Computing
https://doi.org/10.1007/s00607-023-01199-1

SPECIAL ISSUE ARTICLE

A solution for resource allocation through complex systems in fog computing for the internet of things

Fatimah Faraji[1] · Amir Javadpour[2,3] · Arun Kumar Sangaiah[4,6] · Hadi Zavieh[5]

Received: 24 November 2022 / Accepted: 28 June 2023

Abstract
To manage the IoT resources to process the massive amount of collected data in complex systems, Allocation of tasks to nodes to increase user satisfaction has been noticed by researchers. In allocating tasks, the necessity of a powerful auxiliary platform is felt. Cloud and fog networks are two suitable options for performing and completing tasks on the Internet of Things. In the cloud network, due to offloading tasks to remote computing centers, problems such as increasing complete time, increasing traffic in the network, and high energy consumption for the Internet of Things network occur. The fog network solves the problem of completing tasks due to reducing the geographical distance between Internet of Things devices and processor nodes. The devices in the fog network cooperate with the Internet of Things network's diverse and variable nodes with time computing, storage, and processing resources. In this heterogeneous environment, the non-cooperation and dishonesty of malicious fog nodes through misrepresentation of shared resources and non-completion of tasks can disrupt the performance of the Internet of Things network. As a result, two fundamental challenges in the heterogeneous fog network include providing resources to allocate tasks and recognizing valid fog nodes. This paper focuses on creating a resource provider network to allocate tasks to authentic fog nodes with appropriate resources. When an Internet of Things node faces a shortage of resources and link prediction, it can form a resource supply network with neighboring nodes to solve the two stated challenges. For this purpose, in the first stage, efforts are made to select the most appropriate options among the neighboring nodes based on specific criteria. In the second step, reliable nodes are identified based on a trust mechanism, including resource trust and entity trust. In the last step, tasks are divided among reliable devices with sufficient resources. The evaluation of the proposed method in the OMNET++ simulator shows that the trust mechanism can lead to the identification of reliable sources and, as a result, reduce the time to complete tasks.

Keywords Internet of things · Complex systems · Quality of service · Allocating tasks · Fog networks · Resource allocation · Link prediction

Extended author information available on the last page of the article

Published online: 24 August 2023

1 Introduction

The main idea of the Internet of Things is the active and pervasive presence of objects around us. Objects that are able to cooperate and communicate with each other to achieve common goals through unique addressing. In fact, IoT has made it possible to connect billions of devices (such as applications, home appliances, cameras, vehicles, smartphones, etc.) through Internet communication to share information instantly [1–3]. The Internet of things has attracted a lot of attention in various industries. This rapid development has led to an increase in the number of networked objects that generate massive data. The generated massive data requires a lot of resources to process, store and also transfer across the network. In fact, the main goal of the Internet of Things is to process data and return results to end users. Meanwhile, managing network resources, including energy storage, processing capacity, and information storage, plays a vital role, which must be done properly [4, 5] and [6], [7]. One of the resource management solutions in the Internet of Things is the allocation of resources with the aim of energy optimization or the so-called green allocation of resources and energy harvesting. Energy harvesting is an approach to searching for renewable energy sources such as the sun, wind, and radio frequencies to supply the energy needed by Internet of Things devices [8–10]. These methods can help improve the quality of service as well as user satisfaction; But on the other hand, due to things 2 like traffic, limited input, early buffer filling, and variable link quality status, they cannot guarantee the quality of service and, as a result, user satisfaction at a high level [7, 11]. In order to increase the quality of service and user satisfaction, the method of offloading tasks is presented, which is done to reduce the amount of processing and storage in the Internet of Things network [12] and [13], [14]. The temporal and spatial dynamics of applications creates variable computing tasks. To delegate these variable tasks, several solutions have been proposed. One of the methods is to use cloud resources. By allocating part or the entire volume of tasks to cloud servers, the consumption of computing resources can be greatly reduced [15]. On the other hand, offloading to the cloud system is done to servers that usually have a great geographical distance from the Internet of Things devices. This long distance between servers and Internet of Things devices has caused problems such as increasing the time to complete the work, imposing additional traffic on the network, and consuming much energy. In this situation, there is a need for methods that eliminate the delay caused by sending the tasks of Internet of Things devices to centralized servers [16, 17] and [18–20]. Fog computing is considered as a promising technology as an extension to cloud computing at the edge of the network. As shown in Fig. 1, this technology provides computing resources, storage, and network services at a close geographic distance to users. Fog computing and edge network approaches reduce latency and increase battery life by eliminating the long geographical distance between IoT devices and cloud servers [6] and [21–23].

1.1 Problem definition

At first, it should be checked whether the tasks considered equivalent to work in this article are performed locally or assigned to the cloud network for execution [8, 24].

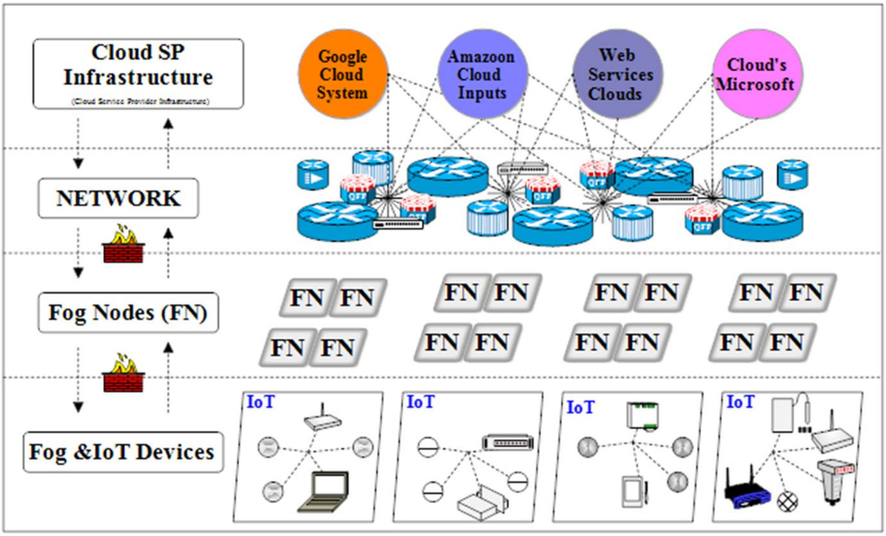


Fig. 1 The position of the fog network in the Internet of Things

This work requires decision-making and optimal planning, and various factors such as time limit, delay, and energy consumption play a role in this decision-making [8, 25]. As a result, the question is raised, what is the best decision to perform the computational task? How to optimally allocate fog resources to improve service quality, including response time, delay, interference, packet loss, and throughput? In this paper, we investigate the problem of allocating tasks to trusted nodes in fog computing. When a fog node suffers from a lack of resources, the tasks of this device should be considered in such a way that they can be allocated to neighboring nodes and computed by them. For this, the first step is to find suitable nodes for offloading. In other words, find a suitable 3 solution based on which the tasks are processed by neighboring fog nodes willing to cooperate, and then the results are delivered to the main device after computation within a certain deadline [15, 26]. Among the fog nodes, there may be nodes that disrupt the network by dropping or changing packets. These nodes are called malicious nodes. The proposed solution should be able to distinguish authentic and reliable fog nodes from malicious ones. For this purpose, a suitable mechanism for calculating trust and high security for resource sharing is required. As a result, both resource allocation and trust issues of fog nodes are considered in the proposed plan.

2 Literature review

In the following section, some basic concepts such as Internet of Things, cloud and fog network and their challenges in Internet of Things will be defined at the beginning. In the following, some of the previous research regarding improving service quality, resource allocation, and allocating tasks in the Internet of Things are reviewed.

2.1 Quality of services in the internet of things

The Internet of Things is a new and growing technology to make people's lives easier. The concept of the Internet of Things is a pervasive presence in which the real (physical), digital and virtual worlds converge and interact, and their communication is either wireless or wireless and through unique addresses [27]. Figure 2 shows an application of the Internet of Things.

