CTR: Cluster based Topological Routing for Disaster Response Networks

Vijay K. Shah, Satyaki Roy, Simone Silvestri, and Sajal K. Das
Department of Computer Science, Missouri University of Science and Technology, MO 65409, USA
{vksr38, sr3k2, silvestris, sdas}@mst.edu

***Abstract*—Large scale disasters require prompt rescue and relief operations to restrict further casualties. To carry out such operations, it is essential to have a communication infrastructure between survivors and responders, which is often impaired due to the disaster. Off-the-shelf wireless devices such as smartphones, PDAs and Laptops offer an effective solution towards the establishment of makeshift communication infrastructure. However, in the absence of bonafide power sources, it becomes imperative to judiciously utilize energy (battery power) of such devices such that the network is functional until primary infrastructure is restored. This paper proposes a novel approach, called Cluster based Topological Routing (CTR) that prolongs the longevity of the network by exploiting the natural gathering of survivors in shelter points. In particular, the clustering algorithm identifies such survivor groups combined with a data forwarding approach, to minimize the number of data transmissions yet guaranteeing the required packet delivery and network latency. Our extensive simulation study shows that CTR yields twice the network lifetime than existing routing approaches in disaster response networks, while ensuring comparable packet delivery and network latency.**

I. INTRODUCTION

In the event of a large-scale disaster, natural (e.g., earthquake) or man-made (e.g., terrorist attacks), a communication infrastructure is essential for prompt rescue and relief operations. However, the partial (or complete) impairment of primary communication infrastructures makes it difficult for the responders to assess the required logistics for recovery efforts, identify and extend rescue/relief to the survivors.

Smart devices such as smartphones, PDAs and Laptops are often available among the survivors in disaster affected areas. For example, after Nepal earthquake in 2015 [11][12] and Haiti earthquake in 2010 [13], there were approximately 23 and 2.8 million active mobile subscribers out of 27 and 10 million inhabitants, respectively. Hence, several research efforts have been directed towards utilization of such smart devices to create ad-hoc communication network in post-disaster scenarios, also termed as *Disaster Response Networks* (DRN). These devices, characterised by limited transmission range and battery supplies essentially form a Delay or Disruption Tolerant Networks [16] with extreme energy constraints.

Smartphone based DRNs are basically opportunistic networks where nodes are mobile and the communication links exist temporarily. Hence, it is extremely challenging to establish end-to-end connections for data delivery. Therefore, existing approaches [1][2][3], have mostly focused on achieving high packet delivery at the expense of multiple data copying and forwarding (flooding message copies), thereby consuming significant amount of energy. In [4][5], intelligent routing approaches have been proposed which attempt to reduce the data copies by determining high delivery path for each message using history of encounters and contact patterns. However, in post disaster scenarios, it is difficult to determine such high delivery paths because there may not exist recurrent and regular contact patterns among survivors.

The work proposed in this paper is based on the key observation that there are naturally occuring survivor groups [6][15] in the form of shelter points, open places or evacuation centers in the aftermath of a disaster, even though the geographical location of such survivor groups may drastically change due to very nature of a disaster. For instance, in the event of an earthquake, the survivors are likely to reside at an open space, whereas during flood, people generally take refuge at relatively elevated lands or rooftops. Nonetheless, irrespective of the nature of disaster, there exist survivor groups that tend to remain stable mainly due to hostile outside environment.

We observe that the existence of survivor groups can be exploited because the survivors within these groups meet more frequently. Also, such survivor groups remain stable for a considerably longer time. To this end, we propose a novel *Cluster based Topological Routing* (CTR) approach that exploits these naturally existing survivor groups to enhance the longevity (energy efficiency) of DRNs. Our approach consists two logical steps: (i) identification of survivor groups, and (ii) data forwarding approach. Such groups can be identified by employing clustering on the survivors. There are several clustering techniques for opportunistic networks [7][8][9][10], which are not a good fit for DRNs because they require highly connected network topology, timely information exchange among nodes, highly predictable mobility patterns, and existence of cliques in the network. The paper also proposes an adaptive and distributed Light-weight Clustering Algorithm (LCA) to determine such extant survivor groups. To the best of our knowledge, our research is the first effort to exploit such survivor groups in disaster scenarios and hence, investigate cluster based routing approach for establishing energy-efficient disaster response networks (DRN).

In summary, this paper makes the following contributions:

- We propose a Cluster based Topological Routing (CTR)

This work is partially supported by Defense Threat Reduction Agency (DTRA) grant HDTRA1-10-1-0085, North Atlantic Treaty Organization (NATO) under SPS grant G4936 "Hybrid Sensor Network for Emergency Critical Scenarios", NSF grants under award numbers CNS-1545037, CNS-1545050, DGE-1433659, and a grant from Intelligent Systems Center at Missouri S&T.

