A Novel Network Architecture for Resource-Constrained Post-Disaster Environments

Krishnandu Hazra[§*] , Vijay K. Shah[†*], Mohd Bilal[§],
Simone Silvestri[†], Sajal K. Das [‡], Subrata Nandi[§], and Sujoy Saha[§]
[§]National Institute of Technology, Durgapur, India, [†]University of Kentucky, Lexington, USA,
[‡] Missouri University of Science and Technology, Rolla, USA
krishnandu91@gmail.com, vijay.shah@uky.edu, mohdbilal2922@gmail.com,
silvestri@cs.uky.edu, sdas@mst.edu, subrata.nandi@gmail.com, and sujoy.ju@gmail.com

Abstract—**Following a large-scale disaster such as an earthquake or a hurricane, existing communication (e.g., cellular towers) and other infrastructures (e.g., power lines, roads etc.) are often critically impaired. This hampers the seamless exchange of information, such as, the status of survivors, requirement of relief materials, supply chain of goods and services, between the rescue/relief teams and the control station in a disaster area, and thereby preventing the timely recovery operations. To address this, several network architectures, utilizing rescue/relief teams equipped with wireless devices and easily deployable towers, have been proposed to set up a temporary communication network. While these works propose novel network architectures, they largely ignore the fact that the availability of network resources are often limited in such scenarios (mainly due to budgetary constraints). Hence in this paper, we design a novel network architecture to specifically address the resource-constrained post-disaster scenarios. The underlying idea is to rationally allocate the constrained network resources in the disaster area such that (i) each shelter point is served by at least one network resource and (ii) the end-to-end network latency, from volunteers to the control station or vice-versa, is minimized. We formulate this resource allocation problem as a non-linear programming (NLP) optimization problem. After proving that such a problem is NP-Hard, we propose an effective sub-optimal heuristic for solving it, and thereby designing the planned architecture. Our extensive experiments based on the real map of Durgapur, India show that, in a resource-constrained scenario, the planned architecture greatly outperforms an unplanned architecture in terms of both delivery probability and end-to-end network latency.**

Index Terms—**Post-disaster environments, Delay tolerant network, Resource allocation, Disaster management**

I. INTRODUCTION

During the great Nepal Earthquake in April 2015, nearly $9,000$ people were killed, more than $22,000$ people were injured, in excess of $600,000$ houses were destroyed and more than $288,255$ houses damaged [1]. Several public infrastructures, such as roads, bridges, power lines, and electricity poles, and communication infrastructures, such as cellular towers and Internet routers, were damaged or impaired in several places (14 out of 75 districts) in the event of Nepal Earthquake [2]. This is one example of many natural disasters that has occurred in recent years. In fact, one recent report [3] has shown that the occurrence of natural disaster has increased *three folds* in $1980 - 2016$ compared to that in $1940 - 1980$.

Following a large-scale disaster, the affected area suffers from sparse or no network coverage during and after the *golden hour* [1] of the disaster (as also evident from Nepal Earthquake). In such a situation, the rescue/relief parties, such as, police vehicles, medical teams, firemen, etc., are *unable* to seamlessly exchange information messages (e.g., status of survivors, requirement of relief materials, supply chain of goods and services, information on collapsed buildings and roads, etc.) among each other. As a result, this leads to an asynchronous coordination of rescue/relief operation and ad-hoc decision making in the disaster area, which further worsens the human lives and economic loss [2]. Hence, there is a need to establish a temporary network that enables the seamless information exchange among various entities in the disaster area, i.e., volunteers gathered in shelter points (or relief camps), rescue/relief parties, and the controlling authority, referred to as *Master Control Station* (MCS).

In the last two decades, several research works [6]–[13] have been proposed for rapidly deploying a temporary communication infrastructure in a post-disaster scenario. For example, the authors in [6]–[11] have proposed utilizing smart devices and sensors to deploy a wireless ad-hoc delay tolerant networking solution for emergency situations. Sakano et al. [12] proposed a network architecture that utilizes movable and deployable resource unit (MDRU) to establish to a disaster-resilient network. Zussman et al. [13] proposed to employ the network formed by smart badges to collect information from trapped survivors. Saha et al. [14], [15] proposed utilizing long-range WiFi towers, mobile vehicles equipped with wireless devices, and information drop boxes to create a delay-constrained network infrastructure for post-disaster environments. A more detailed survey on deploying communication network in post-disaster scenarios can be found in [16], [17].

---

[*] Co-primary author

[1] In emergency medicine, the golden hour (also known as golden time) refers to a time lasting for one hour, or less, during which there is the highest likelihood that prompt medical treatment will prevent death.

[2] Nepal Earthquake: Sindhupalchowk district - The total death toll after a day of Nepal Earthquake was nearly 900. However, within 2 weeks, it rose beyond 2000, standing it as one of the two most affected districts, along with Kathmandu, the capital of Nepal [4]. According to a report [5], the villages in Sindhupalchowk district did not receive any help even after 5 days of the Earthquake. Consequently, several people reportedly died due to the absence of basic relief materials, such as, blankets, food, drinking water, and medicine.

All these works have mainly focused in designing novel network architectures, utilizing varying numbers and types of network resources, such as, UAV drones, smart phones, communication towers, information drop boxes, mobile base stations, vehicles, sensors etc. However, they largely ignore the fact that the availability of such network resources pool are often limited in post-disaster scenarios, particularly in low economy zones. This is mainly because the controlling authority, such as the government disaster management authority, have a fixed funding budgetary constraint for rescue/relief operations [3] in post-disaster environments.

