sensors

MDPI

Article

Multilevel Central Trust Management Approach for Task Scheduling on IoT-Based Mobile Cloud Computing

Abid Ali [1,2,†], Muhammad Munawar Iqbal [1,†], Harun Jamil [3,†], Habib Akbar [1], Ammar Muthanna [4,5], Meryem Ammi [6] and Maha M. Althobaiti [7,*]

1 Department of Computer Science, The University of Engineering and Technology, Taxila 47080, Pakistan; abidali.hzr@gmail.com (A.A.); munwariq@gmail.com (M.M.I.); habibakbar@uoh.edu.pk (H.A.)
2 Department of Computer Science, Govt Akhtar Nawaz Khan (Shaheed) Degree College KTS, Haripur 22620, Pakistan
3 Department of Electronic Engineering, Jeju National University, Jejusi 63243, Jeju-do, Korea; harunjamil@hotmail.com
4 Department of Telecommunication Networks and Data Transmission, The Bonch-Bruevich Saint-Petersburg State University of Telecommunications, 193232 Saint Petersburg, Russia; ammarexpress@gmail.com
5 Department of Computer Science, RUDN University, Peoples' Friendship University of Russia, 6 Miklukho-Maklaya Str., 117198 Moscow, Russia
6 Department of Computational Sciences, Naif Arab University for Security Sciences, Riyadh 13216, Saudi Arabia; mammi@nauss.edu.sa
7 Department of Computer Science, College of Computing and Information Technology, Taif University, P.O. Box 11099, Taif 21944, Saudi Arabia
* Correspondence: Maha_m@tu.edu.sa
† These authors contributed equally to this work.

Abstract: With the increasing number of mobile devices and IoT devices across a wide range of real-life applications, our mobile cloud computing devices will not cope with this growing number of audiences soon, which implies and demands the need to shift to fog computing. Task scheduling is one of the most demanding scopes after the trust computation inside the trustable nodes. The mobile devices and IoT devices transfer the resource-intensive tasks towards mobile cloud computing. Some tasks are resource-intensive and not trustable to allocate to the mobile cloud computing resources. This consequently gives rise to trust evaluation and data sync-up of devices joining and leaving the network. The resources are more intensive for cloud computing and mobile cloud computing. Time, energy, and resources are wasted due to the nontrustable nodes. This research article proposes a multilevel trust enhancement approach for efficient task scheduling in mobile cloud environments. We first calculate the trustable tasks needed to offload towards the mobile cloud computing. Then, an efficient and dynamic scheduler is added to enhance the task scheduling after trust computation using social and environmental trust computation techniques. To improve the time and energy efficiency of IoT and mobile devices using the proposed technique, the energy computation and time request computation are compared with the existing methods from literature, which identified improvements in the results. Our proposed approach is centralized to tackle constant SyncUPs of incoming devices' trust values with mobile cloud computing. With the benefits of mobile cloud computing, the centralized data distribution method is a positive approach.

Keywords: index terms—mobile cloud computing; task scheduling; trust development; energy optimization

check for updates

Citation: Ali, A.; Iqbal, M.M.; Jamil, H.; Akbar, H.; Muthanna, A.; Ammi, M.; Althobaiti, M.M. Multilevel Central Trust Management Approach for Task Scheduling on IoT-Based Mobile Cloud Computing. *Sensors* **2022**, *22*, 108. https://doi.org/10.3390/s22010108

Academic Editor: Ivan Andonovic

Received: 12 November 2021
Accepted: 21 December 2021
Published: 24 December 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Internet of Things (IoT) and mobile cloud computing (MCC) are names given to the emerging concept of establishing a meaningful relationship between the actual objects around us to interrelated things that collectively change our traditional lifestyle with progressive and forward-looking ideas [1]. Smart homes, smart cities, and intelligent transportation systems are practical examples of these ideas, where devices and man's real-time

performance integrate to obtain intelligent access to physical changes in our surroundings [2]. It is the biggest revolution of the 21st century and allows every entity of the natural world to be connected, whether it is a group of people, machines, sensors/actuators, or anything else [3]. Therefore, we can say that it is just a name for our ease, through which we can obtain connection to the internet regardless of our location, expensive connectivity devices, and weak Wi-Fi signal issues [4,5]. MCC provides endless connectivity to the IoT and mobile devices to connect and perform their required tasks with less power consumption and less time stamp. Task scheduling for mobile cloud computing enhances sensor processing in IoT devices and mobile device tasks. We investigated and provided significant task scheduling for mobile and IoT-based sensor devices through mobile cloud computing [6].

Mobile cloud is a network of networks where physical objects such as mobile devices and sensor-equipped devices are connected to process their running tasks and update real-time data [7]. The sensors with mobile cloud network (MCN) act as global architecture for advanced level services connectivity, i.e., virtual and physical, information society, and interoperable ICT [8]. IoT allows communication between different heterogeneous devices by the integration of various technologies, i.e., radio frequency identification (RFID) [9], near field communication (NFC) [10], wireless sensor networks (WSN) [11], and mobile cloud computing (MCC). Thus, we can say it is the technology that allows networked devices to interchange information and perform desired activities without manual assistance. Figure 1 depicts the IoT environment with all the devices to communicate [12].

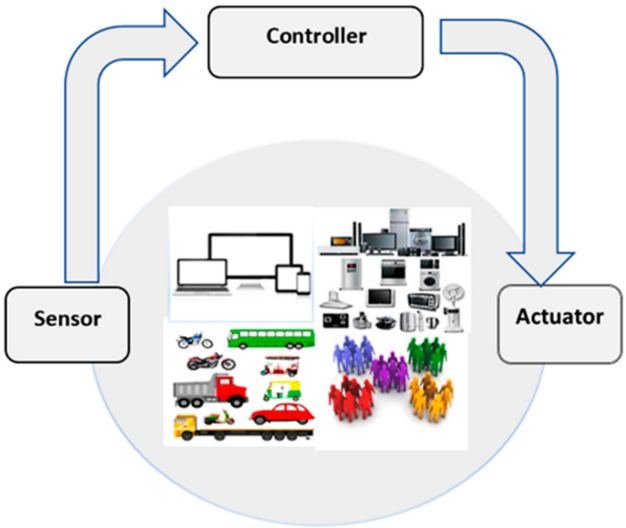

Figure 1. Use of controller, sensors, and actuators in IoT environment [13].

Secure communication among IoT and mobile devices in MCC is not possible without trust evaluation. When two persons come into a relationship, the very first thing developed at that instance is "trust." Their relationship quality is directly proportional to the degree of trust [14]. The more they trust, the more they stay in contact with one another. Similarly, connectivity is a relationship between mobile networked devices [15]. Figure 2 shows IoT and mobile devices' trust working to communication after establishing trust values. To make this connectivity robust and reliable, we must introduce trust among mobile devices, sensors, and actuators. Trust motivates the collaboration between two communicating parties. Belief is a single word that defines trust straightforwardly and concisely. Trust can be used in a different context, so everybody defines this term differently to describe the degree of vigorous confidence in someone's reliability, honesty, and truthfulness [16].

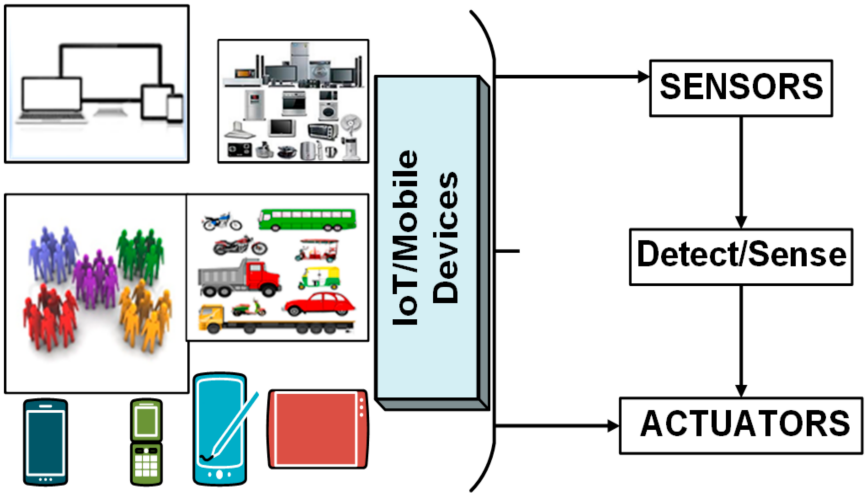

Figure 2. Mobile cloud with IoT and mobile devices for data sensing.