Improving the quality of service is done on the one hand by increasing the processing speed and battery life and on the other hand by reducing the computational and communication delay [14]. These changes lead to user satisfaction, which has become one of the main criteria for improving service quality. Silva et al. [8] have divided IoT technologies into three groups:

1. Technologies that obtain information in objects.
2. Technologies that process information in objects.
3. Technologies that provide security and privacy.

In order to improve the quality of services, all three groups should be examined as important fields in the Internet of Things. In the following, we examine the service quality criteria in these groups [17, 18, 28] and [29].

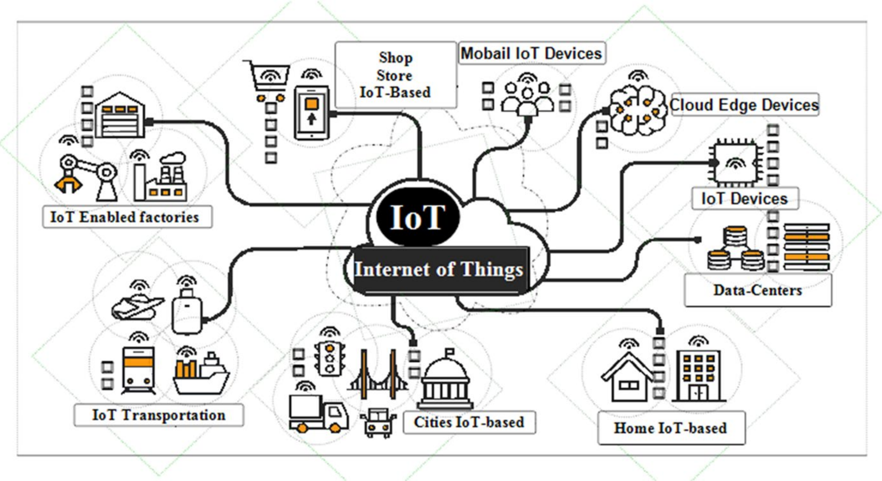


Fig. 2 Application of the Internet of Things

2.2 Research background

Since the distribution and offloading of tasks in the Internet of Things is usually done with the goals of (1) reducing the time to complete the work, (2) reducing the energy consumption, (3) increasing the battery life, (4) increasing the user satisfaction,

The article [1] has focused most on optimizing energy consumption. They have used fog network to increase computing capability and also reduce energy. With their proposed method, the battery life of the devices increases, which improves handover. They have used an online dynamic task scheduling called Lyapanov optimization-based offloading to investigate the energy consumed and the execution delay. Mao et al. used an algorithm called LODCO to reduce computational delays and cost, making handover decisions according to processor frequencies. The advantage of the proposed algorithm is that the decision taken is based only on the system's current state without the need for distributed information. The results show an improvement of energy consumption and the reduction of packet delay and failure [1]. By harvesting energy, Zhang and his colleagues have taken a step toward improving the quality of service and extending battery life. In this article, the energy needed to charge devices is obtained from the surrounding environment. They have used reinforcement learning to delegate computation and Markov decision process to select the appropriate virtual machine. They have tried to minimize the energy consumption by considering the queue stability, buffer size and battery energy stabilization. The results show a compromise between energy consumption and delay reduction [30]. The presence of dynamic traffic among the different locations of IoT users' spatial and temporal dynamics. Fan and his colleagues have spoken of the placement of DBSs in the network's busy locations. The volume of traffic should be dedicated to BS and DBS. In their article, they have two basic challenges (1) location of DBS (2) assigning network traffic loads. Drunks in mobile networks are used to improve the quality of service. These drones are dynamic in the network areas to deliver users' tasks to MBS, which greatly affects network communication delay.

One of the fog network's goals is to reduce load and computational and communication pressure on the cloud network and the network center. In this regard, there have been few reviews of the division of tasks between neighboring nodes and helping nodes in the edge devices. In this regard, we are discussing and presenting a solution.

3 Proposed method

3.1 Proposed method

In fog networks, lack of resources and a safe environment are major challenges. This article focuses on creating a network to allocate tasks among valid fog nodes with limited resources. In this section, the two challenges are attempted by examining and improving the resource provider network. In the scenario presented in the article [31] when the fog node is facing a shortage of resources for a limited task, the neighboring fog nodes form a resource provider network. These nodes are called provider nodes. The node may allocate its computational task to other adjacent devices. This provider network is formed by a node with a resource shortage with fog devices that agree to lend parts of their computing resources to the node. Initially, the fog node sends a message to the nodes that exist in a few or more steps. The nodes that are willing to cooperate and are able to present some of their resources to the host node announce their readiness to form the network by sending a message. Jin et al. In the proposed network, the link quality and connection only examine the nodes. In this article, in the first step, by examining the sent messages, it is attempted to select more suitable nodes in terms of resources. This suitability is evaluated based on the time, memory and processor criteria of the node. Around the fog node, there are different fog nodes with different sources. The host node can choose the most appropriate options as the resource provider nodes among these options based on the above criteria. It is attempted to create a reliable resource provider network with these criteria. The next step is to check the trust of these nodes. By checking their trusts, it tries to understand the reliable nodes. After these reviews, the division of tasks among the reliable devices with sufficient resources. The proposal is based on a two-step trust checking mechanism. In this mechanism, first, examines the sent reports and then examines the trust of the node. In the proposal, using the trustworthiness mechanism, the host node forms the provider network by selecting the best reports and nodes (Figs. 3, 4).

shows that this design consists of three main phases:

- *Phase 1:* formation of resource provider network.
- *Phase 2:* division of tasks.
- *Phase 3:* network update based on time period.

3.2 The first phase: network formation

The selection of nodes to participate in forming a resource provider network is based on a two-step trust assessment. This confidence is composed of two stages.

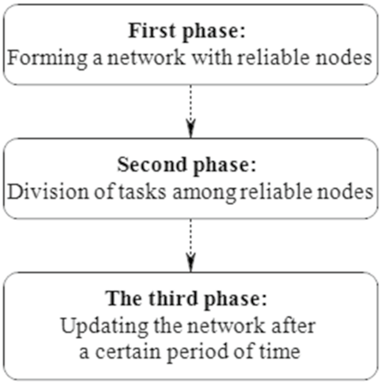

First phase:
Forming a network with reliable nodes
Second phase:
Division of tasks among reliable nodes
The third phase:
Updating the network after
a certain period of time

Fig. 3 Different phases of network

3.2.1 Cooperation motivation and trust measurement

Provider nodes consider the trust of the host node to cooperate. If the trust of the host node is higher than a threshold and the node has the right resources and conditions, they agree to help the host node. Considering the trust level of the host node is a way to encourage the cooperation of a node with other nodes. The more nodes a node interacts with and the better it performs the tasks sent by them, the more trusted the node is. More cooperation with nodes leads to higher recognition and trust of the node. At the time of resource shortage, if the node is reliable, more nodes will cooperate (Fig. 5).

- *Source trust assessment:* selection of nodes with appropriate resources based on the factors specified among nodes that have agreed by sending reports.
- *Entry trust:* the entity of the nodes selected by the specified criteria is examined.

The symbols used in the proposed method are described in Table 1.

The two-step trust assessment process is fully explained in the next two sections.

3.3 Reliability of sources

This is a confidential assessment to determine the node's status in terms of their resources and time. To form a resource provider network, more suitable nodes are selected among other nodes by examining the following factors for each fog node. In this design, there are three factors for each node and its resources in them:

- *Time factor*—in IoT applications, the status of node's resources may change over time, the time factor needs to be considered as one of the influential factors; This is an important issue because of reducing or increasing delay and the quality of service. This factor determines when the report shows the node status.
- *Memory factor*—in IoT applications, the status of nodes' resources may change over time, the memory factor needs to be considered as one of the most effective factors. The amount of memory plays an important role in processing some tasks. Since the device's memory varies over time, this criterion plays an important role in selecting nodes. This factor determines whether the nodes are appropriate for the tasks in question.
- *CPU factor*—since the features and characteristics of each node or device differ from other devices, this feature is considered the third criterion. Here, it shows the same amount of a freedom processor that can be provided to the host node.

