Designing Infrastructure-less Disaster Networks by Leveraging the AllJoyn Framework

Srikant Manas Kala[†], Vanlin Sathya[¶], Suhel Sajjan Magdum[†], Tulja Vamshi Kiran Buyakar[†], Hatim Lokhandwala[†], and Bheemarjuna Reddy Tamma[†]

[†] IIT Hyderabad, India. [¶]University of Chicago, USA.
[†][cs12m1012, cs17mtech11012, cs16mtech11020, cs13m1018, tbr]@iith.ac.in, [¶]vanlin@uchicago.edu.

Abstract

AllJoyn is an open-source framework which has the potential to be the platform for next-generation proximity centric disaster network (DiNet) applications. However, its development is still in nascent stages, and one of the primary challenges is the design of efficient routing algorithms, which can ensure seamless and uninterrupted communication in an unfavorable environment. In this work, we implement a DiNet prototype using AllJoyn to highlight the challenges of multi-hop routing and propose the concept of *extended proximity* (e-proximity) in AllJoyn. As a first step towards solving this challenge, we carry out field experiments by implementing an AllJoyn file-transfer application on a trivial DiNet prototype. We then evaluate the performance of the AllJoyn based disaster network and demonstrate that AllJoyn can support robust and reliable DiNet applications.

Keywords

Disaster Networks, AllJoyn, DiNet, PCN

ACM Reference Format:
Srikant Manas Kala[†], Vanlin Sathya[¶], Suhel Sajjan Magdum[†], Tulja Vamshi Kiran Buyakar[†], Hatim Lokhandwala[†], and Bheemarjuna Reddy Tamma[†]. 2019. Designing Infrastructure-less Disaster Networks by Leveraging the AllJoyn Framework. In *International Conference on Distributed Computing and Networking (ICDCN '19), January 4–7, 2019, Bangalore, India.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3288599.3295596

ICDCN '19, January 4–7, 2019, Bangalore, India

ACM ISBN 978-1-4503-6094-4/19/01…$15.00
https://doi.org/10.1145/3288599.3295596

1 Introduction

With the rise in global warming, the accelerated pace of climate change, and intense seismic activity, the frequency of natural disasters is increasing, *e.g.*, Tsunami in Japan (2011), floods in Uttarakhand, India (2013), earthquake in Nepal (2015), hurricane Harvey (2017), and floods in Kerala (2018). Communication infrastructure in disaster-hit zones is often completely or partially damaged which leads to a disrupted network. Hence, we aim to design a proximity centric disaster network application with minimum requirements of telecommunication infrastructure by leveraging new technological frameworks such as AllJoyn. Our disaster network (DiNet) model envisions *Collaborative Participation* from the local communities as a primary objective, which pivots the success of this project on the end-user being at the center of disaster data gathering and dissemination. Such DiNet architectures are possible today due to the pervasive presence of Smartphones in most parts of the world, regardless of remoteness of the location or extremities of the terrain. Increased penetration of Smartphones also facilitates analysis and prediction of mobility patterns through development of mobility-aware applications [1].

2 The Challenge of Multi-Hop Routing in AllJoyn

AllJoyn is an open source platform [2] that creates ad hoc opportunistic networks of devices which fall within each other's transmission range and facilitates real-time communication between them. AllJoyn applications are cross-platform and user-friendly, which makes it a primary candidate for creating proximity centric DiNets. The high-level architecture of the AllJoyn framework has two primary components, *viz.*, AllJoyn Apps and AllJoyn Routers, or simply put, Apps and Routers. Three common functional topologies exist which are illustrated in Figure 1. AllJoyn performs participant equipment discovery and attachment, session management and data transfer over a single hop between participant mobile devices. Let us consider a trivial 4-node AllJoyn *Proximity*