Trust inside the global mobile useability predicts the other mobile device's behavior. It is a directional relationship between trustor and trustee. The node which desires to communicate with the other party is called the trustor. The other node with which the trustor communicates is called the trustee. Digital trust evaluates past behavior or evidence of the behavior of a device concerning its self-defined level of trustworthiness, which helps perceive its upcoming activities [17]. It is a presupposition to enhance the decision-making for successful cooperation between two agents. Trust between devices occurs at first glance, and it seems very unusual and extraordinary that devices will be expected to show trustworthiness to one another. However, they neither have an intelligence quotient nor an emotional quotient. Figure 3 depicts the relationship between the trustor and trustee devices [18].

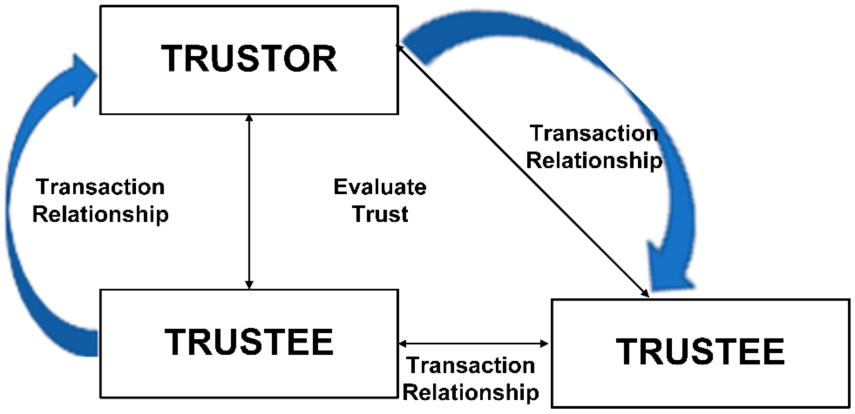

Figure 3. Relationship between trustor and trustee [18].

Today in IoT and MCC scenarios, many heterogeneous devices communicate with one other. The heterogeneity of devices arises from security, privacy, mobility, power consumption, interoperability, artificial intelligence (AI) adoption, trust, task scheduling, cloud computing, and real-time data processing. Trust between two communicating devices is one of the fundamental issues which must be resolved first. Otherwise, the reason for IoT adoption in MCC will become meaningless. This fact has opened a new research door called "TRUST Enables task Scheduling in MCC" to upload the mobile device's tasks using MCC. When we talk about social networks, one of the users' widespread problems is "privacy". No-one is willing to compromise on security or privacy issues. Most users demand the safe conduct of their confidential data while using online services [19].

Mobile cloud computing (MCC) acts as an alternative to the cloud to compute, store, control, and maintain the network near IoT and mobile devices. It is simply a layer between edge and cloud to process the data before sending it to the actual cloud [20]. MCC makes cloud services more effective as it reduces latency, saves bandwidth and storage, and enhances the quality of service (QoS), while reducing power or energy consumption and CPU time utilization. MCC can connect several mobiles and IoT devices and share services and computational resources among those devices on an on-demand basis. These services are accessible through the third-party platform to connect and effectively collaborate among users. MCC structure is demonstrated in Figure 4. Mobile devices connected from remote locations try to connect with clouds to access the cloud services. The local mobile network provider (MNP) initially accepts all the incoming requests from the mobile devices. The MNP contains a central server and database to provide the services to these mobile users for the mobile network [21]; after obtaining the request, it is prepared for internet service providers to control and coordinate for services. After services are not found on the local server, these services are searched from a remote cloud provider with an effective platform [22–24]. In 2012, CISCO suggested MCC to eliminate the shortcomings of cloud computing in IoT and mobile networks.

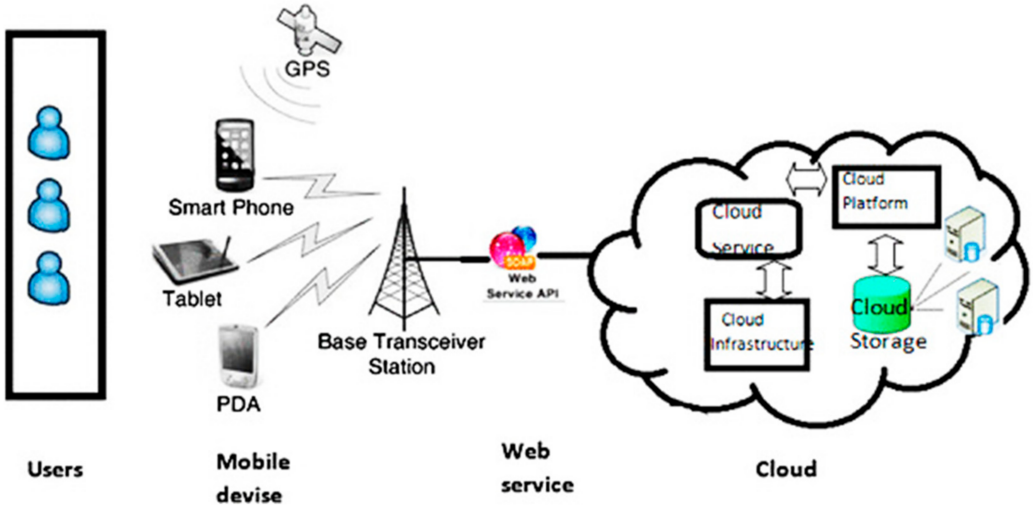

Figure 4. Mobile cloud computing for users, mobile devices, web services, and cloud [25].

Users can enjoy three essential MCC services regardless of their physical presence, background, and powerful computing hardware. They can also use the software running on cloud infrastructure (SaaS) [26]. They can even create their application software using different tools, programming languages, libraries, and several other services to access and control only their own deployed applications on the cloud (PaaS) [27]. Moreover, they can use fundamental storage, network, and computing resources in the area where their application is deployed or running (IaaS).

Trust evaluation and modeling are used among mobile devices and mobile cloud servers to estimate the devices' reliability. The trust improves the cloud computing performance efficiency and enhances secure communication among trustable devices. A model describes whether it computes the trustworthiness of nodes or data. During the modeling phase, it is necessary to decide whether a node's trustworthiness must be checked or the data's trust. In [28], the trust modeling among the MCC task schedule was introduced to only offload the tasks which are trustful. These tasks are trustworthy to enhance the trustworthiness of the devices and other related features. Figure 5 shows the trust model in MCC [29]. Trust management is a service mechanism that self-organizes items based on using their trust status to decide. Trust management constructs a framework where mobile devices and MCC draw closer and form a trust-based relationship to exchange

sensitive data confidently to process in the MCC virtual machines [30]. This can be done by analyzing and computing the degree of trust in their relationship to make better decisions.

- IoT and mobile devices tasks must be scheduled through MCC due to their energy and time constraints. During task scheduling, trustworthiness is one of the important elements because we need to offload only those trustworthiness tasks. This research article focused on this problem faced by the MCC during task offloading. Trust is required to offload the tasks because they execute MCC. The main contributions address time, packet delivery ratio, trustworthiness, and power consumption. The main contributions are the main objectives to adopt in trustable task scheduling in mobile cloud computing through organized algorithms.
- The proposed technique predicts trustable task scheduling to enhance the efficiency of the proposed system.
- Task scheduler updates from trustee and trustor to communicate with each other to exchange trust boundaries and then decides through trust computational algorithm for dynamic decision-making.
- Dynamic trust manager uses trust-based certification to execute and offload only trusted tasks passed from trusted computational models.
- Trust evaluation and development are handled through Algorithm 1, and correspondence and addition of new mobile node for trust evaluation is checked through Algorithm 2.
- Trustable task offloading through Algorithms 3 and 4 effectively offloads the task through effective decision-making.
- We effectively enhance the quality of service (QoS) through a multilevel central trust management approach for task scheduling on IoT-based MCC.
- Finally, to evaluate the system performance, we analyze the results using mobile offloading through simulation. Our proposed technique indicates that the trust development algorithm and task offloading decision algorithm effectively improves the system decision-making, and less power is consumed through the proposed approach.

