IJICTR
International Journal of Information & Communication Technology Research

Volume 10- Number 3 – Summer 2018 (19 -30)

DISOT: Distributed Selfish Node Detection in Internet of Things

Solmaz Nobahary
Department Computer Engineering
Science and Research Branch, Islamic Azad University
Tehran, Iran
solmaz.nobahary@srbiau.ac.ir

Hossein Gharaee Garakani*
ICT Research Institute
ITRC
Tehran, Iran
gharaee@itrc.ac.ir

Ahmad Khademzadeh
ICT Research Institute
ITRC
Tehran, Iran
a.khademzadeh@itrc.ac.ir

Amir Masoud Rahmani
Department of Computer Engineering
Science and Research Branch, Islamic Azad University
Tehran, Iran
rahmani@srbiau.ac.ir

Received: 2 December, 2017 - Accepted: 10 March, 2018

***Abstract*— Internet of Things (IOT) has prepared for a range of small sensors to popular laptops. Wireless communication in IOT systems assumes the nodes as a terminal as well as a router which can transmit the data packets. However, individual nodes may refuse to cooperate with others sometimes, leading to a selfish node behavior. The existence of selfish nodes degrades the network performance. This paper proposes to detect selfish nodes in IOT (DISOT) in three phases: Setup and Clustering phase which identifies and then clusters all the nodes in the network. The global phase which indicates whether a selfish node(s) exists in the clusters or not using the main cluster head and the cluster heads in each cluster must identify the selfish node(s) within the local phase. The proposed scheme is simulated by 2500 IOT nodes in the network and the results show that DISOT reduces end-to-end delay up to 41% and when the percentage of selfish nodes in the network does not exceed 35%, DISOT increases detection accuracy up to 10% and false positive rate decreases down to 5%.**

Keywords- Internet of things; selfish node; False Positive Rate (FPR); Detection Accurate (DA).

I. INTRODUCTION [1]

Internet of Things (IOT) is created to integrate the physical objects into information networks in order to equip human life with advanced and smart services [1]. Sensors or mobile devices and other devices similar to these, named as 'things', collect and aggregate the data of human life. Things, not only can aggregate the data, but also can process and extract beneficial and effective information to provide smart and intelligent services. IOT application developed in

* Corresponding Author

different environments such as smart cities, smart homes, healthcare and medical applications, Emergency networks, cognitive radio vehicular networks, military application [2-4]. Internet of Things includes many applications, but there are many challenges in IOT technologies. Some of the most important challenges include Architecture, Availability, Reliability, Mobility, Performance, Management, Scalability, Interoperability, Security and Privacy [5]. Addressing these challenges can provide the smart and intelligent services that are useful for human beings [6].

IOT system nodes perform not only as a terminal but also as a router that can forward data to other nodes. When a node doesn't forward the data packets and save energy supply, it will decrease performance in the network. The existence of selfish nodes increases energy consumption in the network, so the network in IOT needs to deal with non-cooperation nodes by detecting them and stimulating them to cooperate with other nodes. Existing solutions for detecting non-cooperation nodes in IOT systems fall into different categories. Reputation-based system [7-13], credit-based system [14-17], punishment based system [18-21], acknowledgment based system [22-23], game theory based system [24-27], and hybrid system [28-32] are methods that are used to detect and stimulate nodes to cooperate in networks. All methods and algorithms have their own advantages and disadvantages.

In this study, we propose a novel scheme to detect selfish nodes which are in the hybrid system category. The proposed mechanism uses the clustering method to provide monitoring and all the nodes are in the clusters and we select the cluster head(s) (CH) to monitor the member nodes in each cluster. The cluster head has directly communication with other CHs which are selected as main CH but other cluster heads may have multi hops to communicate with other CHs then main CH monitors other CHs activities like its traffic in the network. The proposed method is called DISOT which increases the network performance by detecting selfish nodes before several levels of the network are affected. An early alarm of detecting the selfish behavior can decrease energy consumption of the nodes in the network. The low false positive rate of the selfish node and high detection accuracy will decrease the average end-to-end delay. Finally, most of the data packets can receive the destination node by detecting the selfish node and may then isolate them from the network.

The relation between WSN and IoT is that the WSN is a subset of IoT because the node that posts the data to the Internet is considered as one Thing in IoT. The Source nodes in WSN shouldn't have any IP address. The source nodes should communicate with sink nodes via some routing protocols. In IoT each and every nodes should have IP address, so that cluster head can exactly know which node is selfish and then for future work can stimulate or isolate the node by cluster head.

Wireless Sensor Networks (WSN) is like the eyes and ears of the Internet of Things. It is the bridge that connects the real world to the digital world. And it is also responsible for passing on the sensed real world values to the Internet (WSN is thus involved with the hardware communication). The Internet of Things in a broad sense is like a brain, it can both store the real world data (in cloud services or databases) and can also be used to monitor the real world parameters, make meaningful interpretation and even make decisions based on the sensed data and also, the node in WSN has limit resources to support the DISOT method but IoT is responsible for the data processing, manipulation and decision making and can done DISOT protocol to detect the selfish node.

The rest of the paper is structured as follows: Section 2 presents an overview of the related work found in the literature. Section 3 presents definition and assumptions that are used in the new scheme. The proposed protocol properties and algorithm operation are described in details in Section 4. Section 5 evaluates the proposed method and discusses the method and compares it with the other similar methods. Finally, summary and conclusions are highlighted in Section7.

II. RELATED WORKS

As mentioned before, IOT is a network consisting of ad-hoc network and wireless networks. Enforcing node cooperation for transferring other node's packets is a major concern in an ad-hoc network and wireless networks. Most of the existing solutions are based on following mechanisms: reputation-based system, credit-based system, punishment based system, Acknowledgment based system, game theory based system, and Hybrid system.