978-1-4673-8999-0/17/$31.00 ©2017 IEEE

which vastly reduces the communication overhead, thus improving the longevity of DRN while achieving the required packet delivery and network latency.

- We present an adaptive and distributed Lightweight Clustering Algorithm (LCA) to determine extant survivor groups in post disaster scenarios.
- We analyze proposed approach on a real disaster prone region in Kathmandu, Nepal through simulation study. The results show that CTR achieves twice the network lifetime than existing approaches while guaranteeing comparable packet delivery and network latency.

The rest of the paper is structured as follows: Section II discusses the system model. The problem is formulated in Section III. Section IV discusses the details of CTR. We discuss simulation results in Section V followed by conclusions in Section VI.

II. NETWORK MODEL AND ASSUMPTIONS

Let us consider a post disaster scenario where the affected people termed as *survivors*, gather in common geographical locations known as *neighborhoods* [6][12], which may include safe places such as schools, parks, office buildings, evacuation centers and shelter points. These neighborhoods tend to remain stable over a considerable time period since survivors, though mobile, rarely tend to leave their neighborhoods due to unsafe outside environment. Such neighborhoods are periodically or aperiodically patrolled by members of medical, police and disaster response teams, called *responders*. These responders report to a *coordination center*, which coordinates entire rescue and relief operations.

We assume that survivors are equipped with smart devices such as smartphones, PDAs and tablets which are defined as the set of *survivor nodes* $S = \{s_i | \forall i = 1, 2 \ldots |S|\}$. Each survivor s_i has an application installed on his device that allows him to (a) establish communication with peers directly through WiFi-Direct or Bluetooth (ad-hoc mode), and (b) exchange situation or rescue/relief need-based information in the form of text, image, audio and video clips. We also define two other entities: (i) a set of responders $R = \{r_k | \forall k = 1, 2, \ldots, |R|\}$ equipped with smart devices, and (ii) a coordination center equipped with a smart device as the *coordination node*, c. (The proposed approach can easily be extended to multiple coordination nodes.)

A. Network Model

We model DRN as a graph $G(V, L)$, where the node set $V = S \cup R \cup c$. The graph G evolves over time as nodes move within the neighborhood. The link set L is a triplet $L = L^{ss} \cup L^{sr} \cup L^{rc}$, where $L^{ss} = \{l_{ij}^{ss}, 1 \leq i, j \leq |S|, i \neq j\}$ corresponds to the set of communication links between the survivors; $L^{sr} = \{l_{ij}^{sr}, 1 \leq i \leq |S|, 1 \leq j \leq |R|\}$ corresponds to the set of communication links between the survivors and responders; and $L^{rc} = \{l_{ic}^{rc}, 1 \leq i \leq |R|\}$ corresponds to the set of communication links between the responders and the coordination node. For simplicity, we consider a single neighborhood in G although our approach can be applied to DRN with multiple neighborhoods.

Node Structure: A survivor node s_i may not be in direct communication with the coordination node c for they may be out of communication range. Additionally, the neighborhood is generally large [12], which makes it impossible to establish direct communication between every pair of survivors within the neighborhood. Consequently, each survivor s_i would have a higher likelihood of meeting proximate nodes rather than the distant nodes in the same neighborhood. We assume each s_i generates data packets at a rate of λ packets per minute. The responders patrol the neighborhood, carry out the delivery of data packets from the survivor nodes, and report to the coordination node c.

Link Structure : We define a metric called *contact fitness* ranging between 0 and 1, to quantify the link reliability in G. Specifically, cF_{ij} is the contact fitness between survivors s_i, $s_j \in S$ if cF_{ij} exceeds a threshold cF_{th} (explained later in Section IV). Similarly, a responder contact fitness is defined between survivors and responders. Given a survivor s_i and a responder r_j, a link exists if the responder contact fitness rF_{ij} is greater than a threshold, rF_{th}. We assume that every r_j aperiodically or periodically reports to c, making it imperative to have a communication link between them.

In this paper, we assume that each survivor $s_i \in S$ has a finite energy (battery power) capability. Let $E_{init}(i)$ and $E_{res}(i)$ be the initial and residual energies of node s_i, respectively. In contrast, we assume that each responder r_j and coordination node c have no energy constraints due to steady access to power sources. Additionally, we assume all s_i, r_j and c have sufficient storage, given the fact that the generated data is usually in the order of kilobytes while these devices have gigabytes of memory.