Section 2 presents the related work on trust development, task scheduling, and fault tolerance. Then, in Section 3, we present the proposed model for the relevant problem presented in Section 1, using mathematical problem formulation and algorithms supported by methodology and flow diagrams. Section 4 presents a simulation environment using the hybrid approach for task scheduling and problem formulation. Section 5 presents the conclusion supported by future directions for better task scheduling.

2. Related Work

According to FCR [12], the cloud is a traditional central server that facilitates almost every customer type by providing ample storage, computations, and network services at very cheap rates. This low cost motivates many users to leverage cloud computing. These are the good aspects of cloud computing. Still, for a moment, focusing on the other side of the picture, it raises some serious problems such as latency, low bandwidth, security/privacy threats, and unnecessary power consumption and time used by the computational resources [31]. To eliminate these shortcomings of cloud computing, we can adopt MCC. MCC acts as an alternate to the cloud to compute, store, control, and maintain the network near mobile and IoT devices. MCC is a nontrivial extension that reduces cloud computing limitations by introducing these features [32,33].

MCC deployment near the mobile devices and IoT layer is responsible for low latency, which is the essential requirement of gaming, video streaming, and augmented reality in mobile and IoT devices. Figure 5 indicates IoT devices with interaction of MCC. A wide geographical distribution of mobile nodes maintains location awareness. This feature reduces mobility issues. The sovereignty of wireless access plays a beneficial role in implementing smart grids and vehicular networks. It introduces real-time interactions as compared to batch processing and the interoperability of the nodes. It gathers the environmental infor-

mation by negotiating with the sensor-equipped devices (data collectors) and responds to a specific situation using actuators.

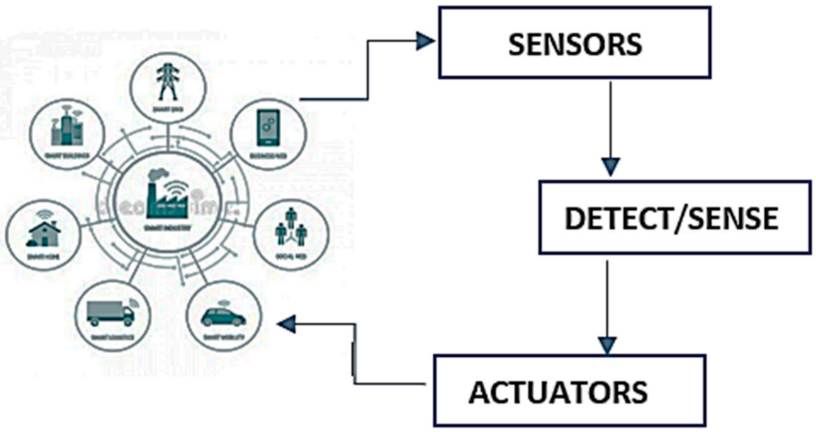

Figure 5. The role of MCC computing in mobile and IoT [34].

Trust is the ability to predict the behavior of another party. The establishment of trust necessitates two or more communicating parties. Trust can be calculated by a multileveled investigation of relationships in different contexts. Otherwise, it may increase additional complexities while increasing the interaction domain [35,36]. Research on multilevel trust, i.e., interorganizational trust, has been minimal compared to individual-level research, and badly affects trust interpretation. After the trust's calculations, only trustable mobile and IoT tasks are ready to upload on the MCC environment.

The trust computation for IoT and mobile devices are discussed and presented in different research areas. The research conducted effectively defines different proposed approaches to enhance the efficiency of task scheduling in MCC. According to [37], planning, commitment, execution, and integration are the significant steps to be taken before trust at multilevel develops from leaders to administration. Reference [38] discusses the relationship between control and confidence and concludes its dependency on institution and situation. The role of collective trust is later examined, and individuals' reactions to changes is investigated. The theoretical and illustrative effect of individuals in shaping their organization has been investigated. In [39], the authors suggested the need for differentiation between trust and distrust. An overall consideration of temporal dynamics is vital as exchange relationships can be changed and affect trust. Lastly, fluidity between people and the environment has been called upon to be explored; refs. [40,41] came up with the idea that the relationship between people and place is elastic and can be worked upon after understanding multilevel trusts. The trust computation is directly linked with MCC. After trust, the next work is to upload the mobile cloud for processing on virtual machines (VM).

After trust computation, the literature is moving towards multilevel trust management frameworks for service-oriented environments. According to [42], IC3 reported a 22.3 percent increase in online fraud to provide trustable MCC services, which is a big reason to distrust online services. Running online businesses demands numerous vendors' and consumers' requirements according to their role to control and provide MCC services to the end-users. Trust is the basic need of every business, whether online or traditionally [43]. Trust in any situation is directly related to the certainty of risk. If the risk is higher than the trust, it is nonsatisfactory [44]. In an open environment of e-commerce, feedback from participating parties plays an important role. So, detection of falsified feedback is necessary. Falsified feedback can make honest participants incredulous, and dishonesty can be considered dubious. The collected feedback can be investigated to accurately picture predicted risks and make consumers feel more confident in online services [45]. This paper proposes a multilevel trust management framework for task scheduling in the MCC environment

to enhance the task offloading efficiency of mobile and IoT devices under the domain of security.

Gupta et al. [46] provide knowledge about on-demand internet access using cloud computing models and ubiquitous computing resources. This model gained popularity to support such models. Moreover, the NIST cloud explains MCC's multiple distinguishing features: rapid elasticity, measured self-services, on-demand services, broader network access, and rapid elasticity. In MCC, the resources are distributed over the network, and distribution provides heterogeneity among resource-sucking devices. Trustable resources are not effectively utilized for efficient workflow and effective features distributions. Fault tolerance is significantly overcome due to trustable communication in MCC. Liu et al. [47] checked the resource distributed with fault tolerance and millions of mobile devices disconnected from any service disasters.

The recent technique for task scheduling in MCC is presented in Table 1. We compare the recent literature for task scheduling in MCC for fault rate, energy optimization, heterogeneity, storage, time, task offloading, control message, and percentage of tasks to be offloaded. The parameters are discussed for IoT and mobile devices to be offloaded to the mobile cloud or executed on the mobile device [48,49]. Table 1 effectively compares the related results obtained through these proposed techniques. Table 1 shows that fault rate, energy optimization, time constraints, and offloading are either not evaluated collectively or are evaluated with lower results than the proposed technique.

Table 1. Comparison of IOT-based task scheduling.

Proposed Papers	Fault Rate	Time	Energy Optimization	Offload	Heterogeneity	Control Messages	Storage	% of Task Executed
Lee et al. [50]	✓	✓	-	✓	✓	✓	-	-
Raju et al. [51]	✓	✓	-	-	✓	-	-	✓
Abd et al. [52]	✓	✓	✓	-	✓	-	-	✓
Park et al. [53]	✓	✓	✓	✓	-	✓	✓	✓
Al-Sayed et al. [54]	✓	✓	-	-	-	-	-	-
Kashanchi et al. [55]	✓	-	-	✓	-	-	✓	-
Peng et al. [56]	✓	-	✓	-	✓	-	✓	-
Tang et al. [57]	✓	-	✓	-	-	-	-	✓
Lin, Xue, et al. [58]	-	-	✓	✓	-	-	-	✓
Guo et al. [59]	-	-	✓	✓	-	✓	-	-
Wei et al. [60]	-	✓	-	-	✓	-	-	-

3. Methodology

3.1. Model Structure

Today, almost everybody is becoming part of social networks. Still, no one is willing to compromise on privacy, one of the most common problems with IoT deployment. The lack of trust plays a vital role in hindering using online services to deal with this problem. We propose developing an environment where users can feel confident to upload the mobile- and IoT-based devices to MCC. This approach works in two ways. First, it computes the trustworthiness of every device and task. The trustful functions should be distributed to the cloud layer and provide the significance of controlling all the automation techniques. Figure 6 depicts the working of task schedular to support for experience based trust computation and trust evaluation.