Hence, in this work, we primarily investigate the problem of designing a planned network architecture suitable for such *resource-constrained* post-disaster environments. Specifically, the proposed architecture intelligently allocates the constrained network resource pool in the disaster area in such a way that (i) each shelter point (residing affected people) in the area is served by at least one network resource, e.g., a communication tower or a rescue/relief vehicle etc., and (ii) the end-to-end network latency (i.e., from volunteers to the MCS or vice-versa) is minimized. We formulate such a network resource allocation problem as a non-linear programming (NLP) optimization problem, and show that it is NP-hard. Following this, we propose a simple yet effective sub-optimal heuristic for solving the problem and thereby, designing a planned network architecture for the resource-constrained post-disaster scenarios. Our extensive experiments based on the real map of Durgapur, India (obtained from Google Maps [4]) on top of the Opportunistic Network (ONE) Simulator [18], demonstrate that the planned network architecture greatly improves upon the delivery probability and end-to-end network latency, compared to that of an unplanned approach, in all considered resource-constrained post-disaster scenarios.

In summary this paper makes the following contributions:

- We propose a novel network architecture specifically designed to form a temporary communication infrastructure in post-disaster scenarios with a limited availability of network resources.
- We formulate a non-linear optimization problem where the objective is to optimally allocate the constrained network resources pool in such a way that the end-to-end network latency is minimized, while ensuring that each shelter point in the area is covered.
- We prove that such a resource allocation problem is NP-Hard, and propose a simple yet effective sub-optimal heuristic to solving it in polynomial-time.
- Through simulation experiments on the real map of NIT, Durgapur, we demonstrate that the proposed architecture greatly outperforms the unplanned approach in terms of two important performance metrics, i.e., delivery probability and network latency.

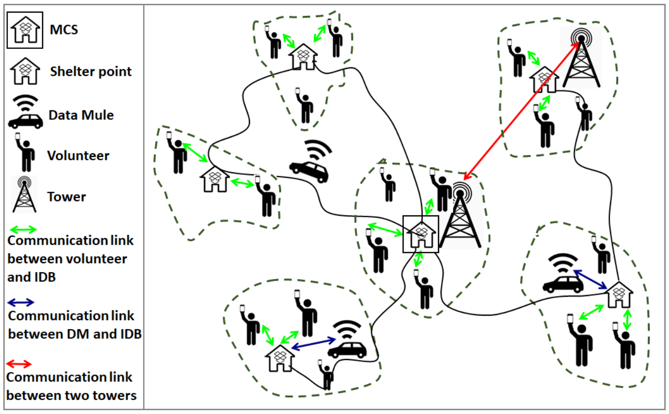


Fig. 1. Network model. The green and blue colored lines denote the communication link between a volunteer and the IDB (via Type 1 interface), and between IDB and DM (via Type 2 interface), respectively. Similarly, the dotted red colored line denote the communication link (via Type 3 LWC interface) between any two communication towers.

The rest of the paper is organized as follows. Section II presents the network model, whereas Section III formulates the resource allocation problem. Section IV discusses the details of the proposed heuristic. Section V discusses the performance evaluations, followed by conclusions in Section VI.

II. NETWORK MODEL

As shown in Fig. 1, we model a disaster affected area as a graph $G(V, E, R \cup C)$ where the node set $V = \{i : 1 \leq i \leq |V|\}$ corresponds to the set of *shelter points* (SPs) in the disaster area. A SP is a small geographical area, such as, a relief camp, school, park, building, evacuation center etc., where the affected people gather and stay for a few days (or even several weeks) in the aftermath of a disaster until the outside environment is safe [8]. We assume that several of them, termed *volunteers*, have a disaster application (such as Surakshit [19]) installed on his smart device which allows him to (i) establish ad-hoc communication with other peers directly via *Type 1 interface* i.e., Bluetooth or WiFi-Direct, and (ii) exchange situation or rescue/relief need based messages in form of text, image, audio and video clips. The edge set $E = \{e_{ij}\}$, where e_{ij} denotes a direct physical pathway (i.e., a road) from a SP i to j, where $i, j \in V$. One specific SP, denoted by $\hat{v}$, is designated as the *Master Control Station* (MCS) from where the entire rescue/relief operations are coordinated.

We assume that each SP $i \in V$ is equipped with an *Information Drop Box* (IDB). An IDB is a fixed laptop or a customized equipment (like a kiosk [20] or throwbox [21]) which has following capabilities: (i) two wireless interfaces - (i.a) *Type 1 interface* (such as, WiFi-Direct or Bluetooth) to communicate with the volunteers residing in SP i, and (i.b) *Type 2 interface* (such as, short-range WiFi interface) to communicate with the patrolling data mules (explained below), and (ii) *memory storage* - to temporarily store messages received either from the visiting volunteers or data mules. Hence, an IDB basically acts as a proxy between a certain SP (and the residing volunteers) and the patrolling data mule.

---

[3] For instance, Calamity Relief Funds (CRFs) are dedicated funds used by the state governments in India, to meet the expenditure for providing immediate relief to victims of large-scale disasters (such as, cyclone, earthquake, flood, tsunami etc), which also includes the funds for network resources pool.

[4] https://www.google.com/maps

Additionally, we consider that the SPs (and IDBs) are usually far from the MCS, and therefore do not lie in the direct communication range of each other (which is usually the case in large-scale post-disaster scenarios [7], [15], [22]).

Network Resources. We consider that there exist a limited set of *network resources* [5] pool available in the disaster area – (a) $R = \{r : 1 \leq r \leq |R|\}$ set of *data mules* (DMs) and (b) $C = \{c : 1 \leq c \leq |C|\}$ set of *communication towers*, which can be utilized for establishing a temporary communication network between the SPs and MCS (to carry out timely and efficient recovery operations). A DM $r \in R$ is a moving vehicle (e.g., police or fire vehicle etc.) or a rescue/relief team (e.g., medical team, disaster relief team etc.) equipped with a smart device (with Type 2 interface and a memory storage). Since, a DM patrols one or more SPs lying on its route trajectory, it possesses the capability to collect information messages from the IDBs (belonging to the patrolling SPs) and/or distribute the messages to the IDBs (that it may have received from the MCS). For ease of presentation, we consider that each DM r moves at an average speed of s Km/hour [6]. On the other hand, a communication tower is directly allotted to a certain SP and remains fixed. A tower possesses a Type 3 long-range WiFi communication (LWC) interface, and is capable of communicating with other far distant tower (located at some other SP or MCS itself); thanks to the large transmission coverage of LWC interface.

