Hybrid Communication Architecture for Emergency Response - An Implementation in Firefighter's Use Case

The An Binh Nguyen*, Frank Englert*, Simon Farr†, Christian Gottron*, Doreen Böhnstedt*, Ralf Steinmetz*
Multimedia Communications Lab (KOM)
Technische Universität Darmstadt
Rundeturmstr. 10, 64283 Darmstadt, Hessen, Germany
Email: *{The.An.Binh.Nguyen, Frank.Englert, Christian.Gottron, Doreen.Boehnstedt, Ralf.Steinmetz}@kom.tu-darmstadt.de
†s.farr@me.com

***Abstract*—Emergency response is a critical mission. In an emergency situation, coordination and information dissemination are two of the most important tasks. The rise of the smart mobile devices leads to the tendency to use these devices to facilitate the aforementioned tasks. Despite of that, communication based on mobile devices in an emergency situation is not always straight forward since parts of the networks might not be available at the same time. Nevertheless, it is of utmost importance to keep the information synchronized across multiple stakeholders. In this work, we design and implement a hybrid communication system to support the relief work of the responders, in particular, coordination and information synchronization among the stakeholders through the usage of mobile devices. Our focus lies on the relief works carried out by the firefighters. Our designed system is evaluated regarding its performance and its usability through a field test to show its applicability in practice.**

I. Introduction

In emergency response scenarios, a large amount of relief tasks are accomplished by the firefighters. Firefighters are often one of the first responders to be on-site in an emergency situation. To facilitate the work and the coordination of the firefighters, communication is very important. Until now, the communication among the firefighters are still heavily based on radio signal [1]. Even though radio based communication can disseminate data quickly, it can be inefficient since a message will be disseminated to all firefighters, regardless if the message is relevant for them or not. It raises the question, if there is other option appropriate for the firefighters' communication. The answer can be smart mobile devices. In the last few years, we have seen an increasing number of smart mobile devices. Smart mobile devices provide us several communication possibilities besides classical cellular network, e.g., WiFi, UMTS. Furthermore, smart mobile devices can enhance and allow the firefighters to exchange information in several forms, such as text information, images, videos etc. Thus, smart mobile devices can be exploited in the emergency relief task [2]. Recently, vehicle based computers have been used more to enhance information exchange for the first responders [3]. Naturally, firetrucks as the transportation for the firefighters can also be used in the similar way to facilitate the communication process.

Communication for the firefighters poses several challenges. First, the firefighters might be dispatched from different fire departments. This might lead to different arrival times, which make the coordination and information exchange of the firefighters on-site more difficult. Second, it is not always possible for the firefighters to uphold the connection with the central headquarters. Consequently, it is essential and beneficial for the firefighters to be able to communicate in an ad hoc manner during on-site missions [4]. To address the aforementioned challenges, we design and implement a coordination and communication system for firefighters. The system can be used as well by the first responders in emergency response:

- We propose a hybrid communication system which is based on publish/subscribe queuing [5] on multiple layers of the responders' hierarchy. The system enables the communication 1) between the on-site responders and the headquarters, 2) among the on-site responders via firetrucks, 3) between the on-site responders and the on-site coordinators through an on-board computer system in the firetrucks. Our design supports and permits a delay tolerant communication, which increases the flexibility for the communication.
- We design and construct a communication module based on Raspberry Pi [6] which can be installed into the firetrucks that realizes an on-board computer system. The components of this module can be flexibly exchanged or extended. This construction can provide even more flexible communication options and enhance the on-site coordination.
- We implement the system based on our design, conduct several performance tests and field tests with the fire department of Bad Vilbel[1] to assess the applicability of the system in the real-world scenarios.

The rest of this paper will be structured as follows: in Section II, we review the related work regarding coordination and communication in the field of emergency response. The design of our system is presented in Section III. We show the implementation of the system and the construction of the communication module for firetrucks in Section IV. The system is evaluated via different performance tests and a field test. The result is discussed in Section V. Section VI concludes our work.