III. PROBLEM FORMULATION

In existing approaches [1][2][3], typically a survivor tends to communicate directly or indirectly with the responder resulting in multiple data transmissions and greater energy consumption. However, due to possible existence of stable neighborhoods and higher likelihood of encounters among nodes within same neighborhood, it is possible to reduce the communication overhead by considering each neighborhood as a single entity and electing a subset of nodes to communicate directly with responders on behalf of the neighborhood. We term such elected nodes as the **exemplar** set while all survivor nodes communicating with a unique exemplar node are called **non-exemplars**. In this paper, an exemplar together with all non-exemplars assigned to it, will be called a **cluster**. Note that a single neighborhood may constitute multiple clusters depending on its area of coverage and population size.

Thus given a time evolving DRN graph G, our goal is to partition a neighborhood into a suitable set of clusters that minimizes communication overhead and maximizes network lifetime while meeting packet delivery and network latency requirements.

The determination of set of clusters in a neighborhood poses following two challenges:

(i) The graph G evolves over time due to the mobility of survivor nodes, and thus the set of clusters should also adapt to the dynamic network topology.

(ii) Every survivor node can only communicate with the nodes in its transmission range, and hence can not acquire global information about the entire network. Therefore, a distributed solution is required, where each node autonomously determines its cluster.

The DRN, being a time evolving graph, the total time duration has been slotted into H ($1 \leq h \leq H$) identical time slots, each of duration T where the h^{th} time slot is denoted by t_h. Given above challenges, our problem can be stated: **Determine a set of clusters at each time slot t_h that maximizes network lifetime, while meeting the required packet delivery and network latency.**

Given K clusters and a set of survivor nodes S, we observe that if $K \approx 1$, the network lifetime is improved but the network may fail to meet the required packet delivery and latency since it is unlikely for an exemplar to communicate with every non-exemplars by itself. Even if it does so, it is unlikely to meet them often, causing long communication delays. In contrast, if $K \approx |S|$, we expect significant improvement in the network lifetime. Since in this scenario, all survivor nodes are exemplars, there will be no communication among them. However, such network is bound to fail the required packet delivery and network latency as it is infeasible for responder to communicate with every survivor node directly. Hence, the desired value of K should lie between 1 and $|S|$.

At each timeslot t_h, we formulate the aforestated problem as an *integer linear optimization* problem where **the objective is to minimize the number of exemplars while maximizing link reliability between each exemplar and its corresponding non-exemplars (or clusters)**. This will enhance network lifetime while assuring the required packet delivery and network latency.

Consider $X = \{x_1, x_2, ..., x_K\} \subset S$ be the exemplar set and $\bar{X} = S \setminus X$ be the non-exemplar set. We refer to $D(i)$ as the total amount of generated data packets at a survivor node s_i. Moreover, $E_{tx}(i)$ and $E_{rx}(i)$ denote the energy required for s_i to transmit and receive a data packet, respectively.

Let $e_{ij} = 1$ if a link exists between s_i and s_j i.e. $cF_{ij} \geq cF_{th}$, otherwise $e_{ij} = 0$. Also, let $e'_{ij} = 1$ if there exists a link between s_i and r_j i.e. $rF_{ij} \geq rF_{th}$, otherwise $e'_{ij} = 0$. Consider a variable $y_i = 1$ if s_i is elected as an exemplar, else $y_i = 0$. Similarly, a variable $z_{ij} = 1$ if non-exemplar $\bar{x}_j$ is assigned to an elected exemplar x_i, otherwise $z_{ij} = 0$. Now, the optimization problem can be stated as follows:

$$\textit{Minimize} \qquad \sum_{i=1}^{|S|} y_i + \sum_{i=1}^{|S|} \sum_{j=1}^{|S|} (1 - cF_{ij}) z_{ij} \tag{1}$$

subject to

$$\sum_{i=1}^{|S|} z_{ij} = 1, \qquad \sum_{r=1}^{|R|} y_i \times e'_{ir} \geq 1 \qquad \forall s_j \tag{2}$$

$$\begin{gathered} \sum_{j=1}^{|S|} z_{ij} \times E_{rx}(i) \times D(j) + \sum_{r=1}^{|R|} e'_{ir} \times E_{tx}(i) \times \\ \left(D(i) + \sum_{j=1}^{|S|} z_{ij} \times D(j) \right) \leq E_{res}(i) \qquad \forall s_i \end{gathered} \tag{3}$$

$$z_{ij} \geq e_{ij} \times y_i \qquad \forall\, s_i, s_j \in S \tag{4}$$

The first part of objective function shown in expression (1) represents the set of exemplars while the second part represents link reliability (defined by cF_{ij}) between an exemplar and its corresponding non-exemplars. Minimization of $(1 - cF_{ij})$ has the same implication as maximization of cF_{ij}. Expression (2) restricts a non-exemplar node to belong to a unique exemplar node and also constraints each exemplar to have a link with at least one responder node. Constraint in Eq.(3) enforces that the residual energy at an exemplar must be greater than or equal to energy required in receiving data packets from all assigned non-exemplars and transmitting data packets to the communicating responders. Inequality (4) ensures that a link exists between two survivor nodes only when the link is reliable and one of the nodes is an exemplar.