A. Reputation-based system

One of the most important methods for detection of the node misbehavior is the Reputation-based method. The reputation-based method monitors the nodes which are forwarding all the data packets and then it incites the selfish node to stimulate cooperation in wireless multi-Hop networks [7, 8]. The different reputation methods have different algorithm to monitor the acts of the nodes to forward the other nodes' data packets. Watchdog mechanism was introduced by Marti et al. [9] in which a node overhears and monitors the node's behavior within its transmission range. If the neighbor nodes forward the data packet, it will be identified as the cooperation nodes and other nodes in the network will be informed. Buchegger and Le Boudec [10] designed a novel method to detect nodes with misbehavior which is called CONFIDENT (Cooperation of Nodes: Fairness in Dynamic Ad Hoc Networks). Misbehaved nodes are detected by neighbor nodes where information is sent to a reputation system and in this approach data packet of the low reputation (selfish) nodes, which collect adequate evidence and votes from monitoring nodes, is not forwarded. OCEAN (Observation based Cooperation Enforcement in Ad hoc Networks) was introduced to save energy of nodes by the act of local observation. The new scheme saves the nodes energy supply and doesn't need to send and exchange reputation between nodes. Each node saves

the reputation of its own neighbor nodes. First state of nodes is cooperation and it is changed by forwarding the data packets and the list of noncooperation nodes broadcasts during the route discovery [11]. In [12], Guo et al. introduced and designed a novel method that was modified and was better than the OCEAN. The new method was called HEAD (Hybrid Mechanism to Enforce Node Cooperation) which is composed of sent warning messages. The method uses DSR routing protocol to detect misbehavior of nodes and categorizes the nodes in three groups of malicious, selfish, and capture nodes which are isolated in the network. This method has the advantage of OCEAN method and saves nodes' energy supply and prolongs the network lifetime. Separation of Detection Authority (SDA) is the new scheme and it is proposed to detect the selfish node and trustworthiness of share information of nodes [13]. SDA has different entities: central authority, named reports, agents. When a node is submitted to the central authority as a selfish node, that node is named *"suspected node"* and the node that reported the selfish behavior is called the *"reporter"*. The central authority investigates the report, and then, the other neighbor nodes called "*agents*" will be observing the suspected node. The central authority evaluates the suspected node based on majority votes of agents and can detect the selfish node. SDA has high energy consumption because of having high exchanging message amount and observation to detect selfish behavior.

B. Credit-based system

Credit-based methods are used in currency to pay the nodes to forward the packets and nodes can gain more currency by cooperating with other nodes and that itself may stimulate the nodes to cooperate and forward the node's packets. Many works are studied by researchers about the credit-based incentive mechanisms which are presented as follows. The first method in the credit-based method is introduced as virtual currency by Buttyan and Hubaux and is called NUGLETS [14]. Nuglets are reworded to the forwarded data packets and they have two models, which are namely, Packet Purse Model (PPM) and Packet Trade Model (PTM). The difference between two models is the pain of nuglet which is a source node paid in PPM, but for PTM destination node pays the nuglets. Intermediate nodes will take nuglets according to forwarding data packets and nuglets are appended in the packet and in case of lack of nuglet in the packet, it will be dropped in PPM but the packets are traded between intermediate nodes and the destination node pays the total packet cost. In [15], the authors improved a virtual currency in which the method used a counter for each node. If the packet is forwarded successfully by a node, the node's counter will be increased and otherwise it will be decreased. The nodes will send their packet if their counter value is positive. The negative value of counter means the node will not forward the data packet and will be isolated from the network. In [16], Zhong et al. designed a new credit-based method for Mobile Ad-Hoc Networks which is known as Sprite (Simple, Cheat-Proof, Credit-Based System). The method uses a central agent and that is Credit Clearance Service (CCS). When a node forwards the packet, it will be claimed for payment from CCS. If the CCS received proof from corresponding nodes, the node achieves credits. The scheme prevented the payment from the destination and source, but CCS is the breakpoint of Sprite and the main disadvantage of this method. The sprite is improved by Rekha Kaushik et al. Called as MODSprite and reduced the overhead of sender down to 25% [17]. The protocol clustered all nodes in the network and whenever each member node received the data packets, it saved the receipt of the packet. The payment for each member node of clusters is paid by each of the cluster heads. MODSprite could overcome the breakpoint of Sprite.

C. Punishment based system

Punishment based methods use punishment and reward to let selfish nodes cooperate in the network. When a node saves its energy supply and refuses to forward the data packets it will be known as a selfish node. So, the punishment based system will punish the selfish nodes and the other nodes won't forward the selfish nodes' data packets in the network. Collaborative Reputation (CORE) was proposed by Michiardi and Molva to exclude the non-cooperative behavior of the nodes that persist to send a message [18]. The scheme involves requestors and providers. Both of them are two kinds of protocol entities. The requestor asks the provider for the execution of a function and waits for the result within a predefined time. If the outcome is the same and the two parties behave correctly, the requested function will be correctly executed and the reputation table (RT) will be updated, but when the reputation value in the global table is negative, the requested function will not be executed. Although, CONFIDANT could reduce the node, false positive rate, had more and more communication overhead. In [19], the new method was designed to decrease energy consumption and reduce the effect of misbehaving nodes known as secure and Objective Reputation-based Incentive (SORI). The method shares the nodes' reputation information with other neighbor nodes and punishes non-cooperating nodes by resisting to forward their data packets. A proactive protocol is introduced not only to detect misbehaving nodes in the network, but also to provide their Quality of Service (QoS) [20]. There are two kinds of nodes, the cluster heads and Multi-Point Relay (MPRs) nodes in the QoS-OLSR protocol. The protocol uses a Dempster-Shafer theory to improve the decision between nodes and uses Tit-for-Tat strategy to motivate individual nodes in a cluster to cooperate. In [21], a novel method was introduced to stimulate nodes to cooperate in the VANET. The protocol monitors the neighbor nodes and punishes the misbehaved nodes, the protocol is called Payment Punishment Scheme (PPS). The scheme establishes the stable clusters and uses the high weighted nodes of the cluster head by using the VCG model. PPS uses a modified Extended Dempster-Shafer model to evaluate the reporting of other three watchdogs as the reputation of nodes to detect the selfish and/or malicious nodes.

D. Acknowledgement based system

Another group of detection method applies the hop-by-hop acknowledgment scheme, the path which

is the way data packets passes through all intermediate nodes sending ACK packets back to the source node. TWOACK scheme was proposed by Balakrishnan et al. [22]. The nodes need two hops away from the source to send the ACK packets in this method. If a node doesn't receive an ACK packet in predefined time, it will decrease reputation of the next hop link. If the reputation value of a link is less than the threshold, the link is known as a misbehaved link and will not be used in the routing process. The method could reduce false accusation and increase the detection accuracy of misbehaving nodes, but the traffic congestion has increased in this method. The authors have improved the method by S-TWOACK (Selective-TWOACK) scheme. The S-TWOACK scheme uses less ACK packet, and the nodes send group ACK packet to the source. The S-TWOACK scheme has reduced traffic congestion. In [23], Liu et al. introduced a 2ACK scheme which improve TWOACK and S-TWOACK schemes. The scheme uses a one-way hash chain to provide authenticity of the packet and reduces the routing overhead. The scheme can detect the intruder nodes and links, but it has high traffic overhead.