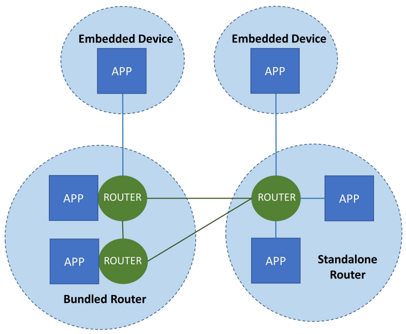


Figure 1: **App and Router Topologies in AllJoyn.**

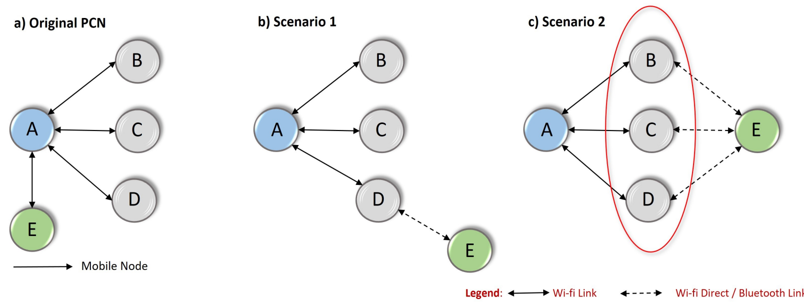
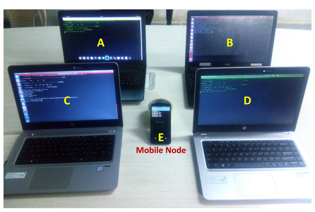
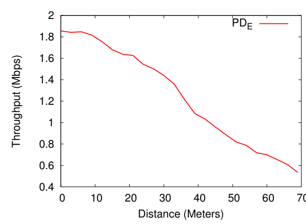


Figure 2: **Multi-hop Routing in AllJoyn.**



(a) **AllJoyn PCN.**



(b) **Throughput : Mobile Node-E.**

Figure 3: **Multi-hop Routing in AllJoyn : An Evaluation.**

Centric Networks (PCN) illustrated in Figure 2 (a). An *advertising device* (AD) initiates an AllJoyn PCN by creating a *channel*, which is a data sharing session in AllJoyn terminology. The mobile devices in its proximity receive the advertisement and may latch on to the channel to share data with the AD. The PCN is sustained as long as AD keeps the channel active. However, a mobile device may move to a location which is beyond the direct transmission range of AD. This leads to scenarios presented in Figure 2 (b), and Figure 2 (c), which represent a disrupted PCN. If the underlying graph of the PCN is connected, it implies that every participant device (PD) is in the proximity of at least one PD.

Currently, for an AllJoyn session to sustain, a direct single-hop communication between the AD and PDs is necessary *i.e.*, all PDs must be in the direct proximity of the AD. This however is seldom the case in real PCNs owing to the inherent dynamic nature of mobile devices. For seamless connectivity and session sustenance, it is imperative that *multi-hop transmission* be a norm in the data communication protocols for the AllJoyn PCN. This idea is illustrated in Figure 2 (b) where a PD_E moves out of the direct proximity range of AD_A, and thus the direct communication link between AD_A and PD_E is disrupted. An extended PCN will mitigate this dynamism in the network by routing the traffic between AD_A and PD_E through PD_B over a two-hop link. Another scenario is depicted Figure 2 (c), where PD_E can choose between three equidistant nodes PD_B, PD_C, and PD_D to communicate with A over a 2-hop connection. To create these scenarios, we implement a 5-node AllJoyn PCN illustrated in Figure 3 (a), where the PCN devices are shown and labeled. The five nodes consisting of four laptops and a smartphone run the AllJoyn framework and create a PCN. Four laptops serve as the stationary nodes *viz.,* AD_A, PD_B, PD_C, and PD_D, all of which run the Ubuntu 16.04 LTS OS. A Google Nexus 5 smartphone running Android KitKat 4.4.4 OS, is the mobile node PD_E.

