


Article

# Efficient Opportunistic Routing Protocol for Sensor Network in Emergency Applications

Mohammed S. Al-kahtani <sup>1</sup>, Lutful Karim <sup>2,\*</sup>  and Nargis Khan <sup>2</sup>

<sup>1</sup> Department of Computer Engineering, Prince Sattam bin Abdul Aziz University, Al-Kharj 16278, Saudi Arabia; alkahtani@psau.edu.sa

<sup>2</sup> School of Information and Communication Technology, Seneca College of Applied Arts and Technology, Toronto, ON M2J 2X5, Canada; nargis.khan@senecacollege.ca

\* Correspondence: lutful.karim@senecacollege.ca

Received: 15 February 2020; Accepted: 3 March 2020; Published: 8 March 2020



**Abstract:** Routing or forwarding information, such as the location of incidents and victims in a disaster, is significantly important for quick and accurate incident response. However, forwarding such information in disaster areas has been a challenging task for the Wireless Sensor Network as existing networks are affected (destroyed or overused) the disaster. Opportunistic information forwarding can play a vital role in such circumstances. Existing opportunistic routing protocols require huge message transmissions for cluster restoration, which is not energy efficient and results in packet loss. Hence, this paper introduces an energy efficient and reliable opportunistic density cluster-based routing protocol that opportunistically transmits data using a density-clustering protocol for emergency and disaster situations. Simulation results show that the proposed protocol outperforms some existing and well-known routing protocols in terms of network energy consumption, throughput and successful data transmissions.

**Keywords:** mobile sensor network; opportunistic routing; density clustering; disaster routing; energy efficient; throughput; packet loss ratio

## 1. Introduction

Mobile Wireless Sensor Networks (MWSN) had been popular for thousands of applications, including disaster recovery. It is evident that these networks are usually destroyed or inaccessible in disastrous situations (e.g., tornados, hurricanes, earthquakes and war) and emergency personnel do not get proper information for quick and proper incident responses. Hence, an alternative solution is greatly required in these types of emergency scenarios.

Traditional solutions [1–4] attempt to build up a Mobile Ad Hoc Network (MANET) by distributing antennas in the affected area. This may be a solution from a communication point of view but, in reality, the feasibility of these solutions for large-scale disasters tends to be zero. Other studies [5–8] suggest wireless opportunistic networks, which are formed using sensors that are used in everyday use devices and objects (e.g., smartphone, tablet, PDA, vehicles, appliances, furniture) for monitoring heart rate, temperature, humidity, pressure, accelerometers. The idea of building an opportunistic recovery network for communication using those sensors may save many lives in disastrous situations. Thus, this type of solution provides improvements and efficiency for emergency rescue teams.

Assuming the existence of a network with an end-to-end connection between the sender and receiver is far from the reality in a disaster scenario. Using the traditional data forwarding methods in such scenarios most likely results in a data delivery failure. Hence, a new type of data forwarding and routing protocol supporting delay tolerance and disruption is greatly required for such disastrous situations. Opportunistic forwarding is a protocol based on the mobility of nodes in a network where

sensors can communicate properly and efficiently manage their energy consumption and forward information to the first responders. In this protocol, nodes carry stored messages while moving around and forward them whenever an opportunity arises. This opportunity is simply making a contact with a node that knows how to route the message closer to the desired receiver. Hence, a route is built dynamically from the sender to the receiver. Any node can possibly be the next hop in an opportunistic forwarding. Figure 1 demonstrates such an opportunistic network for disaster recovery.

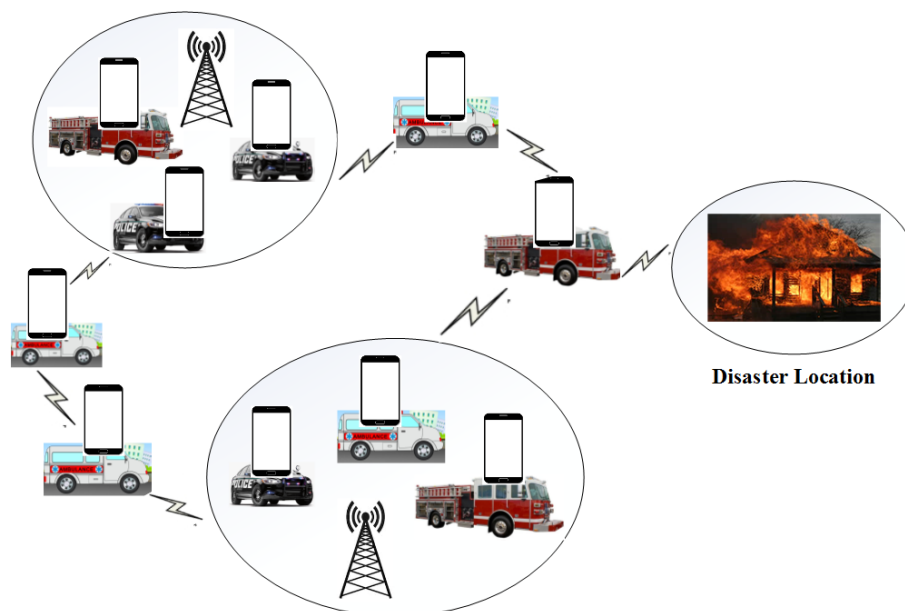


Figure 1. Opportunistic network for disaster recovery.

On the other hand, a type of edge computing [9,10] is emerging that integrates smart devices, e.g., sensors, with the opportunistic network, enabling local data processing along with processing data at the base station or central server. Though edge computing supports decentralized data processing, routing data is significant in such networks for disaster recovery. However, designing an efficient opportunistic routing protocol [8,11–22] in terms of energy efficiency, reliability and successful message transmission is a major challenge because of the mobility of nodes (i.e., rescue team members), the low transmission ranges of devices (i.e., smartphones, sensors, Bluetooth). Hence, this paper introduces an opportunistic density cluster-based routing protocol (ODCR) for emergency/disasterous situations. This protocol supports the efficient management of the energy and an opportunistic communication aspect to tolerate delays and disruption and builds routes in a dynamic and infrastructure-less manner. The number of sensors and quantity of generated data extremely influences the performance of any routing protocol. Hence, we consider edge computing in sensors that reduce the amount of data transmitted. Moreover, we use a density-based clustering model in this routing protocol [7,9–12] so that the number of nodes performing data collection and transmission reduces to achieve energy efficiency.