Intuitively, the above optimization problem will provide a centralized optimal solution, which may not be feasible in disaster scenarios, because (i) There is no centralized node, hence determination of the set of clusters must be performed distributively, (ii) Each node only has local information so the cluster set has to be determined in the absence of global information, and (iii) The nodes are mobile and hence, the cluster set may change over time to adapt to the evolving network topology. This implies that the optimization problem has to be solved multiple times incurring very high complexity.

Thus, the first step of CTR is to propose a light-weight clustering algorithm (LCA) described in Section IV-B which adaptively and distributively determines the best set of clusters and exemplars at any given timeslot t_h.

IV. CLUSTER BASED TOPOLOGICAL ROUTING

The proposed approach *Cluster based Topological Routing* (CTR) consists of two logical steps: (i) determining exemplar and cluster sets i.e., network topology using LCA at every time slot t_h; and (ii) data forwarding to deliver data packets from survivor nodes to the coordination node. Before proceeding any further, let us define the performance metrics and fitness parameters used in our approach.

The important performance metrics are as follows.

Packet Delivery Ratio: Fraction of the total received unique data packets at the coordination node c to the total generated unique data packets at any survivor node $s_i \in S$.

Network Latency: Maximum incurred delay to deliver any data packet from a node $s_i \in S$ to the coordination node c.

Network Lifetime: Instead of defining network lifetime as the time until the first node dies out [14], we define it as the time until which $X\%$ of all survivor nodes die out. In our simulation study, we consider $X = 50\%$.

Overhead Ratio: Ratio of the total number of additional message copies transferred in the network to the total number of unique message copies delivered.

A. Fitness Parameters

Given a DRN graph G at a timeslot t_h, if a node $s_i \in S$ belongs to a cluster with exemplar x_j, then they should meet quite often and for a sufficient length of time (to exchange data). Hence, let us define the following fitness parameters.

1) ***Contact Fitness*** *(cF_{ij}):* The total contact duration between any two survivor nodes $s_i, s_j \in S$ from the beginning until the current time slot. In a time slot t_h with duration T, let F be the number of times s_i meets s_j and d_f be the contact duration at f^{th} encounter then, contact fitness between s_i and s_j in time slot t_h, is given by $cF_{ij}^h = \frac{\sum_{f=1}^{F} d_f}{T}$.

To calculate cF_{ij}, we use an Exponential Weighted Moving Average (EWMA) in which s_i maintains cF_{ij} for every s_j it has met before. Then, cF_{ij} is updated at the end of every time slot as $cF_{ij} = (1-\alpha)cF_{ij}^{pre} + cF_{ij}^h\alpha$, where α is a constant between 0 and 1, and cF_{ij}^{pre} denotes the contact fitness until the previous time slot. In this approach, a link exists between any two survivors s_i and s_j if it exceeds a threshold, cF_{th}.

2) ***Responder Contact Fitness*** *(rF_{ij}):* The total contact duration between survivor $s_i \in S$ and responder $r_j \in R$ from the beginning until the current time slot. Let the responder contact fitness in current time slot, t_h and from the beginning until t_h be denoted as rF_{ij}^h and rF_{ij}, then, they can be calculated in similar way as cF_{ij}^h and cF_{ij}, respectively. Likewise a link exists between s_i and r_j if it exceeds a threshold, rF_{th}. Let $rF(i)$ denotes the overall responder contact fitness between s_i and any responder, then $rF(i) = \sum_j^{|R|} rF_{ij}, \forall r_j$.

3) ***Weighted Fitness*** *($wF(i)$):* The suitability of a survivor node s_i to be chosen as an exemplar in the current time slot. We consider the following three metrics to calculate $wF(i)$.

The first metric is *Residual Energy*, $E_{res}(i)$ which is the remaining energy at current time slot, which is a normalized percentage between 0 and 1. The second metric is *Energy Consumption Rate*, $ECR(i)$ which is the ratio of energy difference between previous and current time slots to the duration of time slot T. For instance, at time slot t_1, consider two nodes s_i and s_j having $E_{res}(i) = 0.7$, $E_{res}(j) = 0.5$, $ECR(i) = 10$ joule/hr and $ECR(j) = 1$ joule/hr. Here, despite less residual energy, s_j is a better candidate for exemplar by virtue of its low energy consumption rate. In other words, both $E_{res}(i)$ and $ECR(i)$ contribute to the fitness of a survivor node to be elected as an exemplar. Finally, the third metric is *Node Connectivity*, $\kappa(i)$ which is the number of communication links possessed by node s_i. This metric ensures that the chosen exemplar is a well connected node which can serve a greater number of non-exemplars.