We divide the evaluation properties into two beneficial and cost characteristics. Useful features such as the amount of memory available, the processor speed, and the cost features in our design include energy consumption, time, and the host knot distance with the supply node. The closer the usefulness is to our maximum

Table 1 Symbols used in the proposed method

Explanation	Symbol
Report i	R_i
k−th reporting node	N_k
Node location	t
The number of Indirect Trust	N_t
The processor agent of node i in the sent report	$\mu_p(R_i)$
The time agent of node i in the sent report	$\mu_t(R_i)$
The memory agent of node i in the sent report	$\mu_m(R_i)$
Weight of report R_i provided by N_k	$W_k(R_i)$
The number of sent tasks	JS
The number of received tasks	JR
The number of dropped tasks	JDR
Node profile	IPT
The amount of direct trust	DT
The amount of indirect trust	IDT
Acceptable dropped tasks	E
Quality of done task	Q
The number of received direct trusts from neighbors	N
The total number of trustworthy nodes	N
Calculated trust base on latest interactions	DT_{new}

value, and the closer the cost feature is to the minimum value, the more convenient and optimal choice. The maximum and minimum value is considered the right option, and the less the other options are with the graceful option, the more desirable they are. Due to these features, the graceful nodes are selected by comparing the graceful nodes among the candidate nodes. The graceful or maximum model is a model that performs the maximum processor and memory and is close to the host knot. For the provider nodes to cooperate, they consider the host's trust. If the confidence of the host node is higher than a threshold and the node has the right resources and conditions, they agree to help the host node. Since important criteria in devices or nodes, memory (M), time (T), and CPU (P) are considered influential factors, resource provider nodes are required to send this information to the host node. As a result, reports should contain processor value, memory capacity, and node time. The report must be evaluated when a node announces its agreement by sending a report about its situation. By examining the received reports, the appropriateness or not of each node is determined. This study shows whether the node is graceful for forming a provider network. These reports contain a message that actually reports on time, processor and memory. Here is the report of the node of i with R_i. The R_i Report CPU agent is specified with $\mu_p(R_i)$. Report time agent with $\mu_t(R_i)$ and the memory agent with $\mu_m(R_i)$. These reports weigh these three factors. This weightlifting determines the amount of suitability of each node. So the first step to examining the tale of the nodes is to weigh the reports of their submitted.

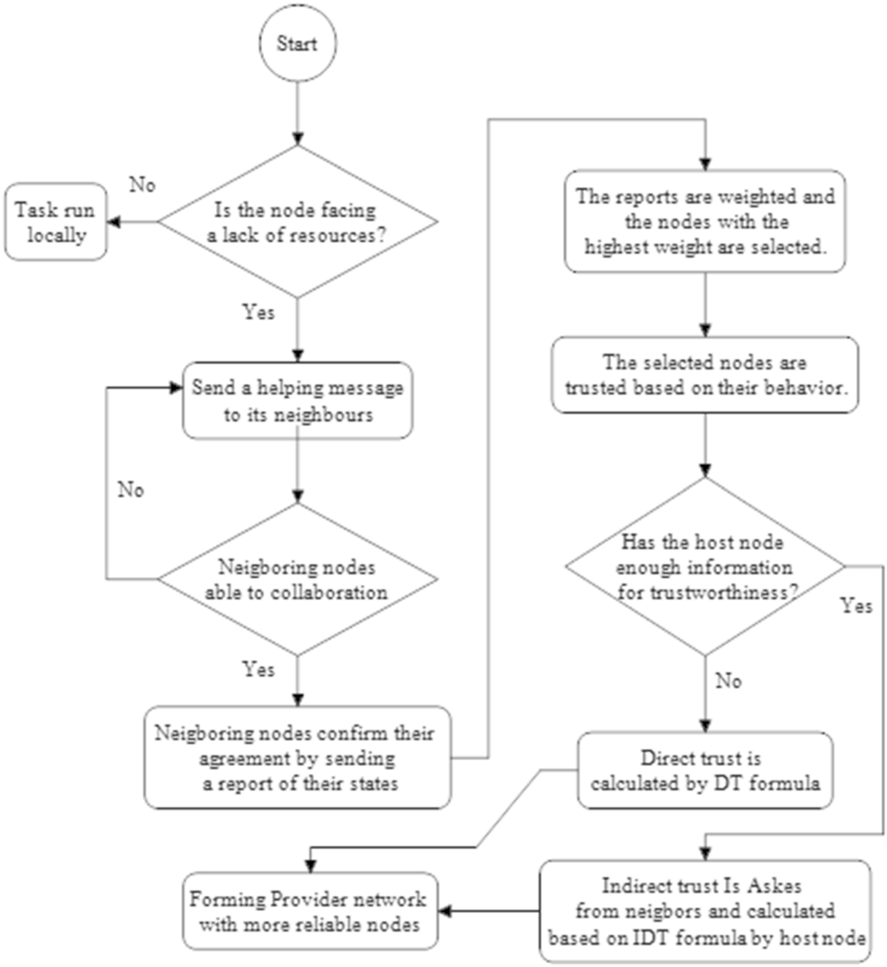
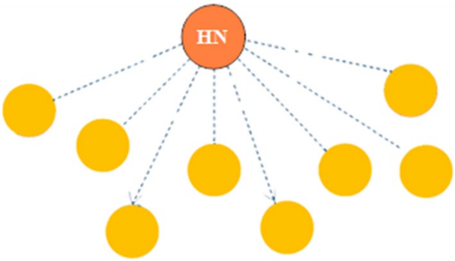

Start
Is the node facing a lack of resources?
No
Task run locally
Yes
Send a helping message to its neighbours
Neigboring nodes able to collaboration
No
Yes
Neigboring nodes confirm their agreement by sending a report of their states
The reports are weighted and the nodes with the highest weight are selected.
The selected nodes are trusted based on their behavior.
Has the host node enough information for trustworthiness?
Yes
No
Direct trust is calculated by DT formula
Indirect trust Is Askes from neigbors and calculated based on IDT formula by host node
Forming Provider network with more reliable nodes

Fig. 4 Flowchart of resource provider network formation

Fig. 5 Sending a help request from the node



3.3.1 Report weighting

At this point, by weighing reports, try to trust their resources. Any report from the node about resources should be examined. By allocating a weight to that report, determine how appropriate this report or node is. It is not certain whether a report is reliable or not. Rather, by providing a random number to that report, it is determined to what extent it is reliable. It means that this report is reliable if the existence of the existence and the high value of $W_k(R_i)$ is reliable and otherwise, it is not necessarily unreliable. In Internet applications, sensors have different accuracy; each entity may provide a more accurate report according to its type or order; That will be the basis for uncertainty in the data [7]. The purpose of this section is to choose the most appropriate nodes among the candidate nodes to perform tasks.

The following formula shows the weight resulting from the aggregation of the field factors for the report of the R_i provided by the N_k node.

$$W_k(R_i) = \alpha(\mu_p(R_i)) + \beta(\mu_t(R_i)) + \gamma(\mu_m(R_i)) \tag{1}$$

It is obvious that $\alpha + \beta + \gamma = 1$, which means that due to the higher importance of the processor than background factors, the equation $\alpha \geq \beta + \gamma$ must be established. After weighing the reports, competitive sorting is used as a selector method to select the nodes with the highest weight. Weighted nodes are entered into the tournament match. Nodes with higher weights win this match. So far the nodes with the best resources and status are selected. These winning nodes enter the next stage, the entity trust measurement stage.