The proposed ODCR protocol uses opportunistic data forwarding and integrates both (i) density-based clustering and (ii) isolated nodes. In density-based clustering, each node is likely to reside in a cluster and exchanges information among the nodes in that cluster. On the other hand, isolated nodes (not part of a cluster) forward data packets opportunistically to a node among a number of nodes, based on their data delivery probability. Figures 2–4 illustrate such an opportunistic network based on the ODCR protocol, where both nodes in cluster and isolated nodes opportunistically create a data transmission path to forward data to the destination. This protocol forms clusters evaluating the Time to Return (TTR) parameter between nodes. The energy consumption of a node varies and depends on the role of each node in the cluster. We perform an experiment (simulation) on a realistic disaster area model [8] to measure the performance of the proposed opportunistic routing protocol. Simulation

results show that the proposed ODCR protocol in opportunistic forwarding or routing, increases the network lifetime and throughput and reduces the packet loss ratio compared to well-known clustering and forwarding protocols, such as LEACH, TEEN and LORA [16,18,21]. In summary, the contributions of the paper are as follows.

- This paper integrates edge computing with opportunistic networking for disaster recovery;
- The sensors in the network process and aggregate data locally to reduce the amount of data transmitted;
- The sensors create density-based clusters where sensors with the most neighboring nodes become cluster heads. This approach reduces the number of data collectors and transmitters and distributes energy consumption equally over sensor nodes;
- The nodes, which are not part of any cluster (e.g., isolated nodes) opportunistically forward data to the forwarder nodes. These nodes, while moving, join to the cluster closest to them;
- If any node moves out of a cluster due to mobility, the cluster restoration process starts based on the node's property, called the minimum period of connectivity (MPC). This property ensures an energy efficient cluster restoration process compared to the cluster restoration process of existing protocols, which involve all nodes in cluster restoration at the end of each round.

The rest of the paper is organized as follows. Section 2 presents the literature on existing opportunistic routing protocols. Section 3 presents the working principle of the proposed opportunistic density cluster-based routing protocol (ODCR). Section 4 evaluates the performance of the proposed ODCR protocol. Section 5 presents the network, energy and mobility models used in the experimental simulation along with experimental setup and results. Finally, Section 6 concludes the paper with suggested future research directions.

## 2. Related Work

The Low Energy Adaptive Clustering Protocol (LEACH) [18] is a well-known clustering protocol of sensor networks. In this protocol, the cluster head (CH) is randomly selected from all nodes based on a metric called probability of CH. Once clusters are constructed, the member nodes of each cluster transmit data to its CH using a time division multiple access (TDMA) scheme. The CH collects all data, aggregates and transmits them to the base station for further processing. In LEACH, the cluster member nodes transmit data periodically and the cluster formation takes place in every round, which is not energy efficient. In the Threshold Sensitive Energy Efficient Sensor Networks Protocol (TEEN) [16], each sensor node senses and transmits an event of interest (e.g., pressure, humidity) based on the minimum attribute value set into to the sensor. The next time the node senses and transmits another event of interest, if the attribute value changes by a certain threshold, that is set into the node. Hence, the number of sensed and transmitted events is less in the TEEN protocol. However, LEACH and TEEN protocols may not work well for emergency and disastrous situations.

The work done in [12] introduced an opportunistic routing protocol, namely EXOR. In this protocol, if the sender needs to transmit data to a destination it first transmits data to a node from its forwarding list, which has the least distance to the actual destination. Once the sender sends the data packet to all forwarding nodes, it determines the best forwarder based on the receiving ACK packet from the forwarder nodes. Although EXOR has higher throughput, it only uses the information during the time of transmission but does not use the updated information during transmission. Another work in [12] introduces an opportunistic routing that selects the next hop forwarding node based on the data delivery ratio and priority of nodes to transmit data.

In [12], the authors present a Trust Opportunity Forwarding Mechanism that considers both distance of forwarding nodes to the destination (cost) and the trust factor/degree. The trust factor of a node is calculated by neighboring nodes through direct observation or recommendations. A node is selected as a forwarder if it has better trust degree as well as less distance to the destination. On the other hand, among several routing protocols Social Relation Opportunistic Routing (SROR) identifies

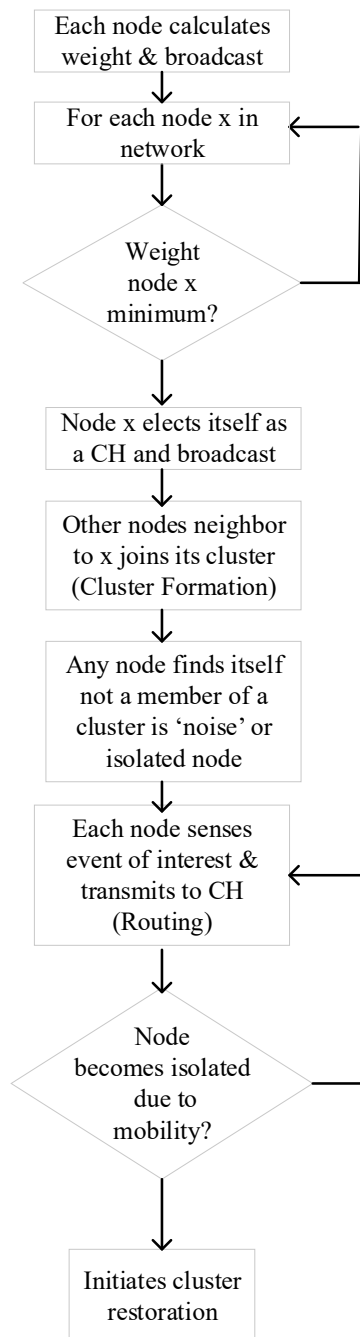
the forwarding nodes based on social relations and profiles among nodes. The work done in [20] introduces a planned opportunistic delay tolerant routing protocol by creating a subset of opportunistic links from source to destination, as opposed to a deterministic single copy or multiple copies of links between source to destination. This paper optimizes the multiple deterministic links to obtain only the best one in terms of lower data delivery rates. However, this approach does not employ clustering and is not efficient, as the creating paths require lot of computations and consumes energy of sensor nodes. The work done in [21] introduces LORA—a load-balancing opportunistic routing for asynchronous duty-cycled WSN. To avoid the flooding of packets from source nodes, through a number of preselected forwarder nodes, this approach creates candidate zones where all possible data forwarder candidates reside. The source node only chooses a candidate based on a metric calculated using direction distribution, transmission distance distribution, perpendicular distance distribution, and residual energy distribution. Again, this approach seems computationally expensive and does not allow clustering to achieve more energy efficiency. In opportunistic routing, each node decides about the forwarder node locally using greedy approaches and collects lot of information (i.e., overhead) for making such decision. To mitigate this problem, the work done in [22] introduces SeeR—a simulated annealing-based routing protocol for opportunistic mobile networks. This approach calculates a cost function for each node using information such as the hop-count and the average aggregated inter-contact time of the node. If the cost of a forwarder node is low, the sender transmits the data to that node. In this approach, a node does not track information about other nodes, thus, reducing privacy and security risks. However, this approach does not consider clustering. This work done in [23] introduces an opportunistic routing protocol (ORR) that finds the optimal number of forwarders based on a cost function using duty cycles and network topology. Moreover, this approach selects the forwarder from a list of forwarders with more residual energy.