In addition, the chosen exemplar should have a link with at least one responder $r_j \in R$. Based on the above metrics, it is evident that a survivor node s_i is the best candidate for an exemplar among all of its neighbors if it has the highest $E_{res}(i)$, lowest $ECR(i)$, highest $\kappa(i)$, and also shares a link with a responder i.e. $rF(i) \geq rF_{th}$. However, since these metrics have different units, we apply inverse exponential function to normalize $ECR(i)$ and $\kappa(i)$ and bound their values between 0 and 1. Hence, $wF(i)$ is calculated as:

$$wF(i) = \begin{cases} \beta_1 E_{res}(i) e^{-ECR(i)} + \beta_2 e^{-\frac{1}{\kappa(i)}}, & \text{if } rF(i) \geq rF_{th} \\ 0, & \text{otherwise} \end{cases} \tag{5}$$

where β_1 and β_2 are weighing factors and $\beta_1 + \beta_2 = 1$. A node s_i, once elected, continues to act as an exemplar unless its $wF(i)$ is greater than or equal to the threshold, wF_{th}. It is important to filter out undesirable re-election of exemplars.

B. Light-weight Clustering Algorithm (LCA)

LCA dynamically determines a set of exemplars (and their corresponding clusters) at the end of each time slot in an adaptive and distributed fashion. The proposed algorithm undergoes three major steps. First, each survivor node s_i computes its weighted fitness, $wF(i)$ and contact fitness, cF_{ij} with every s_j it meets. Also, it estimates the responder contact fitness, $rF(i)$ with the meeting responder node $r_j \in R$. Second, each s_i elects itself as an exemplar if there exists no better candidate in its proximity. Third, each s_i assigns itself as an exemplar to all its neighbors s_j if s_i has a reliable link with s_j i.e., $cF_{ij} \geq cF_{th}$. The exemplar s_i with its assigned neighbors (non-exemplars) constitute a cluster.

1) Local Clustering Information: A survivor node s_i locally maintains its node ID (i), its exemplar ID ($X(i)$), weighted Fitness ($wF(i)$), responder contact fitness ($rF(i)$), current timestamp ($T(i)$) and exemplar-cluster table ($EC(i)$). Entries correspond to any survivor node s_k which s_i has ever met. Thus, entry for node s_k contains its ID (k), exemplar ID (X_i^k), contact fitness (cF_{ik}) with node s_i, weighted fitness (wF_i^k), responder contact fitness (rF_i^k) and timestamp (T_i^k) which is the most recent timestamp when s_i meets s_k. It is to be noted that the parameters for an entry of s_k at $EC(i)$ are updated according to the best knowledge of node s_i, and may not always be accurate. The members of any cluster with exemplar x at node s_i can be obtained as $C_i^x = \{k | X_i^k = x\}$.

2) Clustering Operations: Initially, each survivor s_i assigns itself as its exemplar i.e., $X(i) = i$ and creates an empty EC table. Node s_i performs a set of operations on two distinct events, namely Slot Timeout event and Meeting Node event.

a) ***Slot Timeout Event****:* At the end of every time slot, each s_i updates $ECR(i)$, $E_{res}(i)$ and $\kappa(i)$, and then, recomputes $wF(i)$. It also updates cF_{ik} for every entry s_k in its EC table, as well as $rF(i)$, as defined earlier. Following these updates, two cases may occur:

Case a.1: Consider that exemplar of s_i is s_k. If contact fitness between s_i and s_k goes below the threshold, then s_i updates its exemplar as itself. Notationally, if $X(i) = k$ and $cF_{ik} < cF_{th}$, then update $X(i) = i$

Case a.2: Consider an entry node s_k in $EC(i)$ such that exemplar of s_k is s_i. If contact fitness between s_i and s_k goes below threshold or weighted fitness of s_i goes below threshold, then, s_i updates exemplar for s_k as s_k. Notationally, if $X_i^k = i$ and ($cF_{ik} < cF_{th}$ or $wF(i) < wF_{th}$), then update $X_i^k = k$

At the end of the slot timeout event, each s_i maintains the most updated values of fitness parameters and its exemplar.

b) ***Meeting Nodes Event****:* When two survivor nodes s_i and s_j meet, the following cases may occur.

Case b.1: Consider there exists no entry for s_j in $EC(i)$, then create a new entry for s_j with its information.

Case b.2: Consider there already exists an entry for s_j in $EC(i)$, then the following updates are performed. If timestamp at s_j is more recent than that of entry node s_j in $EC(i)$, then s_i updates weighted fitness for entry node s_j as $wF(j)$. Similarly, s_i also updates exemplar of s_j as $X(j)$. Notationally, if $T(j) > T_i^j$, then update $wF_i^j = wF(j), X_i^j = X(j)$. However, if timestamp at s_j is older than that of entry node s_j in $EC(i)$, then s_j updates its exemplar as X_i^j. Notationally, if $T(j) < T_i^j$, then update $X(j) = X_i^j$.