[1] http://www.ff-badvilbel.de/

II. RELATED WORK

Research on emergency response has gained more attention in the recent years. In this section, we will review several works that focus on the communication aspect and management in an emergency situation.

SmartRescue [7] is a system based on publish/subscribe model which allows the user to exploit the capability of their smart phones as a sensor platform. Each user acts as a publisher of the sensor data. Thus, it is able to get an overview of the current situation by aggregating the data provided by the individual publishers. Data provided by the publishers can be subscribed by different types of subscribers. A subscriber regardless of being a public user or an officer can subscribe to the topic which he has interest to. Furthermore, a web-based broker is developed as a special subscriber to mine information from the public users and store it centrally for further references.

Aloqaily et al. [8] design an audio conferencing system for the firefighters based on MANET. The firefighters in the system are organized into different communication layers. To realize the inter-layer and intra-layer communication, the architecture comprises of one main conference and several clusters as the sub-conferences. The main conference is under control of a main leader. Team leaders and other team members of a cluster need to register with the main leader to join the main conference. Upon joining the main conference, the members can ask to join the sub-conferences maintained by the team leaders. The communication of the system is based on multicast. With this, the hierarchical structure is supported and the communications with different concerns can be separated.

DUMBONET [9] is a collaborative communication system for the emergency rescuers. DUMBONET assumes an emergency scenario with different emergency sites. On each site, the rescuers posses mobile personal devices which allows them to exchange information through MANET. The MANET protocol OLSR is installed on the devices of the rescuers. The communication between different sites and the command center is carried out using a satellite link.

HelpMe [10] is a system which supports the rescue operation carried out by spontaneous responders. The system consists of two parts, the central server and the smart phones. The smart phone client in HelpMe system can switch between the normal client mode, which allows the smart phone to connect to the server and the peer mode, which allows the smart phone to communicate with other smart phones in the close proximity in an ad hoc manner. In the ad hoc peer mode, a smart phone can discover other smart phones and send help request to other smart phones based on the users' profiles. The users' profiles are stored, maintained by the server and can be partially downloaded to a smart phone.

WORKPAD [11] is a collaboration platform for the responders in disaster relief work. The architecture of the system comprises of two levels, i.e., the front-end user and the back-end user. The front-end user is the user working directly at the emergency location. They carry PDAs and communicate with each others over a mobile network. The back-end user is located at the headquarter and operates the server to coordinate the relief tasks and provide information. The front-end user and the back-end user communicate with each other through a gateway of the mobile network. A middleware between the front-end and the back-end allows the back-end to assign tasks to the front-end through a web-service call.

Berioli et al. [12] discuss different architectures for communication in emergency response. The authors highlight the importance of hybrid infrastructures using terrestrial and satellite based communication for cooperation between different stakeholders under emergency situation. On the one hand, the responders communicate with each other over an ad hoc network. On the contrary, the information from the command center is relayed via satellite based connection using multicast.

Compared to the aforementioned works, our system realizes publish/subscribe communication at multiple layers of the responders' hierarchy. Thus, our system provides more flexibility. Our system also supports delay tolerant communication. Furthermore, our on-board communication system allows the firetrucks to form an on-site coordinators network, which permits a more robust communication. The usage of Raspberry Pi as the base for the on-board system in our work also allows for a low cost solution which can bring advantages compared to the other expensive options like the satellite based communication. All in all, our realistic approach can serve as the base communication platform for information acquisition and dissemination which are important challenges for emergency management as mentioned in [13].

III. HYBRID COMMUNICATION SYSTEM FOR FIREFIGHTERS

In this section, we first analyze the requirements in disaster relief missions carried out by the firefighters. Consequently, we will describe the overall design of our system and discuss different components of the system in detail.