E. Game theory based system

One of the economic methods that is used in computer sciences is the game theory which can be an analyzed interactive decision among things, for example, among network nodes. A special topic that uses game theory, stimulates nodes to cooperate in the wireless networks. The game theory proposes a mathematical model to make competition between nodes and hence make them cooperate to gain the highest performance in the game. By definition, a game should be defined the components for wireless networks: players are the nodes of the network, actions or strategies are like the data packets being forwarded, preference is a utility function in the game such as having higher performance, average throughput or low energy consumption, and end-to-end delay. Each player reacts to other players to gain the reward in the whole game. In [24], the authors employed a game theoretic approach for analyzing the interactions among the nodes in the network. The game used in the proposed scheme was an infinitely repeated game and a Worst Behavior Tit-for-Tat (WBTFT) incentive strategy was used to stimulate cooperation among the nodes. The proposed method uses a node to monitor the behavior of the node and makes decisions about the action in the nodes according to WBTFT. The strategy uses the perfect monitor, but analysis the system with imperfect monitoring. In [26], the scheme used game theory to detect the selfish node in MANET. The scheme used game theory to make decisions about credit value of each node. The strategy computes the credit value to detect the selfish behavior of nodes and uses game theory to model the interactions among the nodes of Mobile Ad-hoc network. The method can distinguish between selfish nodes and malicious nodes.

F. Hybrid based system

The hybrid scheme was introduced to benefit the advantage of two or more types of detection and incentive mechanism. The hybrid schemes used centralized and decentralized architecture to have the benefit of both architectures. A hybrid scheme, named 'hybrid incentive mechanism for cooperation stimulation (ICARUS)' was introduced to combine advantages of credits based and reputation based schemes [29]. The ICARUS reputation mechanism is based on the reputation scheme that are used in DARWIN [30]. ICARUS uses a central agent which is Credit Accounts Service (ICAS) similar to Sprite [16]. ICARUS aims to isolate and force selfish nodes to cooperate with other nodes to increase optimal consumption of the node's resources. In [31], a green approach is designed in two global and local phases. The first phase can detect the existence of selfish misbehavior in IEEE 802.11 based wireless networks and local phase can identify the nodes that are acting as selfishness nodes. Both phases frequently examined the network to save the network resources.

Trust-based Energy-Efficient Distributed Monitoring for Mobile Ad-hoc Networks (TEEM) was proposed to detect selfish and malicious nodes in the network [32]. The scheme is based on monitoring strategy and distributed time division. TEEM takes advantage of both trust and link between honest peer nodes by exchanging hello messages. The proposed method can achieve the highest security levels and less energy consumption .

All aforementioned schemes and algorithms are important and cannot be ignored; however, each of them has weaknesses in some circumstances that must be improved. To provide an efficient algorithm to detect selfish nodes in IOT, their strength points can be beneficial. In this paper, we present a new approach in three phases to detect a selfish node in IOT.

III. PROPOSED ALGORITHM

The approach is proposed to detect the selfish node in IOT system to prevent decreasing the network performance and throughput. The network nodes can have more effective and high performance when the nodes cooperate each other as a team in a cluster. Moreover, cluster heads can monitor and control forwarding message in the cluster. Hence, in this paper, a novel scheme to detect the selfish behavior for a cluster based forwarding, in which clusters are provided to detect the selfish nodes in the network. The proposed algorithm is consisted of three phases: Setup and Clustering phase, Global phase and Local phase. During the first phase (setup and clustering), things identify each other by sending hello messages and all the nodes in the network become a member of appropriate clusters and the proper cluster heads in each cluster will be determined to monitor the cluster member's operation during each round. The global phase is conducted by the main cluster head which has the most communication with the other cluster heads and indicates whether a selfish behavior exists in the clusters but does not identify the exact selfish nodes. Based on the results of the Global phase, If the global phase result is positive to detect the selfish behavior in clusters, the cluster heads in each cluster would be identified the selfish node(s) during the local phase. More details of each phase are explained in following. Fig.1 has shown the proposed scheme flowchart. Table 1 has shown the used notations in proposed method.

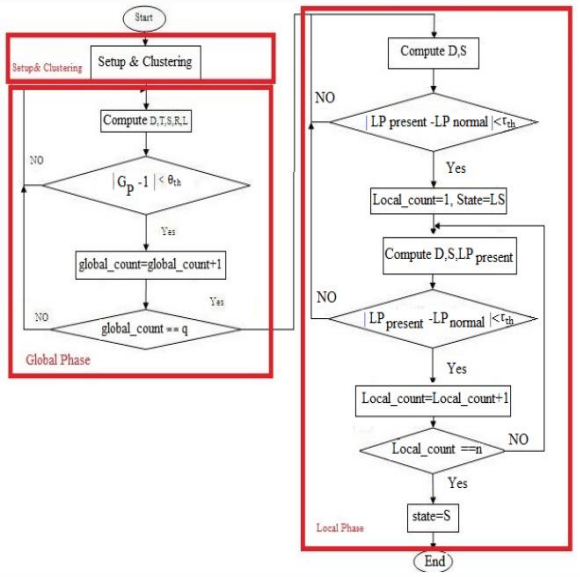

Figure 1. Flowchart of Proposed method

A. *Setup and Clustering phase*

During this phase, all things are randomly distributed in the area. Then, each node broadcasts a "Hello" message and the nodes replying to this message are known as a neighbor node. Each node will store some information in its database regarding its status or that of its neighbor in its database as a table consisting of four fields which is shown in Fig.2

Figure 2. Format of data in cluster heads

Node's ID	#Number of hops to the main cluster head	Node's Data	Node's Status

In the following, more details are discussed about each field as shown in Fig 2.

- Node's ID: It has 16 bits to save node's Identification.
- Number of hops up to the main cluster head: It has 8 bits to save the number of hops between a node and the main cluster head
- Node's Data: It is an array of n bits and saves the data of each node and its neighbors. The amount of n is per byte.
- Node's Status: It is an array of n bits and indicated the status of nodes. Because the nodes can have one of C, S, and LS statues, therefore, the length of each array is considered as 2 bits. The predefined status of this field is C which is assumed as the cooperation node.

After the identification of the neighbor nodes, clusters are established using clustering algorithm and the proper cluster head is selected for each cluster. The clustered network selects the cluster heads based on parameters as shown in Fig.3. After the clustering of the network nodes is finished the first phase of the suggested algorithm will be finished and the clusters will be updated if necessary.