In line with the Scenario 1 in Figure 2 (b), we keep four nodes of the AllJoyn PCN stationary while PD_E is mobile. As PD_E moves away from AD_A, the network performance deteriorates, which is evident from the plots in Figure 3 (b) and Figure 3 (c). Beyond a certain distance (69m), the signal strength quantified by *received signal strength indicator* (RSSI), is too weak to sustain an operational link. Thus, PD_E is disconnected from the AllJoyn PCN, and is no longer a part of the DiNet. At this distance, PD_E moves in an arc, creating Scenario 2 depicted in Figure 2 (c), where it is within the transmission range of PD_B, PD_C, and PD_D. Since AllJoyn routing necessitates a direct AD-PD link, communication over an intermediate hop is not facilitated, and PD_E continues to be disconnected from the AllJoyn PCN despite its proximity to the other three PDs. Thus, results in Figure 3 (b) and Figure 3 (c) do not change beyond the threshold of $69m$ even in Scenario 2. The concept of *extended proximity* (e-proximity) needs to be introduced in the existing AllJoyn framework. Thus, not only the mobile devices that are in the immediate proximity of a PD, but also the devices that lie within the *e-proximity* will be a part of, and benefit from their inclusion in the PCN. Apart from DiNets, enhancing the multi-hop data routing mechanism in AllJoyn will find numerous applications in the domain of PCNs and Mobile Social Networks, such as *Vehicular Communication Systems* and *Tele-medicine*.

2.1 AllJoyn Disaster Network Prototype

AllJoyn framework has been popularly used to create ad hoc proximity centric D2D applications such as *AllChat*, *Raffle*, *DIO*, and *Min-O-Mee* [3–5]. These applications serve as the technology demonstrators for the AllJoyn framework, but fail to highlight the challenges encountered in AllJoyn powered D2D communication. Some works identify these challenges apart from proposing an AllJoyn application, but do not offer a detailed description of these problems. For example, authors propose an AllJoyn PCN prototype called *DirectShare*, and state that AllJoyn is currently incapable of implementing a full-mesh topology, but do not elaborate upon it [6]. We have highlighted this issue in great detail earlier in this section.

Further, none of these works have analysed the performance of the their AllJoyn prototype in terms of observable network metrics, such as end-to-end latency, signal strength, or battery consumption. For general D2D applications such an analysis may not be necessary, but it is imperative to carry out a thorough performance analysis for a DiNet application. The primary reason is to test the feasibility of creating a proximity centric disaster network using the AllJoyn framework. We intend to test the performance of AllJoyn DiNet application in terms of several factors such as reliable data transfer in disaster scenarios, network connectivity, and battery consumption. An experimental evaluation of an AllJoyn Dinet Prototype will also offer us insights into the challenges specific to a disaster network. In addition, the observed data of important network metrics will be vital in incorporating multi-hop routing in the AllJoyn framework and providing solutions to the challenge of e-proximity.

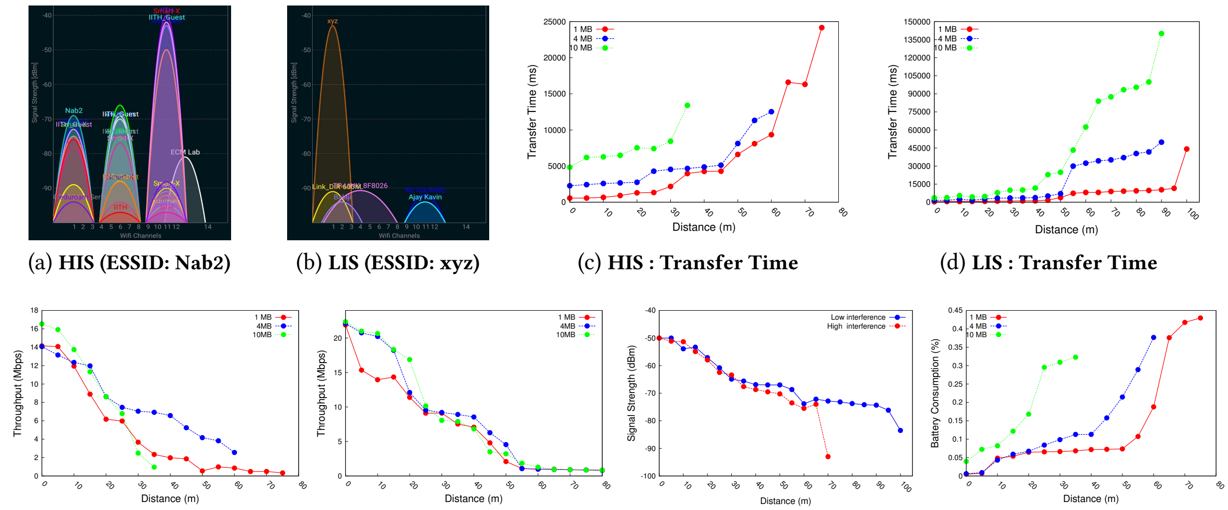
2.1.1 Field Experiment Setup We consider a two node trivial DiNet prototype, comprising of Node-A and Node-B illustrated in Figure 2. We implement a basic AllJoyn application capable of reliable file exchange between devices in proximity, which we call the *DiNetApp*. Node-A serves as AD_A, which advertises the AllJoyn session while Node-B that joins this session as the participating device PD_B.