A. Requirements Analysis

In a disaster relief mission, the firefighters can be dispatched from different fire departments, which results in different arrival times at the location. In addition, under circumstances it is not always possible to preserve the connection with the headquarters. Furthermore, the relief tasks require the firefighters to be able to communicate and exchange information with each other on-site. Altogether, a reliable communication both centrally and locally, is necessary for the firefighters during relief works. The detailed analysis of the aforementioned requirements for communication in firefighters' use-case is as follows:

- To cope with the different arrival times, the first arrival will take care of the reconnaissance and share the information with other firefighters who arrive later. The communication between the firefighters needs to support time decoupling
- The firefighters need to share information with all other firefighters regardless from which departments they come from, thus the communication needs to be independent from the affiliation of the firefighters
- Last but not least, firefighters may want to carry out other works using the mobile devices in parallel while collecting information, thus synchronization decoupling from communication is also important.

The above observations lead us to the decision to use publish/subscribe [5] as the communication instrument in our sys-

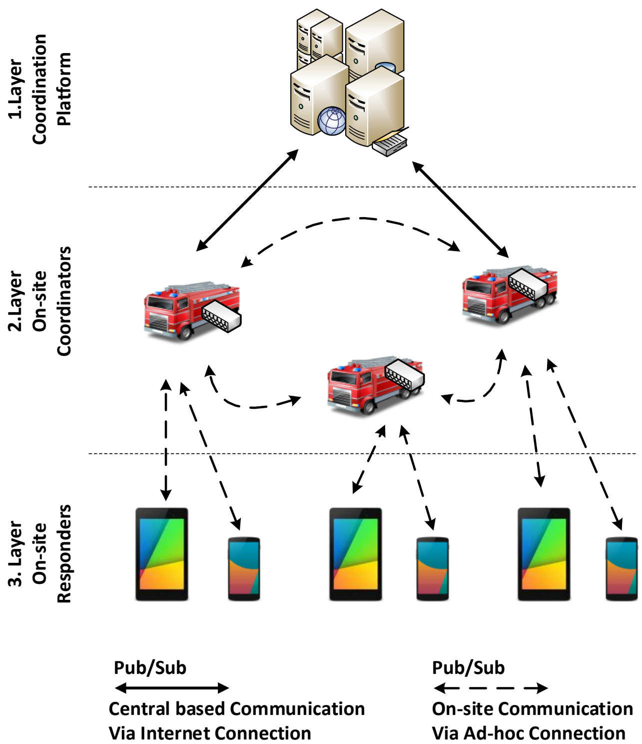

Fig. 1. Overall architecture

tem, since publish/subscribe model supports time decoupling, space decoupling and synchronization decoupling.

B. Overall Architecture Design

Our design focuses on the use-case of the firefighters. In the relief works, firefighters are divided in different groups. The hierarchy can be divided in three layers as shown in Figure 1. The highest layer represents a central headquarter which also functions as a coordination platform and intends to keep a current overview of all groups. When an emergency call is received, the central headquarter will be in charge of dispatching the firetrucks from different locations to the notified locations and update the current situation to the firetrucks if possible. In the second layer of our illustration, different firetrucks installed with an on-board communication system form a network of on-site coordinators. The second layer provides different services for the firefighters carrying mobile devices, such as data buffering, information synchronization among firefighters, and information dissemination between firefighters and headquarters. Furthermore, the second layer also bridges the communication between on-site firefighters and the headquarters. Last but not least, the third layer represents the individual firefighters that carry capable mobile devices such as smart phones, tablets etc. The firefighters form a network of on-site responders. The on-site responders network is in charge of obtaining information about the emergency situation, share it with the other responders as well as report it back to the headquarters for storage. To send a report back to the headquarter, the mobile client first connects to the on-site system in the firetruck. This on-site system will take care of the message forwarding to the server. The forwarding mechanism allows the message to be stored/buffered at the firetrucks. Hence, the system is more flexible and can enable delay tolerant communication.

