Enabling Reliable and Resilient IoT Based Smart City Applications

Thaha Muhammed[1(✉)], Rashid Mehmood[2], and Aiiad Albeshri[1]

[1] Department of Computer Science, FCIT, King Abdulaziz University, Jeddah 21589, Saudi Arabia
m.thaha.h@ieee.org, aaalbeshri@kau.edu.sa

[2] High Performance Computing Center, King Abdulaziz University, Jeddah 21589, Saudi Arabia
rmehmood@kau.edu.sa

Abstract. Internet of Things (IoT) has emerged as a revolutionary technology that has become an integral part of smart cities. It has numerous, longstanding, economical and safety-critical smart city applications. Data acquisition for IoT applications requires sensor networks. Reliability, Resilience, and Energy conservation are the three most critical wireless sensor requirements. Fault tolerance ensures the reliability and the resilience of the sensor network in case of failures. In this paper, we propose a new taxonomy for fault tolerant technique for wireless sensor networks deployed in an IoT environment, and qualitatively compare some major existing methods and propose a new fault-tolerant routing technique for hierarchical sensor networks. The algorithm is a heterogeneous technique based on Dynamic source routing (DSR), vice cluster heads, energy thresholding and hierarchical sensor networks. The proposed technique was simulated and is compared with current techniques to evaluate its validity and performance.

Keywords: Smart-city · IoT · WSN · Reliability · Fault tolerance Energy · Routing

1 Introduction

Urban population has increased vastly in the recent years. The United Nations Human Settlements Program (UN-Habitat) [1] has foreseen it to be 10 billion by 2050, which is two-thirds of the current population on earth. The cities will have to deal with pressing issues such as public safety, efficient transportation, energy consumption, environmental sustainability, and expense reduction. These pressing issues have led to smart city paradigm which aims to plan and develop efficient urban cities in future.

The past decade has witnessed the advancement of Internet of Things (IoT) [2] especially the sensing technology [3]; In addition to the widespread development of sensors, improvement in Big data computing infrastructure has enabled

R. Mehmood et al. (Eds.): SCITA 2017, LNICST 224, pp. 169–184, 2018.
https://doi.org/10.1007/978-3-319-94180-6_17

the collection of huge amount of heterogeneous data produced daily by urban spaces [4]. Urban spaces produce data related to temperature, weather, pollution, traffic control, the mobility of people, resource consumption (water and electricity) which can be analyzed to improve the services provided and make the environment greener. Smart cities rely on sensors, webcams, IoT systems, wireless sensor networks, databases, ubiquitous devices, and many other frameworks that collect, process and take informed decisions based on the data [5]. A survey on Data Fusion and IoT for smart ubiquitous environments can be seen in [6].

Wireless Sensor Networks (WSNs) are one of the atomic components of IoT. Data acquisition for IoT applications require wireless sensor network and are the link between the real world and the digital world. WSNs play a major role in building interconnected urban territories and are critical to smart cities. WSNs consists of small low power sensor nodes that can sense, process, and wirelessly communicate with each other. The sensors are devices with limited battery, storage, size, and computational power. The sensors nodes sense data and forward it to a base station known as sink for further processing of data by IoT systems. Intelligent monitoring and management of smart cities are possible through IoT. WSNs are used in a number of time-critical smart city applications such as agriculture monitoring [7], intruder detection [8], disaster management, health care, mobile object tracking, environment monitoring [8], Intelligent Transport System (obstacle detection, collision warnings and avoidance, traffic monitoring) [9–11], Vehicular AdHoc Networks [12], energy monitoring in smart grids [13], and Home/Office Automation Systems (HOS) [14].

Since WSNs are deployed in harsh and hostile conditions they are susceptible to frequent errors. The occurrence of faults results in disruption of the network or worse in the failure of the network. This might lead to human, economic, environmental loss since the sensors are used in many safety critical applications. Another source of a fault in WSN is the power [3]. Since the WSNs work unattended in a hostile environment it is not feasible to replenish the batteries of the sensors. Moreover, various hazard might cause the power to run out, which results in a node failure. Data transmission consumes a major portion of energy [15]. Hence prolonging energy in WSN becomes a critical and challenging issue [16,17]. A detailed discussion on possible faults in wireless sensor networks has been discussed in [18]. It is required that the data collected by the sensors on critical events should not be of low quality [19,20] that might lead to important information loss. But often random link failures occurs that disrupt communication in the network. All these issues point to the necessity of fault tolerance techniques that would provide techniques to mask these faults and provide the expected services, in the presence of faults. Major disadvantages of existing techniques are a high dissipation of energy, large Mean Time to Repair (MTTR), and use of extra software and hardware [21].

