address enabling us to calculate packet losses of directly broadcasted packets and rebroadcasted packets on each node.

In Figure 9(a), average packet losses are given for the case with ping packets having a payload of 50 bytes and for two different group sizes, 12 nodes and 8 nodes, respectively. It can be seen that by increasing the broadcast frequency, the directly broadcasted packet losses increase drastically: from 1% for 3.33 Hz and group of 8 nodes to 48% for 33.33 Hz. For the group with 12 nodes, these values are even higher, starting at 9% for 3.33 Hz up to 66% for a frequency of 33.33 Hz. The rebroadcasted packets exhibit even higher losses because they are more prone to collisions as they will be retransmitted by all nodes nearly at the same time. However, all the nodes received all the packets at least once with just small losses ($\sim 2\%$) in case of a frequency of 33.33 Hz. From this, we can conclude that by increasing the broadcast frequency and number of nodes that are in the range of each other, packets will still be able to go through but will more likely take a path with more than one hop. For example, in case of a frequency of 20 Hz and packet payload of 50 bytes, 57% of packets arrived through a path with more than one hop in case of group with 12 nodes, while in case of a group with 8 nodes, this was 40%.

Also the packet size has an impact on the packet losses. In Figure 9(b), the average packet loss for ping packets with a payload size of 20 bytes is shown. It can be noticed that all values are lower than those in the previous case. In our use case, the packets payload will be low since the mobile nodes will just transmit their positions and other data that are related to their battery life.

7.4. Wireless Infrastructure Network Only. In this scenario, we assume the presence of fixed access points and do not make use of any meshing capabilities. Every mobile robot is connected to an access point, and selection of the most suitable access point is based on signal strength. Mobile robots move around in the environment covered by access points and get attached and detached to/from access points. As mobile robots can drive at relatively high speeds, such handovers may take place frequently and will affect the communication performance. To quantify this effect on the performance of unicast and broadcast traffic, we set up an

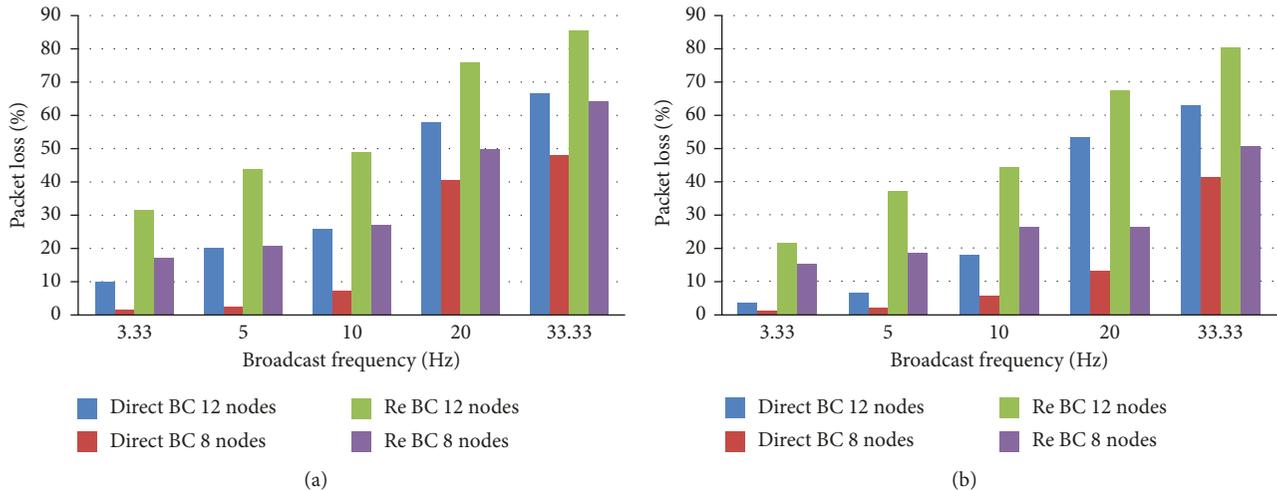


FIGURE 9: Average packet losses for broadcast scalability tests. (a) Packet payload size 50 bytes. (b) Packet payload size 20 bytes.

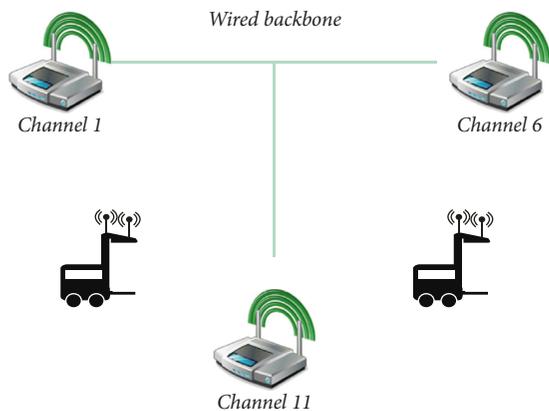


FIGURE 10: Experimental setup to assess the impact of handovers on the communication performance.

experiment in the w-iLab.t testbed [42] as shown in Figure 10. Three APs and two mobile robots are used.