In both these cases, s_i updates timestamp for entry node s_j in $EC(i)$ as current timestamp. Similarly, s_j does the same.

Case b.3: Consider an entry node s_k in $EC(j)$ also exists in $EC(i)$. Now, if timestamp for s_k in $EC(i)$ is older than that in $EC(j)$, then s_i updates exemplar of s_k in $EC(i)$ to that of s_k in $EC(j)$. It also updates weighted fitness of s_k in $EC(i)$ to weighted fitness of s_k in $EC(j)$. Finally, it updates timestamp for s_k as current timestamp T_{curr}. Notationally, if $T_i^k < T_j^k$, then update $X_i^k = X_j^k$, $wF_i^k = wF_j^k$ and $T_i^k = T_{curr}$.

Case b.4 (Special Case): Both meeting nodes s_i and s_j are exemplars and their contact fitness is greater than threshold. Notationally, $X(i) = i$, $X(j) = j$ and $cF_{ij} \geq cF_{th}$. Then, the following sub-cases may occur.

Case b.4.1: Node s_j and its members can be assigned to s_i. However, s_i and its members cannot be assigned to s_j, then assign s_j and its cluster members to s_i. Notationally, if $cF_{ik} \geq cF_{th}, \forall k \in C_j^{X(j)}$ and $\exists k' \in C_i^{X(i)}$ s.t. $cF_{ik'} < cF_{th}$, then update $X(j) = i$, $X_j^k = i$, $T_j^k = T_{curr}$, $\forall k \in C_j^{X(j)}$

Case b.4.2: If both s_i and s_j, along with their respective cluster members, can be assigned to each other, then the node with higher cluster stability is chosen as the exemplar to all other nodes. Stability of cluster refers to the minimum contact fitness value of its exemplar to any of its member. Notationally, if $cF_{jk} \geq cF_{th}, \forall k \in C_i^{X(i)}$, $cF_{ik'} \geq cF_{th}, \forall k' \in C_j^{X(j)}$ and $min(cF_{jk}) \leq min(cF_{ik'}), \forall k \in C_j^{X(j)}, \forall k' \in C_i^{X(i)}$, then update $X(j) = i, X_j^k = i, T_j^k = T_{curr}\ \forall k \in C_j^{X(j)}$

Since the clustering algorithm is distributed, any two nodes s_i and s_j may maintain different cluster information even if they have the same exemplar. Hence, these cases are imperative to synchronize their cluster information to the latest timestamp. Additionally, they also synchronize the values for fitness parameters at each node s_i for all its meeting nodes s_j to the latest timestamp.

C. Data Forwarding Approach

This section presents the second step of CTR wherein we lay out the rules for data communication among different nodes in the DRN. Fig. 1 illustrates three kinds of communications allowed in our approach. First, data forwarding is permitted from each non-exemplar node $\bar{x_j} \in \bar{X}$ to its exemplar $x_i \in X$. Second, x_i communicates directly with the responders $r_k \in R$, it shares a link with. Finally, responders communicate directly with the coordination node c. In Fig. 1, a set of non-exemplars ($\bar{x_1}$, $\bar{x_2} \ldots \bar{x_i}$) communicate directly with their unique exemplar x_1, which in turn communicates directly with responders r_1 and r_R, which then directly forwards data to c.

We believe direct transmission of data packets from $\bar{x_j}$ to x_i in any cluster should be sufficient to ensure the required packet delivery and network latency. However, values of control parameters cF_{th} and wF_{th} should be decided considering the aforementioned requirements. Analysis of cF_{th} and wF_{th} is given in section V-A3. Similar argument holds for direct transmission of data packets from x_i to r_j.

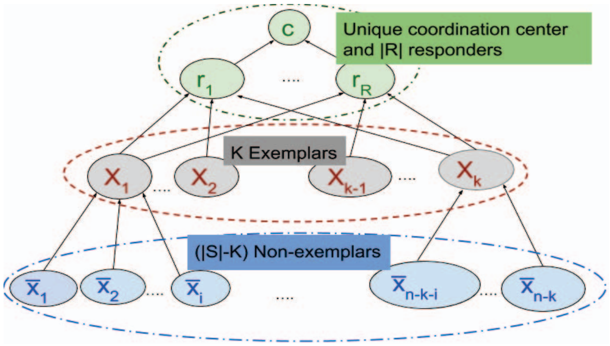


Fig. 1. Data Forwarding Approach

V. PERFORMANCE EVALUATION

We simulate the proposed approach using Opportunistic NEtwork Simulator (ONE) [17] simulator for an area 2 km ×2 km in a real disaster prone region in Kathmandu, Nepal. We have created a recognizable abstraction of earthquake map (replicating shelter points and camps) rather than attempting high degree of realism. All the experiments, unless otherwise stated, are performed with one coordination center, three responders and five neighborhoods, each with 20-60 survivors.