Opportunistic networks use epidemic routing protocols [11] where a node broadcasts or forwards messages to all other nodes, which results in high message overhead. To mitigate this problem, the work done in [13] proposes an energy efficient opportunistic routing protocol where a node forwards a packet based on its remaining energy and delivery predictability factor. However, most routing protocols of opportunistic networks do not consider real-time message delivery. The work done by Santos et al. [14] introduces a real-time message propagation model with a maximum data transmission latency for opportunistic networks containing heterogeneous devices.

While existing works mostly consider delay tolerant networks (DTN) to forward messages opportunistically to the destination node, they do not consider transmitting messages to the first responders, governmental agency, hospitals and other related organizations in response to disaster situations. In [15], the authors introduce a routing protocol comprising DTN and an Information-Centric Network (ICN). The DTN transfers data to the destination and the ICN propagates interest messages to the responding authorities. However, these routing protocols may not work well in emergency or disastrous situations. Hence, an efficient opportunistic routing protocol is greatly required for emergency disastrous situations with a higher packet deliver ratio and lower latency.

### 3. Proposed Opportunistic Routing Protocol

The proposed opportunistic density cluster-based routing (ODCR) protocol uses density-based clustering as well as the opportunity to forward or broadcast its stored data to the nearby or closest node to the destination, as the nodes are mobile and may lose the network connectivity. The following subsections present the proposed ODCR protocol. Section 3.2 defines some terminologies that are used in the proposed ODCR protocol. Sections 3.2–3.4 present different phases of the ODCR protocol, such as cluster formation, routing and cluster restoration. Figure 2 illustrates the flowchart of the working principle of the proposed ODCR protocol.



**Figure 2.** Flowchart of proposed approach.

### 3.1. Terminologies

Minimum Period of Connectivity—in opportunistic networks, the minimum period of connectivity (MPC) represents the connectivity strength between two nodes while they are moving. This property also represents successful data packet transmission by a node. The MPC (represented in time) between two nodes  $a$  and  $b$  is denoted as

$$MPC_{ab} = \frac{R_c - d}{V_a + V_b} \quad (1)$$

where  $R_c$  is the communication range of nodes  $a$  and  $b$ ,  $d$  is the distance between  $a$  and  $b$ .  $V_a$  is the velocity of node  $a$  and  $V_b$  is the velocity of node  $b$ .

The Global Minimum Period of Connectivity ( $MPC_g$ ) is a predefined parameter given to the proposed protocol in order to set up a network having clusters with the same  $MPC_g$  between its nodes.

Weight—the weight of a node represents whether the node is stable or not. Equations (2)–(4) calculate the weight of a node as follows.

$$Y = \text{Total number of neighbors of node } a = |\text{neighbors}(a)| \quad (2)$$

$$X = \sum_{b \in \text{neighbors}(a)} \frac{MPC_{ab}}{MPC_g} \quad (3)$$

$$\text{Weight} = X - Y. \quad (4)$$

### 3.2. Form Density-Based Clusters

Initially, each node calculates its weight, defined by Equation (3). If the weight of a node  $x$  is close to zero, the minimum period of connectivity (MPC) of that node to all its neighbors is close to global MPC ( $MPC_g$ ). That means the node  $x$  is more stable and a good candidate to become CH. Once all opportunistic nodes or sensors calculate their weight based on Equations (2)–(4), they broadcast their weight. The node  $x$  with the minimum weight will elect itself as a CH and broadcasts a “hello” message to let other nodes know that  $x$  is a CH. Other nodes will join to a cluster whose CH is closer to the node. Thus, clusters of the network are constructed. Algorithm 1 presents the pseudocode to select CH, cluster members and border nodes of a cluster  $C$ .

---

#### Algorithm 1 (cluster formation)

Input: node  $x$ ,  $MPC_g$ , minimum number of nodes in a cluster ( $minNode$ ).

Output: a cluster with a cluster head CH.

---

**begin**

1:  $N \leftarrow \text{GetNeighbors}(x)$  //find number of neighbors of node  $x$

2:  $\text{weight} \leftarrow \text{GetWeight}(x)$  //find weight of node  $x$  and all other nodes using Equations (2)–(4)

3: **if**  $|N| \geq minNode$

4:   Select  $x$  as a core point //i.e., a good candidate to be a CH

5:   **if**  $\text{weight}(x) \leq \text{weight}(y), \forall y \in N$

6:     select  $x$  as CH of cluster  $C$

7:      $minNode = |N|$    //  $minNode$  is set to the number of neighbors of node  $x$

8:   **end if**

9:   **else**

10:   select  $x$  as border point // if weight of  $x$  is greater than all its neighboring nodes

11:   **end if**

12:   **for each**  $y \in N$  **do**

13:     **if**  $y$  is not CH nor member of any cluster

14:       **if**  $(MPC_{x,y} \geq MPC_g)$

15:          $C \leftarrow \{y\}$ ; include  $y$  in the cluster  $C$    //  $y$  is in cluster  $C$  if its MPC greater than global MPC

16:       **else**

17:         select node  $y$  as a noise to  $C$

18:       **end if**

19:   **end if**

20:   return CH

21:   Repeat steps 1–20 until all clusters are formed

22: **end**

---

In Algorithm 1, each node  $x$  calculates its weight and number of neighboring nodes and broadcasts this to its neighboring nodes (presented in lines 1–2 of Algorithm 1). Then, the weight of the node  $x$  and all its nodes are compared. The node that has the lowest weight becomes the cluster head (CH) of cluster  $C$  (presented in lines 3–8 of the Algorithm 1). All other nodes  $y$  who are neighboring nodes of  $x$

but are not selected as the CH or member of any cluster will compare  $MPC_{x,y}$  with global  $MPC_g$ . If the  $MPC_{x,y}$  of the node  $y$  is greater than or equal to  $MPC_g$ , node  $y$  is included as a member node of cluster  $C$  (presented in lines 12–15 of the Algorithm 1). This process continues until all nodes become the CH or member node of a cluster or noise of a cluster.

### 3.3. Routing

Once clusters are constructed, nodes start to collect data and, hence, data transmission or routing starts. If a node becomes isolated from a cluster due to mobility (known as a noise node) cluster restoration is required before the node can transmit data.

Scenario 1: Whenever a node  $x$  has some data to transmit, if it is a member node of a cluster, it transmits data to its CH. If node  $x$  is isolated or border node of a cluster it first tries to restore the connectivity (Section 3.4) and then transmits data. Otherwise, it opportunistically forwards data to a neighboring node.

Scenario 2: Whenever a node  $x$  receives a data packet from other nodes, (1) if  $x$  is a member node of a cluster it communicates with the CH and transmits data to the CH. Then, the CH transmits data to the next hop towards the destination (2) if the node  $x$  is a border node, it is a noise, which is either a member of a cluster or not. If  $x$  is a member of a cluster it transmits data to the CH. If  $x$  is not a member of a cluster it must be within the communication range of two neighboring clusters. Then, the node  $x$  broadcasts a message to find out the closest neighboring cluster and joins a cluster by restoring connectivity (presented in Section 3.4).

