# CRIMP: Here Crisis Mapping Goes Offline

Partha Sarathi Paul[a,*], Bishakh Chandra Ghosh[a], Hridoy Sankar Datta[a], Kingshuk De[a], Arka Prava Basu[a], Prithviraj Pramanik[a], Sujoy Saha[a], Sandip Chakraborty[b], Niloy Ganguly[b], Subrata Nandi[a]

[a]*Department of CSE, National Institute of Technology Durgapur, India*
[b]*Department of CSE, Indian Institute of Technology Kharagpur, India*

**Abstract**

Online *Crisis Mapping Systems* (CMS) are de facto tools to facilitate disaster response, although they may fail to work during the initial days of large-scale (Level 3) natural disasters, because of the unavailability of Internet due to the breakdown of existing communication infrastructures and the power-grids. In this paper, we present a novel end-to-end application system, called CRIMP, which can run both as an Android App in users' smartphones as well as an application in custom portable units, referred as Information Storage Boxes (ISB) which are designed as networking building-blocks to combat network-outage after large-scale disasters. In the absence of Internet, it leverages (a) the presence of active smartphone users as crisis mappers, (b) inherent mobility of rescue vehicles as data mules, and (c) presence of few pre-deployed ISBs in a disaster-hit locality. CRIMP creates a Wi-Fi ad-hoc network to opportunistically aggregate captured information, processes it, and finally embeds it on maps to generate semi-real-time 'Local Crisis Map' (LCM) in a decentralized way. It considers issues unique to an offline crisis mapping system and implements a channel-sensed, hands-free, scalable, ad-hoc connectivity among devices in proximity for 'role-based' seamless syncing of situational messages, addressing the 'consistency-coverage-quality trade-off'. Extensive lab-scale testing and analysis from some field-level mock drills reveal that CRIMP can speed-up the need assessment and emergency response in time-sensitive situations.

*Keywords:* Crisis Mapping, Disaster Response System, Disruption-Tolerant Network, Ad Hoc Network, Opportunistic Network

## 1. Introduction

During and after large-scale disasters, the rescue and relief stakeholders like Government officials, volunteers from humanitarian organizations, and general public who happen to be affected in the course of disaster are often keen to get the information related to disaster affected areas in a easily-perceivable format. It should be in such a way that the zone-wise information of disaster impacts on the general biodiversity as well as on human lives may be embedded on a properly-scaled map of the affected area, often known as *Crisis Map*. Immediately after a natural disaster strikes a region, such maps provides answers to queries like (a) which of the roads are still passable? (b) which patches of land can serve as helipads, or, drop zones? (c) where are the landslides? or (d) how many structurally sound buildings will aid workers fine once they arrive?

Several studies and reports like [13] by International and Public Affairs (SIPA), [44] by UNISDR of the United Nations, etc. have revealed that an effective two-way communication between the victims and the rescue/relief team involving Government and NGOs is essential for disaster risk reduction and mitigation. However, a suitable end-to-end system is still unavailable especially when a large-scale disaster strikes in some developing countries.

Internet based crowdsourced *Crisis Mapping System* (CMS) are in place from 2010 when Ushahidi [8] released *Crowdmap* harnessing crowd-fed streams of

*I am corresponding author
*Email addresses:* mtc0113@mail.com (Partha Sarathi Paul), ghoshbishakh@gmail.com (Bishakh Chandra Ghosh), hridaydutta123@gmail.com (Hridoy Sankar Datta), de.kingshuk@gmail.com (Kingshuk De), arkaprava94@gmail.com (Arka Prava Basu), prithvirajpramanik007@gmail.com (Prithviraj Pramanik), sujoy.ju@gmail.com (Sujoy Saha), sandipc@cse.iitkgp.ernet.in (Sandip Chakraborty), niloy@cse.iitkgp.ernet.in (Niloy Ganguly), subrata.nandi@gmail.com (Subrata Nandi)

information. Since then organizations like Sahana, Humanitarian OpenStreetMap (HOSM), The International Network of Crisis Mappers, Humanity Road, Crisis Commons, Google, etc. started developing CMS [46]. Rescue operators and volunteers use CMS application installed in their smartphones to collect and visualize various disaster related information in the form of text messages, images or audio/video clips. It revolutionized the way situational awareness is perceived during a crisis, whereby crisis mapping has become a de facto tool for crisis response in several scenarios like the Kenyan riots, 2010 Tsunami, Chile earthquake, 2011 Pakistan floods, Fukushima nuclear disaster, Hurricane Irene, 2015 Nepal earthquake, etc (Starbird [36], Zook et al. [47], Poiani et al. [25]). The key assumption which all such existing CMS considers is the availability of the Internet (refer Fig. 1 for a generalized architecture) through which the crisis information propagates from the end-users to the CMS database.

*Objective* – However, in a generalized large scale (Level 3 or L3) disaster scenario, wide spread communication outage is generally experienced during the first few days (Nemoto and Hamaguchi [22], Sakurai et al. [29], Shibata et al. [33]), which severely restricts the use of conventional CMS. The main objective of this work is *to develop a novel end-to-end system (CRIMP) which even in absence of Internet (a) can aggregate the crowd-sourced crisis information collected by victims and volunteers using opportunistic contacts of their mobile phones, and (b) create an offline 'Local Crisis Map' (LCM) in mobile phones to provide a visualization of the collected information in pre-cached maps.*

In LCM, the information entered by one user gets locally synchronized among other users in the region, and gives a local view of the situation over a CMS application, even when the network is *offline* (i.e. no backbone Internet connectivity is present). The information may also eventually get synchronized with the central CMS database when the backbone connectivity is restored. The basic concept behind this system is that the smart phones of the end users create an ad hoc network among themselves as well as with the infrastructure, and the crisis information is synchronized among all the users over this network, following the principles of Delay/Disruption-Tolerant Networking (DTN)[9]. From various recent studies like Legendre et al. [17], George et al. [10], Martín-Campillo et al. [19], DTN may be a savior in challenged networking situations like post-disaster communication, rural Internet, or battlefield communication, where

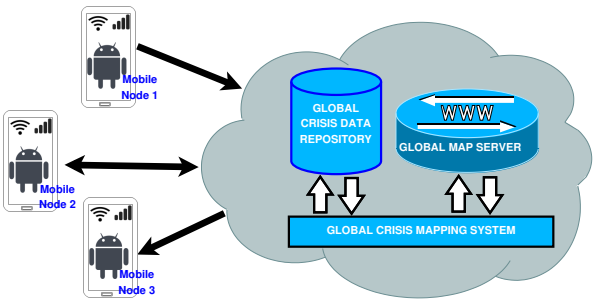



Figure 1: A generalized architecture of the state-of-the-art crisis mapping systems which assumes the availability of Internet

conventional TCP/IP networking become paralyzed due to shortage of networking resources, and intermittent, short-lived contacts among communicating nodes. Unlike TCP/IP paradigm that transfers data packets using an end-to-end node connectivity, DTN follows a *store-carry-forward* model of exchanging data packets by exploiting the physical node movements and their occasional contacts in between. Such networks are also termed as *Opportunistic Networking* as it exploits the node contact opportunities during node movements to exchange data packets.

*State-of-the-art Systems in Place & their Limitations* – There had been several network architecture/systems that has been used to establish network connectivities after a large-scale disaster (Gomes et al. [11], Miranda et al. [21]). People tried wireless mesh network (Portmann and Pirzada [26]), cellular network with disaster backup (Doumi et al. [7]), long range wireless (Saha et al. [28]), tethered balloons (Qiantori et al. [27]), vehicle mounted base stations (Jalihal et al. [15]), exploring aerial base stations (Gomez et al. [12]), exploring ad-hoc, delay-tolerant networks (Nishiyama et al. [23]) and so on. These systems aim to enable backbone connectivity in absence of conventional communicational infrastructure during a large scale disaster. However, these are neither easy-to-deploy nor cost-effective, and hence in general it is difficult propagate crowd (victims/general public) feed information. On the other hand, several software protocols/systems for peer-to-peer connectivity (like Wi-Fi Direct, WLAN-Opp [40], etc.) or peer-to-peer file synchronization (like BitTorrent Sync [30], Syncthing, etc.) that are developed to exploit contact opportunities to enable information transfer in an ad hoc setup fail to address some of the critical requirements of LCM like message transfer based on user role, message priority, etc.

*Issues & Challenges* – As a result, the following key issues and novel challenges need to be addressed to develop a LCM using an underlying Disruption-Tolerant Network.

**First,** to enable effective aggregation of information contained in messages that are time-stamped (time of creation), location-tagged (using GPS), and type-tagged (implies nature of the event) for creation of an LCM, we need to design and implement a policy involving a *consistency-coverage-quality tradeoff* of messages. Here, *consistency* implies time consistency in the sense that a set of devices in proximity must have time consistency in information they acquired. The term *coverage* refers to spatial information diversity in the sense that the devices should try to accumulate information from the entire affected area at the same time; and the same should keep an eye on the qualitative information diversity (*quality*) of the acquired messages, with priority to recent as well as some specific *important* types. The key challenge is to ensure the tradeoff given the constraint of contact opportunity.

**Second,** noting that there are devices of heterogeneous types with multiple roles during a crisis/disaster we need to develop a novel *role-based message synchronization (sync) protocol* which in contrast to conventional sync protocols will implement the above mentioned policy which considers the tradeoff. Here the '*role*' of a device need to be defined based on certain factors like the nature of the device, the quality of the messages it is carrying, its mobility profile, and so on.

**Third,** finally a suitable *role-based DTN connectivity protocol* need to be developed that enhances the existing ad hoc DTN connectivity protocols such that seamless connectivity to a device with a specific role may be preferred over other devices in proximity. Here several other factors like Wi-Fi signal strength, Wi-Fi channel load, residual battery power of the device, and so on need to be considered to make an effective energy-efficient peer-to-peer connection.

*Contribution* – In a nutshell, here we developed an end-to-end local offline **Cri**sis **M**a**pp**ing System (***CRIMP***) which can take care of (i) encapsulating input information from end-users into LCM messages considering the tradeoff in (*LCM application layer*) (details in §5), (ii) message forwarding based on user roles, 'priority' and 'importance' associated with the messages, while minimizing unnecessary message flooding during peer to peer (P2P) information syncing across LCM supported devices (*LCM P2P sync layer*) (details in §6), and (iii) the network connectivity and the device associations in DTN mode, based on application layer requirements and roles of the devices (*LCM connectivity layer*) (details in §7). Every layer needs feedback from other layers, and building up an end-to-end system considering this application specific cross layer optimization is the major contribution and novelty of this work (refer §2). The social challenges and motivation behind developing such an end-to-end system is discussed in §3.