C. Central based Communication

The central entities in our scenarios are the servers at the central headquarters. The main task of the servers is to coordinate the different groups in an emergency situation. The communication between the servers and the firetrucks is bi-directional. On the one hand, the server needs to be able to receive update about status information from different firetrucks. Such status information can be, e.g., the current location of the firetrucks, new firetrucks will arrive at the location, firetrucks cannot arrive to the location due to unforeseen reasons, etc. On the other hand, the server also needs to be able to disseminate new information and notifications to the firetrucks. Also, servers should be able to remove an outdated notification. Examples of notification may be obstacles and objects at a particular location, reinforcement from another group of firefighters etc. Each server runs a message queue which allows the firetrucks to subscribe to receive notification about a specific events. The firetrucks can also publish information for further dissemination/storage through the central servers. The connection between the firetrucks and the servers can be supported using different wireless communication technologies such as UMTS, LTE etc. Therefore, the on-board computer system in the firetrucks is extended by a Raspberry Pi. Further details will be provided in Section IV.

D. On-site Communication

On each firetruck, an on-board communication system is installed. The on-board system runs a message queuing server which is the central heart of our on-board communication system, as it will bridge the communication between the on-site location and the central headquarters. The first firetruck that arrives at the location becomes an anchor of the on-site coordinators network. The anchored firetruck will create an ad hoc network, that allows the later arriving firetrucks to join. This network can be used to subscribe and synchronize the information already stored in the anchored firetruck. Each firetruck is capable of both becoming an anchor node and joining an ad hoc network created by another firetruck. Consequently, the firetrucks form an information exchange network and an on-site coordination platform based on the firetrucks' mesh network. Furthermore, with the ad hoc network created by the firetrucks, the firefighters can communicate with each other. The communication is also facilitated by the queuing server on the firetruck. The firefighters with the mobile devices can subscribe to receive only relevant information. Due to this fact, the firefighters can be organized through the firetrucks as the on-site coordinators. Additionally, the firefighters can publish information which they collect to the queuing server on the firetrucks. The published information will be disseminated to the corresponding entities whether these are other firetrucks, firefighters, or the central servers. As a result, the firefighters form a network of direct on-site responders that can communicate with each other in an ad hoc manner through the on-board communication system in the firetrucks. This set-up conforms with the fact that there are many volunteers within firefighters who might need coordination from the more

experienced firefighters. Such coordinators can be those who operate the firetrucks' on-board system.
Besides the ad hoc communication, the firefighters can use the mobile devices to send reports back to the headquarter by forwarding the message through the on-board system of the firetruck. This permits the on-board system to aggregate and process the information before forwarding to the central server. Thereby, the data throughput sent back to the server can be reduced. The on-board system of the firetruck is also able to store information for later retrieval, thus it supports delay tolerant communication.

IV. System Implementation

In this section, we will describe the construction of the on-board communication component for the firetrucks and the implementation of the server as well as the client on the mobile devices.

A. Communication Component for Firetrucks

The communication system on the firetrucks is designed to support location determination, W-LAN ad hoc for on-site communication, UMTS or similar technology for communication with the central headquarters and a publish/subscribe queuing server. The implementation of this on-board communication system is shown in Figure 2. Our on-board system for firetrucks consists of a Raspberry Pi [6], a USB hub, a GPS receiver, a W-LAN USB adapter, and a UMTS surf stick. A Raspberry Pi is a flexible small computer which can be extended to support multiple purposes. In this case, our Raspberry Pi functions as the heart of the on-board system. The Raspberry Pi runs the message queuing system, and determines the location of the firetruck through GPS receiver reading. The Raspberry Pi is also able to send information back to the central server through Internet connection via the UMTS surf stick. There are different options to implement the communication with the central server. It is possible to extend the Raspberry Pi with LTE or a satellite connection via the USB Hub. Last but not least, the W-LAN adapter allows the Raspberry Pi to work as an on-site server to support ad hoc communication between different on-site entities. As the result, the on-board system operates in two modes at the same time, i.e., ad hoc mode and Internet based mode. The queuing server running on the Raspberry Pi can incorporate the communications from both network types. The box shown in Figure 2 is also wired with a current transformer, which enables the electricity supply for all the components directly from the firetrucks.