Message Transmission. The generated messages at each volunteer (residing in a certain SP) are transmitted towards the IDB (via Type 1 interface), which are then transmitted towards the MCS, either physically by the patrolling DM (via Type 2 interface) or over the communication tower's network (via Type 3 LWC interface). Similarly, the MCS may generate certain response messages, which are transferred towards the volunteers in the similar fashion (in the opposite direction). In this work, we assume that the latency incurred in message exchange between any two SPs i and j, via communication towers is comparatively much smaller than physically carrying the message by a DM. This is a reasonable assumption as unlike towers, a DM has to physically travel from SP i to j, which may take several minutes, or even hours [15], [22].

To meet the key objective of the paper, i.e., *minimize the end-to-end network latency in message exchange between each volunteer and the MCS*, it is important to design a planned approach for the optimal allocation of the constrained network resources pool (i.e., $|R|$ DMs and $|C|$ communication towers). Hence, in rest of the paper, we primarily investigate the network resource allocation problem. In the following sections, we first formulate the network resource allocation problem as an optimization problem, show that such a problem is NP-Hard, and finally propose a simple yet effective sub-optimal heuristic for solving it in polynomial-time.

III. PROBLEM FORMULATION

This section formulates the network Resource AlloCation (ResAlloC) problem as a Non-Linear Programming (NLP) optimization problem. ResAlloC aims at optimally allocating the limited network resource pool, i.e., $|R|$ data mules and $|C|$ communication towers, in a disaster area such that the end-to-end network latency in message exchange between each volunteer and the MCS is minimized.

Notice that the end-to-end network latency may constitute several contributing factors (such as, volunteers' mobility routes, placement of towers, DMs' route trajectories, message transmission between – (i) volunteer and the IDB via Type 1 interface, (ii) IDB and DM via Type 2 interface, if served by DM, and (iii) IDB and IDB/MCS via Type 3 interface over towers etc.), however there are only two factors [7] those influence the ResAlloC problem are – (i) placement of $|C|$ communication towers, and (ii) determination of route trajectories for $|R|$ DMs. Moreover, from our discussion in Section II, recall that the message transmission via towers (if possible) are always be preferred than via the DMs. Keeping these in mind, the ResAlloC problem can be stated as follows:

Problem Statement: *Given a disaster area $G(V, E, R \cup C)$, latency cost l_{ij} incurred by any data mule $r \in R$ in traveling a pathway $e_{ij} \in E$, the ResAlloC problem is to determine (i) the optimal SPs for the placement of $|C|$ communication towers, and (ii) the least latency cost trajectory (i.e., the suitable set of SPs and pathways) for each DM r; while ensuring that $|C|$ towers form a connected network over Type 3 interface and each SP is either served by a tower, or by a DM.*

In the following, we discuss in detail the formulation of the ResAlloC problem as NLP optimization problem.

As discussion in Section II, let e_{ij} denote a direct pathway from a SP i to SP j. That is, $e_{ij} = 1$ if there exists a direct pathway from SP i to SP j and 0 otherwise. Moreover, let l_{ij} denote the incurred latency by any DM $r \in R$ in traveling a pathway e_{ij} from SP i to j. Similarly, let e^a_{ij} denote a direct link between a SP i and j. Note that, such a link e^a_{ij} exists (i.e., $e^a_{ij} = 1$) only if (i) both SPs i and j are allotted a communication tower and (ii) the euclidean distance between SP i and SP j is less than or equal to the transmission range of the communication tower (over Type 3 interface). Finally, let P^r denote the path route trajectory of a certain DM $r \in R$.

Decision Variables. We introduce a binary decision variable $x^r_{ij} = 1$, if DM r currently at SP i visits another SP j, otherwise $x^r_{ij} = 0$. We introduce another binary decision variable $y^r_i = 1$, if a SP $i \in V$ is served by a DM $r \in R$, otherwise $y^r_i = 0$. Note that a data mule visiting a certain SP may or may not serve that SP (It just passes by the SP while traveling on its path trajectory). We introduce a third binary variable $z_i = 1$, if SP i is allotted a communication tower,

---

[5] Though in this work, we only consider C set of communication towers and R set of DMs as network resources pool, it can be easily extended to other types of network resources, e.g., UAVs, mobile base stations etc.

[6] It is straightforward to extend our approach to a disaster scenario where each DM $r \in R$ has a varying traveling speed s^r Km/hour.

[7] Though the other factors' contribution to the end-to-end network latency are non-negligible and varying (as a result of distinct message generation rate at each volunteer, mobility of each volunteer, waiting time at IDB, and data rates of Type 1, Type 2 and Type 3 interfaces), these factor do not affect the optimal solution of the ResAlloC problem, and hence are not considered in the ResAlloC problem formulation.

otherwise $z_i = 0$. We require a fourth binary variable $s_i^r = 1$, if SP i is the start and end node SP (i.e., acts as a *depot*) for the route trajectory of DM $r \in R$, otherwise $s_i^r = 0$. Finally, we introduce a fifth variable $a_{ij} = 1$, if an edge e_{ij}^a exists between a SP i and SP j, otherwise $a_{ij} = 0$.

Objective function. As shown in Eq. 1, the objective of the ResAllocP problem is to determine the least latency cost route trajectory P^r for each DM $r \in R$.