In this paper, we propose a new fault-tolerant routing algorithm based on modified Dynamic Source Routing (DSR) on a clustered, hierarchical sensor network for IoT applications. We use a vice cluster head that takes over the duties of the CH on the failure of a CH. Moreover, we use multiple paths that have been prioritized and sorted on the basis of a cost function, that takes into consideration the total energy in a path and the distance from the source to sink. Furthermore, we use energy thresholds to decide the CHs that would participate in the routing process. One of the major advantages of the technique is that the Mean Time To Repair (MTTR) for this technique is small. We simulate our algorithm and compare our algorithm with DFTR [22], a distributed fault tolerant algorithm and LEACH [3], a well-known routing algorithm. Metrics such as the number of alive nodes, total energy consumption of the network, and total packets transmitted to the Sink are compared measured for all the three techniques. Based on these metrics it was observed that HMDSR performs better than the other techniques.

Our contributions in this article can be summarized as follows:

- We propose a new up to date taxonomy for fault-tolerant strategies for wireless sensor networks.
- We propose a new energy efficient fault-tolerant routing strategy called Heterogeneous Modified Dynamic Source Routing (HMDSR).
- We simulate the proposed technique.
- The results from the proposed technique are compared with two current techniques [3,22] to validate the benefits of the algorithm.

The rest of the article is organized as follows. Section 2 discusses the proposed taxonomy for fault tolerant techniques in WSN. In Sect. 3, we discuss the state of art fault tolerant techniques for WSN. We also do a qualitative analysis of FT techniques in WSN. In Sect. 4, we discuss the System model and Sect. 5 introduces our proposed FT routing technique. Section 6 presents the simulation of the proposed technique. It also presents the comparison with techniques to validate our proposed technique. Section 7 concludes the paper.

2 Taxonomy

Fault tolerance techniques in wireless sensor networks can be classified according to two criteria, namely based on the phase at which the fault tolerant technique triggers and based on the origin of faults in WSN. Based on these criteria fault detection techniques in WSN can be classified as (1) Proactive and (2) Reactive as shown in Fig. 1.

2.1 Proactive Techniques

Proactive techniques in WSN proactively and sensibly uses the existing resources of the wireless sensor to extend the lifetime of the network or prevent the fault from occurring. These techniques take preemptory action against potential faults. Based on the origin of faults these techniques can be classified (1) Node based techniques, (2) Network-based techniques and, (3) Holistic techniques.

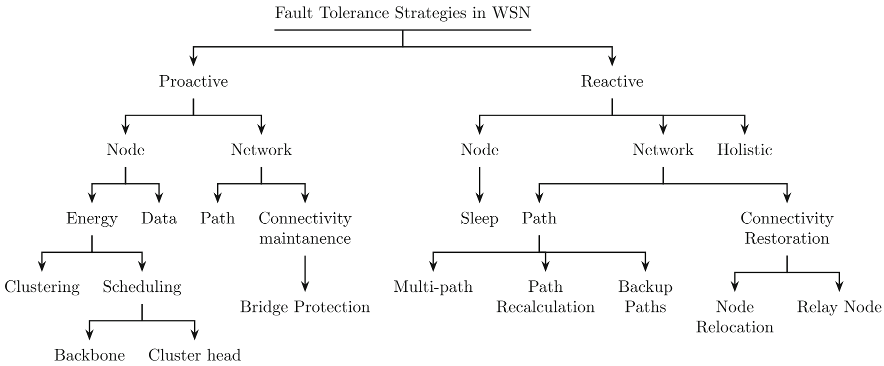

Fig. 1. Proposed taxonomy of fault tolerant techniques in WSN

Node Based Techniques. The node based proactive techniques can be further classified as (1) Energy based fault tolerance technique and (2) Data fault tolerance. Energy based fault tolerance increases the mean time to failure and the lifetime of the network. This strategy uses techniques such as clustering of sensor nodes, hibernation of nodes, and scheduling nodes and backbone of the WSN. Proactive data fault tolerant techniques helps in recovering from data faults. One of the major techniques of data fault tolerance is the dual transmission of the same value and comparison of these data to detect the faults.

Network Based Techniques. It comprises of mainly two techniques namely (1) Connectivity maintenance technique and (2) Multi-Path routing. Connectivity maintenance techniques increase the lifetime of network using various algorithms. Bridge protection algorithm is an example of connectivity maintenance algorithm that increases the lifetime of WSNs comprising bridged nodes. Data is sent through multiple paths to increase the redundancy and tolerate network fault in multi-path techniques.

2.2 Reactive Techniques

Reactive techniques trigger the fault tolerant strategy on the occurrence of the faults. This strategy waits for the faults to occur and then adjusts or reacts to the faultby starting the recovery process. These techniques can also be further classified based on the origin of the faults as (1) Node based, (2) Network-based and, (3) Holistic based technique.

Node Based Techniques. Node based reactive techniques are used to recover from node failures. It consist of strategies like switching to the sleeping backup node on the occurrence of node failure.