B. Coordination Platform and Mobile Devices

The server as the coordination platform is implemented using PHP on an Apache server. As the coordination platform, the server supports different roles in relief works since different roles lead to different duties, i.e., the on-site relief workers, the on-site coordinators, the coordinator at the headquarter and, the administrator of the platform. The server provides an interface for the users to create and edit tasks, notifications, deployments, different types of objects such as firetrucks, obstacles. Google Map service [14] is integrated which allows the user to move objects interactively on the map. An instance of RabbitMQ [15] queuing server is up and running on the

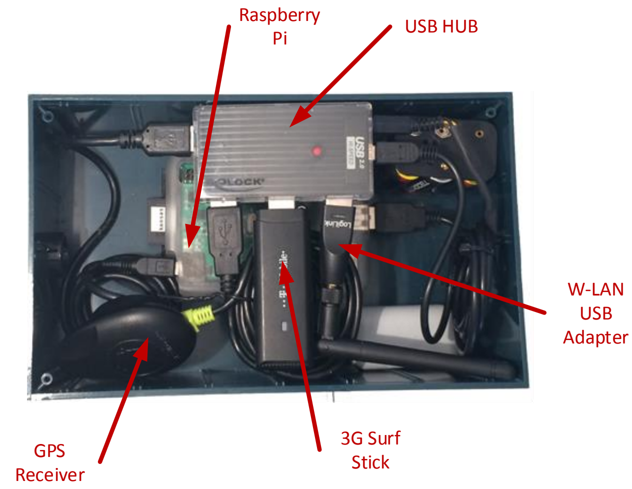

Fig. 2. Communication Component built upon Raspberry Pi

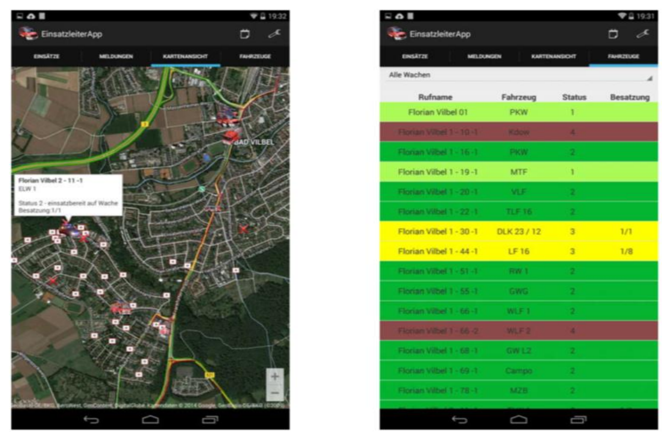

Fig. 3. Graphical User Interface on the Android App

server. This queuing server functions as the message broker for publish/subscribe communication between firetrucks and the server. MySQL is used as the persistent database.

We implement the prototyped client on the mobile devices as an Android application. The application is tested on two types of tablets, i.e., Nexus 7 [16] and Getac Z710 [17]. The client on the Android mobile devices also runs a message queue which is used to store and buffer information to report back to the on-site coordinators at the later time. We also implement a graphical user interface for the user to easily retrieve information and create report. The interface on the mobile devices is created resembled that of the server to enhance the usability. A snapshot of the implementation is shown in Figure 3. In Figure 3, a list of current firetrucks and objects is projected on the map, a list of several notifications is also shown with different colors according to their priority to boost up the work of the firefighters.