We have deployed CRIMP on a specialized network architecture [28] with a device pool consisting of heterogeneous roles. Here, the smartphones create intermittent DTN connectivity among themselves to sync information locally. Further, to ensure that the system spans a large coverage area the architecture optionally accomodated the use of specialized devices like static *Information Storage Boxes* (ISB) as well as *Data Mules* (DM) to sync and ferry the data. CRIMP has been implemented as an Android App for the smartphones, and as a standalone service for the Raspberry Pi based devices (to support ISBs and DMs). Extensive lab-scale testing results (§8) and feedback from some field trials in Government organized mock drills (§9) reveal that CRIMP can significantly improve message syncing time, coverage of information from an affected zone and message diversity for developing a rich local crisis map, while ensures energy efficiency for the end devices.

## 2. Related Works

The existing works on post-disaster situation analysis and management have seen many-fold studies from different directions, as summarized in this section.

***Peer-to-peer Connectivity for Disaster Information Network:*** DTN based connectivity setup has been well studied in the literature, and several works have exploited that for establishing disaster information network. One may consider Miranda et al. [21] regarding a survey on the same. Shahin and Younis [32], Conti et al. [6] studied limitations of Android-based platforms regarding the usability of Wi-Fi Ad hoc and proposed frameworks for P2P networking of using Wi-Fi Direct. Trifunovic et al. [39] proposed WiFi-Opp, a simple and energy-efficient opportunistic networking set-up based on smartphones' existing communication features and APIs, by using the open stationary access points (APs) in the proximity and allowing some smartphones to create hotspots spontaneously to serve nearby devices. The authors find their WiFi-Opp to perform ten times more energy-efficiently than Wi-Fi Ad-Hoc for comparable dissemination performance. In a succeeding work of the above, Trifunovic et al. [40] extended WiFi-Opp to WLAN-Opp. Through a simulation based

study by replaying real contact traces in this work the authors find that WLAN-Opp could utilize up to 80% of the contact time while saving up to 90% of the energy that Wi-Fi ad-hoc would consume. Ólafur Helgason et al. [48] presented *Fram*, which is a middleware architecture to allow applications on mobile devices to share contents. Turkes et al. [42] proposed *Cocoon*, a lightweight middle-ware for smart mobile platforms to support opportunistic communication, where the connectivity service operates on top of the universal Wi-Fi and Bluetooth interfaces without violating their physical and MAC layer standards. However, such connectivity setup and device association protocols do not consider message priority and importance, and therefore may result in sub-optimal message delivery due to contention from devices generating low priority messages.

***Peer-to-peer Sync:*** *rsync* by Tridgell and Mackerras [38] is the first-of-the-kind P2P file sync algorithm to update and synchronize files among different devices assuming every pair of them are connected by a low-bandwidth high-latency bi-directional communication link. Basic limitation of rsync is that the algorithm works best when files are similar; for dissimilar files, which is very likely to occur in case of multimodal file exchange, its performance is not found satisfactory. An interesting P2P sync application for devices of heterogeneous platform and genre is *BitTorrent Sync*. Scanlon et al. [30] discussed various aesthetic issues and potential challenges of the peer-to-peer file sharing systems like *BitTorrent Sync*. Lareida et al. [16] presented *Box2Box*, a new P2P file synchronization application which supports novel features not present in BitTorrent Sync. Paul et al. [24] proposed a reliable and scalable P2P sync application for DTN. Lindblom et al. [18] proposes a distributed application using Named Data Networking, which can synchronize user files across devices in P2P fashion. Another interesting application in this category is *Syncthing*[1]. Senftleben et al. [31] proposed a decentralized privacy-preserving micro-blogging infrastructure like Twitter which is based on a distributed P2P network of mobile users. The proposed solution relies on a secure distribution of encrypted messages over local radio links to physically close peering nodes. There exists conventional Wi-Fi enabled Android Apps used for sharing data when mobile phones are in proximity. Shahin and Younis [32] Turkes et al. [42] Turkes et al. [43] cover some of the above type of applications. Arnaboldi et al. [1] presented a middleware platform called *CAMEO* to collect and share multidimensional context information from physical as well as virtual worlds using personal mobile devices. *CAMEO* allows mobile phones that occasionally meet at various physical locations to automatically discover users' common interests, available services, and resources through opportunistic communications in order to create a form of mobile social networks (MSN) via opportunistic networking. However, there are certain features and design issues such as the application specific cross layer optimization, which are unique in generating an LCM. Due to intermittent and often short-lived contact intervals, a node cannot share anything or everything to all its neighbors.

***Disaster Management Services:*** Various attributes of CMS have been discussed in related literatures, such as Middleton et al. [20], Birregah et al. [3], and the references therein. One relevant application in this connection is person-finding. An interesting work in this field may be obtained from Stiegler et al. [37]. Relief resource management may be another important issue after a disaster; people often need to know the location for getting relief services. Chou and Zahedi [5] discussed issues involved with the above. Another critical problem after any large-scale disaster is the destruction/blockade of existing pathways, which may change the road networks of the affected zone. Trono et al. [41] and Silva et al. [34] proposed strategies to create pathway maps for the affected region through sharing of mobility trails of the victims and rescue personnel, however, they rely on backbone Internet connectivity. Solmaz and Turgut [35] proposed and developed a method for tracking pedestrians and emergent events during disasters by opportunistic ad hoc communication between smart-phones of pedestrians and a limited number of mobile sinks. Another interesting direction in this connection is the use of smartphone based DTN in various geospatial applications. Bhattacharjee et al. [2] has proposed a post-disaster resource management scheme using smartphone based DTN. Boldrini et al. [4] proposed a Context- and social-aware DTN middleware that autonomically learns context and social information on the users, and uses the same to predict users' future movements.

***Limitations and Open Research Areas:*** It is revealed from the study so far that existing crisis mapping services neither consider the absence of Internet connectivity into account nor provide any mechanism for alternate connectivity to be used in such situations. Although there exists a number of emergency communication proposals in the literature, none of them could be plugged readily with the existing crisis mapping applications to enhance the capacity of crisis mapping in the

[1] https://bit.ly/2EG36Bx (last accessed: July 31, 2017)

local and offline mode, when backbone Internet connectivity is absent.

## 3. Motivational Field Study & Gap Analysis

As we shall observe in section 2, there exist a multitude of solutions that could serve the purpose if Internet is present; but they fail to function if Internet goes down. To best of our knowledge, very few works in the domain throws light in this side of the problem, and is our prime focus in the present work.

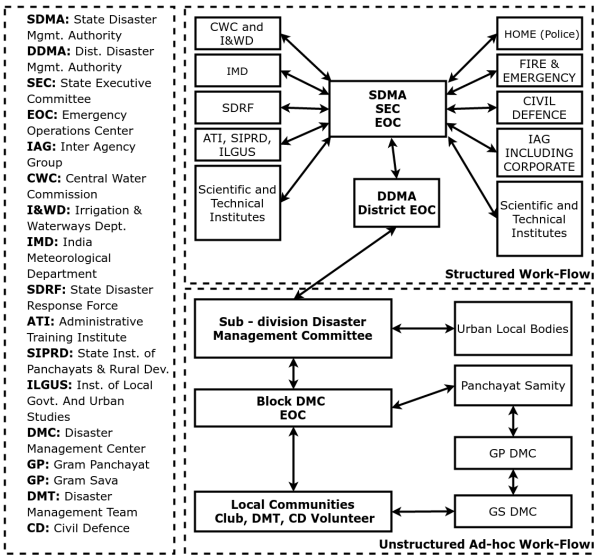



Figure 2: Typical organization chart of a state disaster management department

### 3.1. Disaster Management in Developing and Underdeveloped Countries

To ensure smooth dissemination of aforesaid crisis information even after a large-scale disaster, an integrated emergency information management system for collecting, processing and disseminating in-situ information is highly warranted for "enhancing disaster preparedness for effective response and to Build Back Better in recovery, rehabilitation and reconstruction" [45]. The desired system should gather information through participatory process; should be tailored for the needs of users at varying level, including social and cultural requirements; and needs to ensure that it remain effective and operational during and after disasters in order to provide live-saving and essential services. Since, disruption of conventional communication is a very common phenomena after a large-scale disaster (due to destruction, power cuts, etc.), the system needs to operate seamlessly in presence as well as in absence of conventional communication systems.

As part of field survey and ground-truth collection, we come in contact of various government and non-government voluntary bodies actively connected with such activities in our state. Feedback from the personnels of those bodies and other allied sources compel us to believe that a typical organization structure of a state disaster management department of an arbitrary state in India is like Figure 2. A close look at components in the chart, and our physical interaction during our field survey, made us to believe that the work-flow in top-order (state and district level authorities) of the organizations is quite structured and streamlined; whereas those who are at the level of sub-division and rural administration follow quite an unstructured and ad-hoc form of work-flow. For sub-divisions/villages that are situated in the remotest part (like terrains, coastal area, etc.) are often found to be excluded from the mainstream network infrastructure. When a large-scale disaster strikes such a region, the information related to damage and rescue-relief needs reach the district/state head-quarters after a long delay.

### 3.2. Critical Service Needs

In view of all the above we identify the following critical services that may be beneficial to the stakeholders after the occurrence of a natural disaster.

- Setting up an alternative communication infrastructure for post-disaster situation that may rapidly be deployed with portable and readily-available devices.
- Setting up a crisis mapping service that would function on top of the above alternate network infrastructure following a cooperative map-data sharing model to provide route information for the neighborhood of the affected region.
- Setting up an infrastructure to provide an alternative road network from the movement trajectory of the participating users that would be overlaid on the map for the guiding the the other users about the sustained pathway network.
- Setting up an information management scheme that may be applied to provide the maximum utilization of the networking resources of the networking infrastructure for users from every nook-and-corner of the affected region, so that more important messages get through earlier than the less important ones, and the every user get a fair chance to utilize the networking resources.

The proposed system CRIMP is designed to mitigate some of the gaps that we identified during our field and literature studies.

## 4. CRIMP: Design Issues & Architecture

This section gives the system and network architecture on top of which CRIMP works. We also mention various design issues that have been addressed in the design of CRIMP.