3.4 Entities trust measurement

entity trust measurement stage. H. Entities trust measurement Since there may be a malicious node among the valid nodes that sends false and hostile reports about the nodes and causes disruption to the network and the performance of tasks, the trustworthiness of entities becomes necessary [32]. To calculate the trust, the trust of the fog node itself is considered as the entity's trust. The purpose of entity trust calculation is to distinguish malicious nodes from well-behaved nodes. The purpose of entity reliability calculation is to determine the correctness or incorrectness of an entity about which reports have been received and will participate in the provider network; to be determined. Here, the amount of trust assigned to the node varies between −1 and +1. The closer this number is to +1, the more reliable the node is. In creating this network, we try to choose more reliable nodes. This trust assessment consists of two parts: direct and indirect trust assessment:

3.4.1 Direct trust measurement

This type of trust measurement is used when the host node has information about the provider node. For each node, a profile is created. This profile, named Packet Traffic Information or IPT is placed in each node separately to keep the node's behavior and essential information [33]. Since tasks are considered as transferable units. Trust measurement is done based on these tasks. In fact, their trust level is measured by monitoring the behavior of nodes with these tasks. This trust metric determines whether the node is trustworthy or malicious. When a node interacts with other nodes, they are evaluated based on parameters. The following parameters are included in this profile:

- JS: number of tasks sent by the sender node.
- JR: number of tasks received by the receiver node.
- JDR: rate of dropped tasks between sender and receiver nodes.

Direct validation is done by evaluating these parameters and using the following equation.

$$DT_{i,j} = \frac{JR * Q}{JS} - \frac{JDR * E}{JS} \tag{2}$$

In the formula (3–2), the value of the task drop rate is obtained from the Eq. (3)–(3)

$$JDR = JS - JR \tag{3}$$

The value of E is the number of normal acceptable tasks based on network problems such as noise. If the packet drop rate is higher than E, it indicates that the node is malicious. The value of Q is the quality of received tasks. Each node gives a score to the quality of the work done. This quality varies between 0 and 1. The direct trust value is calculated for each node and stored in the node profile using these parameters. This trust value is used in indirect trust measurement.

3.4.2 Indirect trust measurement

The trust of the unknown node is calculated using the average of the direct trusts that the neighboring nodes obtain from the desired node. When a node does not have information about another node, it sends a message to its neighboring fog nodes, asking them to perform a direct evaluation and send the result to the nodes with sufficient information. Based on the obtained trusts, the overall trust is obtained with the average.

$$IDT = \frac{DT_i + DT_j + \cdots + DT_n}{n} \tag{4}$$

In the formula (3)–(4), n represents the number of direct trust obtained about the node. The obtained values are compared with a threshold. As a result, the nodes with the most trust are selected to participate in the resource provider network. Also, in order to give a chance to the nodes that do not have enough trust, a node is randomly selected among them. The host node sends a confirmation message to the trusted nodes in this step. This confirmation is for resource reservation; and finally, the host node creates a resource provider network with the selected nodes, and tasks are divided between these nodes. On the other hand, due to an adjustable trust threshold, it is possible to change the degree of strictness towards the trust of nodes.

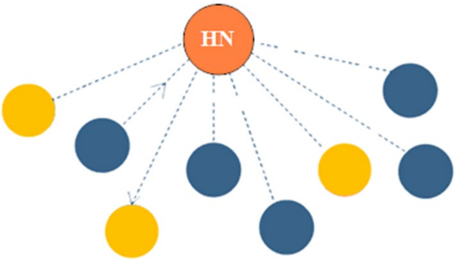


Fig. 6 Declaration of consent to help the host node

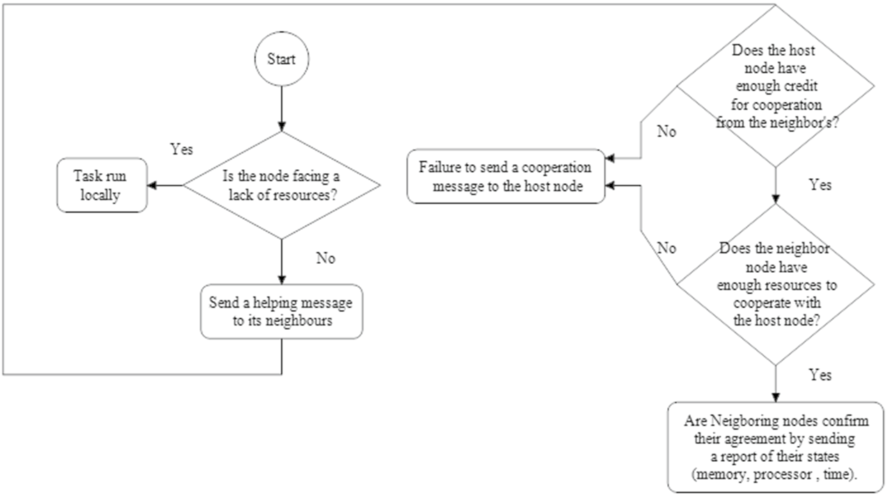

Start
Is the node facing a lack of resources?
Yes
Task run locally
No
Send a helping message to its neighbours
Does the host node have enough credit for cooperation from the neighbor's?
No
Failure to send a cooperation message to the host node
Yes
Does the neighbor node have enough resources to cooperate with the host node?
No
Yes
Are Neigboring nodes confirm their agreement by sending a report of their states (memory, processor , time).

Fig. 7 Flowchart of the first and second steps of the resource supplier network

Depending on the type and content of the sent information, the host node can adjust the network strictness. In Fig. 6, the workflow of the formation phase of the resource supplier network is explained:

Below are the general steps of interaction between nodes:

- *The first step*—Figs. 7 and 8 shows that a node that faces a lack of resources or the host node sends a help request to its neighboring nodes. Neighboring nodes are shown in yellow.
- *The second step*—nodes that are willing to cooperate and can provide part of their resources to the host node, by sending a report of their status, declare their readiness to form this network. Figure 9 also explains the flow chart of the first and second step of forming the resource supplier network.
- *The third step*—The host node performs a two-step trust measurement on these nodes. The host node sends a message to the valid nodes to announce their authentication. They are obliged to reserve these resources. Figure 10 shows valid nodes in green and malicious nodes in black. Also, in figure, the flow chart

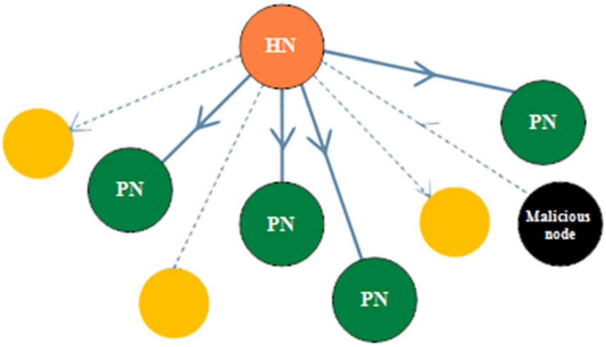


Fig. 8 Detection of malicious and valid nodes

of the first and second step of forming the resource supplier network is shown (Fig. 11).

The entity's trustworthiness pseudo code is presented in Table 2.

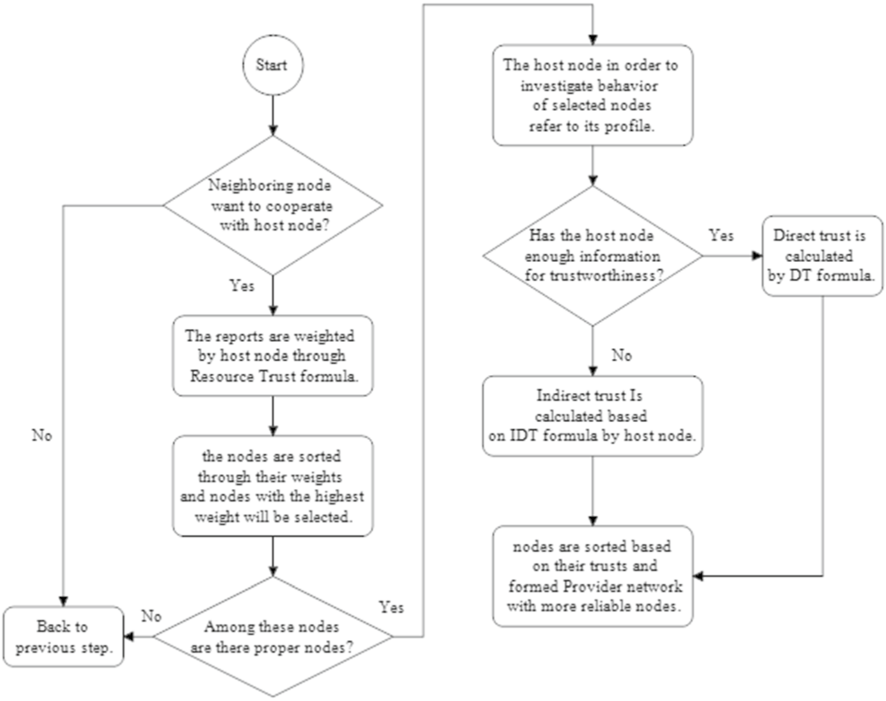

Start
Neighboring node want to cooperate with host node?
Yes
No
The reports are weighted by host node through Resource Trust formula.
the nodes are sorted through their weights and nodes with the highest weight will be selected.
Among these nodes are there proper nodes?
No
Yes
Back to previous step.
The host node in order to investigate behavior of selected nodes refer to its profile.
Has the host node enough information for trustworthiness?
Yes
Direct trust is calculated by DT formula.
No
Indirect trust Is calculated based on IDT formula by host node.
nodes are sorted based on their trusts and formed Provider network with more reliable nodes.