Network Based Faults. Network-based reactive techniques consist of using multiple paths, backup paths, and path recalculation in case of network/link failure. Moreover, for restoring the connectivity, extra nodes are deployed or the existing nodes are repositioned.

Holistic Techniques. These are the techniques that can deal and recover from both network and node based faults. They provide a complete fault tolerance for various faults.

3 Previous Work

In this section we shall discuss the existing work related to WSN fault tolerance techniques.

Zhao *et al.* [23] proposes a sleep scheduling technique called virtual backbone scheduling (VBS). Multiple backbones that overlap with each other are used to transmit data to the sink. The other nodes does not participate in the transmission to save energy. Selection of the backbones is an NP-hard problem. However, node failures might require recalculating the backbones.

Khan *et al.* [24] propose a fault-tolerant algorithm for bridge protection in WSNs. The authors propose a bridge protection algorithm to prevent the bridge node(s) from prematurely exhausting the energy and to maintain the minimal functionality of the network with minimal interference. But this technique has a trade of between time and residual energy.

Boucetta *et al.* [25] propose an energy efficient fault tolerant scheduling algorithm called PASC_AR. The network is clustered geographically based on node location. The cluster head is selected in rotation from the nodes based on a TDMA schedule. The rest of the nodes are put to sleep. However, the sensing accuracy is reduced due to the sleep mode of majority nodes.

Azharuddin *et al.* [22] propose an energy saving and FT routing technique called DFTR that not only deal with energy utilization of cluster heads but also their fault-tolerance. The routing is done based on following criteria: (1) gateway to next hop gateway distance, (2) next-hop gateway to base station distance, (3) energy remaining at the next-hop gateway. But, fixed gateways are not always suitable or plausible. Moreover, clustering is difficult when there are fixed gateways.

Rana [26] proposes a modified dynamic source routing algorithm (DSR) offering energy-efficient, fault-tolerant routing. The major features of this technique are (1) non-usage of nodes below certain energy threshold in the routing process, and (2) two routes cached between source and destination. There is a reduction in throughput of network for any given time period as all nodes are not involved in transmission

Gupta *et al.* [27] propose an energy-efficient fault tolerant clustering algorithm named B^3FT. In this technique, the authors discuss fault tolerance for cluster heads without the redundant usage of cluster heads. This scheme requires the use of extra hardware as gateways.

Azharuddin *et al.* [28] discuss a fault tolerant clustering based routing algorithm based on particle swarm optimization. They maximize the lifetime of the gateway with minimum lifetime by minimizing the routing load over the gateway. This is achieved with the help of particle swarm optimization. However, this scheme does not handle fault tolerance if no gateway is in range.

Hezaveh *et al.* [29] propose a technique called Fault-Tolerant and Energy Aware Mechanism (FTEAM). They put overlapped nodes with highest residual energies to sleep so they can be used as a cluster head (CH) in case of CH failures. It is only reliable when rate of change of sensed value is small inside the cluster.

Dima *et al.* [30] propose an integrated fault tolerance framework (IFTF) which holistically detects and diagnose application level faults, network layer faults, and establish the cause of the fault. However, there is a 4% increase in message overhead and does not consider the computation overhead.

4 Network and Radio Model

We consider a clustered WSN which consists of a single base station/sink and multiple clusters of sensor nodes. The sensor nodes in each cluster are normal nodes that are responsible for sensing and transmitting the data to their respective clusterheads (CH). All the nodes and the clusterheads are considered to be homogeneous with identical initial energy levels. The CHs are also normal nodes with the same energy constraints as that of sensing nodes. The CHs receive the sensed data, aggregates the data and forwards it to the base station. Direct data transmission occurs if the base station is one hop away from the CH else the aggregated data is forwarded to the CH closer to the base station. The nodes are deployed randomly as in smartdust model. The sensor nodes and CHs are considered immobile. There is only a single base station which is stationary and has an inexhaustible power supply. All sensor nodes have equivalent bi-directional communication range. All the wireless links are assumed to be symmetric so as to compute the distance between the nodes based on the received signal strength [31]. CSMA/CA MAC protocol is used by the CHs for communicating with base station [31]. For energy consumption analysis we only consider the energy used due to transmission and receiving of data since radio is the most power consuming parte as the consumption due to sensing and computing is negligible.

In this technique we use a radio model that is used in [3]. The energy dissipated E_T due to the transmission of a message of size l-bit, between two nodes separated by a distance d is given by

$$E_T(l,d) = \begin{cases} l(E_{elec} + \varepsilon_{FS} \times d^2) & for\,(d < d_0) \\ l(E_{elec} + \varepsilon_{MP} \times d^4) & for\,(d > d_0) \end{cases} \tag{1}$$

where $d_0 = \sqrt{\varepsilon_{FS}/\varepsilon_{MP}}$, E_{elec} is the electronic energy required by the circuit, ε_{FS} and ε_{MP} are the transmit amplifier parameters that represents the energy required by the amplifier in free space and multipath models respectively. The energy dissipation at the receiver sensor node for a message of size l-bit is given by

$$E_R(l) = l \times E_{elec} \tag{2}$$

Moreover, the energy consumed for fusing $l-$bits can be given by

$$E_F(l) = l \times E_{df} \tag{3}$$

where E_{df} is the energy incurred due to fusing of one bit data.