### *4.1. System and Network Architecture for Situational Information Propagation*

An L3 disaster often creates isolated localities (e.g. Loc1, Loc2 & Loc3 in Fig. 3), where victims are trapped or sheltered, restricting the movement of local volunteers within a locality. Vehicles, ferries or helicopters for rescue and first-aid generally move across such localities. Accordingly, CRIMP uses three different types of devices as shown in Fig. 3 – (i) *DTN nodes* – smartphones of the end-users that create local network based on DTN connectivity, (ii) static *Information Storage Boxes* (ISB) – customized battery-powered storage devices with WiFi connectivity, which are placed at strategic locations like the shelter points to aggregate and offload messages from DTN nodes, (iii) *Data Mules* (DM) – ISB mounted on the vehicles or helicopters, etc., that can ferry messages across static ISBs and transfer them to the backbone network via conventional Internet, whenever or wherever available. CRIMP services are developed to automatically sync the messages among these three types of devices and to populate the LCM at the static ISBs and smartphones of the end-users.

The end-users provide information as an input to the CRIMP application as crisis mappers. Messages generated by crisis mappers are considered as the primary data units of CRIMP, and are used to generate LCM through 'aggregation'. Due to distributed nature of the underlying network, and the non-availability of centralized server, the data aggregation is performed in participating nodes in local manner with the objective of finding 'consistent' information snapshots with decent 'spatial coverage' at the control stations in regular intervals. To achieve this, 'important' messages are picked and dispatched by CRIMP interface for better aggregation. A message in our case is important when: (a) It is fairly recent; (b) It reports rarer event; and (c) It talks of a less-reported region. Since, an user node in such a distributed network is unlikely to know the global situation, CRIMP in participating nodes approximates the global situation locally by exchanging their local knowledge of the global situation in the form of 'type-wise' and 'location-wise' distribution of messages, and their 'importance' values. A node estimates its global situational view by aggregating its own knowledge with the received knowledge from peers, and update its local knowledge at regular intervals. With a proper definition of 'importance' of messages to have a decent situational view at the control station form these shared local knowledges is serious design challenge.

The situational information supplied by the end users gets synced among the nearby smartphones, static ISBs and DMs within wireless coverage. This forms the LCM at those smartphones and ISB. When a DM or end-user moves from one locality to another, the LCMs from different localities get synced, aggregated and updated.

### *4.2. LCM in Offline Mode: The Design Requirements*

The two major design requirements, that CRIMP addresses, for generating LCMs in offline mode are as follows.

***The 'role' of a device:*** Three types of devices, DTN nodes, static ISBs and mobile DMs, are considered in our system. The role of a device and accordingly its priority vary depending on its mobility, battery, processing power, etc. Smartphone users are considered to be 'crisis mappers'. ISBs have much better battery backup and processing power compared to smartphones. Hence crisis data is preferably aggregated at the static ISBs (if available) which also hosts the local map server. Within a group of smartphones in the proximity, the one with the higher mobility (for example, smartphones carried by volunteers) can be given higher priority in transferring data. Further, DMs are given the highest priority to exploit their larger mobility pattern. Crisis data thus flows from phones to DMs via ISBs and vice verse. Unlike conventional peer-to-peer mobile file syncing system, the role of the devices is a key design issue in CRIMP, for both in establishing peer connectivity and syncing of information among devices.

***Consistency-Coverage-Quality trade-off of crisis data:*** CRIMP enables a crisis mapper to capture time-stamped (time of creation), location-tagged (using GPS), and type-tagged (implies nature of the event; medical & health-care, relief & food-stock, shelter & warehouse, damage & death-toll etc.) data. The messages received by a particular device at any point of time vary widely in terms of their creation-time, creation-location and type due to several factors like opportunistic forwarding, non-uniform distribution of mappers, mobility patterns, human factors, etc. Given a finite short duration during device connectivity in ad hoc mode, it is tricky to decide whether to give priority to recent messages (time consistency), or to messages from a locality with less information (spatial information diversity), or messages of less known types (qualitative

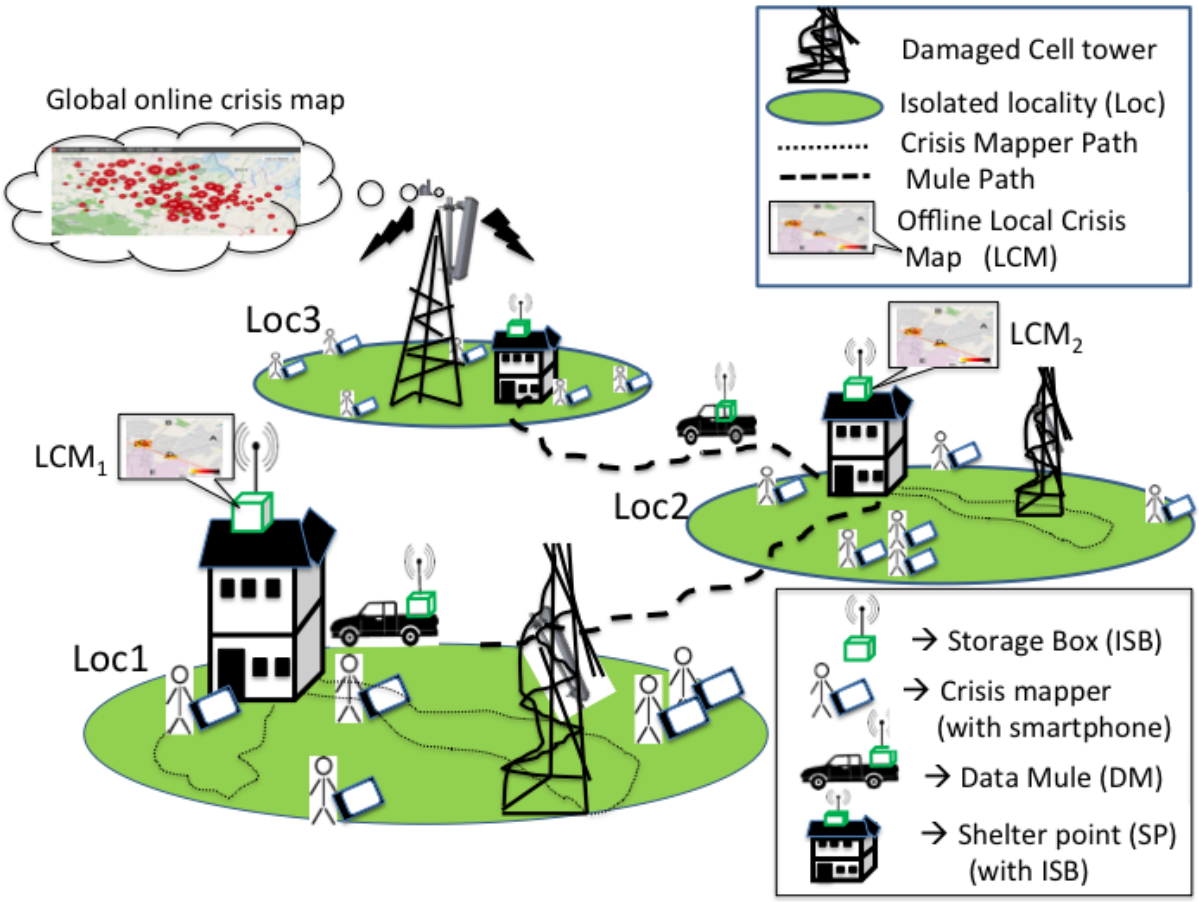



Figure 3: Network and message forwarding architecture for CRIMP

information diversity). To address these inherent trade-offs, CRIMP proposes a strategy to assign an *'Importance'* to each message. The underlying sync protocol forwards messages according to their importance.

### *4.3. Aggregation of Crisis Data in CRIMP*

We have observed that crisis mappers are the end users of CRIMP who supply the lowest level crisis data units to the system using the DTN nodes in the system. These data units are the pieces of the localized crisis data that they encounter in course of their movement through the affected area. These pieces of data are propagated through various nodes of CRIMP as they encounter other nodes (other DTN nodes, ISBs, DMs) in due time. On the other hand, ISBs and DMs do not collect crisis data by themselves. They store (for ISB) and carry the crisis data across nodes (for DMs). The underlying data flow in CRIMP is portrayed in Figure 4, where we have pointed out how data flows from one node to other in CRIMP during various node encounters.

At any point in time, any node in the CRIMP possesses crisis data from various sources including the ones it has collected by itself (for DTN nodes). The LCM in any node are generated by aggregating automatically the pieces of crisis data possessed by a node at an instance. For user convenience, we run an automatic aggregation routine in nodes of CRIMP at regular intervals (of an hour, say), where we simply plot all the data units like geographical shapes (for marking a location of an event), message texts, images, etc. on a map interface of the region of interest based on the associated geographical tag of the data units. This is possible since every data in CRIMP are geographically tagged. Messages of similar nature are aggregated if they are found on the same map tile (of the current zoom level) based on their geographic coordinate tag. Unlike online

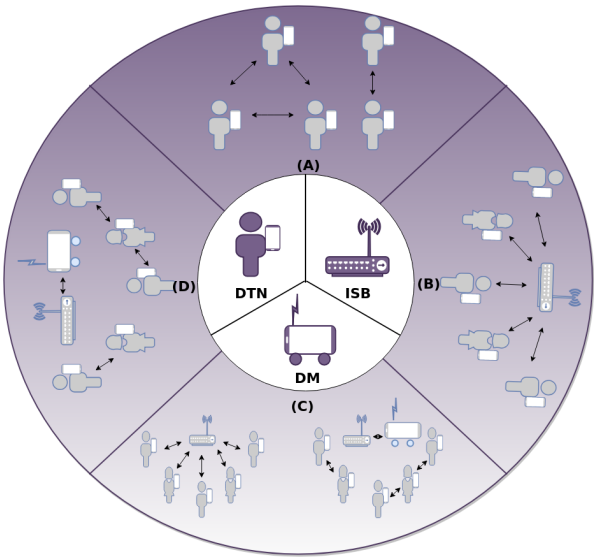



Figure 4: Data flow Scenarios in CRIMP: (A) Multiple DTN nodes meet with no ISB/DM in proximity, (B) Multiple DTN nodes meet with an ISB in proximity, (C) Multiple DTN nodes meet with an ISB, and a DM is in the proximity of another ISB, (D) DM appears in proximity of the current ISB.

mapping systems, such an automatic data aggregation is very much desirable, as the existence human moderator in remote locations during a crisis situation cannot be assumed in practice. On aggregation, each node in CRIMP has its own LCM based on the crisis data volume it possesses. Clearly, LCM in one node may differ from LCM of the other, if they differ in their data possession. However, their LCMs get synced during an encounter if they exchange their data possession in full with each other.