Notation	Description
P	*The total number of things*
q	*The total number of rounds*
n	*number of nodes in each cluster*
C	*cooperation node*
LS	*likely selfish node*
S	*selfish node*
r	*number of network rounds*
N_{Chi}	*cluster head i*
$N_i\_state_r$	*State of node i in each round*
N_i_Fstate	*State of node i at the end of detecting phase*
$local\_count$	*Counter of data collection round in local phase*
$global\_count$	*Counter of data collection round in Global phase*
τ_{th} ,θ_{th}	*Two predefined threshold values to detect network variation*

TABLE I. NOTATION

B. *Global phase*

In this section, we present the global phase of the proposed method which is the main idea of our technique. The global and local network properties of the network are changed in the presence of selfish nodes. Global phase detects whether the network contains selfish nodes or not. None of the selfish nodes are identified in the global phase. If there is any selfish node the global phase will call upon the local phase. The local phase is costlier than global phase since each cluster head should monitor all member nodes in the cluster. The global phase is conducted by the main cluster head and can provide early alarm in case of a selfish behavior of a node in the network. The benefit of early alarm is that the selfish behavior won't be able to affect several levels of the network. In case of encountering a selfish node, the cluster heads will be wasting their energy on monitoring the member nodes action and member nodes will be wasting their energy on generating data packets while being unable to send these packets to the destination. As a result, our scheme is decreasing energy consumption in all the nodes of the network.

The existence of selfish behavior may lead to varying of the network parameters such as the network throughput, network load, the number of sent/received packets, packet delay and etc. considering the different factors involved there is a need for metric alignment. Our suggested method will combine these parameters together in one global parameter (GP).

In this phase, we present the amounts of the global properties in the proposed technique to detect selfish behavior. We combine the mentioned parameter to compute the G_P by using the following equation (1):

Where$\alpha,\beta,\gamma,\theta,$ and ϑ are weight factors, $\alpha+\beta+\gamma+\theta+\vartheta=1$ is the factor condition, D is the average end-to-end delay in μs, L is the network traffic in bit/s

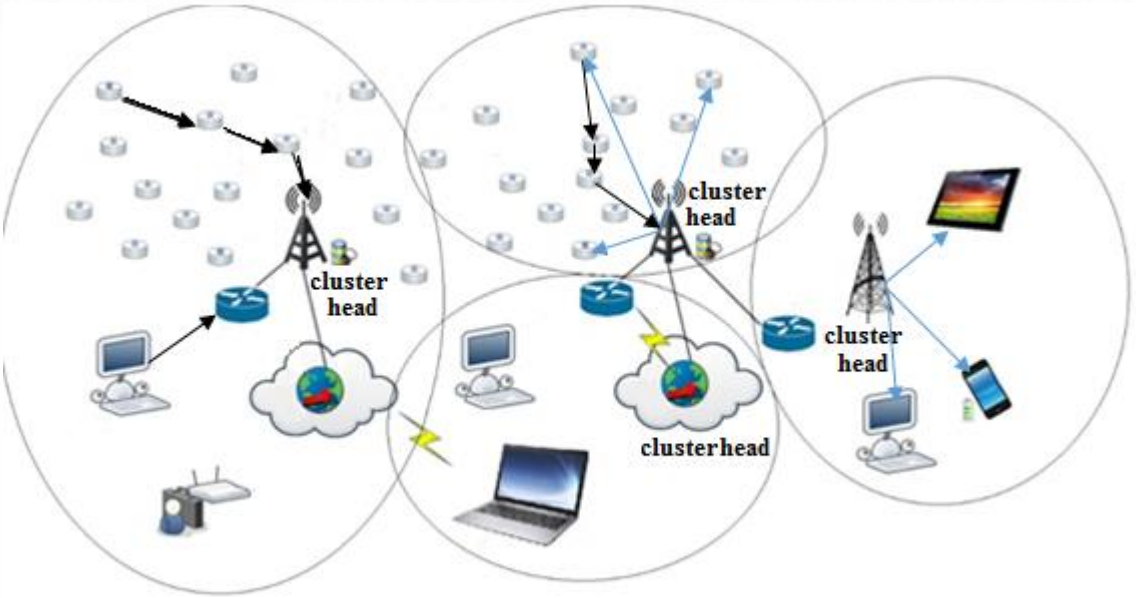

Figure 3. Clustered network at the end of the first phase

$$G_P = \alpha * \frac{1}{\frac{T_{present}}{T_{normal}}} + \beta * \frac{1}{\frac{R_{present}}{R_{normal}}} + \theta * \frac{D_{present}}{D_{normal}} + \gamma * \frac{L_{present}}{L_{normal}} + \vartheta * \frac{S_{present}}{S_{normal}},$$

$$\alpha + \beta + \theta + \gamma + \vartheta = 1 \qquad (1)$$

packets is shown as R. The subscript "present" correspond to present measured value for these parameters and "normal" correspond to normal value of network without any selfish behavior.The average network throughput and the total number of received the data packet have a negative relation with the number of selfish behaviors, so that as the selfish behavior increases the network load and the total received data packet decrease but as the selfish behavior increases the average end-to-end delay, total number of sent data packets and the network load will increase too. The value of factors is selected based on the importance of each weight factor parameter. In equation (1) traffic is one of the parameter and other parameters are involved to know deviate. If we are worry about dynamic network, we can select α as little value to reduce the effect of dynamic traffic in network.

The main cluster head monitors the global parameters in all clusters during measure time and it has the most direct communication with other cluster heads. If the main cluster head finds the selfishness behavior in the network, it will send a message to all cluster heads to run the local phase and identify the selfish node in their clusters. Simi-code of the selfish node detection in global phase is shown in Fig.4

Algorithm. Global Phase

Step 0: Clustered nodes in setup phase
Step 1: global_count=0
Step 2: main cluster head monitors the network for each q (predefined) period
START: Compute D, T, S, R, L
Compute G_P in present state
$G_p = \alpha * \frac{1}{\frac{T_{present}}{T_{normal}}} + \beta * \frac{1}{\frac{R_{present}}{R_{normal}}} + \theta * \frac{D_{present}}{D_{normal}} + \gamma * \frac{L_{present}}{L_{normal}} + \vartheta * \frac{S_{present}}{S_{normal}}$
if present= normal **then**
    G_p=1
else
  if $|G_P - 1| < \theta_{th}$ **then**
    global_count=1, the cluster "k" has deviation and main cluster head should monitor it
      for i=1 to q-1 **do**
          Compute D, T, S, R, L
          Compute G_P
          if $|G_P - 1| < \theta_{th}$ **then**
              global_count= global_count +1
          Endif
      Endfor
      if global_count=q **then**
          go to local phase
      else
          go to START
      endif

```
24:  Endif
25:  Endif
```

Figure 4. Simi-code of global phase

After setup and clustering phase, the main cluster head monitors the values of global parameters in the network during q rounds where the main cluster head has been selected among other cluster heads with the most communication with others in the network. In addition, the proposed method requires reference values for both G_P and L_P by which the presently measured parameters are compared to judge the nodes' behavior. G_P in normal condition indicate the parameters of the nodes and the clusters in the network working under a normal condition where there is no selfish node and present condition will be equal as normal condition then G_P value is 1 but if the G_P value deviates from 1 by a predefined threshold value of θ_{th} then this situation is marked as having a selfish node in the cluster. The threshold value is divided into groups soft and hard threshold. The soft threshold is also called *wavelet shrinkage*, as values for both positive and negative coefficients are being "shrinked" towards zero, in contrary to hard threshold which either keeps or removes values of coefficients. In our approach we don't need to shrink towards zero. And also, to suppress the noise hard threshold apply the nonlinear transform to the empirical wavelet coefficients.