$$\textbf{Minimize} \ \max_{r \in R} L(P^r), \ \text{ where} \tag{1}$$

$$L(P^r) = \sum_{i \in V} \sum_{j \in V} l_{ij} x_{ij}^r \tag{2}$$

subject to:

$$\sum_{i \in V} z_i \leq |C| \tag{3}$$

$$s_i^r \leq z_i, \ \forall r, i - (a); \ \sum_{i \in V} s_i^r = 1, \ \forall r - (b) \tag{4}$$

$$\sum_{j \in V} (x_{ij}^r + x_{ji}^r) - y_i^r \geq 0, \ \ \forall r, i \tag{5}$$

$$z_i + \sum_{r \in R} y_i^r = 1, \forall i \tag{6}$$

$$\sum_{i \in V, i \neq p} x_{ip}^r - \sum_{j \in V, j \neq p} x_{pj}^r = 0, \ \ \forall r, p \tag{7}$$

$$\sum_{i \in V} s_i^r y_i^r = 1, \forall r \tag{8}$$

$$x_{ij}^r \leq e_{ij}, \quad \forall i, j, r \tag{9}$$

$$\sum_{i \in V} \sum_{j \in V} a_{ij} \geq \sum_{p \in V} z_p - 1 \tag{10}$$

$$\sum_{i \in S} \sum_{j \in S} a_{ij} \leq \sum_{p \in S} z_p - 1, \ \ \forall S \subset V \tag{11}$$

$$a_{ij} \leq z_i - (a), \ \ a_{ij} \leq z_j - (b) \tag{12}$$

$$z_i, y_i^r, s_i^r, x_{ij}^r, a_{ij} \in \{0, 1\}, \ \ \forall i, j, r \tag{13}$$

Constraints. Eq. 3 constrains the cardinality of SPs to be allotted with a maximum of $|C|$ communication towers. Eq. 4(a) constrains a SP to act as a *depot* for a certain DM only if it is allotted a communication tower. This constraint guarantees that each SP served (not just visited) by a certain DM will have a non-infinity path to the MCS, and hence can exchange messages with the MCS. Eq. 4(b) restricts a unique SP to act as a depot for a certain DM. Eq. 5 ensures a certain SP to be served by a DM only if it is visited by that DM. Eq. 6 shows that each SP must be served by either a DM or a communication tower. Eq. 7 represents the flow conservation constraint which ensures that once a DM r visits a certain SP $p \in V$, then it must also depart from the same SP. Eq. 8 is a non-linear constraint and enforces a certain DM to start its route trajectory from a depot and end at it. Eq. 9 constrains a DM currently at SP i to travel to another SP j only if there exists a physical pathway from SP i to j (i.e., $e_{ij} = 1$).

Constraints 10 and 11 ensure that the $|C|$ communication towers allotted in the disaster area (i.e., allotted at $|C|$ chosen SPs) form a connected topology over Type 3 LWC interface. Specifically, constraint 10 ensures that there are at least $(K-1)$ edges (a required condition for a connected network) between K towers, allotted at certain K SPs (where $K \leq |C|$). Moreover, constraint in Eq. 11 ensures that there is no cycle in the subset S. Constraints 12 (a) and 12 (b) requires both the end SPs i and j to be equipped with a communication tower, to have a communication link e_{ij}^a (over Type 3 interface) between them. Finally, Eqs. 13 represents the binary decision variables, that take values either 0 or 1.

Theorem 1. *The ResAllocP problem is NP-Hard.*

Proof: We provide a reduction for a well-known NP-Hard Traveling Salesman Problem (TSP) [23]. The TSP problem is as follows: Given a set of cities V and distance d_{ij} between every pair of cities i and j, the problem is to find the shortest possible route P^r that a certain salesman r visits every city exactly once and returns to the starting city.

We reduce the TSP problem to an instance of ResAlloC problem as follows: We consider a disaster area with V SPs, latency cost l_{ij} between every pair of SPs i and j (where $i, j \in V$), a unique DM r (i.e., $|R| = 1$) and $|C| = 0$ communication towers. In this instance, the ResAlloC problem determines a least latency cost path route trajectory for the DM r such that the DM r serves each SP exactly once and returns to the starting SP (i.e., depot). Such a path route trajectory also corresponds to the optimal solution i.e., shortest possible route P^r, of the TSP problem. Therefore, if we are able to solve ResAlloC problem in polynomial-time, we are also able to solve TSP problem in polynomial-time. Since, the TSP problem is NP-Hard, the ResAlloC problem is NP-Hard. ■

IV. PROPOSED HEURISTIC

In this section, we discuss a simple yet effective sub-optimal heuristic that intelligently allocates the limited set of network resources i.e., $|C|$ communication towers and $|R|$ DMs, in such a way that the overall latency cost between each SP and the MCS is minimized. The proposed heuristic operates in two steps, which is discussed as follows:

A. Step 1: SP Group Formation (and Tower Placement)

This step primarily (i) clusters the V set of SPs in the disaster area into K set of disjoint *SP groups* (where $|K| = |C|$), (ii) determines a suitable group *center* χ_k for each SP group $k \in K$, and finally (iii) allocates a communication tower $c \in C$ to each group center such that the $|C|$ communication towers form a connected network topology over type 3 LWC interface. Recall V and C are the set of SPs and communication towers, respectively. Now, we discuss the details of the proposed step 1 algorithm (See pseudocode 1), which is inspired from K-medoid clustering algorithm [24]. It works in two phases:

BUILD: In this phase, the heuristic first selects $|K| = |C|$ random SPs (out of V set of SPs) in the disaster area as the initial K set of group centers (or simply *centers*), such that the $|C|$ communication towers deployed at $|K|$ centers form *a connected network topology* over type 3 LWC interface (See line 4). Then, each remaining non-center i (i.e., $i \in V \setminus K$) SP is uniquely assigned to the nearest (or least pathway distant) center. Hence, a *SP group* $k \in K$ constitutes a unique center (denoted by χ_k) and the assigned SPs as its group *members*.