Since ISBs and DMs possess higher storage, computing, and energy resources than DTN nodes, they take additional responsibilities in connection of data aggregation. One major challenge in the current scenario is providing the crisis mappers with the map tiles of the affected area, as we are assuming that the affected region may be out of Internet connectivities. Since ISBs and DMs are likely to be deployed in advance in the affected area, they are preloaded with map tiles of a region covering the affected area. The map tiles get synced with DTN nodes when crisis mappers eventually come across ISBs/DMs in course of their movement.

### *4.4. Software System Architecture*

The functionality of the entire system is handled by a three-layer software suite designed on top of TCP/IP, as shown in Table 1. In a bottom-up fashion, CRIMP implements the connectivity layer (CRIMPCon), P2P syncing layer (CRIMPSync) followed by the crisis map layer (CRIMPApp).

Table 1: A three layer software suite for CRIMP

| Layer | Proposed Solution | Solution Alternatives | Desired Features |
|---|---|---|---|
| CMS | CRIMPApp | Not Available | • Local Crisis Mapping<br>• Crisis data ranked on importance of data |
| P2P Sync | CRIMPSync | • BitTorrent Sync [30]<br>• Syncthing<br>• rsync [38] | • Scalability<br>• Priority and role aware sync<br>• Robustness |
| Connectivity | CRIMPCon | • Wi-Fi Ad hoc<br>• Wi-Fi Direct [32]<br>• WiFi-Opp [39] | • Prompt, role-aware connection establishment<br>• Peer selection based on received signal strength<br>• Energy aware design |

The key idea of segregating connectivity from syncing comes from the following facts. To ensure effective data transfer during connectivity, a node should prefer a peer with better signal strength, residual battery and channel constraints. The CRIMPCon layer handles these issues. However, while syncing data, a node prefers few connected peers over others based on their role and importance of the stored content. The CRIMPSync layer handles these issues. The map layer only deals with application/service level demands which generally are independent of the network layer issues, as handled by the CRIMPApp layer.

## 5. CRIMPApp: Smartphone App for LCM

CRIMPApp is the user interface of CRIMP, through which an end-user can enter situational messages or can navigate through the LCM. CRIMPApp encapsulates the input information into geo-tagged, time-stamped and type-tagged messages, and set message importance. These messages are forwarded to the next layer of CRIMP, CRIMPSync.

Message importance is used in CRIMP to specify the time consistency, spatial information diversity and qualitative information diversity. To capture spatial information diversity, we utilize the map tiles of the `OpenStreetMap` mapping system. Any given region from a geospatial surface can be embedded completely on a map tile of a particular zoom level $z$, depending on the surface area of the region. The corresponding tile may be divided recursively into $2 \times 2$ grids of map tiles of higher zoom level till the maximum zoom level is reached. For spatial embedding of a geo-tagged message within a map tile given by the coordinates $(x, y)$ in the Cartesian coordinate system, we express

$(x, y)$ as a function of the geo-location $(lat, lon)$ (latitude and longitude of the location) of the message and the zoom level $z$ of the current map view. In CRIMPApp, we express $(x, y)$ following the technique described in Slippy Map[2] application, which finds the values of $x$ and $y$ using the formulas: $x = \lfloor \frac{lon+180}{360} \times 2^z \rfloor$, $y = \lfloor (1 - \frac{ln(tan(lat \times \frac{\pi}{180}) + sec(lat \times \frac{\pi}{180}))}{\pi}) \times 2^{z-1} \rfloor$.

CRIMPApp divides the LCM into LCM tiles of fixed size, and let there are $\nu$ numbers of such LCM tiles in a smartphone. We consider that there are $\mu$ different types of information that can be embedded on the LCM tiles. The obtained tile coordinates $(x, y)$ is used to count the number of messages embedded on an LCM tile. The number of messages includes both the forwarded messages from other devices as well as the messages generated in this device. Let $\Delta_i^j$ denote the number of messages tagged over LCM tile $S_i$ and are of message type $T_j$, where $i = 1 \cdots \nu$, $j = 1 \cdots \mu$. Then, we approximate the probability of occurrence of a message of type $\mathbb{T}_j$ from sLCM tile $\mathbb{S}_i$ is $\mathscr{P}_i^j = \frac{\Delta_i^j}{\Sigma_i \Sigma_j \Delta_i^j}$. We finally compute the importance of a message ($Imp(\mathcal{M})$) as $Imp(\mathcal{M}) = -log(\mathscr{P}_i^j) \times exp - (t - t_0)$, where $t$ refers to the current time, and $t_0$ refers to the time when the message was first generated (obtained from the message timestamp). It can be noted that the importance value has an aging factor that helps in ensuring the time consistency of the system as well, apart from spatial information diversity and qualitative information diversity.

This importance value is used in CRIMPSync to sync the messages across various devices, as discussed in the next section. CRIMPApp uses the messages to populate LCM locally under offline mode, which the end-users can utilize for disaster recovery planning.

## 6. CrimpSync: A Peer-to-Peer Sync for DTN Systems

Upon receiving messages from CRIMPApp, it is the responsibility of the next layer of CRIMP, called CRIMPSync, to sync the messages and other relevant data with peer nodes. Based on the available connection setup, CRIMPSync utilizes a P2P file synchronization protocol for maximum information transfer among the various associated devices involved in disaster information collection. The salient features of CRIMPSync are as follows.

1. CRIMPSync uses message importance to sync information in such a way that consistency-coverage-quality trade-off can be ensured.

2. Because of intermittent and ad hoc connectivity, CRIMPSync supports partial message transfer so that even the partial information can be utilized to populate the LCM. If the LCM over a device already has a part of a message, then CRIMPSync supports syncing the rest of the message, when a device with that message gets connected next time. Because of this reason, CRIMPSync uses byte streaming rather than packet transfer to forward a message.

3. CRIMPSync supports two types of message syncing – broadcast syncing (messages for all LCM supported devices) and targeted syncing (messages for a single device or a group of devices – say, information for volunteers).

4. File replication is used to increase the probability of message transfer for targeted syncing under DTN scenario. However, too many replication of the same message may result in network resource underutilization. Therefore, CRIMPSync introduces a concept of *restricted epidemic* routing for information dissemination.

Fig. 5 shows how various modules of CRIMPSync interact with each other to sync data among themselves. The major modules of CRIMPSync are as follows.

**Device Priority Listing:** CRIMPSync assigns a 11 character device identifier to every device, where the first character determines the role of a device (DTN node, ISB or DM). From the broadcast packets received from its peer devices, CRIMPSync generates a priority list of the connected peers by observing their roles.

**Peer Discovery:** CRIMPSync broadcasts a device information packet, consisting of its own IP address and a 11 character long node identifier, to the devices which are associated through CRIMPCon. This broadcast packet is used to determine peer information.

**Summery Vector Exchange:** Each peer maintains a file table, called the *summary vector*, that contains the following fields for each message present in the device: message ID (256 bytes), message size (4 bytes), end byte number (4 bytes), message time-stamp (32 bytes), TTL of the file (4 bytes), message destination for targeted syncing (4 bytes), flag indicating whether the message has been reached the destination for targeted syncing, and importance of the message (8 bytes). The end byte number ($B_e$) helps in partial message syncing, where a device has the first $B_e$ bytes of a message, and need to sync the message starting from byte $B_e + 1$. The typical size of a single entry in the summary vector

[2]https://bit.ly/2JF2qQx (last accessed: 30 July, 2017)

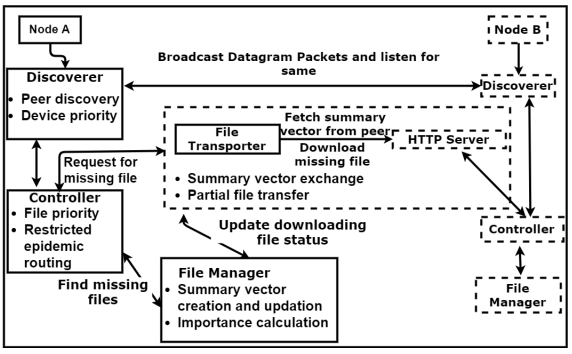



Figure 5: CRIMPSync Peer Interaction: Solid boxes represent transmitting peers, dotted boxes represent receiving peers

is less than 312 bytes. After discovering the peer devices, every device shares the above summary vector to all other peer devices. From the summary vectors, the devices determines which messages (complete or partial) need to be synced.

**Message Ranking:** CRIMPSync ranks all the messages (complete or partial) that need to be synced, based on their importance as tagged by CRIMPApp.

**Sequential Byte Forwarding:** CRIMPSync uses byte streaming to transfer a message from one peer to another. Based on message raking, CRIMPSync transfers bytes of the message from one peer to another in a sequential manner. For partial message transfer, the peer knows that the other peer already has the message up to $B_e$ bytes (as obtained from summary vector), and so starts forwarding the message from $B_e + 1$ byte until the message gets ended, or the connection gets terminated.

**Restricted Epidemic Routing:** To restrict flooding of messages that already find their destinations for targeted syncing, CRIMPSync in the destination device issues an acknowledgment back to the network by setting the 'flag' to true. Whenever a device finds the status flag for a message (for targeted syncing) in the received summary vector to be true, it stops forwarding the message further and sets the flag as true corresponding to that message in its own summary table.

## 7. CRIMPCon: Support for Opportunistic Wireless Connectivity

CRIMPcon provides opportunistic connectivity among different types of devices, depending on their roles and energy budgets. It supports a role based opportunistic and dynamic Wi-Fi infrastructure setup to balance the connectivity establishment and power consumption by individual devices. As ISBs are static devices deployed at strategic locations, they have better power backup compared to others; and so, we set them as Wi-Fi access points (APs). DMs exchange data with the ISBs only, and so we set them in Wi-Fi station (STA) mode. However, the DTN devices are mostly mobile and resource constrained, whereas they need to sync data among themselves as well as with the ISBs. So, to setup wireless connectivity among themselves, CRIMPCon opportunistically creates local Wi-Fi hotspots whenever required, based on the power availability of the devices.