Scenario 3: The node might also loose contact with other nodes and become isolated. Figures 3–5 illustrate this scenario. In Figure 3, the node  $z$  is not a member of a cluster. Whenever it senses an event of interest or receives data from other nodes, it forwards data to the closest neighboring/forwarder node  $y$ . However, the node  $y$  is isolated and has no contact with other nodes. As the node  $y$  moves due to mobility and reaches the communication range of node  $x$ , it immediately forwards data to node  $x$  (Figure 4). Then, node  $x$ , being a noise (border node trying to join to a cluster), forwards data to the member nodes and CH of a cluster towards the destination, as illustrated in Figure 5.

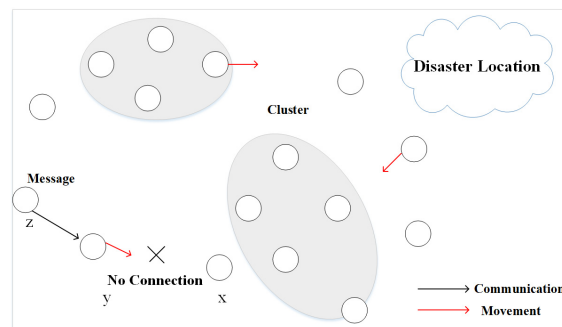


Figure 3. Disaster network topology.

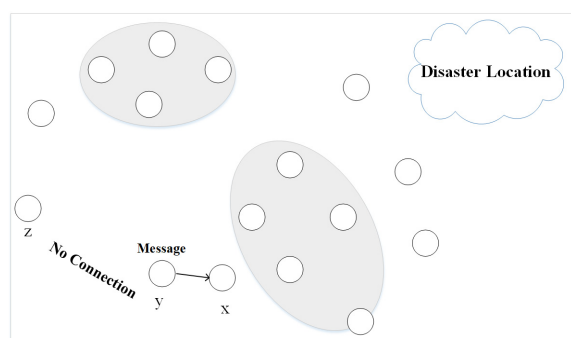


Figure 4. Opportunistic message forwarding.



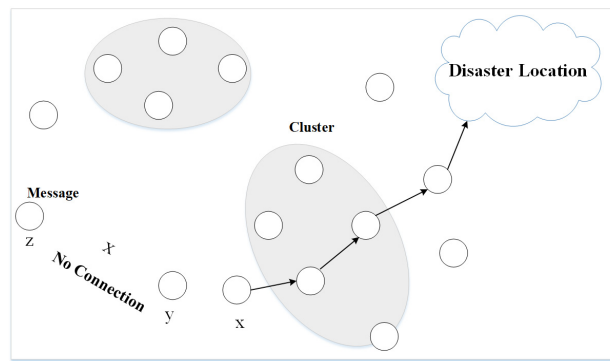


Figure 5. Cluster-based message transmission.

Forwarder node selection: During routing, a node may need to select a forwarder node. To do so, each node keeps the MPC value with all of its neighboring nodes and opportunistically forwards data to the neighboring node with the highest MPC. As this protocol is applicable to a disaster situation where all nodes are moving almost at the same speed towards the location of the emergency or disaster, they are more likely to keep communicating with a number of nodes. Thus, the source node that is the initiator of data transmission should be opportunistically able to transmit data to the destination by communicating with those nodes.

### 3.4. Cluster Restoration

As nodes are most likely mobile in opportunistic networks, they reach the border of the cluster or become isolated. Hence, cluster restoration is required. The border node  $x$  starts cluster restoration by finding the best candidate from its noise neighbors to communicate with for restoring connectivity to cluster  $C$ , as illustrated in Figures 6 and 7. We present two scenarios of cluster restoration below.

Scenario 1: once the node  $x$  becomes a border node of cluster  $C$ , it must run an algorithm of cluster restoration or join other clusters that maintain the global MPC, increasing the communication range,  $R_c$ , of  $y$  as follows:

$$MPC_{y,x} = MPC_g = \frac{R_c - d}{V_x + V_y} \tag{5}$$

where  $V_y$  and  $V_x$  are the velocity of the border nodes  $x$  and  $y$  of two neighboring clusters,  $d$  is the distance between nodes  $x$  and  $y$ , and  $R_c$  is the communication range of nodes in both clusters. Figure 6 demonstrates Scenario 1. In Scenario 1, we use Equation (5) to increase  $R_c$  to restore connectivity.

Scenario 2: If  $x$  becomes isolated, the algorithm finds the minimum MPC between  $MPC_{x,y}$  and  $MPC_{x,C}$  and increases the  $R_c$  of this minimum to maintain  $MPC_g$  (Figure 7).

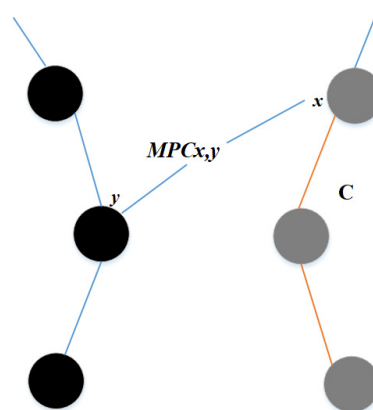


Figure 6. Scenario 1, where  $x$  is a border node of cluster  $C$ .



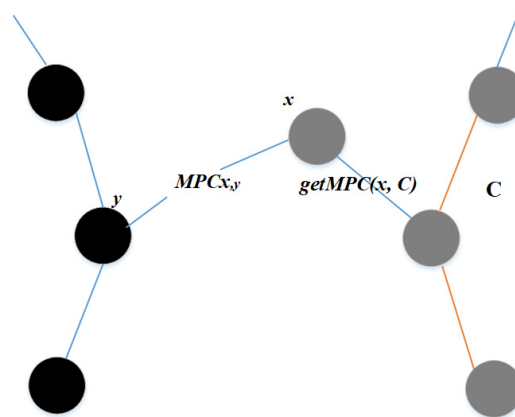


Figure 7. Scenario 2, where  $x$  is not a member of  $C$ .

In Scenario 2 (Figure 7), node  $x$  is considered as noise,  $x \notin C$ , and is in the communication range of communication of both neighboring clusters. The cluster restoration approach tries to find the cluster that will establish connection with  $x$  first, i.e., find the minimum  $MPC$  between  $MPC_{x,y}$  and  $getMPC(x, C)$ . Then, this algorithm increases the  $R_c$  of this minimum to the level of the global  $MPC_g$ . Thus, this guarantees a uniform period of connection in the network.

The cluster restoration process to repair connectivity between two clusters is launched by the border nodes of one of the clusters. The cluster, which has the CH with the minimum weight (calculated using Equations (2)–(4)) will normally activate its border node to establish the connectivity with the other cluster (Figure 7).

Algorithm 2 presents the pseudocode of cluster restoration based on Scenarios 1 and 2.