Algorithm 1 Grouping of Shelter Points

```
1: Input: Disaster area $G$, $C$ communication towers, $I_{max}$
2: Output: $K$ set of SP groups, each with unique group center and group members.
3: procedure GROUPING-SPs(G, $C$)
4:     Select $C \subseteq V$ set of random SPs as initial $K$ set of centers (where $|K| = |C|$) if $|C|$ centers form a connected topology over type 3 LWC interface
5:     Number of iterations, $I_{curr} = 0$, $K_{old} = \phi$
6:     while ($K_{old} \neq K$) and ($I_{curr} < I_{max}$) do
7:         Initialize $Sum = 0$, $K_{old} = K$
8:         for each remaining SP $i \in V \setminus K$ do
9:             Assign SP $i$ to the center, $\chi_k$ with $(\min_{k \in K} d_{i\chi_k})$
10:            $Sum += \min_{k \in K} d_{i\chi_k}$
11:        for each randomly selected SP $i \in V \setminus K$ do
12:            for each SP group $k \in K$ do
13:                $K_{new} = (K \setminus \chi_k) \cup i$
14:                if is-connected-network($K_{new}$, LWC-range) then
15:                    Swap SP $i$ with center $\chi_k$, i.e., $\chi_k = i$
16:                    Initialize $Sum_{new} = 0$
17:                    for each SP $i \in V \setminus K_{new}$ do
18:                        $Sum_{new}$ += $min_{k \in K_{new}} d_{i\chi_k}$
19:                    if $Sum_{new} < Sum$ then
20:                        Finalize Swap and update $K = K_{new}$
21:        $I_{curr} += 1$
22:    Choose any one group center as the MCS.
```

SWAP: In this phase, the algorithm attempts to improve the K set of SP groups by repetitively exchanging the pre-computed group center (in previous iteration) with a non-center SP in the current iteration; while ensuring that the updated centers are connected (over type 3 LWC interface). As shown in lines 6 - 21, this phase keeps repeating until one of the following conditions are met– (i) there is no change in the K set of centers, and (ii) maximum number of iterations I_{max} is done. Note that a non-center SP, say i (See line 11) is chosen randomly as the new center χ_k (for a certain SP group, k) with the hope that it improves (or minimizes) the overall sum of the pathway distances between every non-center SP to their centers. Provided that the overall sum of the pathway distances for the new K set of centers is improved, the swap between non-center i and the center χ_k is finalized. Finally, as shown in line 22, one unique center is chosen as the MCS.

B. Step 2: Determination of DM Trajectories.

The second step of the heuristic determines the suitable (i.e., least latency cost) trajectory for each DM $r \in R$ in the disaster area. As shown in pseudocode 2, the proposed algorithm for second step is as discussed follows:

First, the algorithm allocates $\left\lfloor \frac{|R|}{|K|} \right\rfloor$ DMs to each SP group $k \in K$, except the last one which will constitute the remaining DMs (See lines 5 - 8). Let V_k denote the set of SPs (including the center χ_k) in the group k. Then, the algorithm utilizes Dijkstra's algorithm [25] to compute the shortest (i.e., least pathway distance) path from the group center χ_k to every non-center SP $i \in V_k \setminus \chi_k$ (i.e., belonging to the group k) (See line 11). This returns a P set of shortest paths, which is then sorted in the increasing order of the pathway distance (See line 13). Following this, as shown in lines 14 - 19, the algorithm assigns a certain (in increasing order) shortest path $p \in P$ to a certain DM $r \in R_k$ (i.e., belonging to a SP group k), provided that the chosen path p has at least one unvisited non-center SP. Then, all the SPs belonging to the path p are marked visited. Finally, each of the remaining unvisited SPs in group k are assigned to the nearest route trajectory P^r of DM $r \in R_k$ (See line 20).

Algorithm 2 Calculation of Data Mule Trajectory

```
1: Input: $K$ set of SP groups, $R$ set of DMs
2: Output: Suitable trajectory path $P^r$ for each DM $r \in R$
3: procedure GET-DM-TRAJECTORY()
4:     for each SP group $k \in K$ do
5:         if ($k < |K|$) then //except the last one
6:             No. of DMs, $|R_k|$ = $\lfloor \frac{|R|}{|K|} \rfloor$
7:         else
8:             $|R_k| = |R| - (|K| - 1)\lfloor \frac{|R|}{|K|} \rfloor$
9:         Set of paths, $P = \phi$
10:        for each SP $i \in V_k \setminus \chi_k$ do //$V_k$ is set of SPs in group $k$
11:            Compute shortest (or least pathway distance) path $p$ from center $\chi_k$ to non-center SP $i$ (using Dijkstra's algorithm)
12:            Include path $p$ in the path set $P$, i.e., $P = P \cup p$
13:        Sort $P$ in increasing order of pathway distance
14:        for each DM, $r \in R_k$ do
15:            for each unique path, $p \in P$ do
16:                if at least one unvisited SP in path $p$ then
17:                    Route trajectory, $P^r = p$
18:                    Mark all SPs belonging to path $p$ as visited
19:                    break
20:        Allocate remaining unvisited SPs in group $k$ to the nearest route trajectory $P^r$
```

Time Complexity. Algorithm 1 takes $O(|K|^2)$ to check if $|K|$ centers are connected (using Depth First Search or Breadth First Search algorithm [25] in the worst case). Hence, the total complexity of Algorithm 1 is $O(I_{max} \times |V| \times |K|^3)$, where I_{max} is the maximum number of iterations. The time complexity of the algorithm 2 is $O(|K| \times |R_k| \times |V_k|^2)$. Since I_{max} is comparatively very high, the total complexity of the proposed heuristic is $O(I_{max} \times |V| \times |K|^3)$.

V. PERFORMANCE EVALUATION

In this section, we evaluate the performance of the proposed network architecture in terms of the following performance metrics: (i) *Network Latency* - the worst-case delay incurred in delivering a generated message from a volunteer (residing in a certain SP) to the MCS or vice-versa (i.e., from MCS to the intended volunteer)., (ii) *Delivery Probability* - the fraction of total messages successfully delivered (within the prespecified TTL deadline) to the MCS to the total messages generated at the volunteers or vice versa.

A. *Simulation Setup*

We simulate the proposed planned network architecture on the real map of Durgapur, India in the Opportunistic Network Environment (ONE) simulator [18].

As shown in Fig. 2, we have developed a software tool, termed *Post-Disaster Communication Resource Planning Tool* (PDCRPT), where we select a 3×3 sq. Km area surrounding National Institute of Technology (NIT), Durgapur as the simulation area. Following this, the shelter points are placed in the considered area (See Fig. 2(a)). Then, it extracts the path information from the map in a distributed manner [8] using Google Map APIs (Fig. 2(b)). Using both the shelter point and pathway information, the proposed heuristic is run offline and the suitable shelter points for tower deployment, and route trajectories for data mules are computed (Fig. 2(c)). Finally, we perform the experimental analysis on top of the ONE simulator, as discussed below.

