Floater: Post-disaster Communications via Floating Content

Flavien Bonvin[1], Gaetano Manzo[1,3], Christian Esposito[2], Torsten Braun[3], and Gianluca Rizzo[1]

[1] University of Applied Sciences and Arts Western Switzerland
[2] University of Napoli "Federico II", Italy
[3] University of Bern, Switzerland

Abstract. In the immediate aftermath of nature-based disasters such as earthquakes, fires, or floods, have a clear vision of the situation and the population involved is of main priority for rescue operations—it is a matter of life and death. But these disaster events may cause malfunctions in communication services making the exchange information impossible—the experienced delay sending a message in an overcrowded area is a shred of evidence.

In this demo, we introduce Floater, a mobile awareness-based communication application for the immediate aftermath of a disaster, when ad hoc infrastructure support has not been deployed yet. Floater enables communications between peers in a common area without requiring the support of a cellular network. The application is developed for Android and it does not require an account or an Internet connection. Floater exploits local knowledge and constraints opportunistic replication (peer to peer) of information to build a global view of the involved area efficiently. The app is the first to implement Floating Content, an infrastructure-less communication paradigm based on opportunistic replication of a piece of content in a geographically constrained location and for a limited amount of time.

The demo illustrates the feasibility and the main functionalities of Floater and presents disaster assistance use cases for supporting rescue operations.

1 Introduction

In the immediate aftermath of nature-based disasters such as earthquakes, fires, or floods, have a clear vision of the situation and the population involved is of main priority for rescue operations [3]—it is a matter of life and death. But these disaster events may cause malfunctions in communication services making the exchange information impossible—the experienced delay sending a message in an overcrowded area is a shred of evidence.

In this demo, we introduce Floater, a mobile awareness-based communication application for the immediate aftermath of a disaster, when ad hoc infrastructure support has not been deployed yet. Floater enables communications between peers in a common area without requiring the support of a cellular network. The application is developed for Android and it does not require an account or an Internet connection. Communication is essential for rescue operations at the occurrence of a disaster. When a catastrophic event occurs such as a flood, an earthquake or a wildfire, the exchange of information between people enables the population involved to make decisions and potentially saving lives. But disaster events may disrupt communication networks due to power cuts to prevent further damages to the power lines, to avoid fires, and to reduce the risk of cascading effects on the power distribution network [1].

Portable radio access networks, as well as point-to-point radio communications, are a way to mitigate such problems providing communication when the cellular network is unavailable. However, their deployment naturally takes a few days lowering the likelihood of surviving in such disaster events. It is crucial to facilitate information exchange in the immediate aftermath of a disaster, empowering all people on-site and first respondents with potentially vital information.

Opportunistic communications have been proposed as a mode of communication that fits particularly well the absence of infrastructure support such as in a disaster scenario [2]. The ubiquity of smartphones makes them particularly suited for creating an ad-hoc network that allows content sharing and eventually connects to the Internet in a delay-tolerant mode [4]. Smartphones provide point-to-point communications and not necessarily require infrastructure availability in the form of WiFi Access Point or Cellular Access Networks.

Opportunistic communications connect peers in a delay-tolerant mode exchanging text, video, or real-time voice interactions. With no geographical and time constraints, such a mobile ad hoc network solution does not scale jeopardizing the opportunistic communication channels. Sharing information among groups is indeed a crucial aspect of post-disaster communications, and several existing solutions address this aspect [5, 6]. However, in the majority of the proposed solutions, there is no control over the amount of information shared nor on its relevance. As a result, these approaches are unsuitable for real-disaster cases where resources (e.g., energy and bandwidth) are limited.

Floating Content (FC) is an opportunistic communications paradigm [7]. FC geographically constrains message replication implementing infrastructure-less spatial information storage within an area denoted as Anchor Zone.

In this demo, we present Floater, the prototype of a mobile application that applies the concept of Floating Content in order to implement in a distributed fashion a situation awareness service for post-disaster scenarios. Floater faces the communication difficulties of a disaster scenario enabling infrastructure-less communication between peers only equipped with smartphones. Through Floater, a tagged map of the disaster area is shared among participating peers, in order to exchange critical information. The demo shows how information is produced, categorized, and shared on the platform, and how a simple decentralized mechanism of trust allows users to evaluate the usefulness, relevance, and level of accuracy of a given piece of information.