---

#### Algorithm 2 (Cluster Restoration)

Input:  $y$  is a border point from a neighbor cluster to  $C$ ,  $C$  is a cluster to be connected to  $y$ ,  $MPC_g$  is the minimal period of connectivity.

---

##### Begin

```

1: Find  $x$  such as  $MPC_y, x = \text{Max}$ , where  $x$  is a noise
2: Relay  $x = \text{FindBestCandidate}(y, x)$ , where  $x$  noise &  $MPC_{y,x}$  is a max
3: if  $x$  is Member( $C$ ) then                                //Scenario 1
4:    $y.increaseRc(\text{getMPC}(x, y), MPC_g)$                 // increase  $R_c$  of  $y$  to maintain  $MPC_{x,y}$ , same as  $MPC_g$ 
5: else                                                    //Scenario 2
6:    $temp = \min(\text{getMPC}(x,y), \text{getMPC}(x,C))$ 
7:    $y.increaseRc(temp, MPC_g)$ 
8: end if
9: end

```

---

## 4. Performance Evaluation

This section presents the performance evaluation of the proposed ODCR protocol.

### 4.1. Computation Efficiency

The computational and message complexity of the proposed ODCR protocol is  $O(1)$  per node. Initially, each node broadcasts one message of its velocity and its position to neighbors. Then, once CHs are selected, each CH broadcasts one declaration message (such as the “hello message” presented above). When the  $k$ -hop nodes ( $k \geq 1$ ) hear the declaration message they broadcast an acknowledgement message, join to the cluster and rebroadcast the declaration message to the  $k+1$ -hop nodes to find the shortest path to the destination. Hence, it has been observed that each node sends, at most, three messages to perform the operation.

#### 4.2. Energy Efficiency

In this section, we analyze the energy efficiency of the proposed ODCR protocol, well-known clustering protocols LEACH [18] and TEEN [19] and a very recent opportunistic routing protocol, called LORA [21].

The cluster formation phase is initiated in LEACH and TEEN protocols after a certain number of rounds where all nodes participate in the cluster restoration or reformation. On the other hand, cluster restoration is initiated in the proposed ODCR protocol only when a node becomes a border node or becomes isolated. In addition, only the border nodes and two neighboring CHs participate in this cluster restoration process. Other nodes remain in their cluster and do not participate in the cluster formation process.

Hence, the proposed ODCR protocol seems to be more energy efficient, lower packet loss ratio and higher throughput as compared to the existing LEACH and TEEN protocols. Let us assume that:

The number of nodes in the network =  $n$

The number of clusters in the network =  $c$

Then, the average number of nodes in each cluster =  $\frac{n}{c}$

The number of messages each node transmits and receives for cluster formation =  $m$

$$\text{The average number of message transmission for cluster formation} = n \times m \tag{6}$$

**For cluster restoration in the ODCR protocol:**

The number of nodes participate in cluster restoration = 3, where  $3 < n$

$$\text{The average number of message transmissions in ODCR for cluster restoration} = 3 \times m \tag{7}$$

Hence, the number of transmitted messages for cluster restoration is  $O(n)$ .

Comparing Equations (6) and (7), we expect that the average number of message transmissions in the ODCR protocol is much lower than that of the LEACH and TEEN protocols.

On the other hand, LORA [21] is a non-clustered multi-hop opportunistic routing protocol.

Let us assume that the number of hops from source to destination in LORA

$$= h f_1 + f_2 + f_3 + \dots + f_{h-1} \leq n \tag{8}$$

where  $f_i$  represents the number of forwarder nodes at hop  $i$ .

The total number of message transmissions for selecting forwarder nodes will be about

$$f_1 \times m_1 + f_2 \times m_2 + f_3 \times m_3 + \dots + f_h \times m_h \tag{9}$$

where  $m_i$  represents the number of message transmissions for selecting forwarder nodes at hop  $i$ .

If  $b_1, b_2, \dots, b_h$  represent the number of message transmissions for selecting the best forwarder node at hops 1, 2, ... ,  $h$ , respectively, the total number of message transmissions would be

$$f_1 \times (m_1 + b_1) + f_2 \times (m_2 + b_2) + f_3 \times (m_3 + b_3) + \dots + f_h \times (m_h + b_h). \tag{10}$$

Hence, the selection of forwarder nodes for transmitting a data packet from the source to destination will be  $O(n^2)$ . The process of selecting such a path with forwarder nodes occurs every time a node transmits data to the destination. Hence, it is more likely that the total number of message transmissions in the LORA protocol is much more than that in the proposed ODCR protocol. We will perform an experimental simulation to verify this.

As energy consumption of sensor nodes is directly proportional to the number of transmitted messages, the energy consumption of the ODCR protocol is expected to be much lower than that in the LEACH, TEEN and LORA protocols. Similarly, as the number of message transmissions is much more

in LEACH, TEEN and LORA protocols, the probability of packet loss in these protocols will be higher than that of the proposed ODCR protocol.

## 5. Simulation Model, Setup and Results

We performed a simulation to evaluate the performance of the proposed ODCR opportunistic routing protocol and compare it with some existing well-known routing protocols. We used the network simulator NS-3 [12] to perform the simulation. The following sections present the energy, network and mobility models that are used in our simulation, along with the simulation setup and results.

### 5.1. Energy Model

Whenever nodes transmit and receive messages, the energy consumption of these nodes are calculated using Equations (11) and (12), as presented below.

Energy consumptions to transmit a packet of size  $n$  over distance  $d$  is given by:

$$E_{TX} = n * \varepsilon_{elec} + n * \varepsilon_{fs} * d^{\alpha}. \quad (11)$$

Energy required receiving a packet is:

$$E_{RX} = n * \varepsilon_{elec} \quad (12)$$

where  $\varepsilon_{elec}$  and  $\varepsilon_{fs}$  in Equations (11) and (12) represent energy spent in transmitter electronics circuitry and energy spent in RF amplifiers for propagation loss, respectively. The constant (propagation loss exponent)  $\alpha$  is dependent on the surrounding environment. For free space without any obstruction in the line of sight,  $\alpha = 2$ .

In the proposed ODCR protocol and existing LEACH, TEEN and LORA protocols, each node transmits and receives data packets during cluster formation, routing and cluster restoration phases. Based on this energy model, the energy consumptions of a node are directly proportional to the size of the data and the distance a data packet travels.

### 5.2. Network and Mobility Model

The sensor network comprising  $N$  nodes is represented by a graph  $G(V, E)$ , where  $V$  represents a set of sensor nodes (vertices) and  $E$  represents a set of communication links (edges). Two nodes,  $u$  and  $v$ , communicate with each other if they are within communication range ( $R_c$ ). This property is known as a Unit Disk Graph (UDG) of graph theory. However, since our work is proposed in the context of sensor-based opportunistic networks, we consider the “double range property” [24] in our experiment, defined as  $R_c = n \times R_s$  where  $n \geq 2$  and  $R_s$  refers to the sensing range.