Fig. 9 Flowchart of the third step of the resource supplier network

Table 2 Entity trustworthiness pseudo code

```
Calculate trust for neighboring nodes
Set the parameters $JS, JR, JDR, IPT, N, TR[I, j], B$
Initialize Profile $IPT$ using $JS$, $JR$, $JDR$ for each node
    For $i = 1 to N$ do
    Calculate $JDR$ using Eq. (3-3) for each neighboring nodes
        If Host Node. $JS[i] <> 0$ then
          Calculate Direct Trust using Eqs. (3)–(2)
        Else
          For $i = 1 to N$ do
            Send a message to node[$i$] and ask about unknown node
            If $js[i] <> 0$ then
                Calculate Direct Trust [unknown node] using Eqs. (3)–(2)
              Send the result for host node
            End if
          End for
        End if
    Calculate Indirect Trust [unknown node] using resided results and Eqs. (3)–(4)
    $TR[i] =$ trust
    Sort TR
    Select $B$ from $TR$
    End for
```

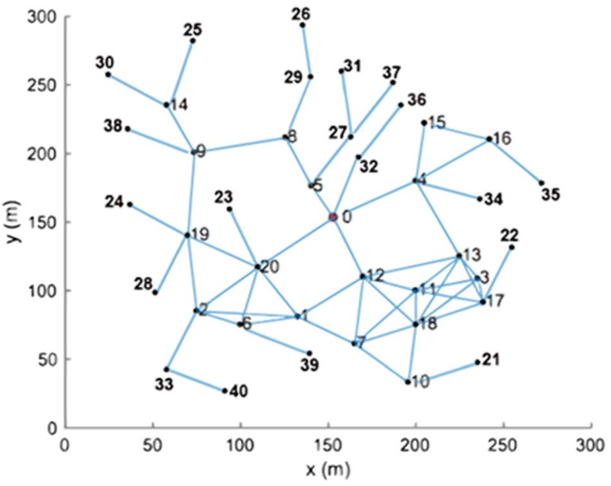


Fig. 10 An example of network topology

3.5 Second phase: division of duties

After forming the network, it is time to divide the tasks among the supplier nodes. This assignment of tasks is done based on the First Fit algorithm. At first, before assigning work, the values of Q, JR, JS, E are set to zero to be replaced with new information at the end of this step. This allocation is done based on processor units and a task allocation algorithm such as first fit. Finally, the Q, JR, JS values are updated based on recent interactions.

3.6 The third phase: network update

Since the status of the nodes may be different, or a node may be disconnected from the network, or the level of trust of the nodes may change, an update is needed after a certain period. In this network, tasks are assigned and executed in seconds. Here, after a certain period, the trust of the nodes and as a result of the network is updated several times. The number of updates within this specified interval is important. Here, step 3 and then step 4 are repeated.

The validation of this step is done with the following formula:

$$DT_{(new)i,j} = \Omega(DT_{old}) + (1 - \Omega)\left(\frac{JR * Q}{JS} - \frac{JDR * E}{JS}\right) \tag{5}$$

where Ω is the weight for which the relation $\Omega + (1 - \Omega) = 1$ must be established. The values of JR, Q, JS, JD are new information that are included in the profile of the nodes during the assignment of tasks. As we have seen in this section, the resource provider network was presented after modeling the parameters affecting the problem of resource allocation to the fog node. In this network, on the one hand, by measuring the reliability of the resources of the neighboring nodes, an attempt was made to divide the tasks among the nodes with appropriate resources and status. On the other hand, by measuring the reliability of nodes, we try to identify malicious nodes and create a safe environment for allocating tasks. Both solutions' formulas, methods and parameters were modeled and the pseudo code related to each was presented.

4 Simulations and results

In the previous section, the proposed method was discussed in the field of the division of tasks in the fog environment and a method was presented to create a secure environment for the division of tasks between fog devices with limited resources in the fog environment. The proposed method examines two types of trust, namely entity trust and resource trustworthiness. In this section, the effectiveness of the proposed method is evaluated and analyzed by simulating a resource supply network in a fog environment. To achieve this goal, the OMNeT++ simulator has been used

Table 3 Evaluation parameters related to the network

Parameter	Amount
The number of neighboring fog nodes	20
The number of host node	1
Size of input tasks	1.5 MB
Size of output tasks	25 KB
Edge computational delay	1 S
Fog network delay	2–4 M
Time frame for each task	10–60 S
Update interval (Tw)	5 M

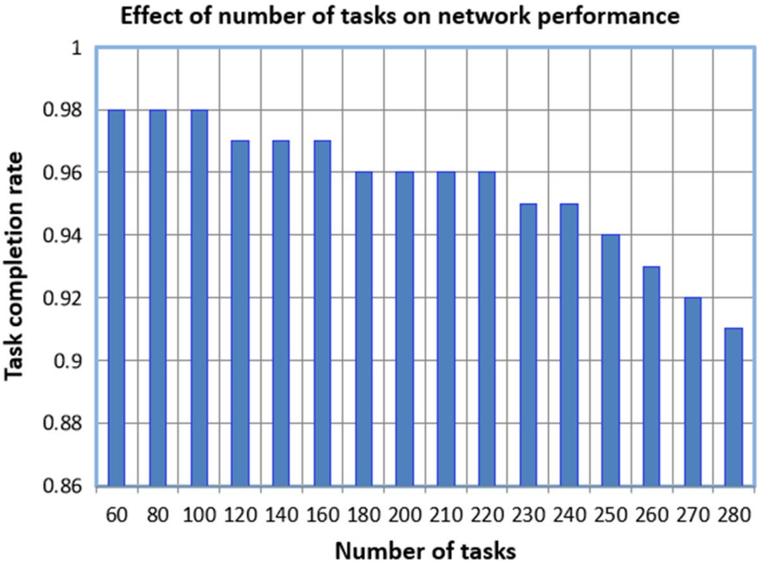


Fig. 11 The effect of the number of tasks on the task completion rate

to examine the strengths and weaknesses of the proposed method. In addition, the phases of the proposed method are compared with the task division method of the article [7] which was reviewed in the second section.

4.1 Network topology

The network topology is an undirected graph. Neighboring fog nodes are directly connected and there is a path from each fog node to another fog node. The way fog nodes are connected is based on how their neighborhood is defined. In order to determine the parameters of the network, the article [7] is used according to the problem space. Other specified parameters have been determined according to the limitations of the problem and the nature of the problem and are shown in Table 2

The fog nodes are randomly distributed in the problem space with an area of 300 × 300. Determining the neighborhood has also been done randomly.

Figure 12 is an example of a resource provider network consisting of a host node and 40 neighbor nodes. The numbers next to the nodes indicate the ID of each node. We assume that fog devices use an 802.1534 radio interface under fixed transmission power. To simulate wireless links, a shadowing model with path loss with a loss power of 0.3 with a reference loss of 46.67777 dB with Gaussian noise N (0 and 32), which is related to the urban area, has been used. We assume that the computational workload is low and the data volume is relatively large. When these tasks are sent to a remote cloud server, bandwidth and network latency increases. The moving speed of mobile nodes is 2.2 m/s, which is the same as human walking speed. The distance between the nodes is up to 60 m. We perform simulation runs on each experiment and report the mean and standard deviation values. 20 simulations are performed for each experiment (Table 3).