The model introduces a way to accomplish a trustworthy interaction between two communicating nodes, i.e., mobile/IoT devices and MCC. The other nodes that allow them to communicate with it are termed the trustor, while the other communicating participant is the trustee. Whenever a new trustee joins a network, its behavior is unpredictable before its positive or negative performance. After starting a communication, the MCC recommends the trustee participate in the network for some specific period under controlled access. After completing the first communication session, the trustee's performance explains why the

trustor can evaluate the trustee's trustworthiness through its experience. The mobile cloud layer controls and provides full access and performs all the tasks' computations. Reliability can be formulated by considering trust properties such as honesty, latency, reluctance, and competence. All properties are evaluated individually to give readings in numeric values. Figure 7 depicts the computation of the trust of the devices and the task scheduler schedule for the MCC tasks.

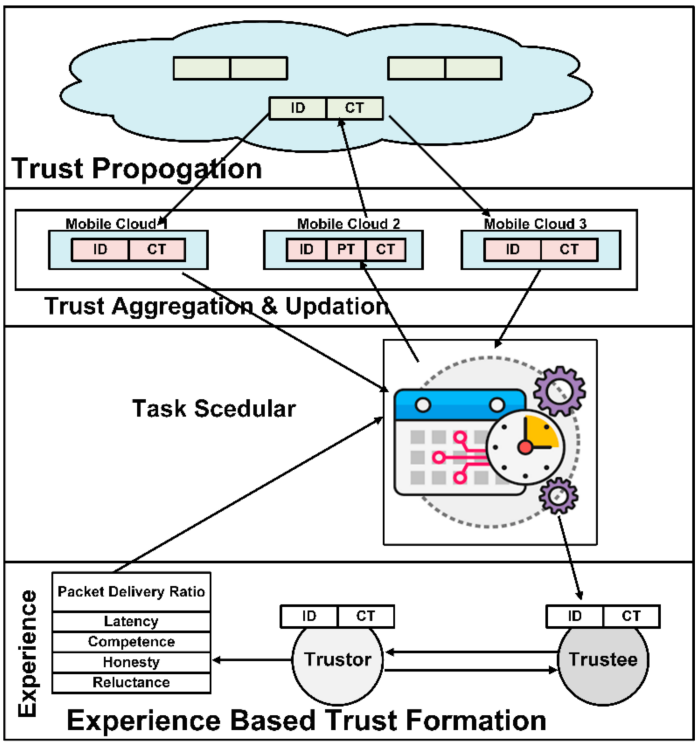

Figure 6. Mobile task scheduling and trust computation.

All values collectively give wholesome weight by performing some statistical operations, and that value is the trustee's trust value. After this calculation, the trustor sends the weight toward the task scheduler to update the trust value based on the threshold. If it has any previous value, the trust values will be updated by simply taking previous and current values. The resultant values are sent to the MCC server. These values are broadcast to all the nodes in the network. This strategy may also contribute to resolving mobility issues and the quality of services. The only trusted tasks from the devices towards the cloud are shifted and provided through trust propagation. All values collectively give wholesome weight by performing some statistical operations, and that value is the trustee's trust value. This strategy may also contribute to the resolution of mobility issues and the quality of services. The working of the task scheduler is enhanced with time as the system becomes mature. The proposed approach builds on the top of the trust levels and provides the significance of these values. The task scheduler schedules only those tasks that passed the trustor and trustee values' IC and CT criteria. When a trustee is an old participant in the network with a trust level history, it has those trust values labeled. Those labels serve as a pass for a trustee to communicate on the web. These values range between 0 to 100, which serves as a scale of trust value to schedule MCC for task processing. Figure 6 elaborates the trust computation with the task scheduler, using experience-based trust evaluation.

If the label on any trustee is less than 50, it can only process on the local machines, i.e., mobile devices and IoT devices.

If the trust value exceeds 50, it schedules through the scheduler towards the MCC.

Trustees with a trust value of more than 90 can be selected as direct service providers to the MCC through direct task scheduling.

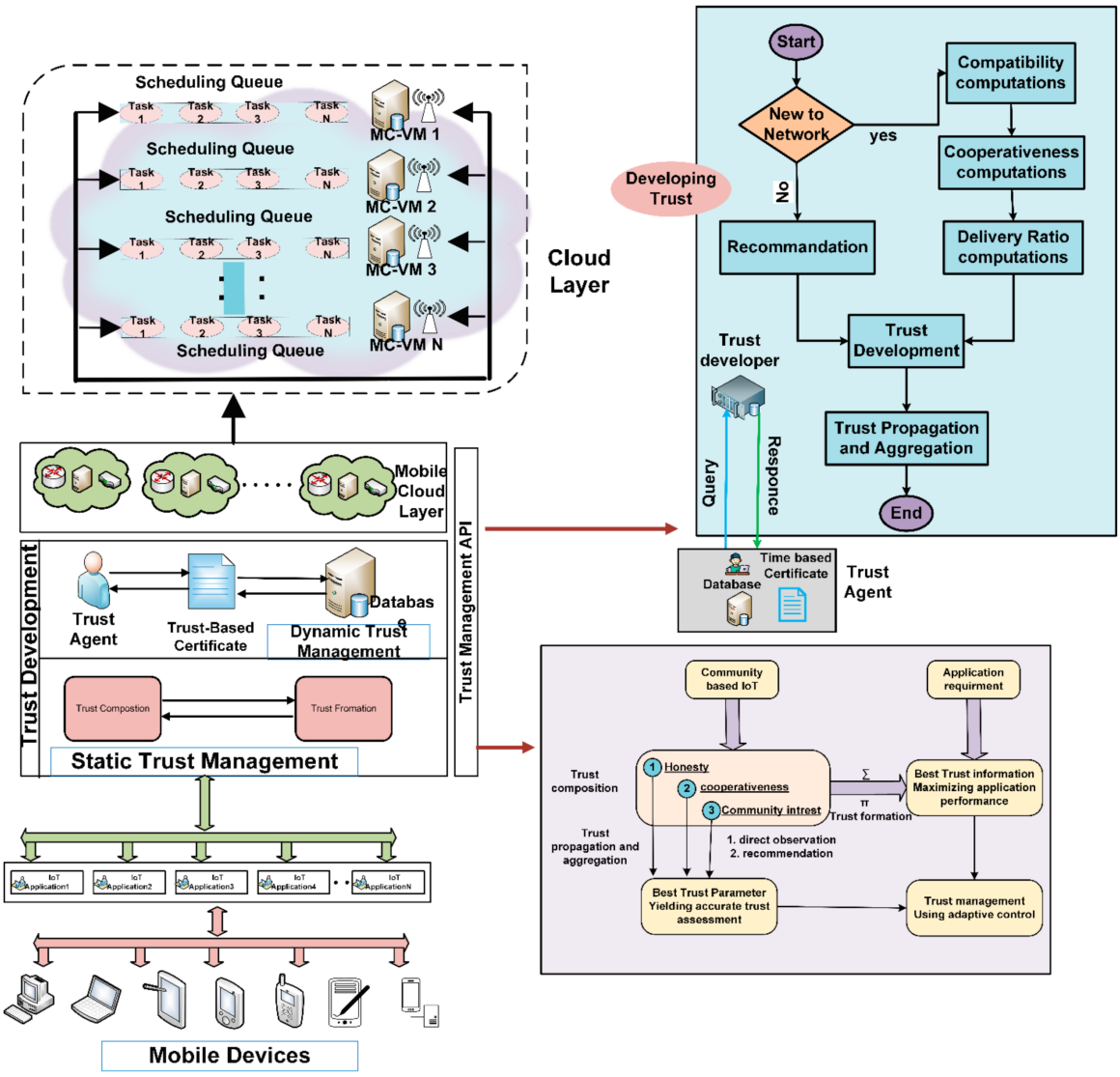

Figure 7. Trust development and task scheduling for mobile cloud computing.

In the situations in which a trustee is to be taken as a service provider by any node, it is required to be a trustful entity. Its label is matched with the centrally propagated trust value to avoid any misleading circumstances. If its value fulfills the threshold requirement, the communication starts; otherwise, the access is denied.