We consider a 4.0 km<sup>2</sup> (approximately 2 km × 2 km) network size as it is a realistic area around disaster zones where rescue personnel need to be connected for recovery operations. Moreover, the mobility speed of rescue personnel is assumed to be about 5 m/s or 300 m/min, considering the scenario that rescue personnel are moving slow or walking in a certain region of that area to perform the operation.

The random mobility of the nodes was generated using mobility generator BonnMotion [25], which places each node randomly in a position in the simulation area of size 2000 × 2000 m<sup>2</sup>. The pattern of mobility is the Random Scenario (RS) provided by the mobility scenario generator BonnMotion [25]. The generated mobility patterns pause each node at its current location for a certain period before it moves to another randomly generated location at a random speed.

The IEEE 802.15.4 standard Zigbee protocol is used in the low data rate and powered sensors. We consider that sensors are used to create the opportunistic network in the simulation. We consider using the libelium Waspote [26] that considers all parameters and specifications mentioned in [26] in our simulation. Table 1 lists the simulation parameters and their respective values.

**Table 1.** Simulation Parameters and Values.

| Parameter                     | Value                      |
|-------------------------------|----------------------------|
| Simulation network area       | 2000 × 2000 m <sup>2</sup> |
| Number of node                | 100                        |
| Communication range of a node | 200 m                      |
| Transport protocol            | UDP                        |
| MAC protocol                  | IEEE 802.11                |
| Mobility Model                | Random                     |
| Mobility Speed                | 5 m/s                      |
| Pause time                    | ½ s                        |
| Duration of a round           | 200 s                      |
| Traffic type                  | Constant Bit Rate (CBR)    |
| Data transmission rate        | 250 Kbit/s                 |
| Data packet size              | 128 Bytes = 1 Kbit         |

### 5.3. Performance Metrics

The performance metrics to measure the performance of the proposed ODCR protocol are defined as follows:

- Energy consumption is the total energy consumed by the sensor nodes in the network over a number of rounds.
- Throughput is the amount of data transmitted by a node per second. We calculate throughput in Gigabytes/second in this paper.
- Packet delivery ratio is the ratio of the total number of packets delivered to the total number of packets transmitted.
- Packet loss ratio is the ratio of the total number of packets lost to the total number of packets transmitted.

### 5.4. Simulation Results

Figures 8–11 illustrate the performance of the proposed ODCR opportunistic routing protocol in terms of network energy consumption, packet loss, throughput, and packet delivery ratio. Each simulation runs for 200 s. We evaluate the performance of the proposed ODCR protocol over a number of rounds. The output is the average of outputs received in a certain number of rounds.

In this simulation, energy consumption is calculated using the data transmission and reception energy consumptions of a node (Equations (11) and (12)), presented in the energy model, and the estimated number of message transmissions for each approach, analyzed in Section 4.2.

Figure 8 demonstrates that the energy consumption of the proposed ODCR protocol is much lower than the existing TEEN and LEACH clustering protocols. This is because, using Equations (6) and (7), we find that the average message transmission in ODCR is much lower than that in the LEACH and TEEN protocols. Similarly, the number of message transmissions in the ODCR protocol is also lower than LORA protocol (as analyzed using Equation (10)). Hence, the network energy consumption in the proposed ODCR protocol is lower than that of the LORA protocol.

Figure 9 demonstrates that the packet loss ratio of the proposed ODCR protocol is much lower than the existing LEACH [18], TEEN [20] and LORA [21] protocols because the proposed ODCR protocol can perform quicker cluster recovery due to node movements compared to the existing algorithms. Hence, the probability of packet loss is much lower in the ODCR protocol. More specifically, in the TEEN protocol, clusters are reconstructed only at the end of each round, regardless of whether any node moves out of the cluster in a round. In LEACH, clusters are reconstructed at the end of a certain number of rounds. Even if a number of nodes move out of clusters and become isolated or noise nodes, the clusters are not formed unless the number of rounds reaches a predefined value. Hence, the probability of packet loss is much higher in LEACH and TEEN protocols. This also results in a lower

throughput in LEACH and TEEN protocols. On the other hand, in the LORA protocol, a candidate zone is constructed with a set of forwarder nodes for each hop. Moreover, the best forwarder is selected for each candidate zone to create a path from the source to the destination for transmitting data packets. As the nodes are mobile, the probability of moving the forwarder nodes in this preselected path is high. Hence, the probability of packet loss is also high in the LORA protocol.

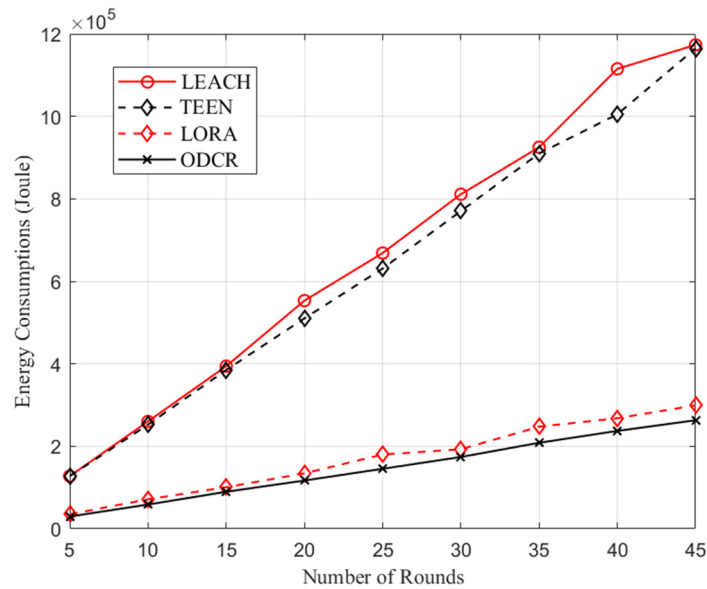


Figure 8. Comparison of energy consumption.

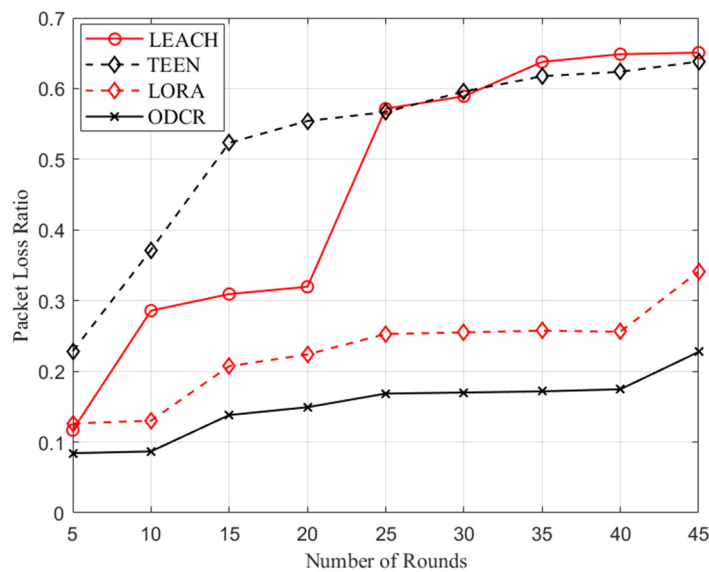


Figure 9. Comparison of packet loss ratio.