4.2 Simulation environment

To investigate the task division method through resource provider network, hardware with Intel Core i5 series processor with 16GB main memory and 256GB disk is used. The software environment for simulating these tests is OMNeT++ version 5.6.1, which is run on Windows 10 operating system.

4.3 Evaluation scenarios and result

In this section, the effect of different conditions and parameters on the efficiency of the proposed algorithm has been investigated. The proposed algorithm is based on two-step reliability measurement. Network efficiency is checked in terms of task

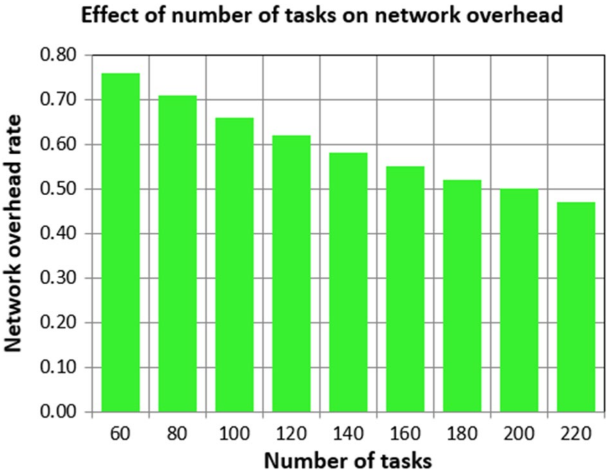


Fig. 12 The effect of the number of tasks on the network overhead

completion rate and network overhead. The security software has checked the optimal result obtained from the implementation of the simulation. The completion rate is measured by counting the number of tasks that have been completed within a limited time limit and is used to measure the network overhead of control messages. To achieve the feasibility of the test results, we have designed the tests so that the performance of our algorithm is checked under various environmental factors such as node status, changing the number of tasks, the effect of two-step trust measurement, and the task deadline. When we compare with the old methods, we see that our algorithm effectively reacts to the lack of resources and instability due to the dynamic nature of the network, and this is due to the creation of the resource supply network using fog nodes.

4.3.1 The effect of the number of tasks on network performance

To investigate how the number of tasks affects network performance. Task completion rate and network overhead are measured concerning total tasks.

The effect of the number of tasks on the rate of work completion To perform this simulation, we increase the number of input tasks to the network. First, as the number of tasks increases, the task completion rate also increases, but when the number of tasks increases to a large amount, the task completion rate decreases. As shown in Fig. 13, our provider network offers up to 210 tasks with good performance. Up to this number, the task completion rate is above 90 completed tasks decreases, and this is due to resource limitations and network overhead.

The effect of the number of tasks on network overhead In this implementation, the amount of network overhead, defined based on the number of control packets to the total number of packets, is checked. Although the number of tasks entered into the network increases, the number of messages to send tasks and receive results increases, but the control messages for two-step trust measurement are only being

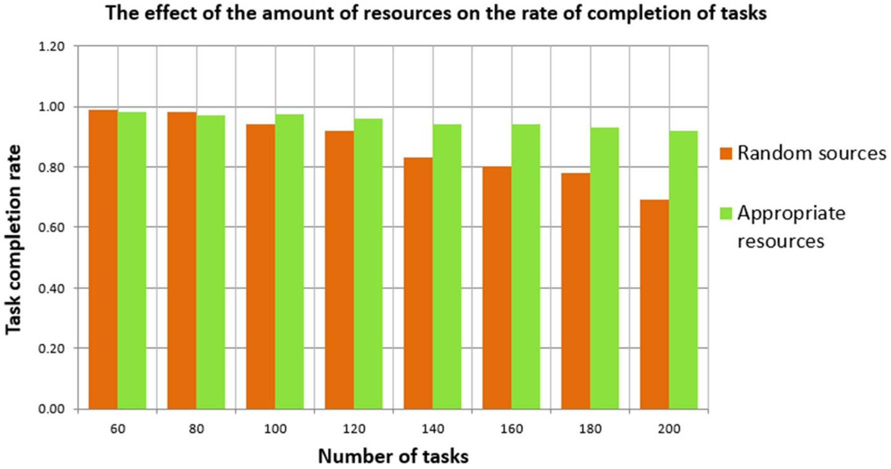


Fig. 13 The effect of resource reliability on network performance

transmitted at the beginning of the network regardless of the number of tasks. In this case, once the nodes undergo a two-step trust assessment, the host node starts dividing tasks. Trust measurement depends on nodes, not tasks. In some algorithms, the trust measurement is performed during the transfer of the task, and with the increase in the number of tasks, the network control messages increase, which leads to an increase in the network overhead. As shown in Fig. 14, the network overhead decreases as the number of tasks increases. According to this evaluation, the proposed plan is suitable for environments where the number of tasks is almost high.

4.3.2 The effect of two-step trust measurement on network performance

This part is for checking phase 1 of the algorithm. As it was investigated, the selection criteria of the node to form the resource provider network are the amount of resources of the nodes and the trust of the entities themselves; that is, nodes that have the most computing and storage resources and are reliable are used to form this network.

The effect of resource reliability on network performance A case where nodes were selected with random resources was investigated to investigate the effect of fog node resources on network performance. While in the second case, among the neighboring nodes, the most appropriate nodes were weighted and selected according to the criteria introduced in Sect. 3, and the resource supply network was formed from nodes with high weights. As seen in Fig. 15, in both cases, we changed the number of tasks between 60 and 200 and checked the network performance. By comparing these modes, it was found that the network with appropriate resources performs better; that is, the task completion rate increases. In case the reliability factor of the sources is not considered in a similar article. In this case, the resources are randomly selected, and the probability of resource shortage increases, and as a result, fewer tasks are completed within

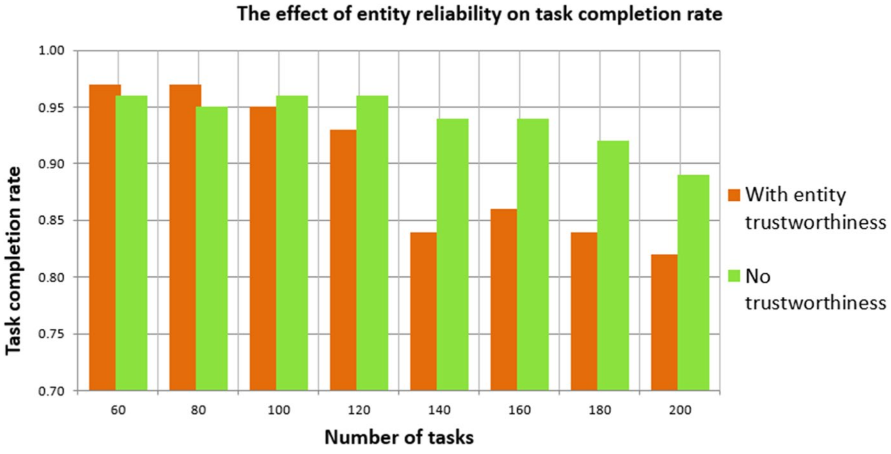


Fig. 14 The effect of entity trust measurement on network performance

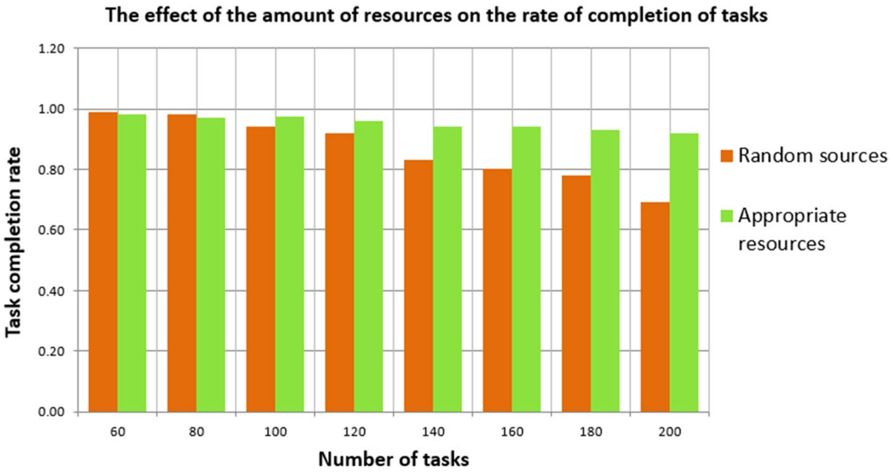


Fig. 15 The effect of resource reliability on network performance

the specified time limit. This review shows the feature of our proposed network compared to resource allocation algorithms that do not care about the number of resources.