5 Proposed Technique

The proposed technique has two phases: (1) Setup phase, (2) Route determination phase, (3) Data communication phase and (4) Fault recovery phase.

5.1 Network Setup

Initially the network will be in setup phase. Initially all the sensor nodes send a HELLO message to the sink. The sink then assigns an ID to all sensor nodes. During the setup phase, we use any of the standard clustering algorithm to cluster the network and assign a cluster head to each cluster. The cluster head in each of the cluster sends a HELLO message to all its node with specific power and based on the strength of the signal received, it finds the nearest node to itself. The nearest node to the cluster head in each cluster is assigned as the vice cluster head. The sink then broadcasts a HELLO message using specific amount of power to all the cluster heads. The sink calculates the distances to each sensor node using the radio strength and this distance is send back to the cluster head. The setup phase ends and the communication phase starts wherein the nodes sends their data to the cluster heads and the cluster heads will fuse multiple identical values into a single value [3]. After certain amount of time the network switches back to the setup phase so as to balance the energy of the nodes in the network. Subsequently, it enters the communication phase and this process continues until the network encounters a fault.

5.2 Route Discovery and Routing Algorithm

We develop our routing technique on top of the foregoing medium access control (MAC) layer. The major steps in our routing technique is given below:

Step 1: Initially in the route determination phase, each cluster head broadcasts an REQ packet similar to that of Dynamic Source Routing (DSR). The REQ initially consists of the source ID, destination ID, the energy of the each cluster head and, the distance to the sink that was obtained during the bootstrap process.

Step 2: This REQ packet is flooded among other cluster heads, and each cluster adds to the packet their respective ID, energy level and, the distance to the base station.

Step 3: We define an energy threshold level. Any cluster head that has energy level below this threshold will not participate in the flooding process.

Step 4: The broadcasted REQ packets reaches the destination. For each cluster head, the sink starts a timer on the arrival of the first REQ packet from that cluster head. The sink will wait for more packets till timer expiry. Once the timer expires the sink will analyze and select the routes for each cluster head based on the remaining energy level and the sum of distance of the all cluster heads in the path to the sink. Based on this the routes for each cluster head is prioritized and are given priority numbers P1, P2,..., Pn.

Step 5: Thereafter, the sink sends a REP message to the cluster heads through all the discovered routes for the cluster heads. The REP message consists of the ID of the nodes that are in the path and the priority of that path.

Step 6: Once all REP messages reach the cluster heads they save the routes on basis of their priority and the route with priority P1 will be used for sending data to base station. The intermediate nodes between the source and destination also saves the routes.

Step 7: Once the routes are selected, the cluster heads pass the route information to their respective vice cluster heads and the vice cluster head resume their sleep state after storing this information.

The routing algorithm has been given in Algorithm 1.

5.3 Fault Tolerance

In this technique we only consider faults in routing especially disruption of route due to failure of cluster heads. We can consider the following cases of failures:

Failure of the source cluster head. When the source cluster head fails, the vice cluster head takes over the job of the cluster head. The routing table is already present in the vice cluster head as explained before.

Failure of the intermediate cluster head. When the data from the source cluster head is send ahead and one of the intermediate cluster head fails then the failed cluster head sends an error message (ERR) to the preceding cluster head. The preceding cluster head will switch its route from primary to secondary route and the faulty cluster head will be replace by the vice cluster head. if the secondary route also fails then it will use the tertiary route and so on. Since the routes have been stored this will save us from recalculating the routes again.

Failure of vice cluster head. On the instance of vice cluster head failure, we go back to the network setup phase we recluster the network and determine new routes.

The fault tolerance algorithm has been given in Algorithm 2.

Algorithm 1. Proposed routing algorithm

Input: $\forall CH_k,\ Energy_k,\ Distance\ to\ Sink_k$
Output: $All\ paths,\ from\ Source\ CH_i\ to\ Sink$