Three nonoverlapping channels (1, 6, and 11) in 2.4 GHz frequency band have been used. To trigger handovers of mobile clients between APs in a small area (limited by the physical space of the testbed), the transmit powers of the APs are configured during the experiment. The mobile robots are limited to scan only over the mentioned channels to prevent time and energy consuming procedure for scanning all available channels. During the experiment, both mobile robots are communicating with each other through the infrastructure wireless network. Figure 11 shows the latency distribution of 10000 unicast packets during a measurement period of 200 seconds. Unicast packets are exchanged every 20 ms and the roaming among access points is configured to be once every 10, 20, and 30 seconds. As can be seen, in most cases, the latency is lower than 4 ms, which is close to the average amount. However, it can become as high as 78 ms during the roaming procedure. Further, the more frequently roaming happens among the access points, the higher the packet latency can become. The reason behind this is that every time a client performs a handover between access

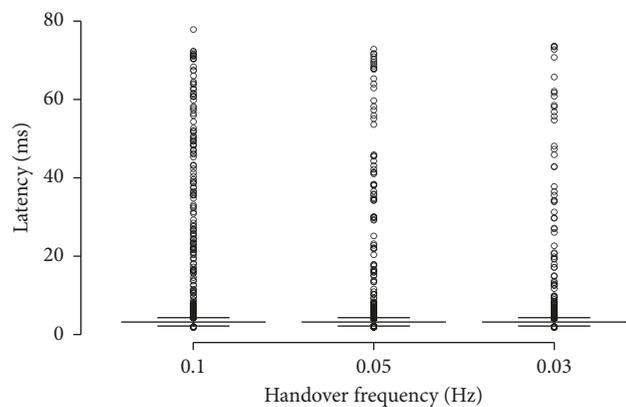


FIGURE 11: Latency of unicast traffic for different roaming frequencies.

points, it gets dissociated, has to look for stronger signal strength, and needs to associate to a new access point. Table 2 shows the latency statistics, presenting the first and third quartile of the results shown in Figure 11.

Figure 12 presents the latency of 10000 broadcast packet transmissions within the same 200 seconds time period. Again, the roaming procedure happens every 10, 20, and 30 seconds. As shown in Table 3, in contrast to the unicast latency, the broadcast latency is now not around the average value but around the third quartile value. The results also show a much more profound negative impact of handovers on the broadcast latencies, due to the way broadcasts are disseminated through the network. Every broadcast from a mobile robot needs to be rebroadcasted to other devices connected to the same access point as well as to all other devices connected to the other access points. This is visible in Figure 13 where every time the mobile robots were connected to the same access point, the latency was around 5 ms while when roaming took place the latency increased up to 100 ms. It is clear that even in this simple setup, our mobile robot solution will never be able to meet the envisioned latency requirements (<20 ms) of broadcast traffic.

TABLE 2: Unicast latency statistics in ms.

Statistics	10 s roaming frequency	20 s roaming frequency	30 s roaming frequency
Average	3.67	3.57	3.52
1st quartile	3.21	3.18	3.13
Minimum	2.13	2.12	2.09
Maximum	78	73	73.8
3rd quartile	3.75	3.66	3.66

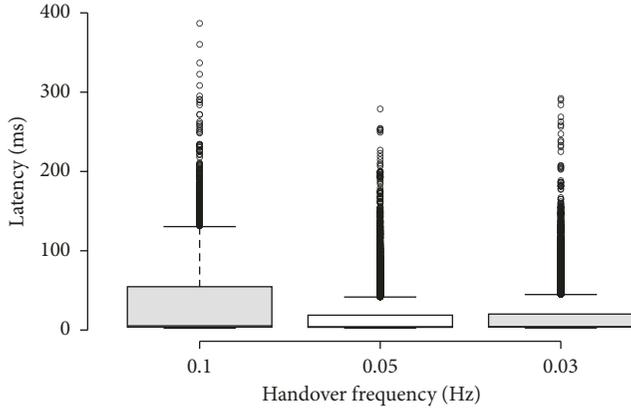


FIGURE 12: Latency of broadcast traffic for different roaming frequencies.

7.5. *Mesh Network Only.* In this scenario, only a mesh network is being used as shown in Figure 2(a). As mentioned, unicast traffic uses a simple reactive routing protocol, whereas broadcast traffic uses blind flooding with duplicate detection. Using this setup, we again measure the impact of mobility of mobile robots on the latency of packet transmissions. In order to be able to mimic a variety of speeds and thus link breaks, we used a forced mobility approach, where MAC filtering is being used to artificially change the mesh topology as shown in Figure 14. While nodes c1 and c5 are communicating, c1 establishes a new link with nodes c2, c3, c4, and c5, respectively, breaking the old link and gradually changing the number of hops over which the packets need to travel.

Figure 15 presents the impact of link breaks and the resulting change in topology and hop count on unicast and broadcast packet transmissions with transmissions being generated every second. In this experiment, latency for unicast and broadcast traffic varies between 17.2 ms and 2.62 ms and 19.9 ms and 3.04 ms, respectively. It is also visible that the latency decreases with the hop count between the sender and the receiver.

In the scenario shown in Figure 15, the beacon interval was set to a very small value (20 ms), making it possible to very quickly react to link breaks in this small topology. In addition, with traffic only being generated every second, no significant unicast packet losses occurred, illustrating only the impact of hop count on latency in a mesh setting. In other settings, the performance of unicast traffic, however, is also strongly affected by the link break detection and routing mechanism.

In reality, the protocol might react slower, traffic generation can happen more frequently or the topology is more

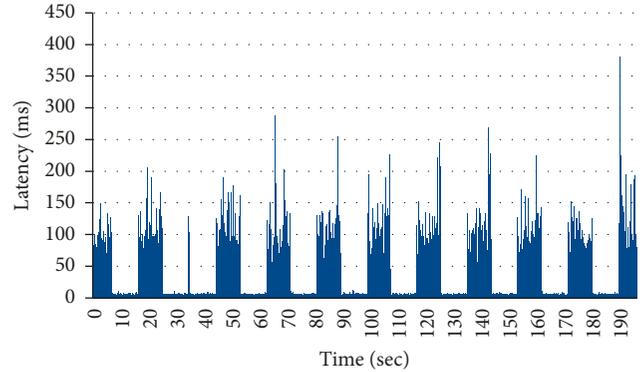


FIGURE 13: Latency of broadcast traffic for a roaming frequency of 0.1 Hz.

complex. These first two aspects are shown in Figure 16, where unicast traffic is being generated every 120 ms. Keep-alive beacons are sent less frequently, that is, every 500 ms, with the detection of a link break in the absence of beacons after 2500 ms. Further, upon the detection of a link break, all traffic for a destination that has become unreachable is being buffered until the route has been established. This has two consequences. First of all, unicast traffic in the presence of link breaks in the mesh network exhibits much higher packet losses than in an infrastructure network, with the amount of lost packets directly related to the efficiency of the underlying link break detection mechanism as shown in Figure 16. Secondly, route recovery takes some time, resulting in higher latencies of the packets that were buffered between the detection of the link break and the moment the route has been recovered. Broadcast traffic does not experience these drawbacks as it can make use of any available link and does not depend on route establishment.