2 Floating Content basics

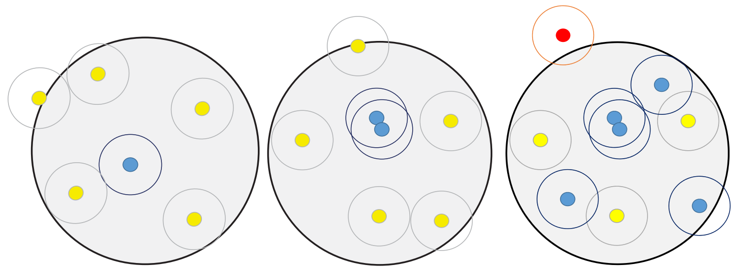
Floating Content [7] or Hovering information [8] is an infrastructure-less communication model that enables context-aware message exchange in an area denoted as Anchor Zone (AZ) and over a limited amount of time. A user interesting in sharing a piece of content, denoted as Seeder node, initializes an AZ based on the application requirement (see the blue node in Figure 1a). Within the AZ, the opportunistic exchange takes place enabling probabilistic content storing. Every time a node with the content comes in contact with a node without it within the AZ, the content is exchanged as shown in Figure 1b. The Floating Content paradigm assumes that the time taken to replicate the content is negligible with respect to contact time. Nodes entering the AZ do not possess a copy of the content, and those exiting the AZ discard their copy as shown in Figure 1c. Hence, the typical behavior of a node consists in entering the AZ with no content, receiving the content from another node, distributing the content to nodes with no content met within the AZ, leaving the AZ and dropping the content. Based on the implementation, the AZ is either static or dynamically adapted to achieve the given application performance.

As a result of such opportunistic exchange, the content *floats* (i.e., it persists probabilistically in the AZ even after the seeder(s) left the AZ) if the critical conditions in [7] are satisfied. In this way, the content is made available to nodes traversing the AZ for the whole duration of its floating lifetime without infrastructure support.

3 Situation awareness via Floater

Floater enables communications between peers in a common area without requiring the support of a cellular network. The application is developed for Android and it does not require an account or an Internet connection [9].

The Floater app enables a user to send, receive, and manage messages in the Anchor Zone —according to the Floating Content communication model. The user that sends a new message can label it as follows:



(a) (b) (c)

Fig. 1: Basic operation of Floating Content. 1a) Seeder (blue) defines the AZ. 1b) Opportunistic message exchange between nodes. 1c) Nodes going out of the AZ (red) discard the content.

- Victims, a user requiring help such as in a car breakdown situation;
- Danger, a user spotting a particular danger such as a rough road surface;
- Resource, a user localizing a new resource such as a restaurant;
- Caretaker, a user localizing health-point such as an ambulance.

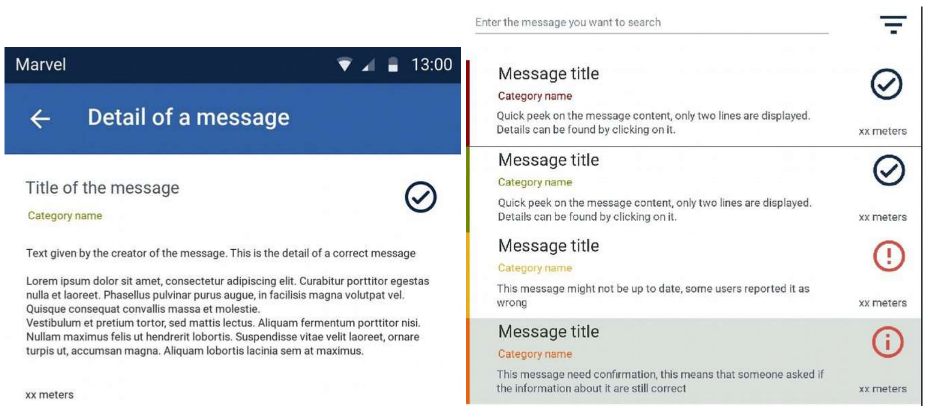
The above categories identify the main application use cases. For instance, a shop owner can advertise particular deals by floating a *Resource* message around the interested shop; a pedestrian can exchange a *Danger* message that spots a malfunction in the cross traffic light; or a skier can request help by sending a *Victim* message aftermath an avalanche.

After selecting the category as shown in Figure 2a, the user can add title, description, and pictures to the message. Floater enables seamless nearby interactions using Bluetooth, WiFi, or ultrasound network interfaces. The transmission radius is based on the message category. User can select the receiver radius in the application setting —displayed on the app map page— discarding messages far away from the actual user position.

Every user in the AZ is allowed to validate a message (as shown in Figure 2b), as a way to confirm the validity of the information carried by the message. In this way, the freshness and accuracy of the information spread via FC is checked. The app automatically eliminates a message once it has been confirmed as wrong multiple times or out of date.