We aim to monitor specific network metrics to identify constraints in a proximity-centric wireless disaster communication network. So we gather relevant wireless metrics of the Wi-Fi channel DiNetApp is operating on such as the IEEE 802.11 mode of operation (b/g/n), signal level, throughput, and link latency. The DiNet prototype is realized through a *Wi-fi ad hoc network* which is setup on both nodes with the help of *Independent Basic Service Set* or IBSS mode of Wi-Fi in 2.4 *Ghz* over the default Wi-Fi channel (Channel 1). The configuration of ad hoc mode is done with the help of linux utilities. The nodes are running the Ubuntu 16.04 OS, the NICs are operating on IEEE 802.11n, and have been assigned static IP addresses.

The purpose of this experiment is to demonstrate the capability of AllJoyn DiNet to provide appropriate metrics for disaster based communication. Post-disaster environment is often characterized by extreme weather conditions, obstacles created by collapsed infrastructure, and harsh terrain. These factors attenuate the signal strength, and to ensure a resilient and robust DiNet, next-hop choices should be determined by SINR and RSSI values [7]. Disaster hit areas are also more prone to *multipath fading* due to collapsed structures of which multiple copies of the same signal may bounce of and may interfere at the receiver [8]. Thus, we opine that in disaster scenarios signal strength and signal quality ought to be the primary factors in designing routing algorithms. For this reason, we consider two scenarios based on ambient wireless interference. We consider a *High Interference Scenario* (HIS) where multiple APs are operating on the same channel, and a *Low Interference Scenario* (LIS) with negligible presence of external interferers. The Extended Basic Service Set Identities (ESSIDs) of the Wi-fi ad hoc network created in HIS and LIS is "Nab2" and "xyz", and their channel graphs are presented in Figure 4 (a) and Figure 4 (b), respectively. Further, we choose a large building undergoing renovation as the site of our experiment. The presence of obstacles such as stair-wells, pillars, furniture on the pathways, *etc.,* creates some semblance of a post-disaster scenario. Also, post-disaster data will range from SoS beacons, pictures of a disaster zone, to a video of a disaster event. To address the challenge of a broad spectrum of file types and sizes, we consider three file sizes of 1MB, 4MB, and 10MB, which represent a picture, a medium resolution short video, and a high-resolution short video being transmitted in a disaster scenario.

In the experimental setup, initially both nodes are stationary. We then slowly move Node-B away from Node-A at 5m intervals. In the HIS, the experiment is repeated to account for the variation in observed values caused by dynamic channel selection in nearby APs. The network metrics observed at every 5m interval are the *Transfer Time* (TT) of each file, Signal Strength, and Throughput.

2.1.2 Results and Analysis Moving away from Node-A, we find that the signal strength gradually reduces and there is a drop in the throughput as well. The results for these two parameters are presented in Figures 4 (c), (d), (e), & (f). It can be discerned that the gradient is not constant and there are mild undulations and plateaus in the slope. This can be explained by the change/reduction in data rate that is triggered by the fall in signal strength with increase in distance between the two nodes. First and foremost observation is that in HIS, the file-transfer for 10MB and 4MB over greater distances becomes more difficult, as time-out errors (ER_TIMEOUT), and bus errors (ER_BUS_BLOCKING_CALL_NOT_ALLOWED) become more frequent. Thus, after 35m, 10MB file-transfer fails repeatedly, while 4MB transfer is reliable upto a distance of 60m. In sharp contrast, AllJoyn DiNetApp performance in LIS is quite reliable in terms of end-to-end data-delivery guarantees, as time-out or bus-errors are rare. From Figure 4 (g),