7.6. *Combined Network.* The third scenario being considered is a mixed setup, where every mobile robot uses one interface to connect to the infrastructure network and one interface to set up a mesh network as shown in Figure 2(b). In order not to overload the wired network with broadcast traffic, the communication system is configured to send broadcast traffic over the mesh interfaces. To avoid frequent routing inside the mesh network, unicast traffic is configured to run over the other wireless interface. Again, we measure the latency of unicast and broadcast traffic in order to investigate the advantages and feasibility of a hybrid configuration with traffic separation. In this scenario, we use three interconnected access points (as in Figure 10) and four mobile robots. Two of them are communicating using unicast traffic

TABLE 3: Broadcast latency statistics in ms.

Statistics	10 s roaming frequency	20 s roaming frequency	30 s roaming frequency
Average	6.1	5.02	4.99
1st quartile	4.44	4.18	4.46
Minimum	3.06	3.17	3.26
Maximum	381	275	288
3rd quartile	54.5	19.2	20.6

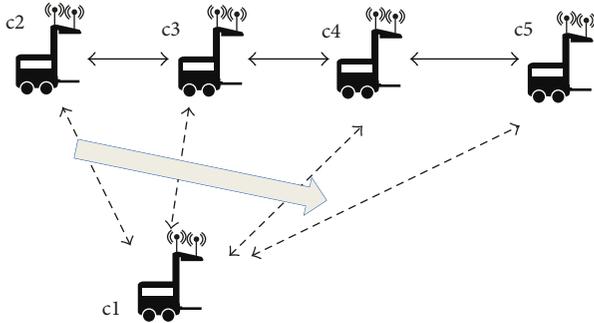


FIGURE 14: Fully mesh network among mobile robots.

through access points while two others are generating broadcast traffic. All of them are connected to one of the APs. One mobile robot is configured to reply to the broadcast packets. Channel 6 is used for communication within the mesh network while channel 1 is used for communication with APs. The handover and link break frequency in this case are both 0.1 Hz.

Figure 17(a) shows the latency of 10000 unicast transmissions during 200 seconds, whereas Figure 17(b) shows the latency of 10000 simultaneous broadcast transmissions. As it is shown in Table 4, the mixed scenario that exploits the possibility to separate different traffic streams combines the best of both worlds. Broadcast traffic can meet the strict latency requirements by using the mesh network, whereas unicast traffic achieves low latency by avoiding the complexity of ad hoc routing. Compared to Table 2, it can be seen that the maximal values for unicast traffic latencies have now dropped from ~ 70 to ~ 11 ms. The same conclusion can be drawn for broadcast traffic where maximal latencies have dropped from ~ 300 to ~ 10 ms.

Compared to solutions proposed in [32, 36], our solution achieves lower latencies. In [36], when using the full QoS feature set of the system with two high priority flows, they reach an average latency of 7.1 ms for unicast traffic compared to an average latency of 3.8 ms with our system. Moreover, their maximal latencies are in the order of 170 ms compared to 11.3 ms in our case.

7.7. Mesh Capabilities for Handling Coverage Problems. In this scenario, we consider the mesh capabilities for handling coverage problems by using the mobile robot as range extenders of the APs, as shown in the network topology in Figure 2(d). All mobile robots are using two network interface cards, one for connecting to the AP and the other one for mesh communication towards other mobile robots.

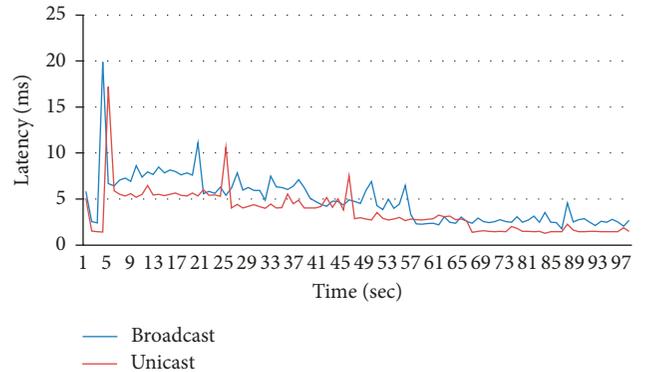


FIGURE 15: Latency of broadcast and unicast traffic with link breaks occurring every 20 s.

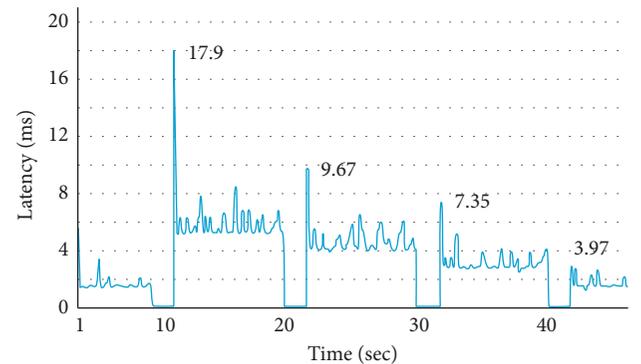


FIGURE 16: Unicast packet transmission latency in the presence of link breaks every 10 s.