The proposed algorithm to detect selfish node starts at line 2 of initializing a global_count with an initial amount of zero for each cluster head. This counter is used to count the number of continuous monitoring during rounds, if the main cluster head is detected to have selfish nodes in one cluster. In lines 4 and 5, the main cluster head monitors the global parameters in the network and computes value of G_P. If the G_P value deviates from 1 by a predefined threshold value of θ_{th} as found on line 10 (where $\theta_{th} > 0$), then this situation is marked as having a selfish node in the cluster "k" in the network. In lines 13-16 the main cluster head monitors the G_P for the next following rounds which I shows the number of rounds and counts the global_count during rounds which the cluster "k" in the network is marked as behaving selfish. The value of *global_count* is compared by q rounds so that, if these are equal it is a confirmation that the cluster "k" in the network contains a selfish node(s) (lines 19-23) and so the main cluster head calls upon a local phase in the cluster "i", otherwise, the counters of clusters will initialize from zero and monitoring is continued again.

C. Local phase

For the local node behavior in each cluster, the local parameters are combined in one round by the cluster heads to compute local parameter (LP) of each node in each cluster. As mentioned before, two main parameters are observed to compute LP: the total number of the sent data packet and the average end-to-end delay. As the selfish behavior of a node more increases, the total number of the sent data packet also increases and the node prefers to send its own data packet and not that of others or it only sends the data packet of other nodes by accident. As result, the other nodes should send the data packet many times to receive the destination so the total number of sent data packet is increased and it makes a higher end-to-end delay of the data packets. To combine the effect of these two positively-related parameters they must be combined with each other. LP is calculated according to the equation 2 for the cluster heads of all clusters.

Algorithm. Local Phase

```
1:  Each N_Chi monitor all its cluster nodes
2:  for ∀N_i ∈ P do
3:  Compute D, S
4:  Compute LP_present
5:  local_count=0
6:  N_i_state_r=C
7:      if |LP_present − LP_normal| < τ_th then
8:          local_count=1
9:          N_i_state_r=LS
10:         for i=1 to n-1 do
11:             Compute D, S, LP_present
12:             if |LP_present − LP_normal| < τ_th then
13:                 local_count= local_count +1
14:             endif
15:         endfor
16:         if local_count=n then
17:             N_i_Fstate=S
18:         endif
19:     endif
20: Endfor
```

Figure 5. Simi-code of local phase.

$$LP_i = \delta * \frac{S_{totali-i}}{S_{totali}} + \omega * \frac{1}{\frac{D_{total-i}}{D_{total}}} \quad (2)$$

Where δ and ω are weight factors with the condition $\delta + \omega = 1$, S is the total number of the sent data packets in the cluster and the average end-to-end delay is shown as D in μs, and the subscripts "totali"

and "totali-i" correspond to the total number of node "i" measured value for these parameters and the total number of nodes except for the node "i", respectively. Although, the values of the weight factors in equation 2 depends on the importance of the parameters, the total number of the sent data packet is the major factor that captures the main difference between the nodes' behavior. Thus, it is suggested to assume δ to be much larger than ω. The Simi-code in Fig.5 illustrates the local phase.

After Detecting selfish behavior in the cluster "k", the main cluster head sends an alarm to the head clusters in order to run local phase in the cluster "k". The local phase follows the same algorithm of the global phase only with different measuring parameters and the cluster head has the ability to monitor each cluster member node in the cluster. In lines 28 and 29 the cluster head monitors the local parameters of the end-to-end delay and the total number of sent data packet in the cluster and computes each node's present value for LP denoted as $LP_{present}$. Each node is assumed to have a cooperator initial state in line 31. If the $LP_{present}$ value deviates from the LP_{normal} by a predefined threshold value of τ_{th} (where $\tau_{th} < 1$) as found in line 32, then this situation is considered of emergence of a selfish node and the node state is changed to likely selfish (LS) in line 34. The cluster head can't make the decision about the node behavior once $LP_{present}$ is less than LP_{normal} and as a result, the cluster head continues monitoring for n rounds and counts it by local_count variable (lines 35-40). Once local_count is equal to n, then the node state will be changed to the selfish node (lines 41-43). The cluster head detects the node as a selfish node and refuses to send the data packet for further relay.

Protocol evaluation

The proposed approach has made decisions about both of the cooperation and selfish nodes by cluster heads. To evaluate the proposed scheme, we were simulated in Window 8.1 basic (64-bit), core i7 processors, 370 M processors, 2.40 GHz of speed with a memory of 8 GB and MATLAB 2015 software. The simulation parameters are shown in Table 2.

TABLE II. USED PARAMETERS IN SIMULATION

parameters	Values
E_{elec}	50 nJ/bit
E_{DA}	5 nJ/bit/signal
$\mathcal{E}_{fs}$	10 pJ/bit/m²
$\mathcal{E}_{amp}$	0.0013 pJ/bit/m⁴
Packet size	512 bits
d_0	87 m
Initial energy	1-5 J
Transmission_Power	1.65 μw
Receiving_Power	1.1 μw
Transition_Power	0.6 μw
Idle_power	1.0 μw
Transition_Time	0.005 ms
N	700

According to the simulated model, the network has the following assumptions:

- All things have been uniformly distributed in the worked area.
- each node has a unique identifier
- transmission energy consumption is proportional to the distance of the nodes

The number of selfish nodes varies from 10% to 50% of the total nodes in the clustered network. We supposed the network normal state is all nodes are cooperating nodes and there is 0% selfish nodes. The simulation result is performed 100 runs and the average result has shown in different metrics. The performance of the proposed method is compared with the QoS-OLSR, PPS, RandomTWOhopAck, and TWOhopAck protocols for evaluation metrics such as detection probability, the percentage of false positive rate and end to end delay.