The effect of entity reliability on network performance This part investigates the effect of entity trust measurement on network performance. Here, a case was investigated where a resource provider network was formed without checking the trust level of nodes. In this network, the drop rate depends on the increase and finally, the rate of completion of tasks decreases. In the next test, all nodes are selected based on their trust level. On the one hand, because reliable nodes are selected, the amount of dropped packets is minimized, but on the other hand, because the number of control messages increases, the completion rate decreases somewhat. Figure 16 shows the difference between network performance with entity trust measurement and without entity trust measurement. Although the completion rate is somewhat reduced due to the overhead, it is suitable for environments where trust is very important. In this network, the secure environment with reliable nodes has priority over the completion rate.

When nodes with higher trust participate in the formation of the provider network, the task completion rate increases until the number of tasks reaches 130; but as the threshold increases, the number of providing nodes decreases. In this case, with the increase in the number of tasks, the network faces a lack of resources and the task completion rate decreases. As can be seen in Fig. 17, when we increase the trust threshold because only nodes with a trust higher than 0.7 have participated, the number of nodes decreases and as a result the amount of resources decreases and the network is formed with only reliable nodes. In this case, when the number of tasks increases, the task completion rate decreases due to lack of resources.

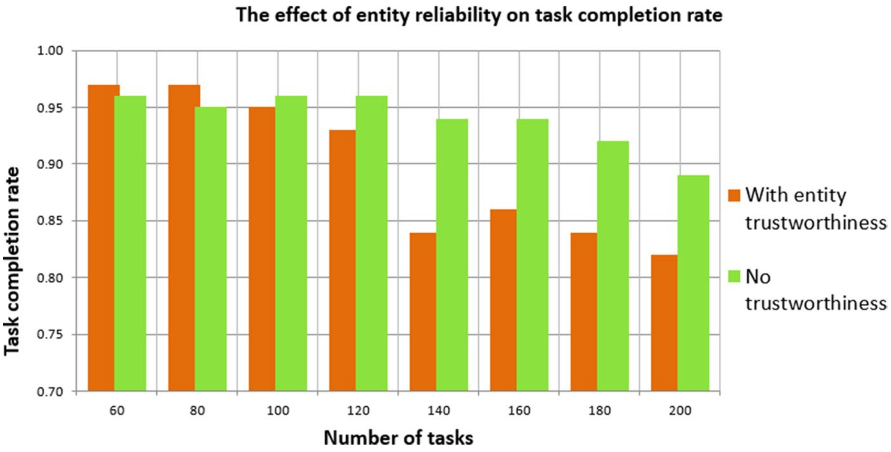


Fig. 16 The effect of entity trust measurement on network performance

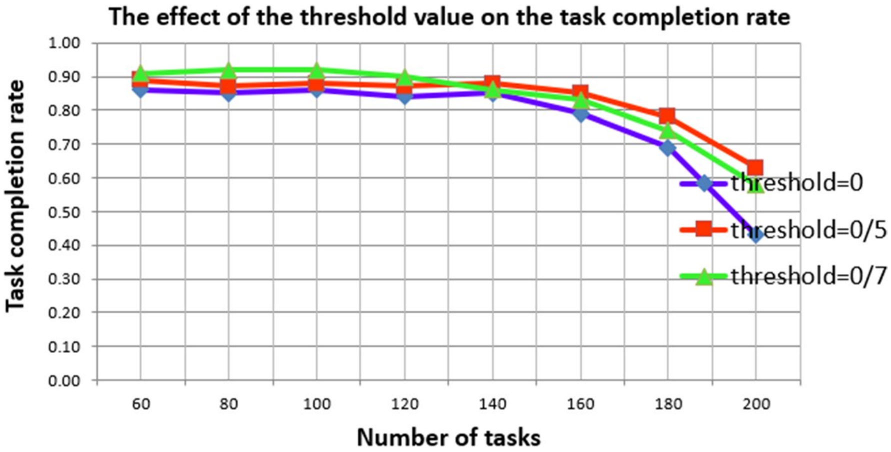


Fig. 17 The effect of entity reliability threshold on task completion rate

4.3.3 The effect of the number of unknown nodes on network performance

in this part, the effect of the number of unknown nodes, the nodes about which the host node does not have any information, on the network is investigated. Here, the number of unknown nodes is counted after each network formation. Finally, their effect on the task completion rate is examined. This is done to find a suitable operating environment and shows that the resource provider network should consider the influence of fixed and mobile nodes. Because as the number of mobile nodes increases, the number of unknown nodes increases. Since information about unknown nodes is not available to the host node, indirect trust measurement is used. The more indirect trust measurement is done, as a result, more control overhead and more time is spent on trust measurement, and finally, the task completion rate decreases. In this case, the more the static environment and nodes are fixed, the less the number of unknown nodes. In this case, the need for indirect reliability measurement is less. Finally, the task completion rate increases. In this case, finding a value with which the number of balanced trust measurements and the task completion rate reach the highest value is necessary. More than this checking amount creates an additional load on the network, which reduces the network's efficiency. The best case happens when maximum nodes are trusted through direct trust. The worst case is when the number of unknown nodes is large. In this case, the network overhead increases due to the increase in the number of messages sent and the number of calculations.

4.3.4 Deadline for doing the work

While we considered the intensity of tasks to be constant and equal to 180, we investigated the effect of the task deadline on the task completion rate. When the time limit is changed from 10 to 60 s, on the one hand, the algorithm has enough time to divide the tasks and receive the results, and on the other hand, the nodes have the right time to perform the tasks. In this case, the nodes complete the tasks more successfully. Here, we consider a confidence threshold of 0.5, which allows more fog devices to participate. By increasing the time limit and reducing the confidence threshold, the completion of the work becomes faster, and the overhead is less. This means our offloading scheme achieves a good balance between time and resources, resulting in a high completion rate. This section discusses the general explanation of the test space and the parameters used in the simulation of the resource provider network using the proposed method. The results of using the two-step reliability method were examined and we examined its results on the network's performance. Then our proposed scheme was compared with the algorithm of the similar article. By fully explaining the assumptions, evaluation criteria and simulation parameters, the results of these simulations were investigated and analyzed in order to evaluate the efficiency of the presented trust calculation method and the proposed method. According to the obtained results and their analysis, the proposed method of division of tasks despite the trustworthiness of resources and entities, at first the resources of the nodes were evaluated and then the malicious nodes were recognized from the reliable nodes. These trust tests divided tasks into a safe environment with suitable resources in terms of location, memory and processor. The next section will discuss the summation and review of future works [27, 30, 31, 32, 33].

5 Conclusion

In this research, a trust calculation method was presented for use in the Internet of Things. For this purpose, the Internet of Things network, fog and cloud computing network and the challenges of these types of networks were first investigated. Even with sufficient computing and storage resources, the cloud network imposes a large delay on the Internet of Things network. This delay is due to the long distance between this network and the Internet of Things devices. To solve these challenges, the fog network is presented as an effective approach. On the other hand, problems such as lack of resources in the fog network have led to the introduction of methods such as assignment and division of tasks between fog nodes. Task allocation methods increase the quality of services in the Internet of Things. Therefore, in this article, an attempt was made to take a step towards solving the lack of resources in the fog network by presenting the design of a network providing resources. In this article, a task division method was presented in a secure environment for use in the Internet of Things, which, in addition to calculating trust for entities in an application, also calculates the reliability of their resources. In the proposed method, two-step trust calculation steps were presented in the first step. The two-step trust calculation consists of two main parts: resource trust measurement and entity trust calculation. The calculation of the trustworthiness of sources consists of the main part of weighting reports. To calculate the trust, the resource nodes that are cooperating with the host node are evaluated based on three specified criteria. At this stage, the state of the nodes is weighted based on the weighted aggregation function.