Figure 10 demonstrates that the throughput of the proposed ODCR protocol is much higher than that of the LEACH, TEEN and LORA protocols. This is because the proposed ODCR protocol performs cluster recovery and creates clusters rapidly based on the worst-case scenario of node movements between two adjacent nodes. Existing protocols lose connectivity in such scenario, which is not the case for the proposed ODCR protocol.

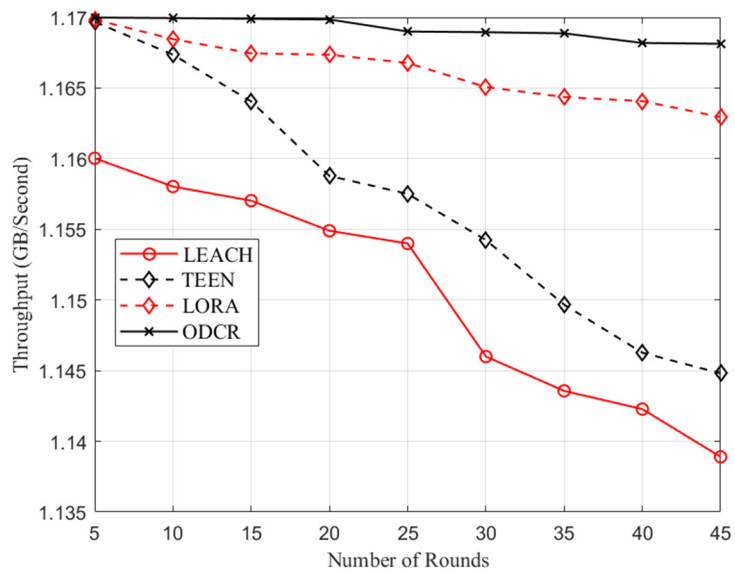


Figure 10. Comparison of throughput.

Similarly, Figure 11 illustrates that the ODCR protocol achieves higher packet delivery ratio compared to LEACH, TEEN and LORA protocols because the proposed ODCR protocol maintains connectivity in most of the mobility scenarios. In the simulation results, we found that, on average, the packet delivery ratio is almost 50% more in the ODCR protocol compared to the LEACH and TEEN protocols and about 15% more in the ODCR protocol compared to the LORA protocol. Table 2 compares several features of the existing LEACH, TEEN, LORA and proposed ODCR protocols.

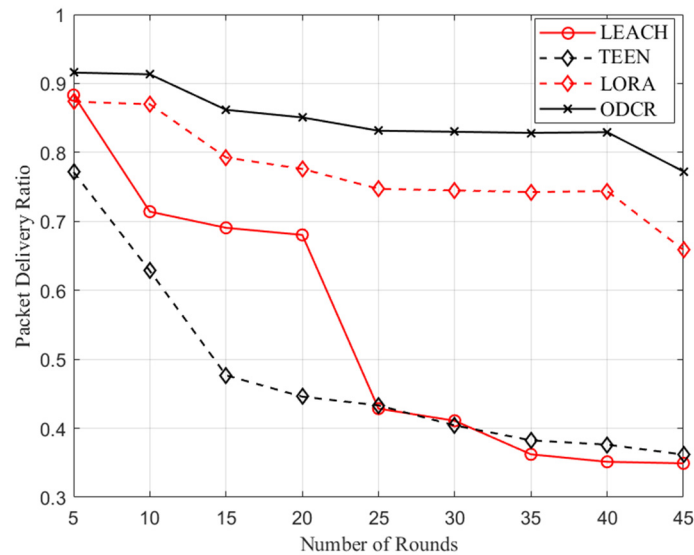


Figure 11. Comparison of packet delivery ratio.

**Table 2.** Comparison of the proposed opportunistic density cluster-based routing protocol (ODCR) protocol with existing protocols.

| Features   | LEACH | TEEN | LORA | ODCR |
|--|-------|------|------|------|
| Cluster formation is done at the end of each round   | √     | √    | X    | X    |
| All nodes in the network involve in cluster restoration or reform  | √     | √    | X    | X    |
| Nodes that connect or communication other nodes for a longer period time are good candidate to be selected as cluster head | X     | X    | √    | X    |
| Consider mobility of nodes   | X     | X    | √    | √    |
| Construct candidate zone at each with a number of forwarder nodes  | X     | X    | √    | X    |
| Requires reselecting forwarder nodes very often due to mobility  | X     | X    | √    | X    |
| Use opportunistic forwarding as a cluster member or isolated nodes   | X     | X    | X    | √    |

## 6. Conclusions

This paper presents a dynamic, energy efficient opportunistic density cluster-based routing protocol (ODCR) that opportunistically transmits data among sensors and other wireless nodes in disaster and emergency situations. In this ODCR protocol, each node joins in a density cluster-based sensor network and exchanges information with other nodes of the cluster, or each node forwards data opportunistically whenever it encounters another node with a higher probability of data delivery to the destination. We evaluated the performance of the proposed ODCR routing protocol in terms of network energy consumption, packet loss, throughput and packet delivery ratio compared with existing well-known LEACH, TEEN and LORA protocols. Experimental results found that the performance of the proposed ODCR routing protocol is much better than that of LEACH, TEEN and LORA protocols in terms of the above described performance metrics. However, we plan to verify the performance of the ODCR protocol in terms of latency, localization error and other performance metrics in the future. We also plan to include a reinforcement learning-based approach (i.e., machine learning and artificial intelligence [27,28]) in the MAC protocol and scheduling mechanism to achieve sensor energy efficiency as part of a future work.

**Author Contributions:** The work presented in this paper is based on the concept provided by M.S.A.-k. Who also contributed to system model, methodology, implementation, validation, writing the paper. Conceptualization and methodology, paper drafting have been done by N.K. L.K. work as a project coordinator, reviewer, proof-reader of this paper. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

- Jang, H.-C.; Lien, Y.-N.; Tsai, T.-C. Rescue information system for earthquake disasters based on MANET emergency communication platform. In Proceedings of the IWCMC'09: Proceedings of the 2009 International Conference on Wireless Communications and Mobile, Leipzig, Germany, 21–24 June 2009.
- Ramesh, M.V.; Jacob, A.; Devidas, A.R. Enhanced Emergency Communication using Mobile sensing and MANET. In Proceedings of the ICACCI'12 Proceedings of the International Conference on Advances in Computing, Communications and Informatics, Chennai, India, 3–5 August 2012.
- Zheng, C.; Chen, L.; Sicker, D. Hybrid cellular-MANETs: An energy-aware routing design. In Proceedings of the 2014 11th Annual Conference on Wireless On-Demand Network Systems and Services (WONS), Obergurgl, Austria, 2–4 April 2014; pp. 9–16.
- Verma, H.; Chauhan, N. MANET based emergency communication system for natural disasters. In Proceedings of the 2015 International Conference on Computing, Communication & Automation (ICCCA), Noida, India, 15–16 May 2015; pp. 480–485.
- Lu, Z.; Sun, X.; Porta, T.L. Cooperative data offloading in opportunistic mobile networks. In Proceedings of the IEEE INFOCOM 2016—The 35th Annual IEEE International Conference on Computer Communications, San Francisco, CA, USA, 10–14 April 2016; p. 19.