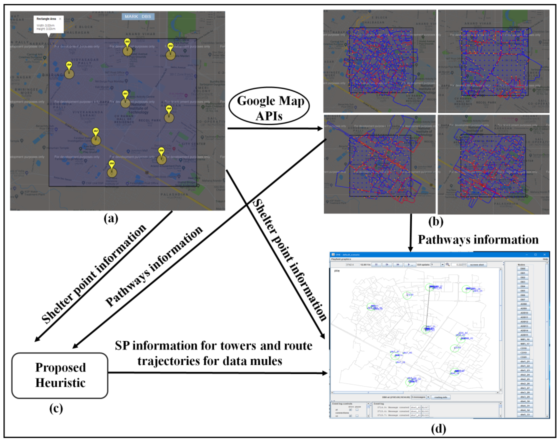


Fig. 2. An overview of PDCRPT tool: (a) Area and Shelter point selection, (b) Path extraction in distributed manner (via four clients) with the help of Google Map APIs, (c) Proposed heuristic offline run, and (d) Experimental analysis of the proposed architecture in ONE simulator

All the experiments, unless otherwise stated, are performed with 8 shelter points, 2 communication towers, and 3 moving vehicles (acting as data mules). Refer to Table I for a complete overview of the considered simulation parameters.

1) Movement model: For volunteers, we consider Post Office Cluster Movement Model (PCM) [15], which is extended from the Cluster movement model [26]. In PCM model, a volunteer visits the IDB (located in its SP) in such a way that the inter-arrival time gap between two consecutive visits to the IDB follows a Poisson or some other standard probability function. See [15] for the details on PCM model. On contrary, for data mules, we utilize External Movement Model (integrated with ONE simulator), where we specify the movement of each DM, which is computed offline by our proposed heuristic (See Fig. 2 (c)).

2) Network Traffic model: We consider two types of message packets in our experiments – (i) Request packet - These packets are generated by volunteers, and are to be delivered to the MCS, and (ii) Response packets - Such packets are generated by MCS, and are to be delivered to the intended IDB (and volunteer). See Table I for the details of request and response packet generation rates and packet sizes.

3) Routing model: For our experiments, we utilize the simplest yet effective epidemic routing protocol [27], for it offers very high delivery probability and network latency, compared to that of other standard routing protocols [7], [8] proposed for post-disaster communication networks.

TABLE I
LIST OF SIMULATION PARAMETERS

Simulation parameters	**Description**
Underlying map information	NIT Durgapur & its surrounding
Simulation area	3×3 sq. Kms
Average. geographical area of a SP	200 m
Disaster impact on pathways	10%
Number of shelter points	8
Number of data mules	3
Average Speed of data mules	10 Km/hr
Number of communication towers	2
No. of volunteers	$3-5$/shelter point
Request packet generation rate:	10 packets/hr
Response packet generation rate:	3 packets/hr
Size of each request packet	$100-300$ KB
Size of each response packet	$50-100$ KB
Time-To-Live (TTL) for each packet	1 hour
(Data rate, Range) of Type 1 Interface	(2 Mbps, 10 m)
(Data rate, Range) of Type 2 Interface	(20 Mbps, 100 m)
(Data rate, Range) of Type 3 (LWC) Interface	(100 Mbps, 8 km)
Data mule waiting time at each DB	$1-2$ min
Simulation time	8 hours

B. *Unplanned architecture*

In order to highlight the effectiveness of our proposed network architecture, we compare our architecture with an *unplanned* network architecture that greedily utilizes the constrained network resources pool for deployment in the disaster area. Such an unplanned architecture largely represents the ad-hoc decision making in rapidly deploying the network resources; in order to form a communication network in the disaster area (as discussed in Section I).

Let us assume the aforestated *resource-constrained* post-disaster scenario $G(V, E, R \cup C)$ with a limited network resource pool of $|R|$ DMs and $|C|$ communication towers, then the steps for the unplanned architecture is as follows:

1) First, choose one random SP (out of $|V|$ SPs) as the MCS and place a *communication tower* to it.
2) Second, allocate the rest ($|C|$-1) communication towers to the farthest ($|C|$ - 1) SPs from the MCS.
3) Third, distribute $|R|$ DMs uniformly to the $|C|$ SPs equipped with communication towers (including MCS), i.e., each such SP (or basically, a depot) will have $\lfloor \frac{|R|}{|C|} \rfloor$ DMs, except the last one which constitutes the remaining DMs. (Notice that this step is similar to that of DM distribution to each center in our proposed heuristic.)
4) Fourth, each DM is uniformly assigned a $\left\lfloor \frac{|V|-|C|}{|R|} \right\rfloor$ subset of SPs, which does not have a communication tower, and are closest to the depot for that DM.

---

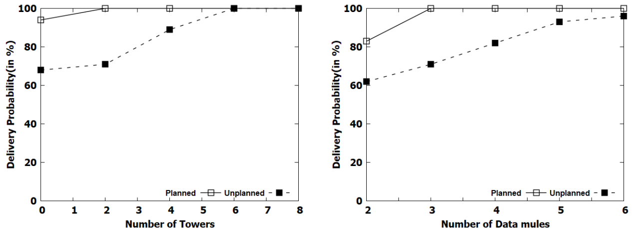
[8] Google limits the extraction of pathways to 10 routes and 23 waypoints (nearly 0.75×0.75 sq. Km map area) per client user. Hence, we developed a distributed algorithm for generating a complete path information, while utilizing limited information from Google Maps, via four clients.

5) Fifth, each of the remaining SPs, say i with no communication towers (if any) are assigned to the nearest SP subset (corresponding to a certain DM). Now, every DM travels to each SP in the assigned SP subset via the shortest path, and finally returns to the depot.

C. Experimental Results

In this section, we present the comparative analysis of the planned network architecture against the unplanned architecture in terms of aforementioned performance metrics (i.e., Network latency and Delivery Probability) against two important parameters: (i) *varying number of data mules*, (ii) *varying number of communication towers*, in the disaster area. These parameters represent the limited network resource pool that may be available for forming a temporary communication network in a given post-disaster scenario.