(a) **HIS (ESSID: Nab2)** (b) **LIS (ESSID: xyz)** (c) **HIS : Transfer Time** (d) **LIS : Transfer Time**

(e) **HIS : Throughput** (f) **LIS : Throughput** (g) **Signal Strength** (h) **Battery Consumption**

Figure 4: **AllJoyn DiNet Prototype : Network Performance.**

it can be inferred that Node-B is unable to join the AllJoyn session advertised by AD_A at 70m and 100m, in HIS and LIS scenarios, respectively. Further, the ratio of fields *charge_now* and *charge_full_design* in the "/sys/class/power_supply/BAT0/" file gives precise battery percentage. Power consumption in data transmission is highest for the 10MB file, and the least for 1MB file, as is evident from Figure 4 (h). In a disaster scenario, external interference is unlikely to be a major hindrance, thereby creating an LIS scenario although the signal strength will be attenuated by obstacles such as debris. The AllJoyn DiNet prototype fares well in all respects in the LIS scenario. Secondly, the data transmission by the DiNet application is more reliable for small and medium sized files, while large files may experience transmission failures. This is not a major setback, as disaster-data is generally comprised of pictures, low-resolution videos, text messages, and SoS beacons, which are files of small and medium sizes. Finally, in the LIS, the battery consumption is negligible for 1MB and 4MB files, while it is only slightly higher in the HIS.

3 Conclusions and Future Work

In this work, we have demonstrated the challenges of multi-hop routing in an AllJoyn PCN. We have introduced the concepts of *extended proximity* and implemented an AllJoyn based DiNet application. We have conducted field experiments to test the reliability and robustness of AllJoyn in two interference scenarios, and demonstrated that AllJoyn can serve as an ideal framework for the development of D2D DiNet applications. However, native support for multi-hop routing is a crucial feature lacking in AllJoyn. Motivated by the findings in this work, we will devise an optimal routing model for disaster networks.

4 Acknowledgement

This work was supported by M2Smart project, JST/JICA SATREPS, Japan.

References

[1] Teruo Higashino, Hirozumi Yamaguchi, Akihito Hiromori, Akira Uchiyama, and Takaaki Umedu. Re-thinking: Design and development of mobility aware applications in smart and connected communities. In *2018 IEEE 38th International Conference on Distributed Computing Systems (ICDCS)*, pages 1171–1179. IEEE, 2018.

[2] allseenalliance. The AllJoyn Framework, url=https://allseenalliance.org/developers/learn/architecture, 2015.

[3] Yufeng Wang, Li Wei, Qun Jin, and Jianhua Ma. Alljoyn based direct proximity service development: Overview and prototype. In *Computational Science and Engineering (CSE), 2014 IEEE 17th International Conference on*, pages 634–641. IEEE, 2014.

[4] Junseok Kwak, Jae-Hwan Jin, and Myung-Joon Lee. A mobile application for information sharing and collaboration among co-located people. *Information Technology and Computer Science*, 106(2015):17–21, 2015.

[5] Hatim Lokhandwala, Srikant Manas Kala, and Bheemarjuna Reddy Tamma. Min-o-mee: A proximity based network application leveraging the alljoyn framework. In *Computing and Network Communications (CoCoNet), 2015 International Conference on*, pages 613–619. IEEE, 2015.

[6] Yufeng Wang, Li Wei, Athanasios V Vasilakos, and Qun Jin. Device-to-device based mobile social networking in proximity (msnp) on smartphones: Framework, challenges and prototype. *Future Generation Computer Systems*, 74:241–253, 2017.

[7] Hyo Won Lee, Wha Sook Jeon, and Dong Geun Jeong. A practical indoor localization scheme for disaster relief. In *Vehicular Technology Conference (VTC Spring), 2017 IEEE 85th*, pages 1–7. IEEE, 2017.

[8] Srikant Manas Kala, M Pavan Kumar Reddy, Ranadheer Musham, and Bheemarjuna Reddy Tamma. Interference mitigation in wireless mesh networks through radio co-location aware conflict graphs. *Wireless Networks*, 22(2):679–702, 2016.