We motivate the traffic separation based on the results from subsection F. In order to assess the solution for handling the coverage problems, we use the setup as shown in Figure 18(a). To emulate the situation in uncovered zones, the node is disconnected from the AP using the wpa-cli [39] disconnection method. Ping packets are sent from node E to node A. Every 10 seconds, we disconnect sequentially the nodes from the AP. This way, we increase the number of hops between the anchored node and the anchor, testing thus also the multihop anchorage process. During the first 10 seconds, node A is connected directly to the AP. After the disconnection from the AP, it is forced to search for an anchor and will select node B as its anchor. After 20 seconds, node B is disconnected from the AP too. This will enforce both nodes A and B to start searching for a new anchor. After the anchor selection process, node C will become the direct anchor for node B and one-hop anchor for node A. After disconnection

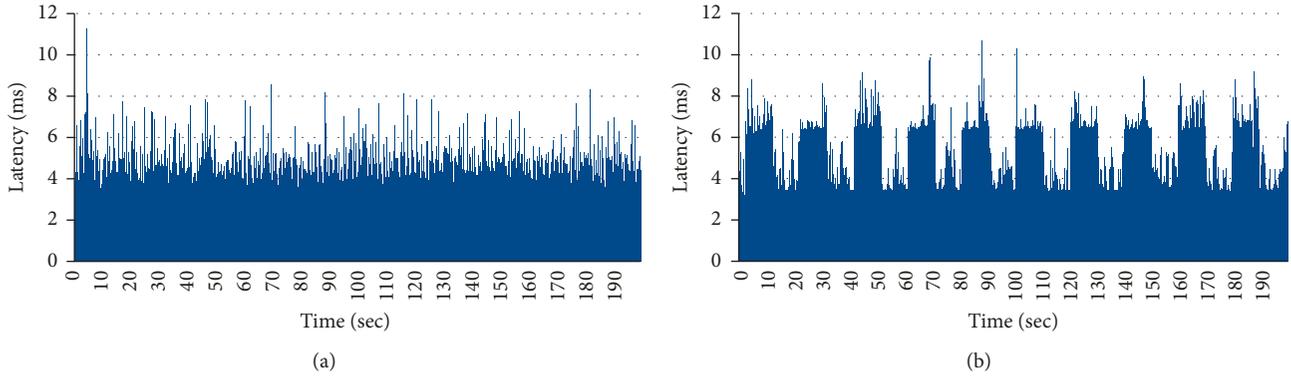


FIGURE 17: Unicast latency and broadcast latency in a mixed scenario with traffic separation. (a) Packet payload size 50 bytes. (b) Broadcast latency.

of node C, there will be three hops between the multihop anchored node A and its main anchor node D. After 40 seconds, we reestablish the AP connections for nodes C, B, and A sequentially every 10 s.

In Section 6, the key parameters of this scenario were introduced: the AP monitoring interval is set to 1s, link break detection time is set to 150 ms, and beacon interval is set to 50 ms. Measurements are performed for two scenarios, one with ping packets sent every 100 ms, and the other one every 200 ms. The expected maximum number of packets to be lost is calculated based on the number of AP disconnections multiplied by the number of packets sent during one AP connection monitor time. We expect packet losses to occur when the AP disconnection happens until the disconnection is detected by the system. In our case, we should have at most 10 or 5 lost packets, per AP disconnection for ping frequencies of 100 and 200 ms, respectively. On the other hand, when the AP connection is reestablished, we should see a higher latency for some packets at the beginning of the connection due to other information packets (GARP and NGARP) that needs to be processed.

In Figure 19, the latency for both packet rates is given. We see that by increasing the hop count between the anchored node and the anchor, the latency increases from 2 ms for one hop up to 6 ms for three hops. Also it can be noticed that every 10 seconds, when the AP disconnection happens, some packets are lost until the disconnection is detected. In the first case (ping packets every 100 ms), in total 18 packets are lost, or $\sim 2\%$ of packets. We have three AP disconnections during the measurement time, so the maximum number of lost packets we expect is 30. For the other case where we send a packet every 200 ms, in total 8 packets are lost, or $\sim 2\%$ of packets, while the maximum number to be expected 15. Moreover, when the AP connection is reestablished for the mobile nodes, we see that the first packet exhibits a higher latency due to the fact that the communication link moves to new anchor. However, in this case, no packets are lost since there is no communication outage time while switching from the old anchor to the new anchor.

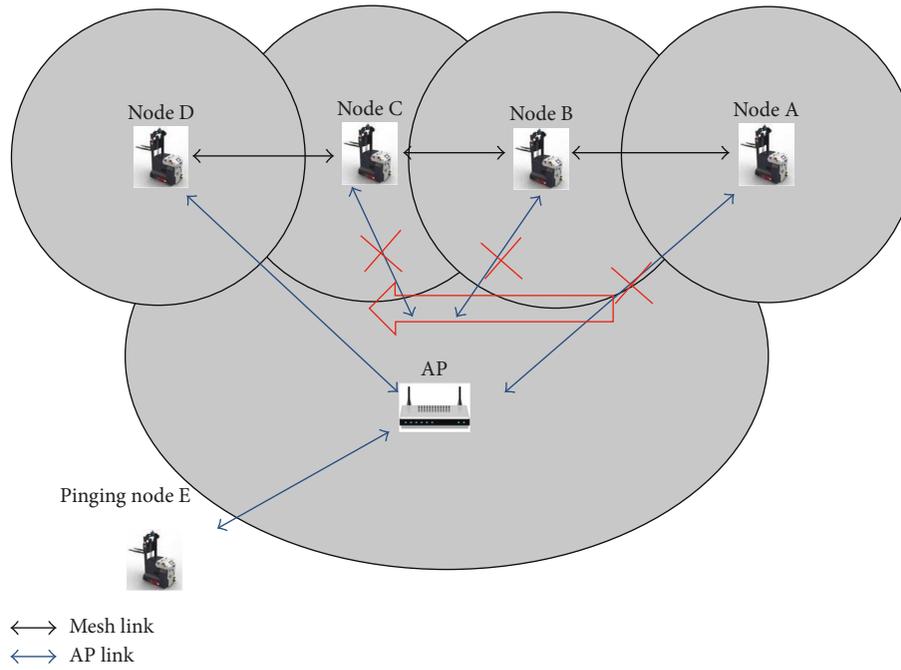
To check the communication outage time during the tests, we parsed the ROUTE table from the databases of all nodes involved in the communication. The total communication outage time was 1.72 s for the first case and 1.67 s for the