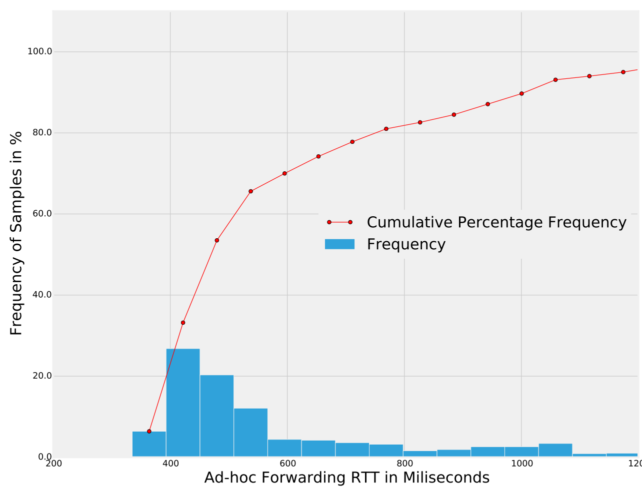

Fig. 4. Message Forwarding from the Mobile Devices to the Server through the on-board System

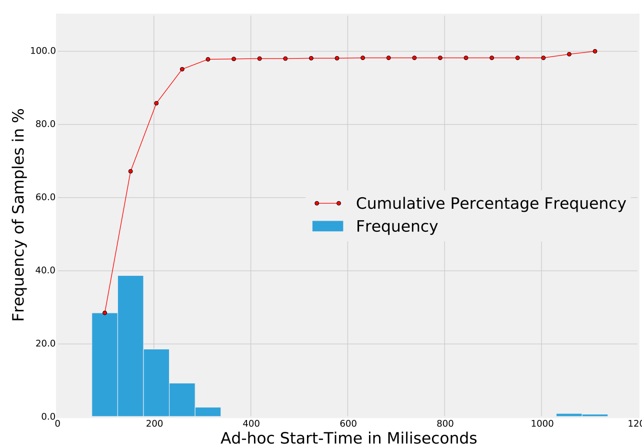

Fig. 5. Ad hoc Start Time between two Firetrucks

V. Evaluation

The prototyped system implemented as described in the previous section was evaluated with regards to its performance and its usability through a field test carried out by the firefighters.

A. Performance Tests

We conducted three test-cases to assess the performance of our system. We used two Raspberry Pis with the same construction as introduced in IV, and one Nexus 7 with the client installed. A Raspberry Pi can be configured to enable UMTS connection with the server and to create an ad hoc W-LAN which allows either the tablet or the other Raspberry Pi to join to. Each test was repeated 1000 times which results in 1000 data samples for each test-case. The test scenarios were designed to simulate the communication at different layers of our system. First, we measured the round trip time (RTT) required to forward a message from the mobile devices to

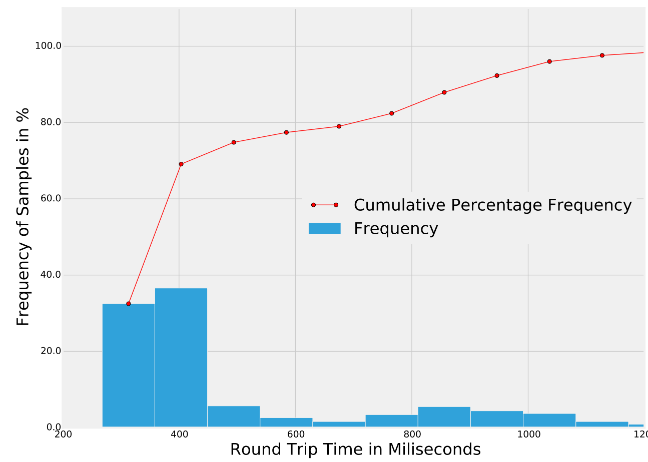

Fig. 6. Message Forwarding from Mobile Device to the Server through two on-board Systems

the server through the on-board communication system of a firetruck. To realize this test, the Nexus 7 sent the message via ad hoc connection to Raspberry Pi, which then forwarded it through UMTS connection to the server. With the first test, we wanted to show that the on-board system is capable of bridging the communication between on-site and the central headquarter. The result of the first test is shown in Figure 4. 57% of the RTT samples lie around 400 milliseconds. Up to 98% of the RTT samples are smaller than 1 second. Thus, the message forwarding through an on-board system happens within second, which should be sufficient for further processing.