1) Delivery Probability Analysis: Fig. 3(a) shows that the proposed planned architecture greatly outperforms the unplanned approach, for *constrained* (< 6) number of available communication towers in the disaster area. This is because, unlike *greedy* approach utilized by unplanned architecture, the planned architecture utilizes the proposed heuristic that intelligently determines the optimal set of SPs for the placement of communication towers in the disaster area. With ≥ 6 communication towers, the delivery probability for both planned and unplanned approach are close to 100%, because there are sufficiently large number of communication towers for 8 shelter points in the disaster area (in addition to 3 data mules, as discussed in the simulation set up), to successfully deliver all the generated (both Request and Response) messages.

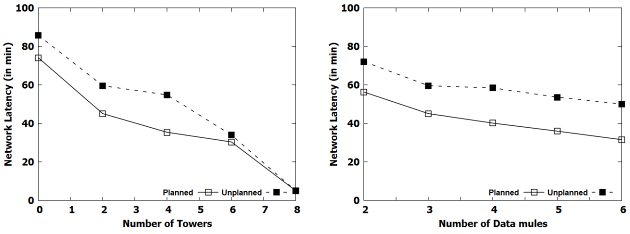


(a) (b)

Fig. 3. Delivery Probability vs Number of (a) Towers, and (b) Data mules

Intuitively enough, as shown in Fig. 3(b), the delivery probability for both planned and unplanned architectures improve with increase in availability of data mules in the disaster area. However, it is noteworthy that the planned architecture greatly outperforms the unplanned one in terms of delivery probability, for *constrained* ($\leq$ 5) number of data mules in the disaster area. This is again because, unlike unplanned one, the proposed heuristic for the planned architecture rationally determines the suitable route trajectories for each data mule.

2) Network Latency Analysis: As illustrated in Fig. 4(a), the end-to-end network latency incurred in the planned architecture is significantly lower than that of the unplanned architecture, especially for limited (< 6) number of communication towers in the disaster area. Moreover, Fig. 4(b) shows that the proposed architecture greatly outperforms the unplanned one, even for varying number of data mules. The reason is that the the proposed heuristic for the planned architecture aims at deploying the constrained set of communication towers (and also data mules) in the disaster area, in such a way that the end-to-end network latency is minimized.



(a) (b)

Fig. 4. Network Latency vs Number of (a) Towers, and (b) Data mules

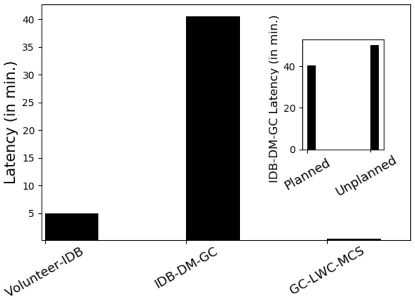


Fig. 5. Mean latency for various communication links between (i) Volunteer and IDB (Volunteer-IDB), (ii) IDB and Group Center via a DM (IDB-DM-GC), and (iii) Group Center and MCS via LWC towers (GC-LWC-MCS)

In Fig. 5, we show, for the planned architecture, the mean latency incurred in various communication links – between (i) *Volunteer and IDB* over Type 1 interface, denoted by (Volunteer-IDB), (ii) *IDB and group center* via a patrolling data mule (over Type 2 interface), denoted by IDB-DM-GC), and finally (iii) *Group center and MCS* via a communication tower (over Type 3 interface). From Fig 5, it is clearly evident that the major portion of the total end-to-end network latency (i.e., between volunteers and the MCS) is due to the latency incurred in exchanging messages between the IDB and the patrolling data mule. Whereas the incurred network latency in message transmission over communication towers is negligible, which further justifies our assumption in Section II (i.e., the incurred network latency in message exchange over communication towers between any two SPs is much smaller than physically carrying the messages via a data mule.).

Also notice the inset plot in Fig. 5, which shows a significant difference in the latency incurred by (IDB-DM-GC) communication links between planned and unplanned architectures. We do not show the plots for other two (Volunteer-IDB) and (GC-LWC-MCS) links for unplanned architecture, as the incurred latency is similar to that of planned architecture.

Discussion. From the above experimental analysis, it is clear that the proposed network architecture clearly outperforms the unplanned architecture, in terms of both the performance metrics, for varying limited network resource pool in the disaster area. Hence, we conclude that the proposed architecture is a promising approach to set up a temporary communication network in resource-constrained post-disaster environments.

VI. CONCLUSION

In this paper, we proposed a novel network architecture specifically designed for resource-constrained post-disaster environments. The proposed architecture intelligently allocated the limited network resource pool such that the end-to-end network latency between the volunteers and the MCS is minimized, while ensuring that each shelter point (and hence, survivor/volunteer) is served by at least one network resource. We formulated the resource allocation problem as a non-linear programming optimization problem, and showed that such a problem is NP-hard. Then, we proposed a simple yet effective sub-optimal planned heuristic that solves the above problem in polynomial-time. Compared to an unplanned approach, our extensive experiments based on the real map of Durgapur, India on top of the ONE simulator, indicated that the proposed planned architecture improves upon both the metrics, i.e., delivery probability and network latency, in all considered resource-constrained post-disaster environments.

As a part of our future work, we intend to create a small testbed surrounding our university (NIT, Durgapur) and simultaneously, conduct large-scale simulation experiments to further validate the effectiveness of our proposed architecture. We would also like to extend our network architecture to take into account the *time-evolving* nature of post-disaster scenarios, for instance, the movement of data mules, and the existence of pathways in the area etc. may evolve over time. Additionally, the wireless devices (including towers and IDBs in the area), may suffer from additional issues, such as, energy scarcity, memory overflow, device failures etc. over time, which we intend to address in future as well.