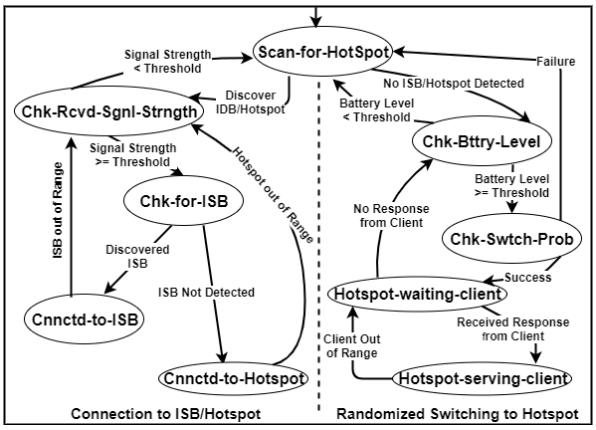



Figure 6: State transition diagram for CRIMPCon

Table 2: States of CRIMPCon state transition diagram

| State | Description |
|---|---|
| Scan-for-Hotspot | Node scans for ISB/Hotspot in proximity |
| Chk-Rcvd-Sgnl-Strngth | Node checks for received signal strength from ISB/Hotspot |
| Chk-for-ISB | Node checks whether nearby hotspot is an ISB |
| Cnnctd-to-ISB | Node is connected to an ISB |
| Cnnctd-to-Hotspot | Node is connected to a mobile hotspot |
| Chk-Bttry-Level | Node, anticipating a randomized switch, checks for its current battery level |
| Chk-Swtch-Prob | Node generates a random number in [0,1], and checks whether it below Swiching Probability |
| Hotspot-waiting-client | Node turns into a mobile hotspot, and keeps waiting for a connection request |
| Hotspot-serving-client | Node, as a mobile hotspot, is connected to a client node |

In CRIMPCon, a DTN device can set itself as a Wi-Fi AP, whenever required. We call this type of APs as *Soft AP*. The working principle of CRIMPCon is illustrated in Fig. 6 as a state transition diagram. A DTN device first checks whether there exists a Wi-Fi AP (ISB AP or Soft AP) in the vicinity (`Scan-for-HotSpot`), which has signal strength over

the minimum required signal strength for good connectivity (`Chk-Rcvd-Sgnl-Strngth`). If there is one, the device sets itself in the STA mode. It can be noted here that the ISB AP has better power backup compared to a Soft AP. Therefore, if there are multiple APs in the vicinity of a STA, it gives priority to the ISB AP compared to the Soft AP (`Chk-for-ISB` for checking ISB AP in the vicinity, `Cnnctd-to-ISB` to connect to ISB AP, and `Cnnctd-to-Hotspot` to connect to Soft AP).

When there are no APs in the vicinity of a DTN device, it can make itself a Soft AP if its battery level is more than a threshold (`Chk-Bttry-Level`). However, to avoid multiple DTN devices turning into Soft APs simultaneously, we use a probabilistic approach, where a DTN device with sufficient battery backup becomes a soft AP with probability $p$ (`Chk-Swtch-Prob`). The Soft AP works with a passive scanning process, where it first checks for all the available wireless channels for ongoing data communications, and then broadcast the beacons to a channel with the minimum load. To save the battery power, a DTN device switches from Soft AP mode to the STA mode, if there is no Wi-Fi STA that gets associated to the AP within a timeout value (`Hotspot-waiting-client`). If one or more STAs get associated with the AP, the devices (both the AP and the STAs) sync data among themselves over the Wi-Fi connection (`Hotspot-serving-client`). The probability $p$ and the thresholds mentioned in this description are taken as design parameters of CRIMPCon.

In a nutshell, CRIMPCon has the following advantages over conventional wireless connectivities,

1. It supports role based connectivity, where the decision of switching from STA mode to AP mode and vice-versa depends on the available energy backup of the devices. This saves average energy consumed across all the energy-constrained DTN devices.

2. The Soft APs work in passive scanning mode over the minimum loaded Wi-Fi channel in the vicinity, and so the connection establishment is fast compared to the Wi-Fi ad hoc or Wi-Fi direct mode.

## 8. Campus Scale Evaluation of CRIMP

The design, development, implementation and testing of CRIMP spanned for more than one and half years from October 2015 to July 2017, out of which, we have done test deployments and evaluation of various modules of CRIMP for the last six months. The deployment and testing have been done at our institute campus, as well as during the Government mock drills under real disaster testing environments. This section analyses CRIMP from the campus scale evaluation, and the results from the Government mock drills have been discussed in the next section.

### *8.1. Campus-Scale Experiment Scenarios*

We carry out three sets of experiments by deploying CRIMP based disaster information management system at our institute campus, as shown in Fig. 8.

**Single ISB scenario:** In the first two scenarios (Fig. 8a and Fig. 8b), 9 DTN nodes (smartphones) [refer table 3for specification] are initially placed at the points marked as A to D in a random manner, and then they start moving from their initial locations following the paths shown by respective user trails. One ISB is kept static at point E. During the initial random placement, at least one DTN node has been placed in every point A to D. These experiments precisely resemble a miniature scale mimic of user movements around a shelter point (E).

The smartphones are preloaded with collections of synthetically-generated situational messages. The total data volume at each location is $1.5 - 2.0$ GB consisting of images, videos and text messages of varying importance values, and are distributed at random to DTN nodes of randomly assigned that location. In all the scenarios, the preloaded messages in a smartphone are set to have geo-locations spanning at the neighborhood of its starting position (refer Fig. 8). They also have a uniformly distributed message creation time within a range of $\sim 30$ hours (prior to the start of the experiments) and a non-uniformly distributed message types. ISB has no preloaded messages, and it receives messages from the devices visiting it as part of the experiment. During the experiments, the users do not collect messages for the sake of uniformity in scenarios. The distribution of messages with their importance values for locations A to D are shown in Table 4.

Table 3: Field Trial 1: Detailed Device List

| Role | #Devices | Device Specifications |
|---|---|---|
| DTN | 9 | OnePlus 2 (1), Gionee F103 (1), Xiaomi Redmi Note 4 (1), Samsung Galaxy Note5 (1), Samsung I9300 Galaxy S III (1), Xiaomi Redmi Note 3 (1), OnePlus 3 (3) |
| ISB | 3 | Custom units (refer table in Fig. 12 for technical specs.) |
| DM | 1 | Smartphone (OnePlus 3) |

**Multiple ISBs scenario:** The third scenario, as illustrated in Fig. 8c, mimics a relatively larger-scale scenario with multiple shelter points with ISBs placed at these points, mobile users mostly stay near their home SP, occasionally visit other SPs, and DMs connect one

Table 4: Message Distribution in smartphones at locations A – D at the beginning of Trial Scenarios 1 - 2

| Importance Range | Total | A | B | C | D |
|---|---|---|---|---|---|
| 2.67059958 - 3.54255305 | 3 | 2 | 0 | 1 | 0 |
| 1.79864611 - 2.67059958 | 9 | 1 | 1 | 3 | 4 |
| 0.92669264 - 1.79864611 | 52 | 14 | 6 | 10 | 22 |
| 0.05473917 - 0.92669264 | 809 | 244 | 82 | 204 | 279 |

ISB with the other. In this scenario, we have 3 ISBs as shown in Fig. 8c, designated ISB1, ISB2 and CMR (abbreviation for *Central Message Repository*). Movement of mobile user around their home SPs forms three virtual clusters C1, C2 and C3, with three mobile users per cluster. The DM is set to move following the path indicated by the arrows in the figure. The detailed specification of devices is given in table 3. Each of the smartphones are preloaded with synthetic data as in the previous scenarios. As before, ISBs and DM do not carry any preloaded data.

The trial begins at time $t = 0$ when mobile users turn their CRIMP on, and starts a random Brownian motion in the neighborhood of their home SP with occasional visits to other SPs. DM turns its CRIMP on at time $t = 10$ minutes in the proximity of ISB1, and remain there for $\sim$ 5 minutes. It then follows the trajectory in Fig. 8c, and waits for $\sim$ 5 minutes near every ISB it visits along it movement. The trial ends when DM visits CMR for the second time, finishes its quota of $\sim$ 5 minutes, and turns of its CRIMP.

These experiments have been executed for at least 20 times at different times of the day, with a different set of volunteers. Used smartphones are of different makes (both low end and high-end smartphones, prices ranging from $200 to $500). We have used the average results for the performance evaluation. The different modules of CRIMP has also been evaluated separately and the performance has been compared with various baselines. We first analyze the individual modules of CRIMP, and then evaluate CRIMP as a whole for propagating disaster information services.

### 8.2. Performance Evaluation: CRIMPCon

We first evaluate CRIMPCon and compare its performance with WiFi-Opp [39]. We have used two variants of WiFi-Opp – one with the default IEEE 802.11 power save mode, and another without the power save mode. It can be noted that we do not use power save mode in CRIMP to fully utilize the device contact opportunities under crisis scenarios. Fig. 7a shows the average connection duration of a DTN device and the average message volume received by a device (inset) during the

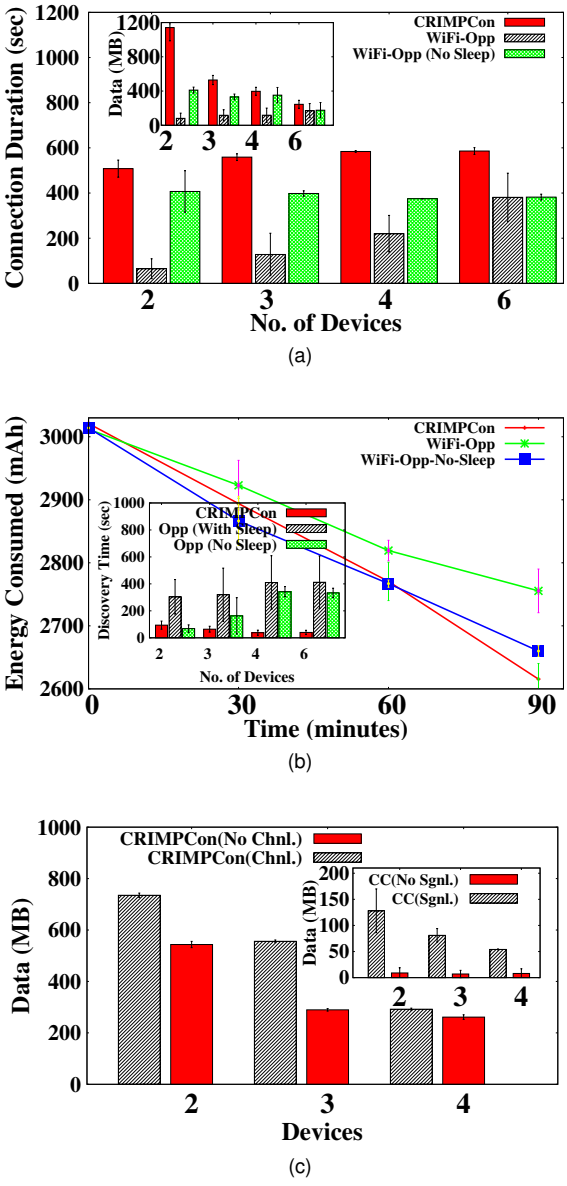



Figure 7: (a) Comparison w.r.t. mean peer contact duration and their mean received data volume (inset). (b) Comparison w.r.t. mean energy dissipation and mean peer discovery time (inset). (c) Comparison w.r.t.mean data volume received by DTN devices when when received signal strength and channel load (inset) is taken into account.

experiments. The figures reveal that CRIMPCon is capable of exploiting P2P contact opportunities better on an average, compared to its competing technologies.