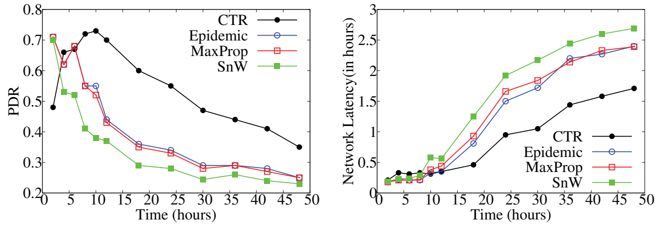
To model survivor mobility, the survivors are confined to move within the boundary of their respective neighorhoods. However, very few of them may move from their neighborhoods to another. For the responders, we consider that a responder move back and forth from the coordination node to its neighborhoods along map based paths.

For data traffic model, we consider that data packets are generated at each survivor node at a rate of one packet per 2-3 minutes. Each generated packet size lies in between 50 to 500Kb. The data packet TTL (time-to-live) is taken as 6 hours. For energy consumption model, we consider the energy consumed in transmitting/receiving data packets per second and scanning other devices per minute as 0.9 and 0.3 joules, respectively. Finally, the simulation duration is 48 hours.

A. Simulation Results

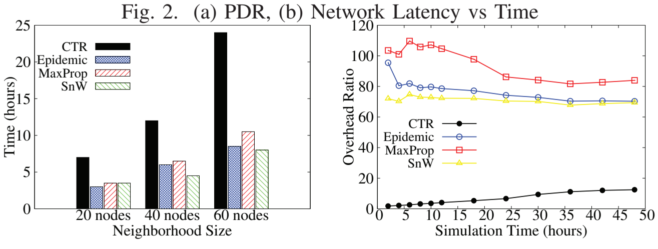
In this section, we evaluate and compare the performance of CTR with three existing approaches (i) Epidemic [1], (ii) MaxProp [3], and (iii) Spray and Wait (SnW) [4] in terms of the following performance metrics: (1) Packet Delivery Ratio (PDR), (2) Network Latency, and (3) Network Lifetime. In our simulation, we have considered network lifetime as the time until 50% nodes die out. Moreover, the values of cF_{th}, wF_{th} and rF_{th} are taken as 0.2, 0.1 and 0.1 respectively.

1) ***Analysis of PDR and Network Latency:*** In early time periods of CTR, each survivor is its own exemplar implying that the only way to transfer information from the survivor to the responder is direct communication. Since it is unlikely for responders to meet every survivors, CTR incurs poor PDR (upto 4 hours), as shown in Fig. 2(a). However beyond that time, when exemplars are elected (and clusters formed), more



(a) (b)

Fig. 2. (a) PDR, (b) Network Latency vs Time



(a) (b)

Fig. 3. (a) Network Lifetime vs Time (b) Overhead Ratio vs Time

survivors communicate with responders via their respective exemplars, resulting in substantial improvement (almost 33%) in PDR. Conversely, existing approaches allow multiple data transmissions; although they exhibit better PDR initially (upto 7 hours), as time progresses, PDR suffers because nodes start dying due to high communication overhead.

Fig. 2(b) shows that CTR improves network latency by almost 25% over existing approaches. It is intuitive because CTR ensures at most two hops data transmission from each survivor to responder as illustrated in Fig. 1.

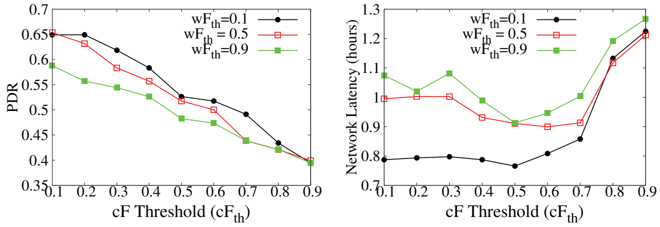
*2) **Analysis of Network Lifetime**:* Fig. 3(a) shows that the network lifetime in CTR is almost twice compared to existing approaches for all neighborhood sizes. The improvement in lifetime is largely due to reduction in communication overhead (see Fig. 3(b)). Unlike existing approaches, CTR yields similar improvement in network lifetime for different neighborhood sizes, owing mainly to lower communication overhead.