After completing the first communication session, the trustor computes the trust values based on its experiences with the trustee. Two different aspects of the trustee are considered to evaluate its trustworthiness in two different levels. In level 1—social trust, Liu's technique [16] is used to determine whether the device is honest or not. If the device comes out, to be frank, in level 1, it goes to level 2 (QoS trust) for further evaluation. At this level, the assessment considers availability and reliability as two standard trust properties. After passing through these trust levels, the task is ready to schedule for MCC's uploaded server. MCC is a directory connected to cloud computing to gather some of the services that are required to process. This technique is novel to provide efficient trust management

entities and significantly enhance the trustworthiness of these entities. Figure 8 depicts the central model diagram from the trust development and task scheduling for mobile cloud computing. Algorithm 1 discusses and computes QoS trust evaluation and development. Trust development enhances the selection of trustable tasks from mobile and IoT-based devices. The output of Algorithm 1 is the trust computation of the new functions from mobile and IoT devices.

$$Trust\_evaluation(i \leftarrow j) \tag{1}$$

Equation (1) computes the trust evaluation for the trustor *J* and trustee *I*. After the trust evaluation, the trust identification is performed and provides the significance to control the trust management.

$$Check\text{: } I \leftarrow j\ (ID + L_i) \tag{2}$$

Equation (2) defines and check the trust from trustee to trustor. The *ID* and *Li* are the required parameters to compute the trust values. Values from trustor *I* and trustee *J* are based on the new trust computation (Algorithm 2) or directly added to the social trust adoption technique (Algorithm 3).

$$T_{avi}\text{: } I \leftarrow j\ (avi_{i \leftarrow j}, rel_{i \leftarrow j}) \tag{3}$$

Equation (3) defines the trustable nodes to be selected for the final computation. The trustee devices enhance the probability of the nodes and significantly reduce the task offloading phenomenon. In Algorithm 3, we adopt the social trust adaptation technique to compute the reliable social level of trust among all the different devices' tasks. The adaptation technique is taken from research to calculate the vehicle's total confidence to control the system's operations with an efficient control view and control mechanism. Social trust is adopted for every node, and every node is responsible for adapting these trust values for efficient resources. If the adaptation technique is not followed, the task is refused access to upload to the cloud server. The jumping from one algorithm towards another makes task scheduling efficient and trustable. Algorithm 4 is used for the trust adaptation and trust computation technique to schedule the MCC tasks. All the information from job nodes and IoT devices is fetched to control the job descriptions. After they finalize the nodes and selection procedure, the nodes are sent back towards the scheduling. After the trust computation, these nodes were completed, and fulfilled the criteria to control these specifications. The main scenario to schedule the task is based on specific parameters to be fulfilled at this task scheduling stage.

$$F \leftarrow \frac{\Delta T_m}{\Delta T_{exc}} \tag{4}$$

In Equation (5), the processing time is computed, and this time rivals the total execution time of the trustable task from the mobile or IoT devices. The job threshold size, which is equal to the job execution time and execution adaptation technique, is similar to the job description time and job running time.

$$Job\_exe_M() \tag{5}$$

$$activeCloud_{(VM)} \tag{6}$$

$$submit_{C(J)}() \tag{7}$$

$$exe_{job}() \tag{8}$$

Algorithm 1. QoS Trust Evaluation and Development.

Input: Mobile Nodes, Sensors, and IoT Devices

Output: Trust Validate

1: *trust_evolution*($I \leftarrow j$)

2: j^{ID} // Trust Identification

3: $J_{req \leftarrow I}$ // Send Request to Trustee i

4: check: $I \leftarrow j$ ($ID + L_i$)

5: if ($j! = L_i$)

Go to Algorithm 2

else

go to Algorithm 3

end if

6: T_{avi}: $i \leftarrow j$($avi_{I \leftarrow j}$, $rel_{I \leftarrow j}$)

// Availability and Reliability

7: T_f: T_avi: $i \leftarrow j$($avi_{i \leftarrow j}$, $rel_{i \leftarrow j}$)

8: if ($T_f > 90\%$)

service_provider ($T_{i \leftarrow j}$)

else if ($T_f > 50\%$ && $T_f < 90\%$)

network_comm ($T_{i \leftarrow j}$)

else

dumble_terminal ($T_{i \leftarrow j}$)

end if

9: trust $_{I \leftarrow j}$ ( ) $\leftarrow$ published ()

Algorithm 2. New Trustee.

Input: New Node(Mobile Device, Sensor, IoT De

vice)

Output: Trustable new Entered Node

2: **Start**

1: $j_{(i)}$

2: **if**($j! = L_i$)

permission_grant(F_n)

go to Algorithm 1 step 8

else

3: Go to Algorithm 3

4: **End**

Algorithm 3. Social Trust Adaptation Technique.

Input: Nodes (trustor, Trustee, Adaptation)

Output: Calculated Social Trust

1: **Start**

2: social_trust($I \leftarrow j$)

3: $j_{(i)}$

4: **if**($j! = h_r$) // Checked through adaptation technique [X. Liu, et al. [16]]

$request\_refuse()$

5: **else**

Go to Algorithm 1 step 8

6: $social\_trust_{calculate( )}$

7: End

Algorithm 4. Task Scheduling Decision.

Input: Input from Table 1 (LEGENDS Table)

Output: Job Scheduling

1: $Mob_{Info}(B, T, L, App, S)$

2: $Job_{Num}(m)$

3: $Node_{Num}(n)$

4: $Fetch_{Info}(T)$

5: $Create_{Node}^{New}(VM, N, Schedular)$

6: Execution of Algorithms 1–3.