Fig. 7b plots average energy dissipation and average node discovery time by a DTN device. Both the results are compared with the two variants of WiFi-Opp. The figure suggests that neighbor discovery is faster in CRIMPCon for typical opportunistic environments, and

consumes energy as per WiFi-Opp, without explicitly using IEEE 802.11 power save mode.

Finally, CRIMPCon is set to disregard high priority nodes when another neighbor shows significantly high signal strength, and also exploit channel load balancing during Soft AP mode. We compare two variants of CRIMPCon – (i) the device connects to high priority node with low signal strength, and (ii) the device connects to low priority node with high signal strength. Fig. 7c (inset) shows that giving the priority to signal strength improves data transfer performance. Similarly, we observe that CRIMPCon can improve the performance by utilizing channel load balancing (Fig. 7c outer).

### *8.3. Performance Evaluation: CRIMPSync*

We evaluate CRIMPSync against BitTorrent Sync, that follows epidemic forwarding. We observe from Fig. 9a that CRIMPSync syncs more data than BitTorrent Sync, and has less (redundant) data overhead. Fig. 9b proves that mean sync time for CRIMPSync is better than BitTorrent Sync. Note that, sync time increases with rise in the number of connected peers, and the rise become steeper for larger clusters. This help is to infer that the effective syncing will be better when network consists of small clusters instead of small number of few large clusters.

### *8.4. Performance Evaluation: CRIMPApp*

We evaluate CRIMPApp with respect to three aspects – time-consistency (computed based on an average delay from message creation to LCM formation), spatial message diversity and qualitative message diversity (computed using Jain Fairness Index [14] with respect to space and message types). In experiment scenarios 1 and 2, where all the messages are destined for ISB (marked as E), we considered only those messages for performance measurement that reached ISB (E) and compared the observed results against *epidemic routing* based message forwarding. As revealed by the observed metric values shown in Table 5, the fairness index values for CRIMPApp for both the scenarios are significantly higher, spatially (17-106% improvement) as well as message quality-wise (10-20% improvement). Also, the average delay for received messages is significantly lower (75-81% improvement) in CRIMPApp compared to epidemic routing. We infer from the results above that CRIMPApp resolves *consistency-coverage-quality trade-off* in message forwarding effectively.

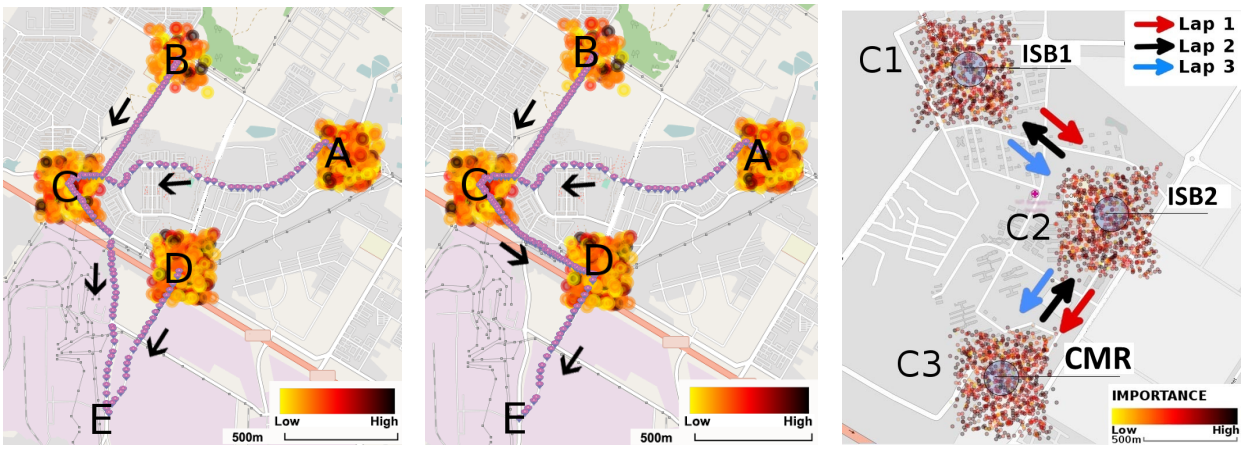
### *8.5. Evaluating CRIMP in Campus-scale*

Finally, we evaluate CRIMP as a complete end-to-end system over Scenario 3 (Fig. 8c). Fig. 10a shows the data volume reached from DTN devices to the cluster ISB, and to other ISBs through opportunistic contacts in a hop-by-hop manner. The figure reveals that CRIMP is successful in fetching multimodal messages to their destinations with a reasonable delay. Inset of Fig. 10a shows the delay distribution of received messages at the CMR. The near-uniform nature of the graph allows us to infer that some of the messages find their destinations fairly quickly. To find out specific messages that are forwarded quickly, Fig. 10b shows that the probability of reaching the destination for a message grows sharply with increase in the importance of the message. Therefore we can conclude that important messages are delivered quickly by CRIMP. Finally, the scatter plot in Fig. 10c shows the system throughput of all the participating devices w.r.t. the length of time they are connected with their peer devices (parallel peer connections are taken as separate peer connections in the figure). The figure shows that ISBs and DM has higher throughput than smartphones.

Figure 11 shows the crisis maps generated in an ISB at intervals of three hours during our campus-scale experiment. Here we have shown only the geographical shapes on the map background. Other information like map annotations, and associated crisis images and audio/video clips are embedded. The diagram highlights how crisis map in a node is getting enriched over time.

## 9. Results from Disaster Mock Drill

Hizole Block, a flood-prone remote rural block in Murshidabad district of the state of West Bengal in India, was selected as the site for a flood and soil erosion mock drill (refer site plan in Fig. 12) by the state disaster management department during October 2016. the mock drill site was set along both the banks of a small river in aforesaid Hizole block. Residents of the village live mostly on huts made of mud along the river embankments. The village and the places surrounding are flooded when the barrage upstream the river releases excess water during rainy-season every year. The village being very remote, and poorly connected with the mainland, the residents have to suffer a lot during the flood, which often, they reported, lasts for even more than a month. At the beginning of the drill, competent authority sent an artificial flood alert message to the block office through proper channel, and the drill began; all the officials and volunteers from participating departments




(a) Scenario 1 (b) Scenario 2 (c) Scenario 3

Figure 8: The campus-scale experiment scenarios with complete ISB locations and DTN/DM movements.

carried out their respective jobs in the same way they are supposed to do during a real disaster.

We participated in the mock drill to test CRIMP during the semi-real environment. During the mock drill, that continued for ~ 3 hours, our volunteers moved along with the participating volunteers from Government Departments and NGOs, sent dummy situational messages using CRIMP in accordance with the mock scenario created by the authority to ISB kept at Flood shelter point. The dotted lines in Fig. 12 shows the movement trails of the volunteers; inset shows a visual of the custom ISB used in mock drill as well as in previously described experiment scenarios; and the attached table shows a brief specification of the ISBs.

In Fig. 13, we plot some performance metrics that reflect the system performance during the mock drill. Fig. 13a shows the delivery status of the created message pool w.r.t. the distance between message origin (i.e. the place at which the message is created) and the location of the ISB. However, we observe that some of the messages are delivered partially, which is due to limited contact opportunities among the smartphones. To perceive the total message delivery (complete/partial) the inset in Fig. 13a shows the delivered data volume to the ISB. Situational messages being multi-modal, they have widely varying message sizes. Also they widely vary spatially, temporally, and type wise, and hence have different 'importance' at any point of time. Due to importance-based message forwarding in CRIMP, we find varying reception pattern of messages at ISB, as shown in Fig. 13b w.r.t. their 'message content' as well as their 'media extension' type. Inset shows volumetric transfer in the same context. Finally, Fig. 13c shows the cumulative delivery ratio at ISB for multimodal messages of different 'media extension' types w.r.t. time. The results help us inferring that CRIMP performs reliably and efficiently even in real-life scenarios.

## 10. Conclusion & Future Scopes

CRIMP is a *first-of-its-kind* end-to-end application system, that implements a crisis mapping system locally even in the absence of Internet, by utilizing few ISBs and mobility of DMs to opportunistically aggregate and

Table 5: Spatial Fairness, Quality Fairness, and Average Delay for Received Messages at ISB for scenarios 1 - 2

| Scenario | Metric | Initial | Proposed | Epidemic | Gain (%) |
|---|---|---|---|---|---|
| Scenario 1 | Jain Fairness Index (Quality) | 0 | 0.76 | 0.63 | 20.6 |
| | Jain Fairness Index (Spatial) | 0 | 0.71 | 0.61 | 17.7 |
| | Average Delay (Received Message) (Minutes) | 0 | 239.7 | 1281.0 | 81.3 |
| Scenario 2 | Jain Fairness Index (Quality) | 0 | 0.80 | 0.73 | 10.52 |
| | Jain Fairness Index (Spatial) | 0 | 0.76 | 0.37 | 106.1 |
| | Average Delay (Received Message)(Minutes) | 0 | 321.1 | 1317.6 | 75.6 |

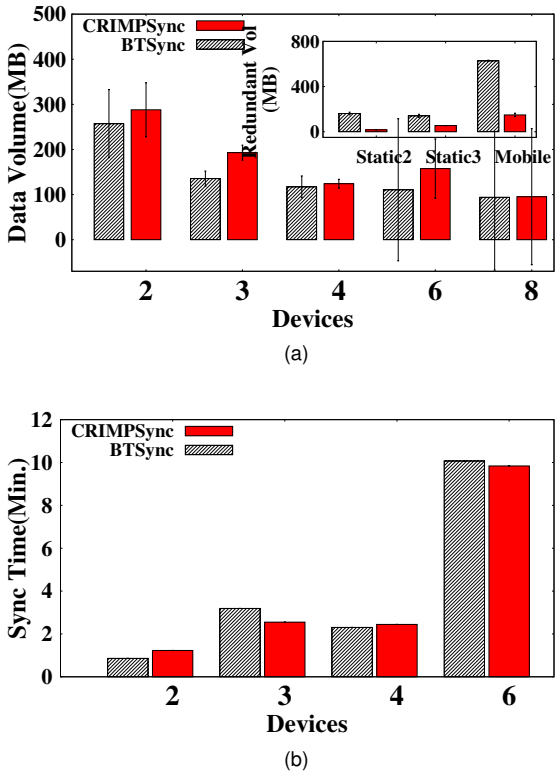



Figure 9: (a) Average message volume exchanged per devices (Inset: Mean volume of redundant messages per device). (b) Average sync completion time

visualize crisis data stored in smartphones. Both lab-scale and field-level trials show that CRIMP can effectively be used to bridge the difference between how a disaster scenario is perceived by the victims and the rescue-relief responders. CRIMP can further be used to develop services like evacuation route recommendation, local need assessment, missing person locator etc. to ensure timely response during the initial stages of disasters. There are instances when the Internet failed to work during man-made disasters like terrorist attacks etc., due to congestion created by a sudden surge of traffic. Moreover, frequently occurring small-scale natural disasters, like flood, landslide, coastal cyclone etc. in remote coastal areas or hilly terrains of developing regions, are often difficult to manage due to extremely poor Internet service quality. Offline functioning of CRIMP makes it feasible to share and visualize situational snapshot in such situations as well.