procedure CH–RouteSelection
  $Node_i$ receives REQ_i packet from it $NeighborNode(i)$
  if $Node_i! = sink$ **then**
    if $Energy(i) < E_thresh$ **then**
      $REQ_i \leftarrow REQ_i + (Id_i, Energy_i, Dist_i)$
      Forward REQ_i to $NeighborNodes(i)$
    else
      $Node_i$ does not broadcast
    end if
  else if $Node_i$ is == Sink **then**
    Start $timer_j$
    while $timer_i < Time_{thresh}$ **do**
      if $REQ == REQ_i$ **then**
        $REQSet(i) \leftarrow REQSet(i) \cup REQ_i$
      end if
    end while
    for each Request $REQ_i \in REQSet(j)$ **do**
      $Cost(j,i) \leftarrow 0.3 \times Dist(Source, Sink) + 0.7 \times Energy(Source, Sink)$
    end for
    for each row $Cost(j,:)$ **do**
      Sort $Cost(j,:)$
      Set Priority in Descending Order in $Cost(j,:)$
    end for
    for each REQ_i in $REQSet(j)$ **do**
      $REP_i \leftarrow REP_i + (Id_i, Path_i, Priority_i)$
      Forward REP_i to $Source_i$
    end for
    Node N_i creates routing table using REP messages.
  end if
end procedure

procedure CH–Routing
  Use the Path with $Priority = 1$
end procedure

6 Simulation Results and Discussion

6.1 Experimental Setup

Th proposed protocol was simulated using MATLAB R2015a on an intel i5 machine with 2.40 GHz and 16 GB RAM running on Ubuntu 15.10. We deployed 400 sensor nodes in a square area of size 300×300 square meters. The topology of the simulated network is illustrated in Fig. 2. The sensor nodes were considered to have a starting energy of 2 J. When the energy level of the node reached 0 J

Algorithm 2. Fault tolerance Algorithm

```
procedure CH–FaultTolerance
    for i ∈ Priority do
        if Path with Priority = i fails due to CH failure then
            if Vice CH not used then
                Awake Vice CH
                Replace CH with Vice CH
                Update routing Tables
            end if
        else
            Use path with Priority = i + 1
        end if
    end for
end procedure
```

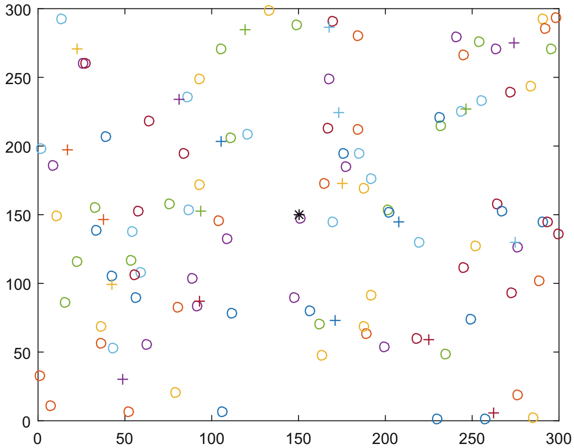

Fig. 2. The simulated network topology

the node was considered dead. We use Weibull reliability function [32] to model the faults in the cluster heads in our network which is given by

$$R(t) = e^{-\left(\frac{t-\gamma}{\eta}\right)^{\beta}} \tag{4}$$

where γ is the location parameter, η is the scale parameter, and β is the shape parameter. We set the values of $\gamma = 0$, $\beta = 3$, and $\eta = 3000$. If β is greater than 1 then the rate of failure increases with time else if β is less than 1 then the rate of failure decreases with time. Moreover, of $\beta = 0$ the rate of failure is constant. β is chosen as per the analysis provided in [33] where it is established that the failure of cluster heads can be represented using a weibull distribution with $\beta = 3$. Furthermore, [22] uses $\beta = 3$ for gateway faults in WSN. The simulation parameters used in the simulation are shown in Table 1. The parameters used are similar to [3].

Table 1. The simulation parameters

Simulation parameters	
Network size	400
Number of clusters	300×300
Initial sensor node energy	2.0 J
E_{elec}	50 nJ/bit
E_F	5 nJ/bit
Communication range	100 m
ε_{FS}	10 pJ/bit/m^2
ε_{MP}	0.0013 pJ/bit/m^4
d_0	88 m
Packet size	4,000 bits
Message size	200 bits
E_{thresh}	20%

The proposed algorithm is compared with DFTR [22] and LEACH [34] in terms of residual energy, number of packets received at sink, number of dead cluster heads, and network lifetime. We discuss the results of the experiments in the following sections.

6.2 Analysis of HMDSR

A wireless sensor network consisting of 120 nodes were simulated and was clustered initially into 20 clusters. These 120 nodes were deployed in a sensing field of size 300×300. The sink was placed at the center of the sensing field at the coordinates (150, 150). The simulated network is depicted in Fig. 2. The total number of alive nodes is compared in Fig. 4. We can see that the total alive nodes after 5000 rounds for the proposed technique are more than the LEACH and DFTR. Nodes are considered dead when their energy reaches 0 J or due to the simulation of faults following the Weibull distribution. We can observe in our proposed technique that initially there is a decrease in alive nodes that stabilizes after a certain amount of rounds.