TABLE III. DETECTION ACCURACY AND FALSE POSITIVE RATE OF DISOT IN COMPARE WITH OTHER METHODS

Selfish node Rate / Algorithms	10%	15%	20%	25%	30%	35%	40%	
PPS [21]	0.76	0.72	0.78	0.79	0.77	0.76	0.74	Detection Accuracy
RandomTWOhopack [22]	0.36	0.42	0.44	0.46	0.51	0.53	0.55	
QOS-OLSR [20]	0.71	0.72	0.79	0.78	0.75	0.74	0.72	
TWOhopack [22]	0.27	0.36	0.48	0.51	0.52	0.55	0.58	
DISOT [proposed-method]	0.85	0.82	0.80	0.82	0.84	0.71	0.69	
PPS [21]	0.25	0.22	0.28	0.29	0.27	0.27	0.31	False Positive rate
RandomTWOhopack [22]	0.09	0.09	0.09	0.08	0.06	0.07	0.09	
QOS-OLSR [20]	0.21	0.22	0.29	0.28	0.25	0.24	0.22	
TWOhopack [22]	0.11	0.09	0.07	0.07	0.08	0.08	0.09	
DISOT [proposed-method]	0.081	0.085	0.07	0.069	0.06	0.058	0.05	

D. Result and discussion

In the proposed scheme, three phases are designed based on clustering method. Based on that, each member nodes are monitored by cluster head and are prevented to send data packets to the selfish nodes. On the other hand, any misbehavior or deviation from the forwarding the packets will be known as selfish node.

Detection probability: the node status is based on cluster heads' decisions. Fig.6 shows the detection probability of proposed scheme with other methods. It can be noted that while detecting selfish nodes, DISOT performs better than QoS-OLSR with above 80% detection probability when the percentage of selfish nodes is not above 35. This is because the global phase can detect the selfish node with importance network factors based on G_p value. Thus, the decision of selfish nodes is done by each cluster head in local phase with importance network factors of clusters, which increases the probability of accurate decisions.

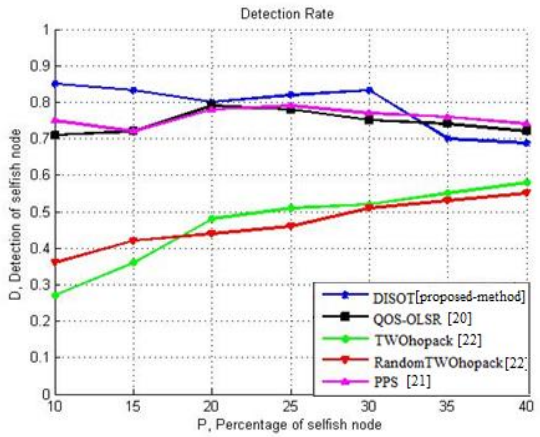

Figure 6. Detection rate of DISOT in compare with other methods

Percentage of false positive rate: Fig.7 describes the impact of false positive rate percentage of DISOT in comparison with PPS, QoS-OLSR, RandomTWOhopack, and TWOhopack. It can be noted that the false positive rate of DISOT scheme deteriorates with increased percentage of selfish nodes.

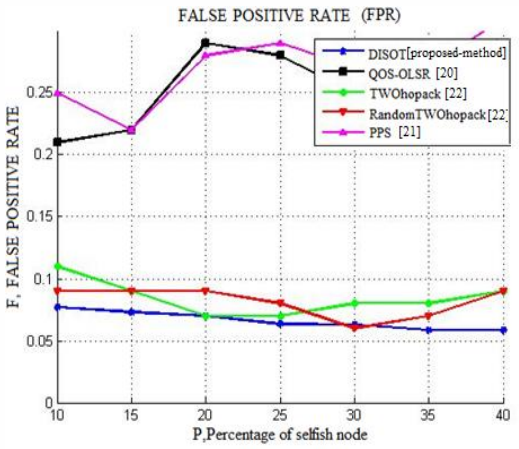

Figure 7. False Positive Rate of DISOT in compare with other

However, DISOT keeps uniform false positive rate in all of percentage of selfish nodes. The percentage of false positive rate is from 8% to 5%. DISOT can detect the selfish nodes with the less false judgment process. Table 3 has shown the detection accurate in first part of the table and the second part of the table has shown percentage of the false positive rate in DISOT in compared by the similar methods.

Energy consumption: IOT system nodes are consisted of sensor nodes and the nodes have mobility similar MANET nodes. So that, each node uses each group energy model. The lower energy consumed in the networks, the efficiency of the proposed method is higher. Due to the random acceleration of mobile nodes in IOT network, the energy consumption varies in a certain range. Average energy consumption in all nodes in IOT network vary 1.2 -1.35 in µjoule. When the nodes change in network and the total number of sent packets is high, the power consumption also increases. Fig.8 has shown average energy consumption in the network during 50 rounds which is the network continues the normal work and the proposed method detects the selfish nodes.

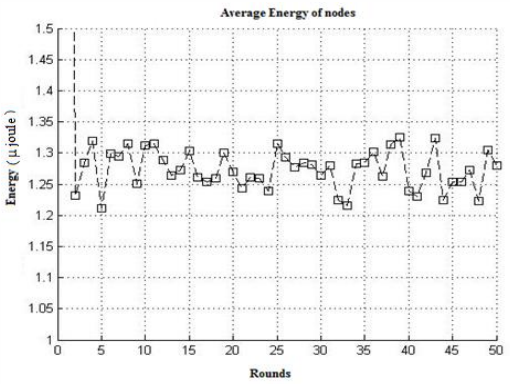

Figure 8. Average energy consumption of the nodes in DISOT during rounds

Average end-to-end delay: Fig.9 depicts average end-to-end Delay in DISOT compared with different methods. The average end-to-end delay is the arrival time of a packet from the source node to the destination. The average end-to-end delay of the proposed method is lower than other algorithms. The increase in the number of the selfish nodes, the average end-to-end delay increases. As the number of selfish nodes increases, it takes a lot of time to get a packet to the destination. Because packets are either discarded by selfish nodes or delayed. Therefore, the network has to resend the data packets. Resending the data packets in the network will cause network power loss and decreases network lifetime and increase the average end-to-end delay. So that, DISOT can detect the selfish nodes soon and it will reduce the end-to-end delay of the data packets. The advantages of the proposed method is detecting selfish nodes at high speeds, so the effects on the network are less. The node is detected and the packets are lost less than other methods at the very beginning of selfish behavior.