TABLE 4: Latency statistics of unicast and broadcast packet transmission in ms.

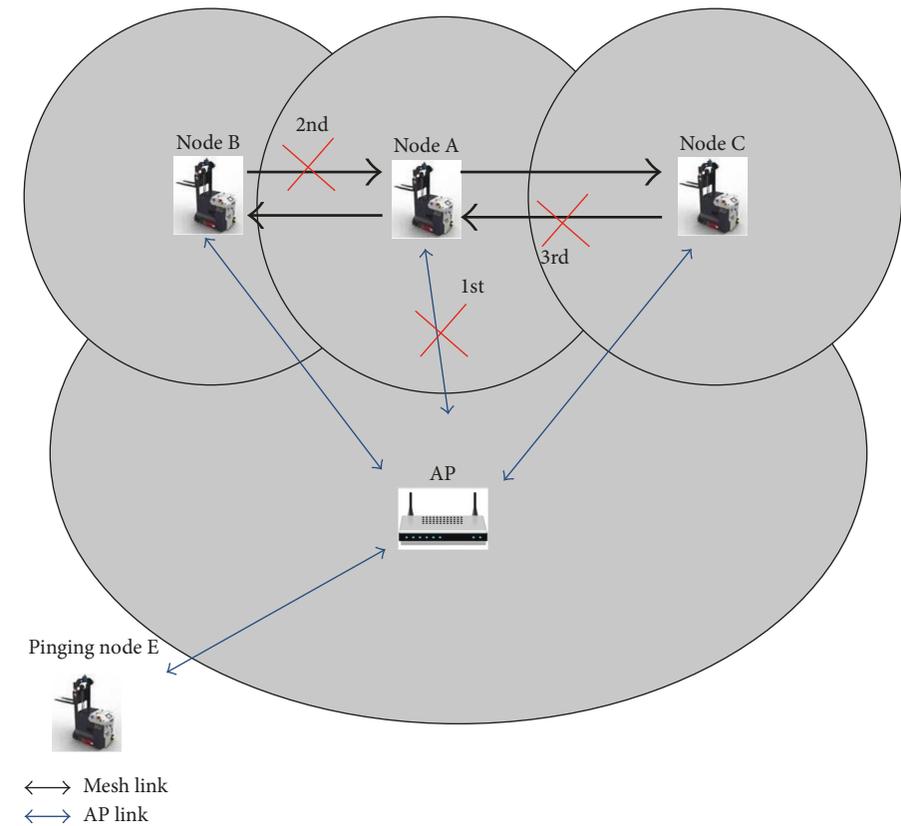
Statistics	Unicast traffic	Broadcast traffic
Average	3.8	4.55
1st quartile	3.2	3.4
Minimum	2.26	2.73
Maximum	11.3	10.7
3rd quartile	3.49	5.3

second case, with the largest continuous outage being 0.63 s and 0.55 s, respectively. This communication outage time fulfills one of the requirements for our system too, namely, RQ7. Based on the time relation between the EVENT table and ROUTE table, we observed that all of the communication outage time happened during the time until the AP disconnection was detected and the anchor was selected. There was no communication outage time due to other reasons. So the communication outage time is related to the configuration parameters: AP monitoring interval, link break detection time, and beaconing time interval.

Another key issue for the proper functioning of the anchor selection process is the existence of a bidirectional link between the anchored node and the anchor itself. In order to test our solution, we use the setup shown in Figure 7. We use a MAC filtering Click element to create unidirectional links between the anchor and the anchored node. This MAC filtering element filters out all the incoming packets with a certain MAC address, emulating thus the absence of beacons from a specific neighbor. We send pings from node D to node A every 100 ms and 200 ms, respectively. The AP monitoring interval is 1 s, link break detection time is set to 150 ms, and beaconing time interval is set to 50 ms. During the first 10 s, node A is connected directly to the AP. After 10 s, we break the link between node A and AP. Since MAC filtering for node C is enabled in node A, node A will choose node B as its anchor. After 10 s of communication, we enable MAC filtering for node B and disable MAC filtering for node C in node A. This way we create a unidirectional link from node A to node B. As such, the anchored node A should initiate a new anchor selection process. In this case, it will select node C as its anchor. After 10 s, we do it the other way around. This way, the anchored node A will alternate



(a)



(b)

FIGURE 18: Mesh capabilities to handle coverage problems. (a) Setup for testing the multi-hop anchor selection process. (b) Setup for testing the anchor selection process in case of unidirectional links.

anchors every 10 seconds due to unidirectional link breakage. After 40 seconds, we reestablish the link towards the AP and stop the measurements.