VII. ACKNOWLEDGEMENT

This Publication is an outcome of the R&D work undertaken partially in the ITRA, Media Lab Asia project entitled "Post-Disaster Situation Analysis and Resource Management Using Delay-Tolerant Peer-to-Peer Wireless Networks (DISARM)", and partially in the project entitled "Developing ICT-based kiosks for post-disaster situational information management using opportunistic networking framework" funded by Department of Higher Education, Science & Technology and Biotechnology, Government of West Bengal, India. It is also partially supported by NATO grant G4936.

REFERENCES

[1] "[online] nepal earthquake response report two years and beyond: https://reliefweb.int/sites/reliefweb.int/files/resources/nepal earthquake - two years on - report (english).pdf."

[2] "[online] nepal earthquake 2015: Post disaster needs assessment vol. a : Key findings: https://reliefweb.int/report/nepal/nepal-earthquake-2015-post-disaster-needs-assessment-vol-key-findings."

[3] "[online] em-dat: The international disaster database https://www.emdat.be/."

[4] "[online] https://www.nytimes.com/interactive/2015/04/25/world/asia/nepal-earthquake-maps.html."

[5] "[online] https://www.theguardian.com/world/2015/apr/28/nepal-villagers-swarathok-aid-beyond-the-capital."

[6] J. Q. Bao and W. C. Lee, "Rapid deployment of wireless ad hoc backbone networks for public safety incident management," in *Global Telecommunications Conference, 2007. GLOBECOM'07. IEEE*. IEEE, 2007, pp. 1217–1221.

[7] V. K. Shah, S. Roy, S. Silvestri, and S. K. Das, "Ctr: Cluster based topological routing for disaster response networks," in *IEEE International Conference on Communications (ICC)*. IEEE, 2017, pp. 1–6.

[8] M. Y. S. Uddin, H. Ahmadi, T. Abdelzaher, and R. Kravets, "Intercontact routing for energy constrained disaster response networks," *IEEE Transactions on Mobile Computing (TMC)*, vol. 12, no. 10, pp. 1986–1998, 2013.

[9] S. M. George, W. Zhou, H. Chenji, M. Won, Y. O. Lee, A. Pazarloglou, R. Stoleru, and P. Barooah, "Distressnet: a wireless ad hoc and sensor network architecture for situation management in disaster response," *IEEE Communications Magazine*, vol. 48, no. 3, 2010.

[10] Z. Lu, G. Cao, and T. La Porta, "Teamphone: Networking smartphones for disaster recovery," *IEEE Transactions on Mobile Computing (TMC)*, vol. 16, no. 12, pp. 3554–3567, 2017.

[11] H. Nishiyama, T. Ngo, S. Oiyama, and N. Kato, "Relay by smart device: Innovative communications for efficient information sharing among vehicles and pedestrians," *IEEE Vehicular Technology Magazine*, vol. 10, no. 4, pp. 54–62, 2015.

[12] T. Sakano, Z. M. Fadlullah, T. Ngo, H. Nishiyama, M. Nakazawa, F. Adachi, N. Kato, A. Takahara, T. Kumagai, H. Kasahara *et al.*, "Disaster-resilient networking: a new vision based on movable and deployable resource units," *IEEE Network*, vol. 27, no. 4, pp. 40–46, 2013.

[13] G. Zussman and A. Segall, "Energy efficient routing in ad hoc disaster recovery networks," *Ad Hoc Networks*, vol. 1, no. 4, pp. 405–421, 2003.

[14] S. Saha, V. K. Shah, R. Verma, R. Mandal, and S. Nandi, "Is it worth taking a planned approach to design ad hoc infrastructure for post disaster communication?" in *Proceedings of the seventh ACM international workshop on Challenged networks*. ACM, 2012, pp. 87–90.

[15] S. Saha, S. Nandi, P. S. Paul, V. K. Shah, A. Roy, and S. K. Das, "Designing delay constrained hybrid ad hoc network infrastructure for post-disaster communication," *Elsevier Ad Hoc Networks*, vol. 25, pp. 406–429, 2015.

[16] K. Miranda, A. Molinaro, and T. Razafindralambo, "A survey on rapidly deployable solutions for post-disaster networks," *IEEE Communications Magazine*, vol. 54, no. 4, pp. 117–123, 2016.

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

[18] A. Keränen, J. Ott, and T. Kärkkäinen, "The one simulator for dtn protocol evaluation," in *Proceedings of the 2nd international conference on simulation tools and techniques*. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2009, p. 55.

[19] "[online] https://itsforkit.github.io/uploaded/surakshitLast accessed on May 14, 2018."

[20] A. Seth, D. Kroeker, M. Zaharia, S. Guo, and S. Keshav, "Low-cost communication for rural internet kiosks using mechanical backhaul," in *ACM International conference on Mobile computing and networking (MobiCom)*. ACM, 2006, pp. 334–345.

[21] N. Banerjee, M. D. Corner, and B. N. Levine, "Design and field experimentation of an energy-efficient architecture for dtn throwboxes," *IEEE/ACM Transactions on Networking (TON)*, vol. 18, no. 2, pp. 554–567, 2010.

[22] M. Y. S. Uddin, D. M. Nicol, T. F. Abdelzaher, and R. H. Kravets, "A post-disaster mobility model for delay tolerant networking," in *Winter Simulation Conference*. Winter Simulation Conference, 2009, pp. 2785–2796.

[23] K. L. Hoffman, M. Padberg, and G. Rinaldi, "Traveling salesman problem," in *Encyclopedia of operations research and management science*. Springer, 2013, pp. 1573–1578.

[24] Q. Zhang and I. Couloigner, "A new and efficient k-medoid algorithm for spatial clustering," in *International conference on computational science and its applications*. Springer, 2005, pp. 181–189.

[25] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, "Introduction to algorithms mit press," *Cambridge MA*, 1990.

[26] M. Romoozi, H. Babaei, M. Fathy, and M. Romoozi, "A cluster-based mobility model for intelligent nodes," in *International Conference on Computational Science and Its Applications*. Springer, 2009, pp. 565–579.

[27] A. Vahdat and D. Becker, "Epidemic routing for partially connected ad hoc networks," *Technical Report CS-200006, Duke University*, 2000.