6: $Send_{(T,\ D,\ C)} \leftarrow Schedular( )$

7: for ($T \geq 0$) do

$calculate_{exe\_time( )}$

$F \leftarrow \Delta T_m / \Delta T_{exc}$

end for

8: while (job_size $\leq$ threshold) do

$C(B, F, M_b, M_{loc}, M_{storage})$

if ($C \leq M$) then

$job\_exe_M( )$

else

$activeCloud(VM)$

$submit_{C(J)}( )$

$exe_{job}( )$

end if

end while

9: $Job\_state_{Store( )}$

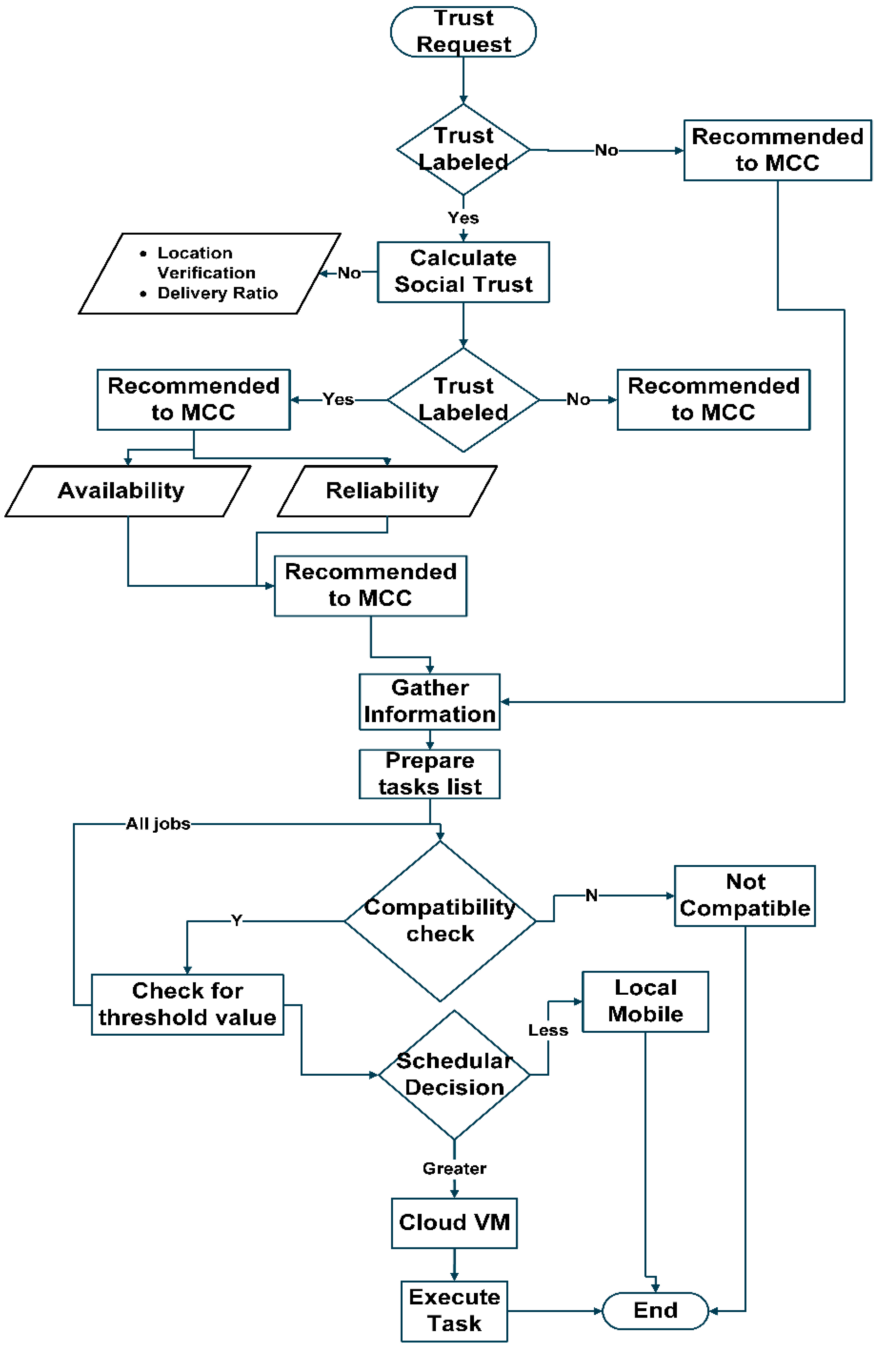

Figure 8. IoT and mobile cloud trust development flow model.

3.2. Trust Factors

Trust factors are checked thoroughly at this level. These two are the critical factors used in avionics to estimate the risk of failure and decide on it. The selection of these factors shows their importance, where any wrong choice has significant, real-life consequences in people's lives. We opted for these features as standards for calculating trust value to cope with vulnerabilities that can cause network failure. Our proposed strategy can develop the fear in users of being penalized in real time by technically controlling their network access based on their behavior. It compels them to behave positively later, if not the first time. Figure 8 shows the complete flow model for trust computation and task scheduling in the proposed technique.

3.3. Reliability

Reliability can show satisfactory performance or the possibility of failure of a system. It can be measured through factors that reinforce the validity of a system. We considered the energy-consumed rate, time taken to respond to a request, and packets delivery ratio as evaluation matrix.

3.4. Availability

Availability is the measure of unpreparedness of trustees during network communication. It can be measured by estimating the possibility of downtimes in the lifecycle.

$$Trustee_{availability} \leftarrow \frac{request\_compt\_time()}{depay() + requestcompttime()} \tag{9}$$

Equation (9) perform trust availability for new tasks. Both properties are considered individually to give numeric value readings used to produce a wholesome value by performing statistical operations. The resultant value is regarded as the trustee's current trust value. After this calculation, the trustor sends these values to the fog node to update current trust values. The fog node checks whether it has any previous trust value, then the trust is calculated by simply taking both current and previous trust values. Otherwise, the trustor's direct observation is considered trust value (this only happens when a new node joins a network). This trust value is then sent to the cloud server to broadcast it for all fog nodes and label it to the device. This label is added to profile information of the device to be available for the time of its next confrontation with other network devices. This way, the impact of the trustee's previous behavior in the form of trust value is publicly visible, and its reputation proceeds it. This approach can also be used to eliminate mobility issues in IoT devices.

4. Results and Discussion

We implemented the algorithm in MATLAB to evaluate our proposed model. MATLAB is one of the best simulators for MCC simulation, specifically for trust computations and energy requirements. The scheduling experiments were conducted on Core i5-CPU/3.0 GHz/8 GB RAM-running PCs and MATLAB R2018b. We design a user-friendly and creative environment using MATLAB to build our experimental model [1]. The experimental model is designed to achieve the reliable and efficient behavior of IoT devices. Two trust parameters are selected; the first is the "availability", and the second is "reliability", which is already discussed in the previous section of the proposed work. We evaluated our work by comparing it with Greedy Perimeter Stateless Routing (GPSR) [4]. We established an IoT environment with 100 edge nodes (T_e and T_r), 10 MCC NodesFn, and a mobile cloud server. The evaluation matrices used to measure the trustworthiness of a device are as follows.

4.1. Time

Proper time management helps us accomplish the top job in the minimum time. Our model performs multiple communicational and computational tasks simultaneously in less time than GPRS. Equation (10) computes total time from task submission to completion. Figure 9 shows the time to manage Trust Request, Trust Development, Trust Upgradation, and Propagation by comparing methodologies with proposed technique.

$$T_{total} = \sum_{req}^{comp} T(T_e + T_r + F_N + C_s) \tag{10}$$

Total is the total time consumed by our model. T is the time taken by the trustee (T_e), the trustor (T_r), mobile cloud node (F_N), and the cloud server. Figure 10 shows the number of tasks increased to 20,000 to check the validity of the proposed system—the proposed system effectively collects makespan time for effective monitoring and validation of the proposed model. The results also elaborate that our system performs effectively

well under such conditions as time passes. In Figure 9, the request elaborates on the request received, and the time it is measured against shows the actual propagated time for request experimentation. T-Development is trust development time. The T-Development time shows the performance of the proposed system, which is better than RGP and MLT techniques. T-Upgradation is trust upgradation. Therefore, trust development time for upgradation of trust enhancement is effective and provides better results than other approaches. Propagation refers to the trust development with upgraded values to show the trust enhancement values for effective analysis and effectiveness. The proposed system shows effective results—overall, 21% better results for GPGR than RGP and MLT approaches.

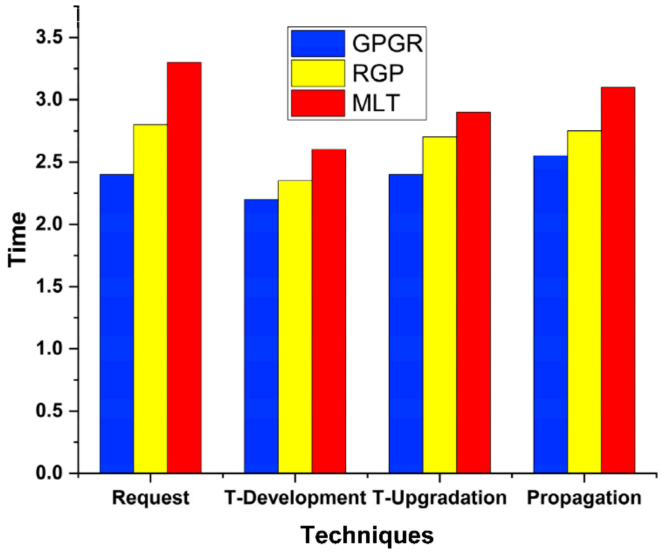

Figure 9. Time for request completion.

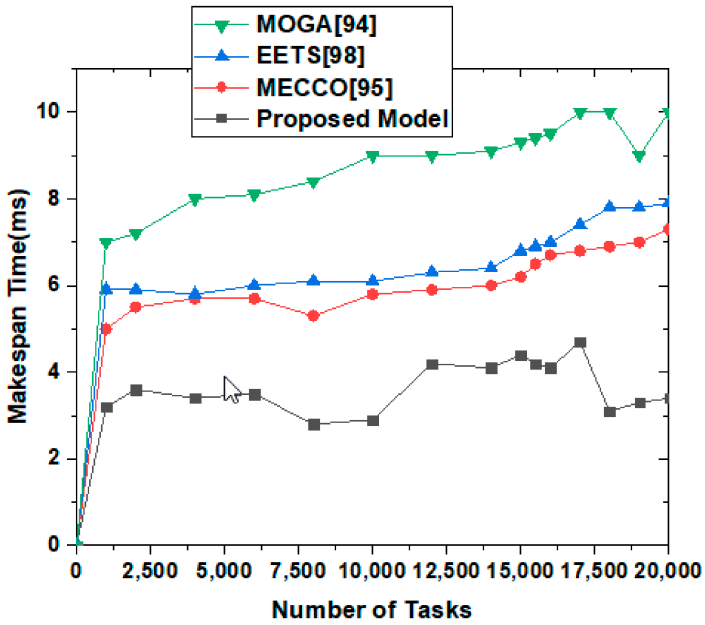

Figure 10. Makespan time consumption after increasing the number of tasks to 20,000.