Presently, CRIMP can upload LCMs to a server (which maintains the global crisis map) whenever the Internet is available. In such cases, it will be interesting to have provisions to merge local ground truth crisis data

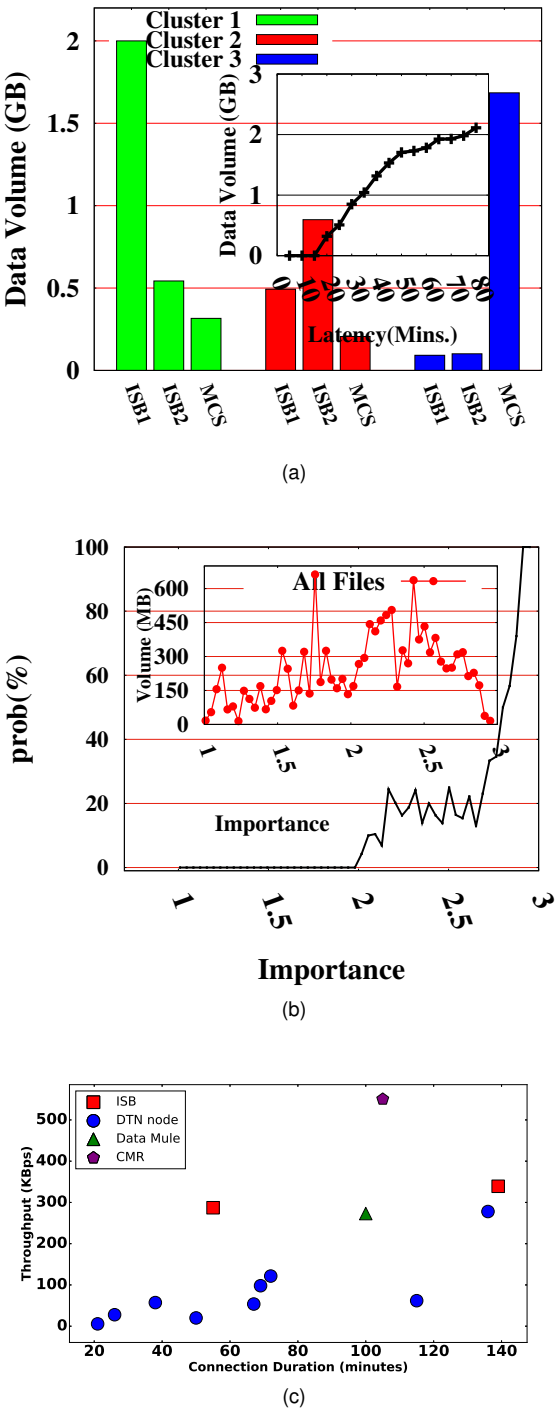



Figure 10: (a) Message volume reached ISBs; (inset) Delay distribution of received messages at CMR. (b) Probability of messages delivery w.r.t importance (c) Throughput of the participating devices.

collected via CRIMP with information from other online sources like Twitter, Facebook, Google crisis map,

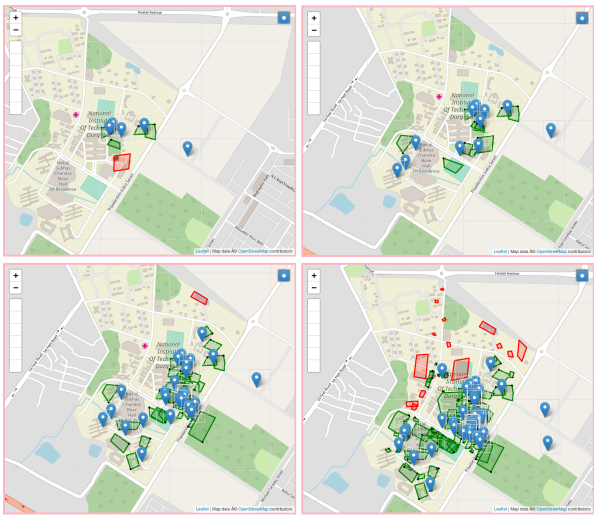
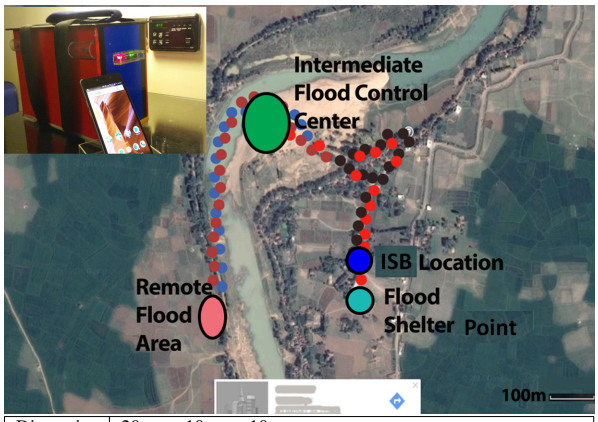


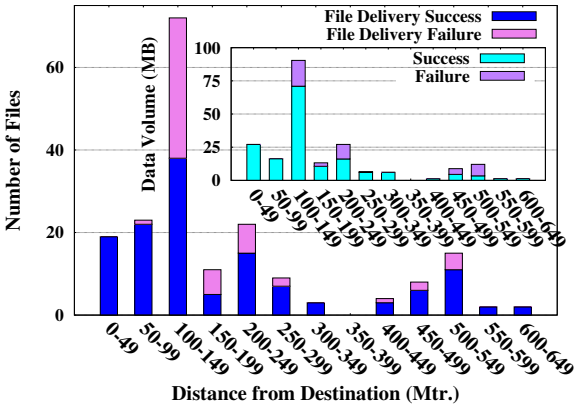
Figure 11: Gradual enrichment of 'localized' crisis map in an ISB at intervals of three hours during our campus-scale field trial



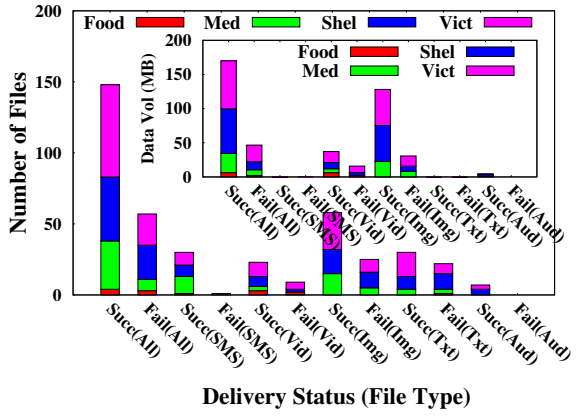

| | |
|---|---|
| Dimension | 20cm × 10cm × 10cm |
| Weight | 3 Kg. |
| Processor | 1-1.2 GHz, ARM64, Quadcore |
| RAM | 1/2 GB LPDDR2 (900 MHz) |
| Battery | 144000 mAh |
| Storage | Upto 32 GB (micro SD Card) |
| Networking | 10/100 Ethernet, 2.4 GHz 802.11b/g/n |
| Bluetooth | 4.1 Classic, BLE |
| GPS | MTK3339 chipset, -165 dBm sensitivity, Accuracy < 3 mtr |

Figure 12: (a) Map view of the disaster mock drill site plan at Hizole Block (Inset: Custom ISB). (b) ISB specification

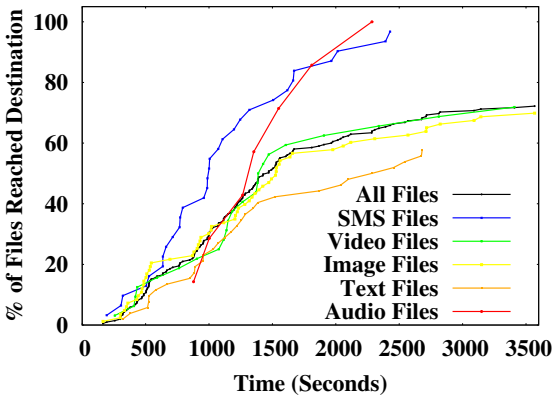
etc. to create a more authentic and effective situational snapshot. Moreover, to improve the naive summarization module, we are presently working on the design of a fast, energy-efficient, lightweight algorithm that can be executed on ISBs or even smartphones for effective summarization of multimodal crisis data.




(a)




(b)




(c)

Figure 13: (a) Distribution of messages and message volume (inset) reached ISB w.r.t. their distance from ISB. (b) Distribution of messages and the message volume (inset) reached ISB w.r.t. their content type and media extension. (c) Temporal distribution of message delivery for different types of messages

## Acknowledgment

This Publication is an outcome of the R&D work undertaken in the ITRA project of Media Lab Asia entitled "Post-Disaster Situation Analysis and Resource Management Using Delay-Tolerant Peer-to-Peer Wireless Networks (DISARM)".

## References

[1] Arnaboldi, V., Conti, M., Delmastro, F., 2014. Cameo: A novel context-aware middleware for opportunistic mobile social networks. Pervasive and Mobile Computing 11, 148 – 167.
URL https://bit.ly/2rvmIVf

[2] Bhattacharjee, S., Roy, S., DasBit, S., 2018. Dpdrm: A decentralized post-disaster resource management scheme using energy efficient smart phone based dtn. Journal of Network and Computer Applications 111, 1 – 16.
URL http://www.sciencedirect.com/science/article/pii/S1084804518300882

[3] Birregah, B., Top, T., Perez, C., Châtelet, E., Matta, N., Lemercier, M., Snoussi, H., June 2012. Multi-layer Crisis Mapping: A Social Media-Based Approach. In: proc. of 21st IEEE WETICE. pp. 379–384.