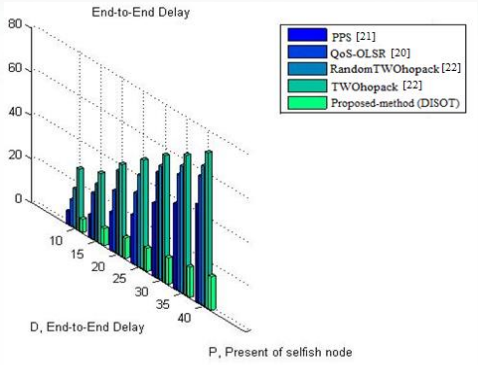

Figure 9. Average end-to-end nodes in DISOT in different percentages of selfish nodes

In most IoT application, the standard deviation for the dataset is not known. In this case, the standard deviation σ is known as the standard error. Since the standard error is an estimate for the true value of the standard deviation, the distribution of the sample mean $\bar{x}$ follows the t distribution with mean μ and standard deviation $\frac{s}{\sqrt{n}}$. The t distribution is also described by its degrees of freedom. For a sample of size n, the t distribution will have n-1 degrees of freedom. The notation for a t distribution with k degrees of freedom is t(k). For a population with unknown mean μ and unknown standard deviation, a confidence interval for the population mean, based on a simple random sample (SRS) of size n, is $\bar{x} \mp t^* * \frac{s}{\sqrt{n}}$, where $t^*$is the upper (1-C)/2 critical value for the t distribution with n-1 degrees of freedom, t(n-1). For example, the estimated standard deviation for the sample mean for one application in the IoT in our simulation is 0.733/sqrt (700) = 0.082, the value provided in the SE mean column of table IV. A 90% confidence interval, then, is approximately ((29.3 - 1.64*0.082), (29.3 + 1.64*0.082)) = (29.3 - 0.13, 29.3+ 0.13) = (29.17, 29.43). The estimated standard deviation for the sample mean is 0.733/sqrt (700) = 0.082, the value provided in the SE mean column of table IV. A 90% confidence interval, then, is approximately ((31.2 - 1.64*0.082), (31.2 + 1.64*0.082)) = (31.2 - 0.13, 31.2+ 0.13) = (31.07, 31.33).

TABLE IV. DESCRIPTIVE STATISTICS

N	Mean($\bar{x}$)	t^*	standard deviation	SE mean
700	29.3	1.64	0.73	0.082
700	31.2	1.64	0.73	0.082

IV. SUMMARY AND CONCLUSION

IOT is the technique that combines physical objects to different technologies to make advanced and intelligent application for the human being. High performance and reasonable security are the most important features in the various IOT applications. Unfortunately, its operation relies on the nodes behaving and cooperating with other nodes, but some of the nodes behave selfishly. A selfish node takes advantage of the network resources, but it doesn't share its resources and cause to decrease the average throughput in the network. In this paper, we use the network attribute which is affected by selfish behavior such as the number of the sent packets, end-to-end delay. In both Global and local phases some parameters of the network are monitored to detect the selfish behavior. We proposed DISOT for selfish misbehavior detection in IOT networks. The solution can save the network resources and can detect the selfish behavior in its early stages. DISOT algorithm is evaluated using MATLAB and the results showed that the suggested mechanism can successfully detect misbehaving in IOT based on wireless networks. Future work involves mathematical analysis to select suitable network metrics to distinguish the selfish behavior. The proposed scheme is evaluated using MATLAB to compare with other methods. The simulation is done with the existence of one or more selfish node in clusters.

REFERENCES

[1] Z. Yan, P. Zhang, and A. V. Vasilakos, "A survey on trust management for Internet of Things," *J. Netw. Comput. Appl.*, vol. 42, pp. 120–134, 2014.

[2] S. H. Shah and I. Yaqoob, "A survey: Internet of Things (IOT) technologies, applications, and challenges," *2016 IEEE Smart Energy Grid Eng.*, vol. i, pp. 381–385, 2016.

[3] J. ROTSTEIN, "Internet of Vehicles: From Intelligent Grid to Autonomous Cars and Vehicular Clouds," *Arch. Phys. Med. Rehabil.*, vol. 46,1965, pp. 198–9.

[4] Y. Saleem, M. H. Rehmani, and S. Zeadally, "Integration of Cognitive Radio Technology with unmanned aerial vehicles: Issues, opportunities, and future research challenges," *J. Netw. Comput. Appl.*, vol. 50, 2015, pp. 15–31.

[5] P. Rawat, K. D. Singh, and J. M. Bonnin, "Cognitive radio for M2M and Internet of Things: A survey," *Comput. Commun.*, vol. 94, 2016, pp. 1–29.

[6] A. Bader, H. Elsawy, M. Gharbieh, M. S. Alouini, A. Adinoyi, and F. Alshaalan, "First Mile Challenges for Large-Scale IoT," *IEEE Commun. Mag.*, vol. 55, no. 3, pp. 138–144, 2017.

[7] Y. Zhang, "Detection and Isolation of Packet Droppers in Wireless Ad-Hoc Networks," 2011.

[8] N. Samian, Z. A. Zukarnain, W. K. G. Seah, A. Abdullah, and Z. M. Hanapi, "Cooperation stimulation mechanisms for wireless multihop networks: A survey," *J. Netw. Comput. Appl.*, vol. 54, 2015, pp. 88–106.

[9] S. Marti, T. J. Giuli, K. Lai, and M. Baker, "Mitigating routing misbehavior in mobile ad hoc networks," *Proc. 6th Annu. Int. Conf. Mob. Comput. Netw. MobiCom 00*, vol. 1, no. 18, pp. 255–265, 2000.

[10] S. Buchegger and J.-Y. Le Boudec, "Performance Analysis of the CONFIDANT Protocol (Cooperation Of Nodes: Fairness In Dynamic Adhoc NeTworks)," *MobiHoc '02 Proc. 3rd ACM Int. Symp. Mob. ad hoc Netw. Comput.*, pp. 226–236, 2002.

[11] S. Bansal and M. Baker, "Observation-based Cooperation Enforcement in Ad Hoc Networks," 2003.

[12] J. Guo, H. Liu, J. Dong, and X. Yang, "HEAD: A Hybrid Mechanism to Enforce Node Cooperation in Mobile Ad Hoc Networks," *Tsinghua Sci. Technol.*, vol. 12, no. SUPPL. 1, 2007, pp. 202–207.

[13] O. León, J. Hernández-Serrano, and M. Soriano, "Outwitting smart selfish nodes in wireless mesh networks," *Int. J. Commun. Syst.*, vol. 23, no. 5, 2010, pp. 633–652.

[14] L. Buttyan and J. Hubaux, "Nuglets : a Virtual Currency to Stimulate Cooperation in Self-Organized Mobile Ad Hoc Networks 1 Introduction," *Technology*, pp. 1–15, 2001.

[15] L. Buttyan and J. Hubaux, "Stimulating Cooperation in Self-Organizing Mobile Ad Hoc Networks 1 Introduction," *Tech. Rep. DSC/2001/046, EPFL-DI-ICA*, pp. 1–23, 2001.

[16] J. Chen and Y. R. Yang, "Sprite: A Simple, Cheat-Proof, Credit-Based System for Mobile Ad-Hoc Networks," vol. 0, no. C, 2003, pp. 1987–1997.