4.2. Packet Delivery Ratio

Measurement of data concealed or dropped by the trustee can predict the trustee's intentions. When a node shows harmful intentions, it can never be considered reliable. Previous works measured packet delivery in a specific period, but we measured it differently.

$$PDR = \frac{DR}{DR + DL} \tag{11}$$

where PDR is packet delivery ratio, DR is data received, and DL is data lost during scheduling in the proposed technique.

We simulated our proposed technique by deploying 100 mobile or IoT nodes and a central server. Figure 11 shows that our proposed technique achieves high accuracy of value. According to the estimated result ratio, the proposed technique achieves 97% of the packet's delivery r. Overall packet delivery ratio value is 97.865%, which is more powerful than other techniques.

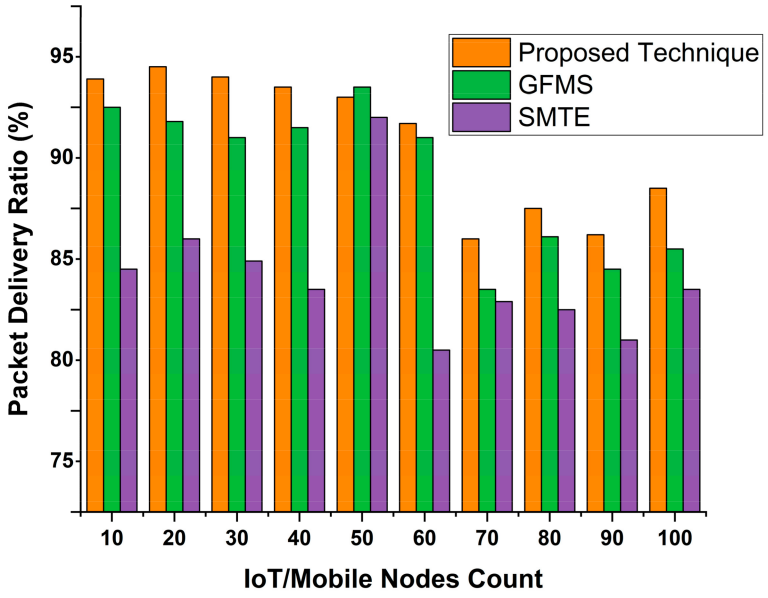

Figure 11. Packet delivery ratio.

4.3. Energy Consumption

Energy consumption plays an essential role in the success or failure of a model, so it should be measured carefully. Econ is the total energy consumed by a model. E is the energy consumed by request submission (RQ), trust development (TD), trust update (tup), and trust propagation. Again, we are comparing our results with MLT and RGP. Figure 12 shows the MLT and RGP comparison with proposed technique.

The results are compared with two approaches, i.e., GFMS and SMTE. Both techniques are designed and developed through proposed task scheduling, and an enhanced version is required to provide effective task processing. Initially, the energy level starts with a high peak time, but the energy consumption remains low with time. The proposed technique shows low energy consumption and enhances energy consumption by providing effective and efficient optimization. The proposed technique shows better results than GFMS and SMTE approaches from the literature. Figure 13 shows these results.

In addition to the time and power consumption, the VM's availability and migration during task scheduling are the other main contributions. Figure 13 shows the comparison of VMs migration during the task scheduling in the proposed technique compared to DRA and MARKOV analysis. We compared the results on multiple VMs, which shows that during the migration, our proposed technique works better to adjust the VM migration from one platform towards another platform for effective and efficient task scheduling. Time and powers are less consumed in the proposed model whenever scheduling a task needed to migrate from one VM to another. This happens when one VM falls, or any other reason for processing and scheduling tasks.

Experiential-based tasks are amplified in consignment (number of functions become batch) and the calculations are excluded, so the tasks' power is smaller than other methods. The chance is calculated throughout Equations (1)–(12). Moreover, the computational possibility calculation, shown in Figure 14, demonstrates the most excellent and minuscule offloading likelihood.

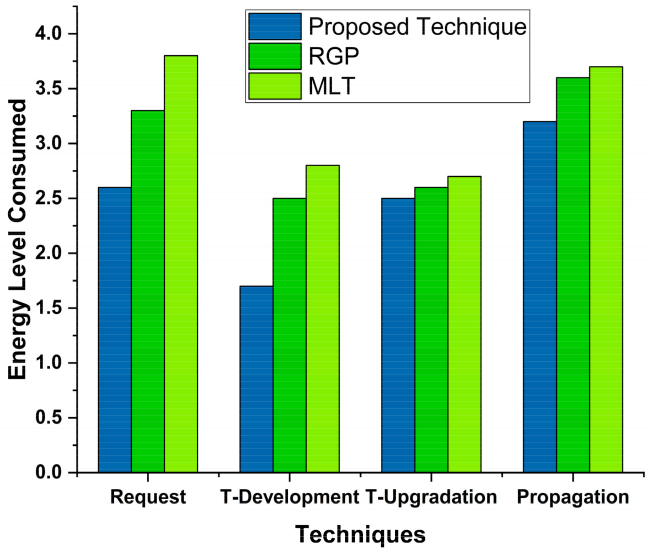

Figure 12. Power consumption while task offloading.

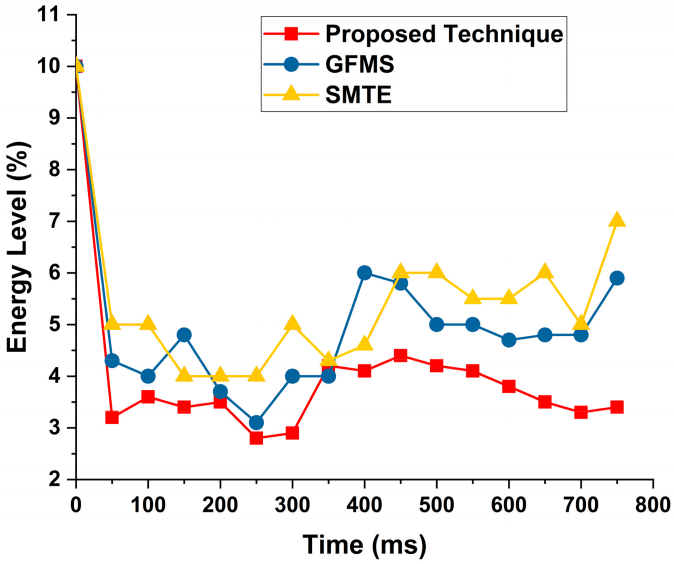

Figure 13. Task energy level consumption while task offloading to MCC VMs.

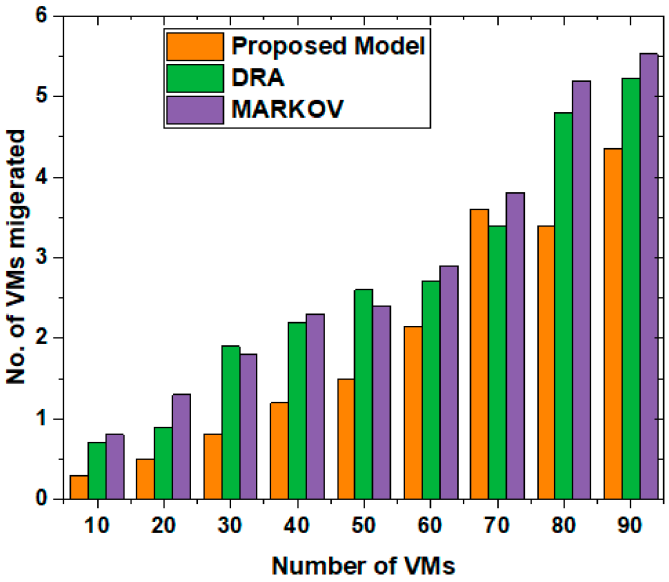

Figure 14. Comparison of VMs for tasks migration.

Figure 15 shows the trust comparison among both techniques. With time, the expected trust from the trust comparison and real trust values shows the enhanced trust obtained from the proposed technique. These trust values are evaluated from Equations (1)–(8) and Algorithm 3. The trust is computed, and results show effective trust computation with efficient and effective trust enhancements.