The follow list summarizes the main actions enabled by the Floater App:

- send messages with a title, category, and position, plus possibly an image;
- receive every message in a configured zone; For instance, a driver can lunch Floater to receive traffic information in the local area avoiding congested roads.
- Ask about the validity of a message; The receiver can validate or not the message. The automatic mechanism implemented in Floater deletes rejected advertise messages avoiding spam and overhead.



(a) (b)

Fig. 2: Floater message category: (a) Mock-up message details. (b) Mock-up message list.

- Filter the messages by categories, distance, date;
- see the messages on a map labeled with different markers;
- see deleted messages in the bin.

We implemented Floater using Google Nearby to establish direct communication channels with other devices without having to be connected to the Internet. Google Nearby increases the possibility of exchanging information between two nodes, as it is based on the use of Bluetooth, WiFi, and ultrasounds interfaces. In addition, whenever cellular communications are available, they are used. This allows to take advantage of the surviving infrastructure available in a post-disaster scenario. Of the four communication technologies, the choice of the one used for exchanging content between two nearby nodes is managed by the Nearby interface in a way which is transparent to the App, and based on considerations on the quality of the channel (and hence also on interference levels), and on the amount of information to be exchanged.

The choice of the communication technologies used for exchanging content between two nearby nodes is managed by the Google Nearby Interface in a way which is transparent to Floater, and based on considerations on the quality of the channel (and hence also on interference levels) and on the amount of information to be exchanged. Table 1 lists the fundamental differences between Floater (implemented with Google API) and the existing solutions on the market. We can see that Floater does not have special requirements, and supports the most message types.

Authors in [10] introduce a cooperative opportunistic alert diffusion scheme for trapped survivors during disaster scenarios to support rescue operations. But they do not implement any specific opportunistic communication model leaving open the network optimization. By implement Floating Content, Floater limits

Feature		Floater	goTenna	FireChat	NearbyChat	Zello
Instant Message		✗	✓	✓	✓	✓
Require specific hardware		✗	✓	✗	✗	✗
Infrastructure-based		✗	✗	✗	✗	✓
Require the Internet		✗	✗	✗	✓	✗
Message supported	Text	✓	✓	✓	✓	✗
	Voice	✗	✗	✗	✗	✓
	Position	✓	✓	✗	✗	✗
	Images	✓	✗	✓	✓	✗
Require an account		✗	✓	✓	✓	✓
Routing type	Text	✗	✓	✓	✗	✓
	Voice	✗	✓	✓	✗	✓
	Position	✓	✓	✓	✓	✗

Table 1: Floater vs. existing communication solutions

the opportunistic communication within the Anchor Zone, preventing network overhead. Drones and vehicles can support Floater increasing the number of peers within the communication area. Authors in [11, 12] presents a vision for future unmanned aerial vehicles assisted disaster management that can be included in the Floating Content scheme.

Figure 3 shows Google Nearby connection procedure:

- Advertise and discovery procedures, respectively *StartAdvertising* and *StartDiscovery* methods, are called as soon as the application starts.
- *ConnectionLifecycle* method handles incoming connection.
- As soon as an advertiser has been found, *EndpointDiscovery* method requires a connection by sending the name of the device and an identifier. Only applications with the same identifier can connect each other —no unwanted third-party peers.
- *Payload* method fires when a new payload is received.

4 Disaster Assistance Use Case

Let us consider an earthquake scenario. Many victims are injured, stuck under ruins, and requiring urgent support. Other such as fireman, police, and volunteer provide the first aid to save as many lives as possible. In such disaster event, we identify two main roles: victims, people requesting help; and, helpers, people assisting victims. We assume that victims and helpers are equipped with a smartphone and able to lunch Floater.

Victims send messages requiring support. Figure 4a shows the message reported by a person stuck in the house with a broken leg. Floater makes available

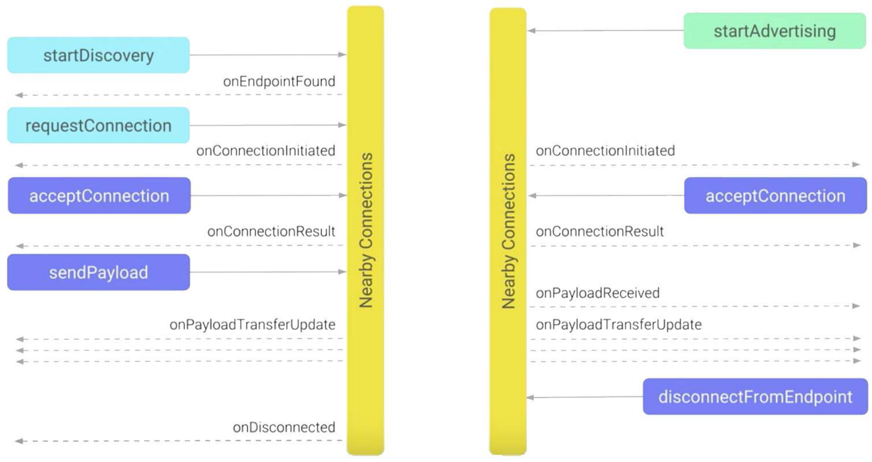


Fig. 3: Google Nearby connection procedure (Android Developers, 2017).

the Victims messages to nearby devices—helpers—through ad hoc communication.