The stability of alive nodes is due to the Energy threshold that was applied which resulted in many cluster heads with lower energy not to participate in the clustering. Whereas in LEACH protocol, we can observe that the rate of dead nodes increases after a certain number of rounds. The DFTR protocol that does not provide an Energy threshold has the least amount of total energy in the network. This is because once the cluster head dies in DFTR technique and nodes of clusters becomes orphan they have to send it to a longer distance. Since DFTR uses special fixed gateways, a failure in cluster head means another normal node doesn't take its place as a replacement, unlike the proposed technique. Figure 3 shows the total energy in the network per round for each of the technique. It is

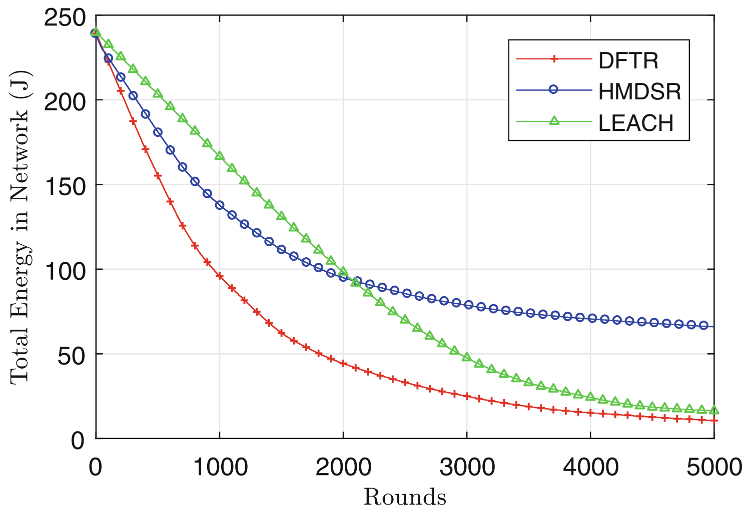

Fig. 3. Comparison of total energy in the simulated network per round among DFTR, LEACH, and HMDSR.

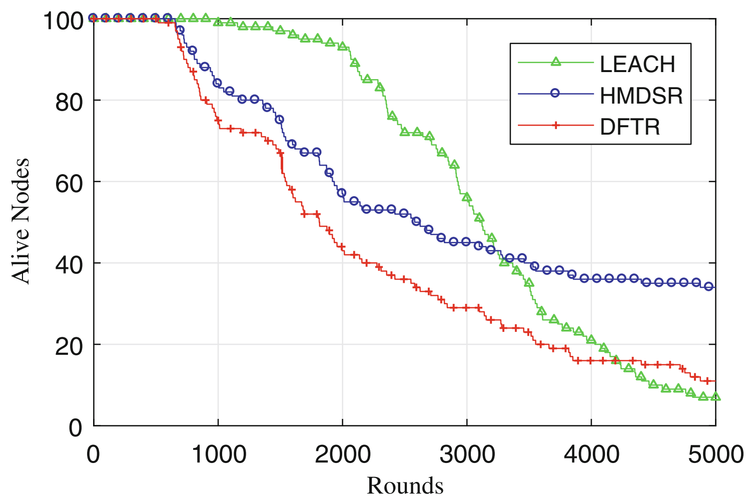

Fig. 4. The number of alive nodes per round in the simulated network among DFTR, LEACH, and HMDSR

also similar to the previous graph where the total energy at the end of 5000 round is highest in the proposed technique. Hence, we can clearly say that the proposed technique increases the overall lifetime of the network as compared to LEACH and DFTR.

The total number of packets that has been transmitted to the sink for 5000 rounds have been compared in Fig. 5. We can see that the proposed technique

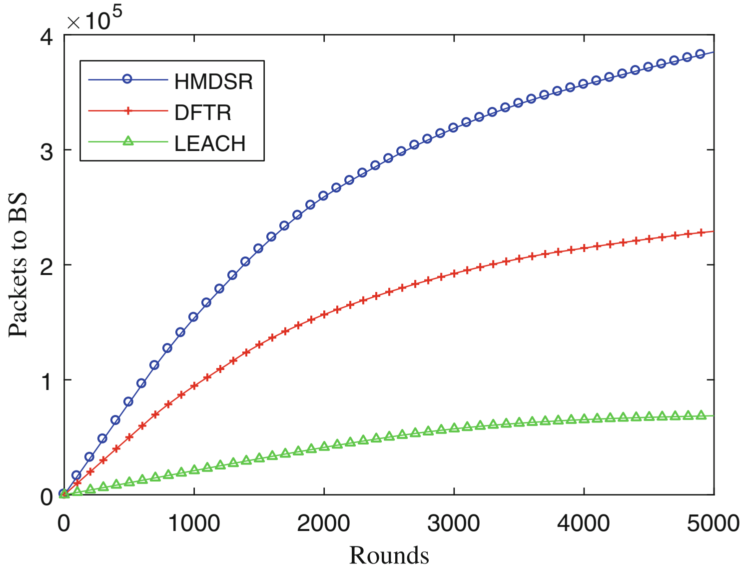

Fig. 5. The total number of data packets received at sink per round among DFTR, LEACH, and HMDSR

transmits the maximum amount of packets. This is due to the presence of vice cluster heads which replaces the failed cluster heads unlike LEACH protocol or DFTR protocol. A Higher number of packets transmitted to the base station indicates the longer life of gateways. We can see that the proposed algorithm performs better than both DFTR and LEACH.