In Figure 20, the latency of unicast packets for this setup is given. Initially the latency is low for direct communication through AP, around 3 ms. When the communication is

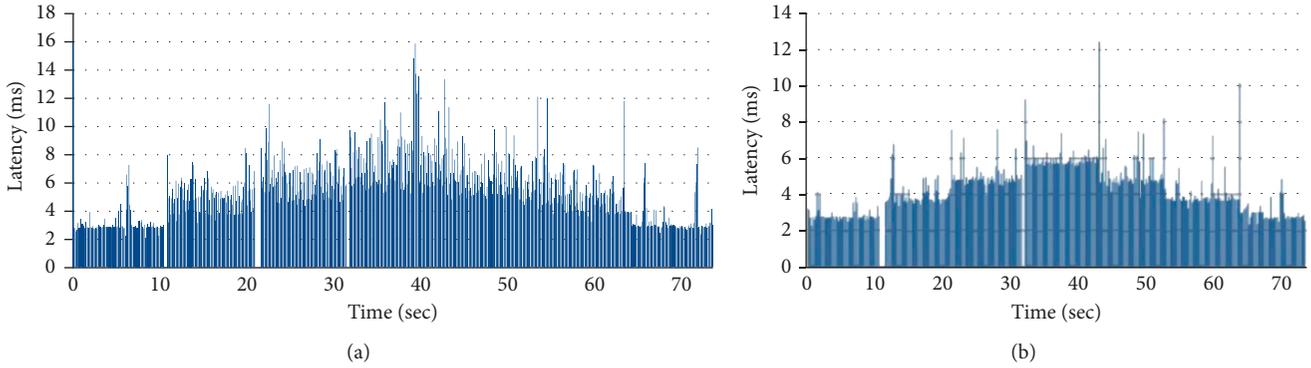


FIGURE 19: Latency for unicast traffic from an anchored node. (a) Packet rate 10 Hz. (b) Packet rate 5 Hz.

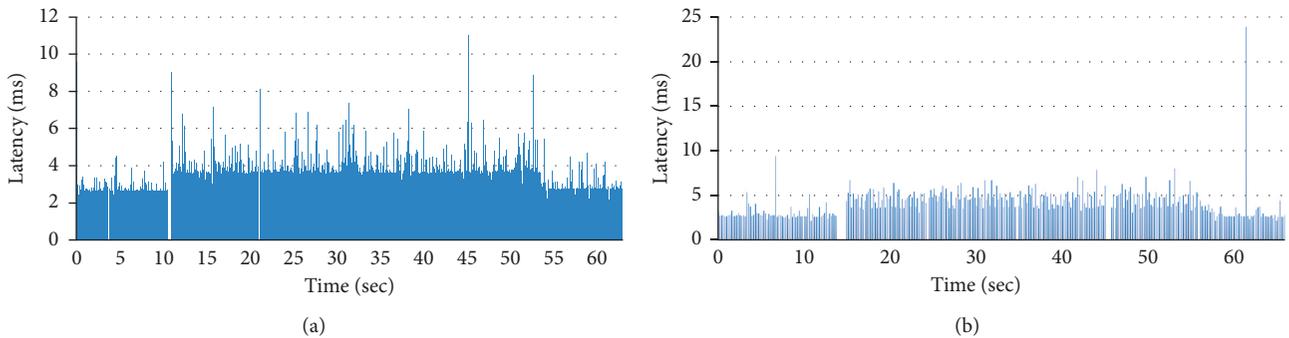


FIGURE 20: Latency for unicast traffic from an anchored node regarding the unidirectional link. (a) Packet rate 10 Hz. (b) Packet rate 5 Hz.

going through an anchor, the latency increases up to 4 ms due to the increase in number of hops. It can be noticed that some packets are lost until the node detects the AP disconnection. Afterwards, every 10 s, when the node will have to switch anchor due to the presence of a unidirectional link, again some packets are lost due to the link break detection mechanism. Since we send packets every 100 ms and the link break detection time is 150 ms, at most 2 packets can be lost in the first case. In the second case (ping packets every 200 ms), we can lose 1 packet at most.

We checked the ROUTE table with the OTM tool, and we found out that the total communication outage time was 0.75 s and 1.11 s respectively. Since we had one AP disconnection, and three link break detections, the maximal total communication outage time to be expected was 1.45 s. In total, 8 packets were lost (~1% of packets) in the first case (ping packets every 100 ms), while in the second case (ping packets every 200 ms) 9 packets (~2% of packets) in total were lost. Correlating the timing between the ROUTE table and EVENT table, the largest communication outage time was during the AP disconnection detection. It was 0.45 s and 0.6 s, respectively, while the rest was due to the link break detection mechanism.

This test shows that the packet loss and communication outage time is a function of the parameters of the mobile communication system we designed. Based on the requirements of the system, these parameters can be easily changed in the configuration file by the administrators.

Regarding packet loss rates, the proposed mobile communication system shows similarities with the system proposed in [36]. Here we have losses up to 2% where in [36], losses are between 1.4 and 3.2%. However, in our communication system, losses depend on configuration parameters.

8. Conclusions

Many existing solutions in industrial settings that make use of mobile robots utilize the existing enterprise network. In this paper, we discussed the potential drawbacks of such an approach. For our particular use case at hand, a key requirement was the ability to deliver broadcast traffic with very low latencies, a requirement that could not be fulfilled in an enterprise network where handovers take place frequently, as shown on our testbed. We proposed flexible and modular system architecture for the mobile node that makes use of two physical interfaces. The mobile node is able to function in different network topologies. It can make use of infrastructure network only; it can use only ad hoc capabilities or use both of them at the same time.

The proposed architecture is able to exploit both the advantages of the presence of an infrastructure network and the advantages of a mesh network. In this paper, we showed the feasibility of implementing such architecture and the advantage of the mixed architecture with traffic separation.

We show that the mixed architecture was able to deal with the occurrence of coverage holes. The mesh capabilities

were used to extend the AP coverage zones by using intermediate robots to enable the communication for robots outside of coverage zones. Moreover, the communication outage time was related solely to the configuration of different parameters, being the AP connection monitor time and link break detection time.