In the second test-case, we simulate the ad hoc communication between two firetrucks. One on-board communication system is switched to the anchored mode, while a second coming firetruck will try to join the ad hoc network created by the anchored firetruck. This was realized using two Raspberry Pis; one Pi turned on W-LAN ad hoc mode that allows the other Pi to communicate with it. We measured the time required to start the ad hoc mode until the message can be exchanged between these two firetrucks. The samples frequency distribution in the second test follows a heavy tailed distribution. The results of the second test-case can be found in Figure 5. In around 82% of the samples, the two Pis can start and exchange message within 200 milliseconds. The time required in the second test is much lower than the time required in the first test, since it is expected that the direct ad hoc communication is faster than the communication with the central server through UMTS connection.

In the last test, a mobile client can communicate with a firetruck without connection to the central server, thus this firetruck will have to forward the message through another firetruck that can connect to the server. The Raspberry Pi without UTMS, the Raspberry Pi with UMTS and the Nexus 7 were used to simulate this scenario. The goal of this test is to show that a firetruck can also be used as a buffer on-site which allows the message forwarding to the server at a later time. The result plotted in Figure 6 also shows similar observation as in other tests. Most of the test samples require around 400 milliseconds RTT. In around 99% of all samples,

the forwarding through the on-board systems can be finished less than 1,2 seconds. We can also observe that, in the third test the RTT is distributed more frequently through out the range between 200 and 1200 seconds than the first two tests. This can be explained by the fact that we conducted the tests in a real-world conditions; different factors such as UMTS link quality, high buildings, location etc. might affect the distribution of the test results.
All in all, the results obtained through the tests carried out in the real-world conditions show, that our system is capable of supporting the relief work carried by the firefighters in practice.

B. Field Tests

Besides the performance tests, we also conducted a field test with the firefighters from Bad Vilbel, Germany to assess the usability and the applicability of the system in practice. The field test took place at a retirement home. It was assumed that the fire broke out in the second floor of the home. Two firetrucks equipped with our system were dispatched to the destination from different locations. The firefighters need to use the system to coordinate and execute different tasks, e.g., extinguish the fire, clear obstructions, etc. For the field test, we installed the application on two types of Android tablets, i.e., Nexus 7 [16] and Getac Z710 [17]. Nexus 7 is designed by Google and targets the general base of the users, while Getac Z710 is made by the company Getac that focuses on development of specialized tablets for emergency responders. There were twenty participants with their age ranging from 18 to 60 years old. The participants were mixed of volunteers and firefighters by profession. 85% of the participants have already used smart mobile devices before. The three most important questions in our questionnaire are concerned with the satisfaction of the users towards the system; i.e., how well can they use the Android application, how well can they use the web server for coordination, whether they consider the communication speed to be sufficient. According to the data obtained from the questionnaire, 45% of the participants consider the Android application to be very good, 55% consider the application to be good, there is no criticism on the Android application. Similarly, 50% find the web server easy to use, 50% find it even very easy to use. Regarding the communication speed, 100% participants think the communication speed between components is sufficient for their work. One interesting fact that came out of the data is regarding the usage of Nexus 7 and Getac Z710. While 100% of the participants find the Getac tablet good or very good, only 95% participants find the Nexus 7 suitable for their application, 5% even find the Nexus 7 not appropriate for their work. The percentage of very good ratings for the Getac lies by 55%, much higher than 15% in the case of the Nexus 7. The overall data from the field test lead us to the conclusion that such smart devices empowered system is well received by the emergency responders; nevertheless, the system still requires special hardware tailored towards the special use-case.