Helpers, in the local area, consult victims messages and take action to them (e.g., reply to the Victims messages or validate them). Helpers, validate messages ensuring the freshness of the information.

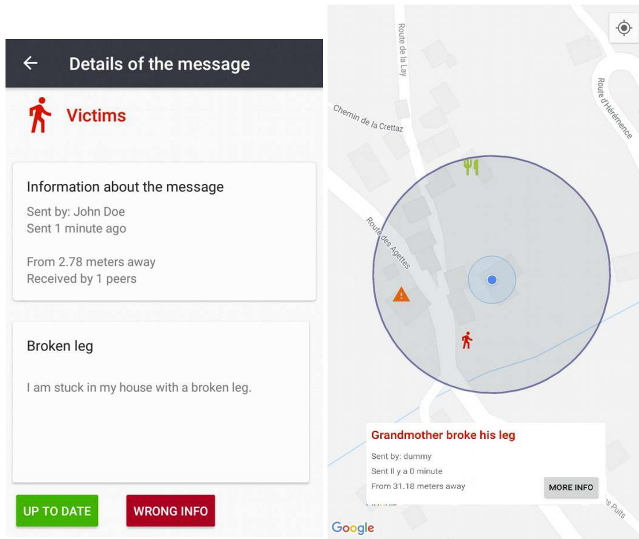
As described above, a message has a title, a description, a category, and the sender position. A message can also include images for content clarity. Floater app filters messages based on category, distance, and date. Both victims and helpers can see their position on the map, as shown in Figure 4b, where each message has a different marker. Finally, the app automatically deletes past useless messages.

5 Acknowledgements

This article is based upon work from COST Action CA15127 (Resilient communication services protecting end-user applications from disaster-based failures - RECODIS) supported by COST (European Cooperation in Science and Technology), and on RCSO MOVNET.

Bibliography

[1] O'Reilly, G., Jrad, A., Nagarajan, R., Brown, T., Conrad, S.: Critical infrastructure analysis of Telecom for natural disasters. pp. 1 – 6 (12 2006)
[2] Mart, R., Crowcroft, A., Yoneki, J., Mart, E.: Evaluating opportunistic networks in disaster scenarios. Journal of Network and Computer Applications (11 2012)



(a) (b)

Fig. 4: (a) Mock-up message details. (b) Mock-up message list.

[3] Cinque, M., Cotroneo, D., Esposito, C., Fiorentino, M., Russo, S.: A Reliable Crisis Information System to Share Data after the Event of a Large-Scale Disaster. Proceedings of the CIT/IUCC/DASC/PICom conference: 941-946, 2015.

[4] R. Schurgot, M., Comaniciu, C., Jaffres-Runser, K.: Beyond traditional DTN routing: Social networks for opportunistic communication. IEEE Communications Magazine **50** (10 2011).

[5] Gutirrez, D., Len-Coca, J., Askalani, M., Toral, S., Barrero, F., Asimakopoulou, E., Sotiriadis, S., Bessis, N.: A survey on ad hoc networks for disaster scenarios.IEEE INCoS, pp. 433–438, Mar 2015.

[6] Miaoudakis, A., Petroulakis, N., Askoxylakis, I.: Communications in emergency and crisis situations, (06 2014)

[7] Hyytiä, E., Virtamo, J., Lassila, P., Kangasharju, J., Ott, J.: When does content float? Characterizing availability of anchored information in opportunistic content sharing. IEEE INFOCOM, 2011.

[8] Castro, A.A.V., Serugendo, G.D.M., Konstantas, D.: Hovering information–self-organizing information that finds its own storage. Springer Autonomic Communication (2009)

[9] Bonvin, F.: Floater, {https://play.google.com/store/apps/floater}

[10] Mezghani, Farouk: An Android Application for Opportunistic Alert Diffusion in Disaster Scenario.IEEE PIMRC 2017 Demo Exhibits, Oct 2017.

[11] M. Erdelj and E. Natalizio and K. R. Chowdhury and I. F. Akyildiz:Help from the Sky: Leveraging UAVs for Disaster Management. IEEE Pervasive

Computing (2017)
[12] M. Erdelj and E. Natalizio:Drones, Smartphones and Sensors to Face Natural Disasters. the 4th ACM Workshop (2018)