7 Conclusions

In this article, we have proposed a reliable and resilient routing technique for wireless sensor networks that forms the atomic component of IoT, for smart city applications. We have proposed a taxonomy for fault tolerant techniques in WSN. Furthermore, we proposed a new fault-tolerant routing algorithms for hierarchical WSN networks based on modified DSR (Dynamic Source Routing) and vice cluster heads. Multiple routes are identified and these routes are prioritized on the basis of residual energy in the path and the distance of the source from the sink. In addition, the proposed technique uses vice cluster heads to tolerate faults during routing. We have shown through simulation that the proposed technique is better than LEACH and DFTR in terms of total energy in the network, the total number of packets transmitted to the sink and the number of alive nodes. Our future work will be based on the mobility of the sensor nodes and the interaction with other IoT components.

Acknowledgments. The authors acknowledge with thanks the technical and financial support from the Deanship of Scientific Research (DSR) at the King Abdulaziz University (KAU), Jeddah, Saudi Arabia, under the grant number G-651-611-38. The work carried out in this paper is supported by the High Performance Computing Center at the King Abdulaziz University, Jeddah.

References

1. United Nations Human Settlement Programme: Global Urban Observatory (GUO) UN-Habitat
2. Mehmood, R., Alam, F., Albogami, N.N., Katib, I., Albeshri, A., Altowaijri, S.M.: UTiLearn: a personalised ubiquitous teaching and learning system for smart societies. IEEE Access **5**, 2615–2635 (2017)
3. Heinzelman, W.B., Chandrakasan, A.P., Balakrishnan, H.: An application-specific protocol architecture for wireless microsensor networks. IEEE Trans. Wirel. Commun. **1**(4), 660–670 (2002)
4. Arfat, Y., Aqib, M., Mehmood, R., Albeshri, A., Katib, I., Albogami, N., Alzahrani, A.: Enabling smarter societies through mobile big data fogs and clouds. Procedia Comput. Sci. **109**, 1128–1133 (2017)
5. Alam, F., Mehmood, R., Katib, I., Albeshri, A.: Analysis of eight data mining algorithms for smarter internet of things (IoT). Procedia Comput. Sci. **98**(Supplement C), 437–442 (2016)
6. Alam, F., Mehmood, R., Katib, I., Albogami, N.N., Albeshri, A.: Data fusion and IoT for smart ubiquitous environments: a survey. IEEE Access **5**, 9533–9554 (2017)
7. Pantazis, N., Nikolidakis, S.A., Vergados, D.D.: Energy-efficient routing protocols in wireless sensor networks: a survey. IEEE Commun. Surv. Tutorials **15**(2), 551–591 (2013)
8. Casey, K., Lim, A., Dozier, G.: A sensor network architecture for tsunami detection and response. Int. J. Distrib. Sens. Netw. **4**(1), 27–42 (2008)
9. Wang, R., Zhang, L., Sun, R., Gong, J., Cui, L.: EasiTia: a pervasive traffic information acquisition system based on wireless sensor networks. IEEE Trans. Intell. Transp. Syst. **12**(2), 615–621 (2011)
10. Nikam, S.S., Mane, P.B.: Swarm intelligent WSN for smart city. In: Satapathy, S.C., Bhateja, V., Joshi, A. (eds.) Proceedings of the International Conference on Data Engineering and Communication Technology. AISC, vol. 469, pp. 691–700. Springer, Singapore (2017). https://doi.org/10.1007/978-981-10-1678-3_66
11. Bottero, M., Chiara, B.D., Deflorio, F.: Wireless sensor networks for traffic monitoring in a logistic centre. Transp. Res. Part C Emerg. Technol. **26**, 99–124 (2013)
12. Mehmood, R., Nekovee, M.: Vehicular ad hoc and grid networks: discussion, design and evaluation. In: Proceedings of the 14th World Congress on Intelligent Transport Systems (ITS), Beijing, October 2007
13. Morello, R., Mukhopadhyay, S.C., Liu, Z., Slomovitz, D., Samantaray, S.R.: Advances on sensing technologies for smart cities and power grids: a review. IEEE Sens. J. **17**, 7596–7610 (2017)
14. Cetinkaya, O., Akan, O.B.: Use of wireless sensor networks in smart homes. In: Emerging Communication Technologies Based on Wireless Sensor Networks: Current Research and Future Applications, pp. 233–258, April 2016