VI. Conclusion

We have presented a hybrid communication system that facilitates both centralized communication and ad hoc communication. Our system, which is based on publish/subscribe at different layers allows for flexible coordination, both centrally and on-site. Furthermore, delay tolerant communication is also supported by our designed concept. We successfully show the applicability of our design in practice through the performance tests and the field test. For the future work, we intend to study more options for long range communication as well as short range communication, e.g., Bluetooth to extend the on-board system of the firetrucks. In general, the use of Raspberry Pi provides us the flexibility to adjust or extend our system. The on-board communication system can also be adapted and configured to match the special requirements of different stakeholders in an emergency situation.

Acknowledgment

This work has been co-funded by the LOEWE initiative (Hessen, Germany) within the NICER project and by the LOEWE initiative (Hessen, Germany) within the SOLIN project.

References

[1] M. Aloqaily, F. Belqasmi, R. Glitho, and A. Hammad, "Multiparty/Multimedia Conferencing in Mobile Ad Hoc Networks for Improving Communications between Firefighters," in *Computer Systems and Applications (AICCSA)*, 2013.

[2] N. Jing, Y. Li, and Z. Wang, "A Context-aware Disaster Response System using Mobile Software Technologies and collaborative Filtering Approach," in *Computer Supported Cooperative Work in Design (CSCWD)*, 2014.

[3] M. van der Lee, J. van de Ven, R. van Rijk, and R. Bras, "Situational Awareness for First Responders: Evaluation of the BIMS Field Trial," in *Technologies for Homeland Security (THS)*, 2009.

[4] X. Wu, M. Mazurowski, Z. Chen, and N. Meratnia, "Emergency Message Dissemination System for Smartphones during Natural Disasters," in *ITS Telecommunications (ITST)*, 2011.

[5] P. T. Eugster, P. a. Felber, R. Guerraoui, and A.-M. Kermarrec, "The Many Faces of Publish/Subscribe," *ACM Computing Surveys*, vol. 35, 2003.

[6] "Raspberry Pi," http://www.raspberrypi.org/.

[7] J. Radianti, J. J. Gonzalez, and O.-C. Granmo, "Publish-Subscribe Smartphone Sensing Platform for the Acute Phase of a Disaster: A Framework for Emergency Management Support," in *Pervasive Computing and Communications Workshops (PERCOM Workshops)*, 2014.

[8] M. Aloqaily, S. Otoum, and H. T. Mouftah, "A Novel Communication System for Firefighters using Audio/Video Conferencing/Subconferencing in standalone MANETs," in *Computer Science and Information Technology (CSIT)*, 2013.

[9] K. Kanchanasut, A. Tunpan, and M. Awal, "DUMBONET: a Multimedia Communication System for collaborative Emergency Response Operations in Disaster-affected Areas," *International Journal of Emergency Management*, vol. 4, 2007.

[10] O. Mokryn and D. Karmi, "Help Me: Opportunistic Smart Rescue Application and System," in *Ad Hoc Networking Workshop (Med-Hoc-Net)*, 2012.

[11] T. Catarci, M. D. Leoni, and A. Marrella, "The WORKPAD Project Experience: Improving the Disaster Response through Process Management and Geo Collaboration," in *International Conference on Information Systems for Crisis Response and Management, ISCRAM*, 2010.

[12] M. Berioli, A. Molinaro, S. Morosi, and S. Scalise, "Aerospace Communications for Emergency Applications," *Proceedings of the IEEE*, vol. 99, 2011.

[13] E. Gelenbe and F. Wu, "Future Research on Cyber-Physical Emergency Management Systems," *Future Internet*, vol. 5, 2013.

[14] "Google Map," https://developers.google.com/maps/.

[15] "Rabbit MQ," http://www.rabbitmq.com/.

[16] "Nexus 7," http://www.google.com/nexus/7/specs/.

[17] "Getac Z710," http://de.getac.com/tablets/z710/features.html.