Since the use case under consideration relies on broadcast traffic communication between mobile robots, we evaluated the broadcast scalability in the mesh network. We showed that by increasing the broadcast frequency and number of nodes per group, every node will be able to receive all packets at least once; however, most of the packets will take a route that spans more than one hop.

The solution was validated in a testbed, and the outcome figures were benchmarked according to the initial requirements. As future work, other tests in larger scale setup need to be done to prove the feasibility further.

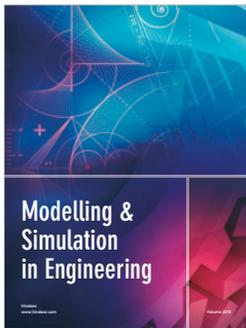
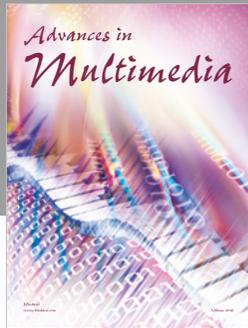
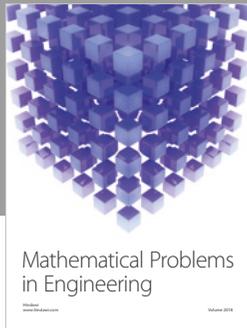
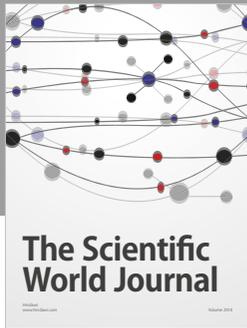
Conflicts of Interest

The authors declare that there are no conflicts of interest.

References

- [1] S. Arumugam, R. K. Kalle, and A. R. Prasad, "Wireless robotics: opportunities and challenges," *Wireless Personal Communications*, vol. 70, no. 3, pp. 1033–1058, 2013.
- [2] R. Drath and A. Horch, "Industrie 4.0: hit or hype? [industry forum]," *IEEE Industrial Electronics Magazine*, vol. 8, no. 2, pp. 56–58, 2014.
- [3] M. Essers, *Design of a Novel, Hybrid Decentralized, Distributed, Modular Architecture for Manufacturing Systems*, Universiteit Twente, Enschede, Netherlands, 2016.
- [4] M. S. Essers and T. H. J. Vaneker, "Design of a decentralized modular architecture for flexible and extensible production systems," *Mechatronics*, vol. 34, pp. 160–169, 2016.
- [5] "Manual of AGVs from motum producer," 2017, http://www.motum.be/wp-content/uploads/2015/10/Br_2Move16_en.pdf.
- [6] E. A. Jarchlo, J. Haxhibeqiri, I. Moerman, and J. Hoebeke, "To mesh or not to mesh: flexible wireless indoor communication among mobile robots in industrial environments," in *Proceedings of the International Conference on Ad-Hoc Networks and Wireless*, pp. 325–338, Lille, France, 2016.
- [7] P. Vijayakumar, M. Azees, A. Kannan, and L. J. Deborah, "Dual authentication and key management techniques for secure data transmission in vehicular ad hoc networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 4, pp. 1015–1028, 2016.
- [8] W. Jiang, F. Li, D. Lin, and E. Bertino, "No one can track you: randomized authentication in vehicular ad-hoc networks," in *Proceedings of the IEEE International Conference on Pervasive Computing and Communications (PerCom)*, pp. 197–206, Kona, HI, USA, 2017.
- [9] P. Li, C. Zhang, and Y. Fang, "Capacity and delay of hybrid wireless broadband access networks," *IEEE Journal on Selected Areas in Communications*, vol. 27, no. 2, pp. 117–125, 2009.
- [10] B. Liu, P. Thiran, and D. Towsley, "Capacity of a wireless ad hoc network with infrastructure," in *Proceedings of the 8th ACM International Symposium on Mobile Ad Hoc Networking and Computing*, pp. 239–246, Montreal, QC, Canada, September 2007.
- [11] A. Zemlianov and G. De Veciana, "Capacity of ad hoc wireless networks with infrastructure support," *IEEE Journal on Selected Areas in Communications*, vol. 23, no. 3, pp. 657–667, 2005.
- [12] H.-N. Dai, R. C.-W. Wong, and H. Wang, "On capacity and delay of multichannel wireless networks with infrastructure support," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 2, pp. 1589–1604, 2017.
- [13] L. Chen, W. Luo, C. Liu, X. Hong, and J. Shi, "Capacity-delay trade-off in collaborative hybrid ad-hoc networks with coverage sensing," *Sensors*, vol. 17, no. 2, p. 232, 2017.
- [14] W. Lee, J. Kim, and S.-W. Choi, "New D2D peer discovery scheme based on spatial correlation of wireless channel," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 12, pp. 10120–10125, 2016.
- [15] C. F. M. e Silva, T. F. Maciel, R. L. Batista, L. Elias, A. Robson, and F. R. P. Cavalcanti, "Network-assisted neighbor discovery based on power vectors for D2D communications," in *Proceedings of the IEEE 81st Vehicular Technology Conference (VTC Spring)*, pp. 1–5, Glasgow, UK, May 2015.
- [16] Q. Sun, L. Tian, Y. Zhou, J. Shi, and X. Wang, "Energy efficient incentive resource allocation in D2D cooperative communications," in *Proceedings of the IEEE International Conference on Communications (ICC)*, pp. 2632–2637, London, UK, June 2015.
- [17] J. Zheng, B. Chen, and Y. Zhang, "An adaptive time division scheduling based resource allocation algorithm for D2D communication underlying cellular networks," in *Proceedings of the IEEE Global Communications Conference (GLOBE-COM)*, pp. 1–7, San Diego, CA, USA, December 2015.
- [18] S. Xiao, X. Zhou, D. Feng, Y. Yuan-Wu, G. Y. Li, and W. Guo, "Energy-efficient mobile association in heterogeneous networks with device-to-device communications," *IEEE Transactions on Wireless Communications*, vol. 15, no. 8, pp. 5260–5271, 2016.
- [19] M. N. Tehrani, M. Uysal, and H. Yanikomeroglu, "Device-to-device communication in 5G cellular networks: challenges, solutions, and future directions," *IEEE Communications Magazine*, vol. 52, no. 5, pp. 86–92, 2014.
- [20] <http://read.cs.ucla.edu/click/click>, 2017.
- [21] V. Firoiu, B. T. Declene, M. M. Leung, S. Nanda, and C. Tao, "Mobile infrastructure assisted ad-hoc network," US Patent 9,596,619, 2017.
- [22] M. Shneier and R. Bostelman, *Literature Review of Mobile Robots for Manufacturing*, National Institute of Standards and Technology, US Department of Commerce, Gaithersburg, MD, USA, 2015.
- [23] Z. Yan, N. Jouandeau, and A. A. Cherif, "A survey and analysis of multi-robot coordination," *International Journal of Advanced Robotic Systems*, vol. 10, no. 12, p. 399, 2013.
- [24] T. Qiu, N. Chen, K. Li, D. Qiao, and Z. Fu, "Heterogeneous ad hoc networks: architectures, advances and challenges," *Ad Hoc Networks*, vol. 55, pp. 143–152, 2017.
- [25] A. Wichmann, B. D. Okkalioglu, and T. Korkmaz, "The integration of mobile (tele) robotics and wireless sensor networks: a survey," *Computer Communications*, vol. 51, pp. 21–35, 2014.
- [26] F. B. Saghezchi, A. Radwan, and J. Rodriguez, "Energy efficiency performance of wifi/wimedia relaying in hybrid ad-hoc networks," in *Proceedings of the Third International Conference on IEEE Communications and Information Technology (ICCIT)*, pp. 285–289, Beirut, Lebanon, June 2013.
- [27] H.-J. Im, C.-E. Lee, M. Jang, Y.-J. Cho, and S. Kim, "Implementation of cooperative routing robot system for long