*3) **Analysis of** cF_{th} **and** wF_{th}:* Figs. 4(a) and 4(b) show that a lower cF_{th} value yields better PDR and network latency. This is because a lower cF_{th} will enable exemplars to serve more non-exemplars, which in turn allow more survivors to communicate with responders directly or indirectly via exemplars. Whereas for higher cF_{th} value, the exemplars will be restricted to serving only a few well-connected non-exemplars, hence an overwhelming majority of non-exemplars will be denied an opportunity to communicate with the responders. Similar to cF_{th}, a low rF_{th} value is preferred as it enhances the existence of communication links between survivors and responders. On the other hand, a large rF_{th} restricts the existence of links to a very few well-connected survivors and responders. (Plot for rF_{th} not shown due to space constraint.)

Figs. 4(a) and 4(b) show that a low wF_{th} yields better PDR and network latency because it facilitates multiple nodes to act as exemplars, thus improving the likelihood of forming clusters.

VI. CONCLUSION

In this paper, we propose a Cluster based Topological Routing (CTR) which exploits inherently extant survivor groups in a given disaster area. Our proposed approach improves network lifetime by two folds compared to existing non-clustering counterparts, while guaranteeing comparable packet delivery and network latency. One of the emergent properties visible in our simulation results is that CTR can be scaled up to arbitrarily large neighborhood sizes. As a future work, we plan to extend this approach to more challenging scenarios where the responders are few and unlikely to visit every survivors groups. We will also analytically investigate the contact fitness, responder contact fitness and weighted fitness thresholds.



(a) (b)

Fig. 4. (a) PDR vs cF_{th} (b) Network Latency vs cF_{th}

REFERENCES

[1] Vahdat A., and Becker D., *Epidemic Routing for Partially Connected Ad Hoc Networks*, Duke Tech Report CS-2000-06, 2000.

[2] Lindgren A., Doria A., and Schelén O., *Probabilistic Routing in Intermittently Connected Networks*, ACM SIGMOBILE mobile computing and communications review, 2003.

[3] Burgess J., Gallagher B., Jensen D., and Levine B.N., *MaxProp: Routing for Vehicle-Based Disruption-Tolerant Networks*, IEEE Int'l Conf. on Computer Communications (INFOCOM), 2006.

[4] Spyropoulos, T., Psounis K., and Raghavendra C. *Spray and wait: an efficient routing scheme for intermittently connected mobile networks.*, ACM SIGCOMM workshop on Delay-tolerant networking, 2005.

[5] Uddin M.Y.S., Ahmadi H., Abdelzaher T., and Kravets, R., *Intercontact Routing for Energy Constrained Disaster Response Networks*, IEEE Transactions on Mobile Computing, 2013.

[6] Uddin M.Y.S., Nicol D.M., Abdelzaher T.F., and Kravets R.H., *A post-disaster mobility model for Delay Tolerant Networking*, Winter Simulation Conference (WSC), 2009.

[7] Chatterjee M., Das S. K., and Turgut D., *WCA: A Weighted Clustering Algorithm for Mobile Ad hoc Networks*, Cluster Computing, 2002.

[8] Zhaowen X., Gruenwald L., and Phang K.K., *A Robust Clustering Algorithm for Mobile Ad Hoc Networks*, Handbook of Research on Next Generation Networks and Ubiquitous Computing, 2008.

[9] Dang H., and Wu H., *Clustering and ClusterBased Routing Protocol for Delay Tolerant Mobile Networks*, IEEE Transactions on Wireless Communications, 2010

[10] Dabideen S., Kawadia V., and Nelson S.C., *CLAN: An Efficient Distributed Temporal Community Detection Protocol for MANETs*, IEEE Int'l Conf. on Mobile Ad hoc and Sensor Systems (MASS), 2014.

[11] *https://www.ushahidi.com/blog/2015/04/25/supporting-online-volunteer-response-to-the-nepal-earthquake/*

[12] *https://data.hdx.rwlabs.org/group/nepal-earthquake*

[13] Bengtsson L., Lu X., Thorson A., Garfield R., and Schreeb J.V., *Improved response to disasters and outbreaks by tracking population movements with mobile phone network data: A post-earthquake geospatial study in haiti*, Public Library of Science (PLoS) medicine, 2011.

[14] Zussman G., and Segall A., *Energy efficient routing in ad hoc disaster recovery networks*, IEEE Int'l Conf. on Computer Communications (INFOCOM), 2003.

[15] Lu Z., Cao G., and LaPorta T., *TeamPhone: Networking Smartphones for Disaster Recovery*, IEEE Int'l Conf. on Pervasive Computing and Communications (PerCom), 2016.

[16] Fall K., *A delay-tolerant network architecture for challenged internets*, ACM Int'l Conf. on Applications, technologies, architectures, and protocols for computer communications (SIGCOMM), 2003.

[17] Keränen A., Ott J., and Kärkkäinen T., *The ONE simulator for DTN protocol evaluation*, ACM Int'l Conf. on simulation tools and techniques (ICST), 2009.