[4] Boldrini, C., Conti, M., Delmastro, F., Passarella, A., 2010. Context- and social-aware middleware for opportunistic networks. Journal of Network and Computer Applications 33 (5), 525 – 541, middleware Trends for Network Applications.
URL http://www.sciencedirect.com/science/article/pii/S1084804510000524

[5] Chou, C.-H., Zahedi, F. M., 2013. When natural disasters strike: managing individual and organisational needs with web-based systems. International Journal of Business Continuity and Risk Management 4 (1), 75–91.

[6] Conti, M., Delmastro, F., Minutiello, G., Paris, R., 2013. Experimenting opportunistic networks with WiFi Direct. In: proc. of IFIP WD.

[7] Doumi, T., Dolan, M. F., Tatesh, S., Casati, A., Tsirtsis, G., Anchan, K., Flore, D., February 2013. LTE for public safety networks. IEEE Communications Magazine 51 (2), 106–112.

[8] Duc, K. N., Vu, T.-T., Ban, Y., 2014. Ushahidi and Sahana Eden Open-Source Platforms to Assist Disaster Relief: Geospatial Components and Capabilities. In: Geoinformation for Informed Decisions. Springer, pp. 163–174.

[9] Fall, K., 2003. A Delay-tolerant Network Architecture for Challenged Internets. In: Proceedings of the 2003 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications. SIGCOMM '03. ACM, New York, NY, USA, pp. 27–34.
URL http://doi.acm.org/10.1145/863955.863960

[10] George, S. M., Zhou, W., Chenji, H., Won, M., Lee, Y. O., Pazarloglou, A., Stoleru, R., Barooah, P., 2010. Distressnet: a wireless ad hoc and sensor network architecture for situation management in disaster response. IEEE Communications Magazine 48 (3).

[11] Gomes, T., Tapolcai, J., Esposito, C., Hutchison, D., Kuipers, F., Rak, J., de Sousa, A., Iossifides, A., Travanca, R., André, J., et al., 2016. A survey of strategies for communication networks to protect against large-scale natural disasters. In: Proceedings of the 8th International Workshop on Resilient Networks Design and Modeling. IEEE, pp. 11–22.

[12] Gomez, K., Kandeepan, S., Vidal, M. M., Boussemart, V., Ramos, R., Hermenier, R., Rasheed, T., Goratti, L., Reynaud, L., Grace, D., et al., 2016. Aerial base stations with opportunistic links for next generation emergency communications. IEEE Communications Magazine 54 (4), 31–39.

[13] Internews, Sept 2012. Communicating During Disasters: Examining the Relationship between Humanitarian Organizations and Local Media.
URL https://bit.ly/2qq2U51

[14] Jain, R., Chiu, D.-M., Hawe, W. R., 1984. A quantitative measure of fairness and discrimination for resource allocation in shared computer system. Vol. 38. Eastern Research Laboratory, Digital Equipment Corporation Hudson, MA.

[15] Jalihal, D., Koilpillai, R., Khawas, P., Sampoornam, S., Nagarajan, S. H., Kalpalatha, S., Bhavani, R., Takeda, K., Kataoka, K., Aug 2014. A stand-alone quickly-deployable communication system for effective post-disaster management. In: proc. of R10 HTC'14. pp. 52–57.

[16] Lareida, A., Bocek, T., Golaszewski, S., Luthold, C., Weber, M., 2013. Box2Box - A P2P-based file-sharing and synchronization application. proc. of IEEE P2P. pp. 1–2.

[17] Legendre, F., Hossmann, T., Sutton, F., Plattner, B., 2011. 30 Years of Ad Hoc Networking Research: What About Humanitarian and Disaster Relief Solutions? What Are We Still Missing? In: Proceedings of the 1st International Conference on Wireless Technologies for Humanitarian Relief. ACWR'11.

[18] Lindblom, J., Huang, M., Burke, J., Zhang, L., 2013. FileSync/NDN: Peer-to-peer file sync over Named Data Networking. NDN, TR 12.

[19] Martín-Campillo, A., Crowcroft, J., Yoneki, E., Martí, R., 2013. Evaluating opportunistic networks in disaster scenarios. Journal of Network and Computer Applications 36 (2), 870 – 880.
URL http://www.sciencedirect.com/science/article/pii/S1084804512002275

[20] Middleton, S. E., Middleton, L., Modafferi, S., Mar 2014. Real-Time Crisis Mapping of Natural Disasters Using Social Media. IEEE Intelligent Systems 29 (2), 9–17.

[21] Miranda, K., Molinaro, A., Razafindralambo, T., 2016. A survey on rapidly deployable solutions for post-disaster networks. IEEE Communications Magazine 54 (4), 117–123.

[22] Nemoto, Y., Hamaguchi, K., 2014. Resilient ICT research based on lessons learned from the Great East Japan earthquake. IEEE Communications Magazine 52 (3), 38–43.

[23] Nishiyama, H., Ito, M., Kato, N., 2014. Relay-by-smartphone: realizing multihop device-to-device communications. IEEE Communications Magazine 52 (4), 56–65.

[24] Paul, P. S., Ghosh, B. C., De, K., Saha, S., Nandi, S., Saha, S., Bhattacharya, I., Chakraborty, S., 2016. On design and implementation of a scalable and reliable Sync system for delay tolerant challenged networks. In: proc. of IEEE/ACM COMSNETS.

[25] Poiani, T. H., dos Santos Rocha, R., Degrossi, L. C., de Albuquerque, J. P., 2016. Potential of collaborative mapping for disaster relief: A case study of openstreetmap in the nepal earthquake 2015. In: System Sciences (HICSS), 2016 49th Hawaii International Conference on. IEEE, pp. 188–197.

[26] Portmann, M., Pirzada, A. A., Jan 2008. Wireless Mesh Networks for Public Safety and Crisis Management Applications. IEEE Internet Computing 12 (1), 18–25.

[27] Qiantori, A., Sutiono, A. B., Hariyanto, H., Suwa, H., Ohta, T., Feb 2012. An Emergency Medical Communications System by Low Altitude Platform at the Early Stages of a Natural Disaster in Indonesia. Journal of Medical Systems 36 (1), 41–52.

[28] Saha, S., Nandi, S., Paul, P. S., Shah, V. K., Roy, A., Das, S. K., 2015. Designing delay constrained hybrid ad hoc network infrastructure for post-disaster communication. Ad Hoc Networks 25, Part B, 406 – 429.

[29] Sakurai, M., Watson, R. T., Abraham, C., Kokuryo, J., 2014. Sustaining life during the early stages of disaster relief with a frugal information system: learning from the great east Japan earthquake. IEEE Communications Magazine 52 (1), 176–185.

[30] Scanlon, M., Farina, J., Kechadi, M.-T., 2014. Bittorrent sync: Network investigation methodology. In: proc. of ARES.

[31] Senftleben, M., Bucicoiu, M., Tews, E., Armknecht, F., Katzenbeisser, S., Sadeghi, A.-R., 2014. MoP-2-MoP – Mobile Private Microblogging. Springer Berlin Heidelberg, pp. 384–396.

[32] Shahin, A. A., Younis, M., 2014. A framework for P2P networking of smart devices using Wi-Fi direct. In: proc. of IEEE PIMRC. pp. 2082–2087.

[33] Shibata, Y., Uchida, N., Shiratori, N., 2014. Analysis of and proposal for a disaster information network from experience of the great East Japan earthquake. IEEE Communications Magazine 52 (3), 44–50.

[34] Silva, A., Marques, D., Duarte, C., Viana-Baptista, M. A., Carriço, L., 2014. LOST-map: a victim-sourced rescue map of disaster areas. In: proc. of CYTED-RITOS International Workshop on Groupware. Springer, pp. 311–318.

[35] Solmaz, G., Turgut, D., 2017. Tracking pedestrians and emergent events in disaster areas. Journal of Network and Computer Applications 84, 55 – 67.
URL `http://www.sciencedirect.com/science/article/pii/S1084804517300784`

[36] Starbird, K., 2011. Digital Volunteerism During Disaster: Crowdsourcing Information Processing. In: proc. of ACM CHI.

[37] Stiegler, R., Tilley, S., Parveen, T., Sept 2011. Finding family and friends in the aftermath of a disaster using federated queries on social networks and websites. In: Web systems evolution (WSE), 2011 13th IEEE international symposium on. WSE'11.

[38] Tridgell, A., Mackerras, P., 1996. The rsync algorithm. The Australian National University Technical Report.

[39] Trifunovic, S., Distl, B., Schatzmann, D., Legendre, F., 2011. WiFi-Opp: Ad-hoc-less Opportunistic Networking. In: proc. of ACM CHANTS. pp. 37–42.

[40] Trifunovic, S., Kurant, M., Hummel, K. A., Legendre, F., 2015. Wlan-opp: Ad-hoc-less opportunistic networking on smartphones. Ad Hoc Networks 25, Part B, 346 – 358.

[41] Trono, E. M., Fujimoto, M., Suwa, H., Arakawa, Y., Takai, M., Yasumoto, K., March 2016. Disaster area mapping using spatially-distributed computing nodes across a DTN. In: PerCom Workshops.

[42] Turkes, O., Scholten, H., Havinga, P. J., 2016. Cocoon: A lightweight opportunistic networking middleware for community-oriented smart mobile applications. Computer Networks 111, 93–108.

[43] Turkes, O., Scholten, H., Havinga, P. J., 2016. Friend-to-friend short message service with opportunistic Wi-Fi beacons. In: proc. of PerCom Workshops.

[44] UNEP, 2013. Cyclone Phailin in India: Early warning and timely actions saved lives.
URL `https://bit.ly/2GTQOaE`

[45] UNISDR, 2015. Sendai Framework for Disaster Risk Reduction 2015 - 2030.
URL `https://bit.ly/1hj93Jk`

[46] Zastrow, M., 2014. Crisis mappers turn to citizen scientists. Nature 515 (7527).

[47] Zook, M., Graham, M., Shelton, T., Gorman, S., 2010. Volunteered geographic information and crowdsourcing disaster relief: a case study of the Haitian earthquake. World Medical & Health Policy 2 (2), 7–33.

[48] Ólafur Helgason, Kouyoumdjieva, S. T., Pajević, L., Yavuz, E. A., Karlsson, G., 2016. A middleware for opportunistic content distribution. Computer Networks 107, 178 – 193, mobile Wireless Networks.
URL `https://bit.ly/2I8qJFu`