15. Shnayder, V., Hempstead, M., Chen, B.r., Allen, G.W., Welsh, M.: Simulating the power consumption of large-scale sensor network applications. In: Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems, SenSys 2004, pp. 188–200. ACM, New York (2004)
16. Abbasi, A.A., Younis, M.: A survey on clustering algorithms for wireless sensor networks. Comput. Commun. **30**(14–15), 2826–2841 (2007). Network Coverage and Routing Schemes for Wireless Sensor Networks
17. Li, Y., Xiao, G., Singh, G., Gupta, R.: Algorithms for finding best locations of cluster heads for minimizing energy consumption in wireless sensor networks. Wirel. Netw. **19**(7), 1755–1768 (2013)
18. Muhammed, T., Shaikh, R.A.: An analysis of fault detection strategies in wireless sensor networks. J. Netw. Comput. Appl. **78**(Supplement C), 267–287 (2017)
19. Gillies, D., Thornley, D., Bisdikian, C.: Probabilistic approaches to estimating the quality of information in military sensor networks. Comput. J. **53**(5), 493–502 (2010)
20. Gelenbe, E., Ngai, E.: Adaptive random re-routing for differentiated QoS in sensor networks. Comput. J. **53**(7), 1052–1061 (2010)
21. Ramanathan, N., Kohler, E., Girod, L., Estrin, D.: Sympathy: a debugging system for sensor networks [wireless networks]. In: Proceedings of the 29th Annual IEEE International Conference on Local Computer Networks, Washington, DC, USA, pp. 554–555. IEEE Computer Society, November 2004
22. Azharuddin, M., Jana, P.K.: A distributed algorithm for energy efficient and fault tolerant routing in wireless sensor networks. Wirel. Netw. **21**(1), 251–267 (2015)
23. Zhao, Y., Wu, J., Li, F., Lu, S.: On maximizing the lifetime of wireless sensor networks using virtual backbone scheduling. IEEE Trans. Parallel Distrib. Syst. **23**(8), 1528–1535 (2012)
24. Khan, S.A., Bölöni, L., Turgut, D.: Bridge protection algorithms - a technique for fault-tolerance in sensor networks. Ad Hoc Netw. **24**, 186–199 (2015)
25. Boucetta, C., Idoudi, H., Saidane, L.A.: Adaptive scheduling with fault tolerance for wireless sensor networks. In: 2015 IEEE 81st Vehicular Technology Conference (VTC Spring), pp. 1–5. IEEE (2015)
26. Ahmed, R.E.: A fault-tolerant, energy-efficient routing protocol for wireless sensor networks. In: 2015 International Conference on Information and Communication Technology Research (ICTRC), pp. 175–178, May 2015
27. Gupta, S.K., Kuila, P., Jana, P.K.: E3BFT: energy efficient and energy balanced fault tolerance clustering in wireless sensor networks. In: 2014 International Conference on Contemporary Computing and Informatics (IC3I), pp. 714–719. November 2014
28. Azharuddin, M., Jana, P.K.: A PSO based fault tolerant routing algorithm for wireless sensor networks. In: Mandal, J.K., Satapathy, S.C., Sanyal, M.K., Sarkar, P.P., Mukhopadhyay, A. (eds.) Information Systems Design and Intelligent Applications. AISC, vol. 339, pp. 329–336. Springer, New Delhi (2015). https://doi.org/10.1007/978-81-322-2250-7_32
29. Hezaveh, M., Shirmohammdi, Z., Rohbani, N., Miremadi, S.G.: A fault-tolerant and energy-aware mechanism for cluster-based routing algorithm of WSNs. In: 2015 IFIP/IEEE International Symposium on Integrated Network Management (IM), pp. 659–664, May 2015
30. Hamdan, D., Aktouf, O.E.K., Parissis, I., El Hassan, B., Hijazi, A.: Integrated fault tolerance framework for wireless sensor networks. In: 2012 19th International Conference on Telecommunications (ICT), pp. 1–6. IEEE (2012)

31. Xu, J., Liu, W., Lang, F., Zhang, Y., Wang, C.: Distance measurement model based on RSSI in WSN. Wirel. Sens. Netw. **02**(08), 6 (2010)
32. Antle, C.E., Bain, L.J.: Weibull distribution. In: Encyclopedia of Statistical Sciences, July 2004
33. Lee, J.J., Krishnamachari, B., Kuo, C.C.J.: Aging analysis in large-scale wireless sensor networks. Ad Hoc Netw. **6**(7), 1117–1133 (2008)
34. Balakrishnan, H., Heinzelman, W.R., Chandrakasan, A.: Energy-efficient communication protocol for wireless microsensor networks. In: 2014 47th Hawaii International Conference on System Sciences, vol. 08, p. 8020 (2000)