Funding Open access funding provided by FCT|FCCN (b-on).

Declarations

Conflict of interest No conflict exists: (Fatimah Faraji) declares that he has no conflict of interest. (Amir Javadpour) declares that he has no conflict of interest. (Arun Kumar Sangaiah) declares that he has no conflict of interest. (Hadi Zavieh) declares that he has no conflict of interest.

References

1. Mao Y, Zhang J, Letaief KB (2016) Dynamic computation offloading for mobile-edge computing with energy harvesting devices. IEEE J Sel Areas Commun 34(12):3590–3605
2. Pirozmand P, Javadpour A, Nazarian H, Pinto P, Mirkamali S, Ja'fari F (2022) GSAGA: A hybrid algorithm for task scheduling in cloud infrastructure. J Supercomput 78(15):17423–17449
3. Cheng X, Hu Y, Varga L (2022) 5G network deployment and the associated energy consumption in the UK: a complex systems' exploration. Technol Forecast Soc Change 180:121672
4. Javadpour A (2020) Providing a way to create balance between reliability and delays in SDN networks by using the appropriate placement of controllers. Wireless Personal Communications 110:1057–1071
5. Javadpour A., Wang G, Rezaei S (2020) Resource management in a peer to peer cloud network for IoT. Wireless Personal Communications 115:2471–2488
6. Javadpour A., Wang G (2022) cTMvSDN: improving resource management using combination of Markov-process and TDMA in software-defined networking. The Journal of Supercomputing 78:3477–3499

7. Li S, Zhai D, Du P, Han T (2019) Energy-efficient task offloading, load balancing, and resource allocation in mobile edge computing enabled IoT networks. Sci China Inf Sci 62(2):1–3
8. Silva BN, Khan M, Han K (2018) Internet of things: a comprehensive review of enabling technologies, architecture, and challenges. IETE Tech Rev 35(2):205–220
9. Mirmohseni SM, Tang C, Javadpour A (2020) Using Markov learning utilization model for resource allocation in cloud of thing network. Wireless Personal Communications 115:653–677
10. Javadpour A, Wang G, Rezaei S, Chend S (2018) Power curtailment in cloud environment utilising load balancing machine allocation. In: 2018 IEEE SmartWorld. Ubiquitous intelligence computing, advanced trusted computing, scalable computing communications, cloud big data computing, internet of people and smart city innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI), pp 1364–1370
11. Sangaiah AK, Javadpour A, Pinto P, Ja'fari F, Zhang W (2022) Improving quality of service in 5G resilient communication with the cellular structure of smartphones. ACM Transactions on Sensor Networks (TOSN) 18(3):1–23
12. Chen M, Chen Z, Zhu L, Hao J, inventors; Verizon Patent, Licensing Inc, assignee. Distributed authentication for internet-of-things resources. United States patent US 10,164,983. 2018 Dec 25.
13. Zahoor S, Mir RN (2021) Resource management in pervasive Internet of Things: a survey. J King Saud Univer Inf Sci 33(8):921–935
14. Nguyen QH, Do NH, Le H-C (2018) Development of a QoS provisioning capable cost-effective SDN-based switch for IoT communication. In: International conference on advanced technologies for communications (ATC), pp 220–225
15. Bera S, Misra S, Vasilakos AV (2017) Software-defined networking for internet of things: a survey. IEEE Internet Things J 4(6):1994–2008
16. Sobin CC (2020) A survey on architecture, protocols and challenges in IoT. Wirel Pers Commun 112(3):1383–1429
17. Fremantle P (2015) A reference architecture for the internet of things. WSO2 White paper 02–04
18. Silva BN, Khan M, Han K (2018) Towards sustainable smart cities: a review of trends, architectures, components, and open challenges in smart cities. Sustain Cities Soc 38:697–713
19. Javadpour A, Abadi AMH, Rezaei S, Zomorodian M, Rostami AS (2022) Improving load balancing for data-duplication in big data cloud computing networks. Cluster Computing 25(4):2613–2631
20. Singh M, Baranwal G, 11Quality of Service (QoS) in Internet of Things'', in Proceedings - (2018) 3rd international conference on internet of things: smart innovation and usages. IoT-SIU 2018:2018
21. Chen Y, Kar S, Moura JMF (2018) The internet of things: secure distributed inference. IEEE Signal Process Mag 35(5):64–75
22. Mishra D, Alexandropoulos GC, De S (2018) Energy sustainable IoT with individual QoS constraints through MISO SWIPT multicasting. IEEE Internet Things J 5(4):2856–2867
23. Bellavista P, Berrocal J, Corradi A, Das SK, Foschini L, Zanni A (2019) A survey on fog computing for the internet of things. Pervas Mob Comput 52:71–99
24. Ni J, Zhang K, Lin X, Shen X (2017) Securing fog computing for internet of things applications: challenges and solutions. IEEE Commun Surv Tutor 20(1):601–628
25. Gao X, Huang X, Bian S, Shao Z, Yang Y (2019) PORA: predictive offloading and resource allocation in dynamic fog computing systems. IEEE Internet Things J 7(1):72–87
26. Ghobaei-Arani M, Souri A, Rahmanian AA (2020) Resource management approaches in fog computing: a comprehensive review. J Grid Comput 18(1):1–42
27. Sangaiah AK, Javadpour A, Pinto P, Rezaei S, Zhang W (2023) Enhanced resource allocation in distributed cloud using fuzzy meta-heuristics optimization. Computer Communications
28. da Silva ACF, Hirmer P, Schneider J, Ulusal S, Frigo MT (2020) Mbp: not just an IoT platform. In: IEEE international conference on pervasive computing and communications workshops (PerCom workshops), pp 1–3
29. Liu L, Chang Z, Guo X (2018) Socially aware dynamic computation offloading scheme for fog computing system with energy harvesting devices. IEEE Internet Things J 5(3):1869–1879
30. Zhang Qin, Yutang Liu (2023) An energy cooperation method of wireless sensor networks based on partially observable Markov decision processes. Sustain Energy Technol Assess 55:102997
31. Gupta S, Singh N (2023) Toward intelligent resource management in dynamic Fog Computingbased Internet of Things environment with Deep Reinforcement Learning: A survey. Int J Commun Syst 36(4):e5411
32. Hazra A, Rana P, Adhikari M, Amgoth T (2023) Fog computing for next-generation internet of things: fundamental, state-of-the-art and research challenges. Computer Science Review 48:100549

33. Rehman A, Awan KA, Ud Din I, Almogren A, Alabdulkareem M (2023) FogTrust: Fog- Integrated Multi-Leveled Trust Management Mechanism for Internet of Things. Technologies 11(1):27

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Authors and Affiliations

Fatimah Faraji[1] · **Amir Javadpour**[2,3] · **Arun Kumar Sangaiah**[4,6] · **Hadi Zavieh**[5]

✉ Amir Javadpour
A.javadpour87@gmail.com

Arun Kumar Sangaiah
aksangaiah@ieee.org

[1] Department of Computer Engineering, Yazd University, Safaeih, Yazd 100190, Iran

[2] Department of Computer Science and Technology (Cyberspace Security), Harbin Institute of Technology, Shenzhen 10587, China

[3] ADiT-Lab, Electrotechnics and Telecommunications Department, ISMAI, and INESC TEC, Instituto Politécnico de Viana do Castelo, ISMAI, Porto 10587, Portugal

[4] International Graduate School of Artificial Intelligence, National Yunlin University of Science and Technology, Yunlin 64002, Taiwan

[5] Department of Economic and Statistics, Guangzhou Higher Education Mega Center, Guangzhou University, Guangzhou 510006, China

[6] Department of Electrical and Computer Engineering, Lebanese American University, 13-5053 Byblos , Lebanon