6. Lu, Z.; Cao, G.; Porta, T.L. Networking smartphones for disaster recovery. In Proceedings of the 2016 IEEE International Conference on Pervasive Computing and Communications (PerCom), Sydney, Australia, 14–19 March 2016; pp. 1–9.
7. Ester, M.; Kriegel, H.; Sander, J.; Xu, X. A density-based algorithm for discovering clusters in large spatial databases with noise. In Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining, Shanghai, China, 26–28 April 2019; Volume 1996, pp. 226–231.
8. Sanchez-Iborra, R.; Cano, M.-D. JOKER: A novel opportunistic routing protocol. *IEEE J. Sel. Areas Commun.* **2016**, *34*, 1690–1703. [[CrossRef](#)]
9. Higashino, T.; Yamaguchi, H.; Hiromori, A.; Uchiyama, A.; Yasumoto, K. Edge computing and IoT based research for building safe smart cities resistant to disasters. In Proceedings of the 2017 IEEE 37th International Conference on Distributed Computing Systems (ICDCS), Atlanta, GA, USA, 5–8 June 2017; pp. 1729–1737.
10. Casadei, R.; Fortino, G.; Pianini, D.; Russo, W.; Savaglio, C.; Viroli, M. A development approach for collective opportunistic Edge-of-Things services. *Inf. Sci.* **2019**, *498*, 154–169. [[CrossRef](#)]
11. Said, J.E.; Karim, L.; Almhana, J.; Anpalagan, A. Heterogeneous mobility and connectivity-based clustering protocol for wireless sensor networks. In Proceedings of the 2014 IEEE International Conference on Communications (ICC), Sydney, Australia, 10–14 June 2014; pp. 257–262.
12. Jadhav, P.; Satao, R. A Survey on Opportunistic Routing Protocols for Wireless Sensor Networks. *Procedia Comput. Sci.* **2016**, *79*, 603–609. [[CrossRef](#)]
13. Kang, M.W.; Chung, Y.W. An energy-efficient opportunistic routing protocol in delay tolerant networks. In Proceedings of the 2016 International Conference on Information and Communication Technology Convergence (ICTC), Jeju, Korea, 19–21 October 2016; pp. 655–659.
14. Santos, R.M.; Orozco, J.; Ochoa, S.F.; Meseguer, R.; Mosse, D. Providing Real-Time Message Delivery on Opportunistic Networks. *IEEE Access.* **2018**, *6*, 40696–40712. [[CrossRef](#)]
15. Kang, M.W.; Kim, Y.; Chung, Y.W. An opportunistic forwarding scheme for ICN in disaster situations. In Proceedings of the 2017 International Conference on Information and Communication Technology Convergence (ICTC), Jeju, Korea, 18–20 October 2017; pp. 948–950. [[CrossRef](#)]
16. Manjeshwar, A.; Agrawal, D.P. TEEN: A Routing Protocol for Enhanced Efficiency in Wireless Sensor Networks. In Proceedings of the 15th International Parallel and Distributed Processing Symposium, IPDPS 2001, San Francisco, CA, USA, 23–27 April 2001; Volume 22, p. 189. [[CrossRef](#)]
17. Yassein, M.B.; Khamayseh, Y.; Mardini, W. Improvement on LEACH protocol of wireless sensor network (VLEACH). *Int. J. Digit. Content Technol. Appl.* **2009**, 132–136. [[CrossRef](#)]
18. Heinzelman, W.; Chandrakasan, A.; Balakrishnan, H. Energy-Efficient Communication Protocol for Wireless Microsensor Networks. In Proceedings of the 33rd Hawaii International Conference on System Sciences, Maui, HI, USA, 4–7 January 2000.
19. Senturk, I.F.; Akkaya, K. Connectivity restoration in disjoint Wireless Sensor Networks using centrality measures. In Proceedings of the 39th Annual IEEE Conference on Local Computer Networks Workshops, Edmonton, AB, Canada, 8–11 September 2014; pp. 616–622.
20. Han, Y.; Wu, H.; Yang, Z.; Li, D. A New Data Transmission Strategy in Mobile D2D Networks—Deterministic, Greedy, or Planned Opportunistic Routing? *IEEE Trans. Veh. Technol.* **2017**, *66*, 594–609. [[CrossRef](#)]
21. Hawbani, A.; Wang, X.; Sharabi, Y.; Ghannami, A.; Kuhlani, H.; Karmoshi, S. LORA: Load-Balanced Opportunistic Routing for Asynchronous Duty-Cycled WSN. *IEEE Trans. Mob. Comput.* **2019**, *18*, 1601–1615. [[CrossRef](#)]
22. Saha, B.K.; Misra, S.; Pal, S. SeeR: Simulated Annealing-Based Routing in Opportunistic Mobile Networks. *IEEE Trans. Mob. Comput.* **2017**, *16*, 2876–2888. [[CrossRef](#)]
23. So, J.; Byun, H. Load-Balanced Opportunistic Routing for Duty-Cycled Wireless Sensor Networks. *IEEE Trans. Mob. Comput.* **2017**, *16*, 1940–1955. [[CrossRef](#)]
24. Xing, G.; Lu, C.; Pless, R.; Huang, Q. Impact of sensing coverage on greedy geographic routing algorithms. *IEEE Trans. Parallel Distrib. Syst.* **2006**, *17*, 348–360. [[CrossRef](#)]
25. Aschenbruck, N.; Ernst, R.; Gerhards-Padilla, E.; Schwamborn, M. BonnMotion: A mobility scenario generation and analysis tool. In Proceedings of the 3rd International ICST Conference on Simulation Tools and Techniques (SIMUTools'10); ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering): Brussels, Belgium, 2010; Article 51; pp. 1–10. [[CrossRef](#)]
26. Libelium Waspote. Available online: <http://www.libelium.com/waspote> (accessed on 13 January 2020).

27. Ghatge, V.V.; Vijayakumar, V. Machine Learning for Data Aggregation in WSN: A Survey. *Int. J. Pure Appl. Math.* **2018**, *118*, 1–12.
28. Savaglio, C.; Pace, P.; Aloisi, G.; Liotta, A.; Fortino, G. Lightweight reinforcement learning for energy efficient communications in wireless sensor networks. *IEEE Access* **2019**, *7*, 29355–29364. [[CrossRef](#)]



© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).