[17] R. Kaushik and J. Singhai, "MODSPIRITE : A Credit Based Solution to Enforce Node Cooperation in an Ad-hoc Network," vol. 8, no. 3, 2011, pp. 295–302.

[18] P. Michiardi and R. Molva, "CORE : a collaborative reputation mechanism to enforce node cooperation in mobile ad hoc networks," *Adv. Commun. Multimed. Secur. IFIP TC6/TC11 Sixth Jt. Work. Conf. Commun. Multimed. Secur. Sept. 26-27, 2002, Portorož, Slov.*, p. 107, 2002.

[19] Qi He, Dapeng Wu, and P. Khosla, "SORI: a secure and objective reputation-based incentive scheme for ad-hoc networks," *2004 IEEE Wirel. Commun. Netw. Conf. (IEEE Cat. No.04TH8733)*, p. 825–830 Vol.2, 2004.

[20] O. A. Wahab, H. Otrok, and A. Mourad, "A dempster-shafer based tit-for-tat strategy to regulate the cooperation in VANET using QoS-OLSR protocol," *Wirel. Pers. Commun.*, vol. 75, no. 3, 2014, pp. 1635–1667.

[21] A. Jesudoss, S. V. Kasmir Raja, and A. Sulaiman, "Stimulating truth-telling and cooperation among nodes in VANETs through payment and punishment scheme," *Ad Hoc Networks*, vol. 24, no. PA, 2015, pp. 250–253.

[22] K. Balakrishnan, J. D. J. Deng, and V. K. Varshney, "TWOACK: preventing selfishness in mobile ad hoc networks," *IEEE Wirel. Commun. Netw. Conf. 2005*, vol. 4, no. C, pp. 0–5, 2005.

[23] Kejun Liu, Jing Deng, P. K. Varshney, and K. Balakrishnan, "An Acknowledgment-Based Approach for the Detection of Routing Misbehavior in MANETs," *IEEE Trans. Mob. Comput.*, vol. 6, no. 5, 2007, pp. 536–550.

[24] B. Niu, H. V. Zhao, and H. Jiang, "A cooperation stimulation strategy in wireless multicast networks," *IEEE Trans. Signal Process.*, vol. 59, no. 5, 2011, pp. 2355–2369.

[25] D. Das, K. Majumder, and A. Dasgupta, "Selfish Node Detection and Low Cost Data Transmission in MANET using Game Theory," *Procedia Comput. Sci.*, vol. 54, pp. 92–101, 2015.

[26] A. S. Sani and R. Syeda, "Defending Selfish node in MANET using Game Theory Approach," 2016.

[27] Z. Ji, W. Yu, and K. J. R. Liu, "A game theoretical framework for dynamic pricing-based routing in self-organized MANETs," *IEEE J. Sel. Areas Commun.*, vol. 26, no. 7, 2008, pp. 1204–1217.

[28] M. J. Shamani, H. Gharaee, S. Sadri, and F. Rezaei, "Adaptive energy aware cooperation strategy in heterogeneous multi-domain sensor networks," *Procedia Comput. Sci.*, vol. 19, pp. 1047–1052, 2013.

[29] D. E. Charilas, K. D. Georgilakis, and A. D. Panagopoulos, "ICARUS: HybrId inCentive mechAnism for coopeRation stimUlation in ad hoc networkS," *Ad Hoc Networks*, vol. 10, no. 6, 2012, pp. 976–989.

[30] J. J. Jaramillo and R. Srikant, "DARWIN: distributed and adaptive reputation mechanism for wireless ad-hoc networks," *Proc. 13th Annu. ACM Int. Conf. Mob. Comput. Netw. - MobiCom '07*, p. 87, 2007.

[31] T. Hayajneh, G. Almashaqbeh, and S. Ullah, "A Green Approach for Selfish Misbehavior Detection in 802.11-Based Wireless Networks," *Mob. Networks Appl.*, vol. 20, no. 5, 2015, pp. 623–635.

[32] A. Lupia, C. A. Kerrache, and F. De Rango, "TEEM : Trust-based Energy-Efficient Distributed Monitoring for Mobile Ad-hoc Networks," no. 1,2017, pp. 133–135.

Solmaz Nobahary received her B.Sc. in Computer Software Engineering and M.Sc. in Computer System architecture engineering from the Islamic Azad University, Tabriz Branch, Iran, in 2008 and 2011, respectively. Currently, she is Ph.D. student at the Islamic Azad University Science and Research Branch, Tehran, Iran. Under supervision of Dr. Hossein Gharaee Garakani and Dr Ahmad Khademzadeh. Her research interests include Interconnection Network, Fault Tolerant, Ad hoc Network, Sensor Network and Computer Architectures. She authored scientific articles in national and international peer-reviewed conference proceedings and journals.

Hossein Gharaee received B.Sc. degree in electrical engineering from K.N. Toosi University, of Technologhy in 1998, M.Sc. and Ph.D. degree in electrical engineering from Tarbiat Modares University, Tehran, Iran, in 2000 and 2009 respectively. Since 2009, he has been with the Department of Network Technology in ICT Research Institute (ITRC). His research interests include VLSI with emphasis on basic logic circuits for low-voltage low-power applications, DSP, crypto chip and Intrusion detection and prevention systems.

Ahmad Khademzadeh was born in Mashhad, Iran, in 1943. He received the B.Sc. degree in Applied Physics from Ferdowsi University, Mashhad, Iran, in 1969 and the M.Sc. and Ph.D. degrees respectively in Digital Communication and Information Theory and Error Control Coding from the University of Kent, Canterbury, UK. He is currently professor in ICT Research Institute. He is a member of the Iranian Electrical Engineering Conference Permanent Committee. Dr. Khadem Zadeh has received four distinguished national and international awards including Kharazmi International Award, and has been selected as the National outstanding researcher of the Iran Ministry of Information and Communication Technology. His research interests include VLSI Design, Interconnection Network, Fault Tolerant and Computer Architectures.

Amir Masoud Rahmani received his B.S. in computer engineering from Amir Kabir University, Tehran, in 1996, and his M.Sc. in computer engineering from Sharif University of Technology, Tehran, in 1998 and Ph.D. degree in computer engineering from Islamic Azad University, Tehran, in 2005. He has been Post-doctoral researcher at the University of Algarve, Portugal in 2013. He is a professor in the Department of Computer and Mechatronics Engineering at the Islamic Azad University University. He is the author/coauthor of more than

120 publications in technical journals and conferences. He served in the program committees of several national and international conferences. His research interests are in the areas of distributed systems, Grid computing, Cloud computing, Ad hoc and wireless sensor networks and Evolutionary computing.