- distance teleoperation,” in *Proceedings of the Autonomous Mobile Systems*, pp. 129–136, Stuttgart, Germany, 2012.
- [28] H.-J. Im, C.-E. Lee, and Y.-J. Cho, “Implementation of a seamless communication scheme using cooperative routing robots for teleoperation,” in *Proceedings of the IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*, pp. 310–311, Gyeongju, South Korea, August 2013.
- [29] H. Sugiyama, T. Tsujioka, and M. Murata, “Integrated operations of multi-robot rescue system with ad hoc networking,” in *Proceedings of the IEEE 1st International Conference on Wireless Communication, Vehicular Technology, Information Theory and Aerospace & Electronic Systems Technology (Wireless VITAE)*, pp. 535–539, Aalborg, Denmark, May 2009.
- [30] W. Vandenberghe, I. Moerman, and P. Demeester, “Adoption of vehicular ad hoc networking protocols by networked robots,” *Wireless Personal Communications*, vol. 64, no. 3, pp. 489–522, 2012.
- [31] R. Tahar, A. Belghith, and R. Braham, “A generic mobile node architecture for multi-interface heterogeneous wireless link layer,” in *Proceedings of the IEEE Third International Conference on the Network of the Future (NOF)*, pp. 1–5, Tunis, Tunisia, November 2012.
- [32] T. Lindhorst and E. Nett, “Dependable communication for mobile robots in industrial wireless mesh networks,” in *Co-operative Robots and Sensor Networks*, pp. 207–227, Springer, Berlin, Germany, 2015.
- [33] E. Khorov, A. Kiryanov, A. Lyakhov, and D. Ostrovsky, “Analytical study of neighborhood discovery and link management in OLSR,” in *Proceedings of the IEEE IFIP Wireless Days (WD)*, pp. 1–6, Dublin, Ireland, November 2012.
- [34] F. Tian, B. Liu, H. Cai, H. Zhou, and L. Gui, “Practical asynchronous neighbor discovery in ad hoc networks with directional antennas,” *IEEE Transactions on Vehicular Technology*, vol. 65, no. 5, pp. 3614–3627, 2016.
- [35] A. Herms, E. Nett, and S. Schemmer, “Real-time mesh networks for industrial applications,” *IFAC Proceedings Volumes*, vol. 41, no. 2, pp. 9314–9319, 2008.
- [36] T. Lindhorst, G. Lukas, and E. Nett, “Wireless mesh network infrastructure for industrial applications: a case study of teleoperated mobile robots,” in *IEEE 18th Conference on Emerging Technologies & Factory Automation (ETFA)*, pp. 1–8, Cagliari, Italy, September 2013.
- [37] C. Perkins, S. Ratliff, J. Dowdell, and Internet Draft, “Dynamic MANET on-demand (dymo) routing,” Technical Report draft-ietf-manet-dymo-26, 2006.
- [38] E. Kohler, R. Morris, B. Chen, J. Jannotti, and M. F. Kaashoek, “The click modular router,” *ACM Transactions on Computer Systems (TOCS)*, vol. 18, no. 3, pp. 263–297, 2000.
- [39] [https://wiki.archlinux.org/index.php/WPA supplicant](https://wiki.archlinux.org/index.php/WPA_supplicant), 2017.
- [40] S. Cheshire, B. Aboba, and E. Guttman, “Dynamic configuration of IPv4 link-local addresses,” Technical Report, 2005.
- [41] S. Cheshire, “IPv4 address conflict detection,” 2008.
- [42] <http://wilab2.ilabt.iminds.be>, 2017.
- [43] <https://w1.fi/hostapd/>, 2016.




Hindawi

Submit your manuscripts at
www.hindawi.com