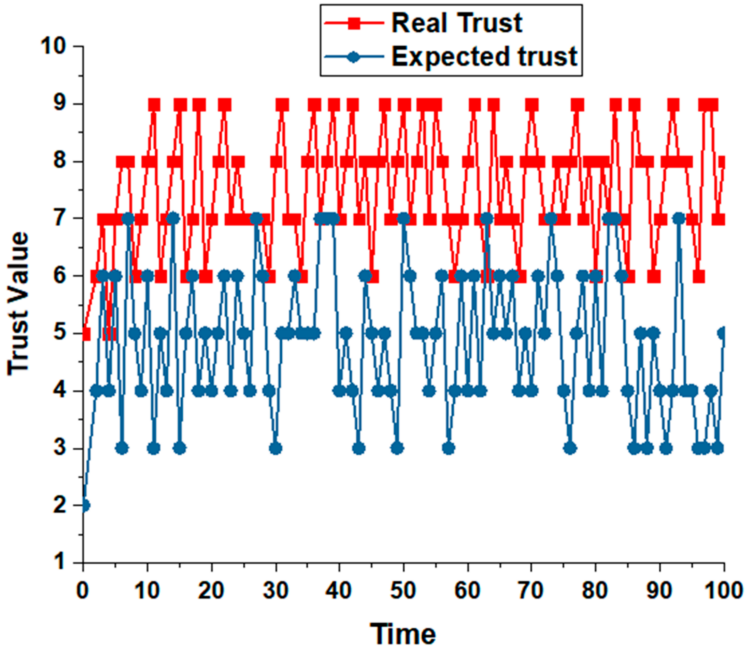

Figure 15. Trust value and trust computations based on time.

They are grounded on the findings of the task's possibility and statistics of mobile device structures shown all through Figure 15. Indicators such as battery information, storage, offloading time, bandwidth, and job completion rate are powerful task offloading frameworks. The time and power consumption are shown in Figure 16. The figure enhances both parameters with effective time and power management. On the other hand, Figure 17 depicts the results of tasks submitted towards the cloud after the final decision. The decision is purely made based on probability and results obtained after Algorithm 4. The results

show that tasks requiring more power, time, and cost must offload towards MCC VM after the computation of trust values computed through Algorithms 2 and 3.

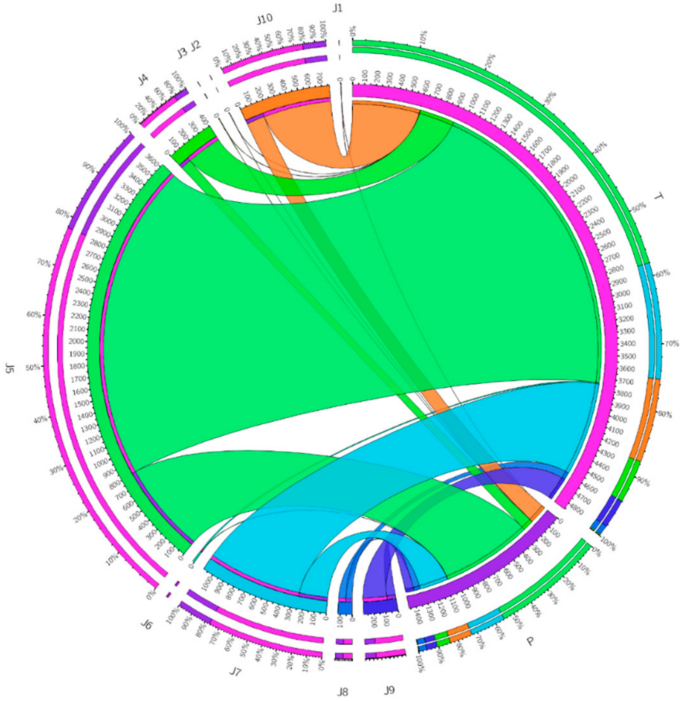

Figure 16. Time and power consumption in the proposed approach.

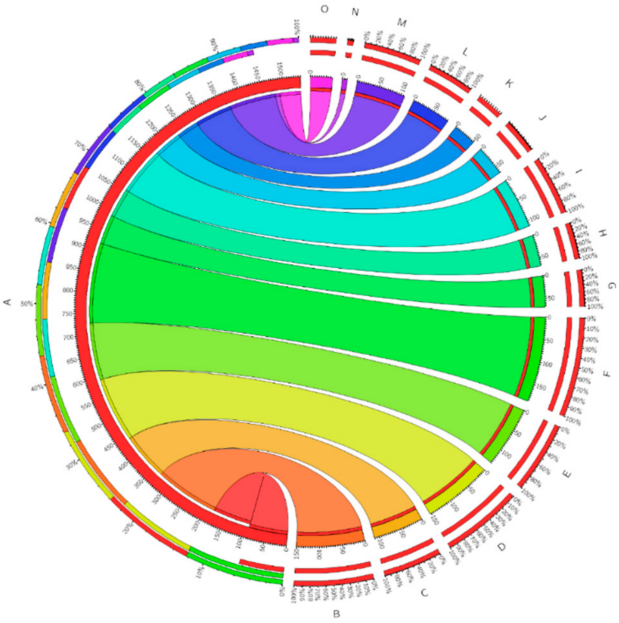

Figure 17. Request submitted to the cloud of the proposed system.

5. Conclusions and Future Work

We proposed a centralized multilayered trust management model for task scheduling in MCC to prove our two-layer trust evaluation model with simulations. The results proved to be better than MOGA [10], EETS [11], MECCO [12], GPSR [4], DRA [13], RGR, and MARKOV [15] (previously accepted techniques), demonstrating our selection of availability and reliability as the trustee's trust evaluation's exemplary standards. Our selected parameters and our model of centrally synchronized MCC nodes such as mobile devices

and IoT devices appear to be the best option for trustable task scheduling. Nontrusted tasks from mobile devices and IoT nodes cannot schedule through the cloud when it can achieve low latency from task submission, centrally synchronizing all fog nodes with trustee trust values, triggered at the interaction of a trustee in a network. The proposed model effectively enhances the results by 20% less than previous techniques from literature, makespan time by 26%, packets delivery and trust computation by 21%, trust values by 17%, and our devices consume 23% less power than the proposed technique.

In the future, the work can consider more security through modern security parameters and provide the significance to handle the resources provided. Additionally, we plan to compute the trust through deep learning and task scheduling through AI constraints.

Author Contributions: Conceptualization, A.A., and M.M.I.; methodology, A.A.; software, H.A.; validation, H.J., and A.M.; formal analysis, M.A.; investigation, M.M.A.; resources, A.A.; data curation, M.M.I.; writing—original draft preparation, A.A.; writing—review and editing, M.M.I.; visualization, A.A.; supervision, M.M.I.; project administration, A.M.; funding acquisition, M.A., and M.M.A. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Taif University Researchers Supporting Project and RUDN University Strategic Academic Leadership Program under grant number (TURSP-2020/328). And The APC was funded by RUDN University Strategic Academic Leadership Program.

Institutional Review Board Statement: Not Applicable.

Informed Consent Statement: Not Applicable.

Data Availability Statement: Not Applicable.

Acknowledgments: We deeply acknowledge Taif University for supporting this research through Taif University Researchers Supporting Project Number (TURSP-2020/328), Taif University, Taif, Saudi Arabia.

Conflicts of Interest: Authors have no conflict of Interest.

References

1. Parajuli, N.; Alsadoon, A.; Prasad, P.W.C.; Ali, R.S.; Alsadoon, O.H. A recent review and a taxonomy for multimedia application in Mobile cloud computing based energy efficient transmission. *Multimed. Tools Appl.* **2020**, *79*, 31567–31594. [CrossRef]
2. Ali, A.; Iqbal, M.M.; Jamil, H.; Qayyum, F.; Jabbar, S.; Cheikhrouhou, O.; Baz, M.; Faisal, J. An Efficient Dynamic-Decision Based Task Scheduler for Task Offloading Optimization and Energy Management in Mobile Cloud Computing. *Sensors* **2021**, *21*, 4527. [CrossRef] [PubMed]
3. Huh, S.; Cho, S.; Kim, S. Managing IoT devices using blockchain platform. In Proceedings of the 2017 19th International Conference on Advanced Communication Technology (ICACT), PyeongChang, Korea, 19–22 February 2017; pp. 464–467.
4. Meneghello, F.; Calore, M.; Zucchetto, D.; Polese, M.; Zanella, A. IoT: Internet of threats? A survey of practical security vulnerabilities in real IoT devices. *IEEE Internet Things J.* **2019**, *6*, 8182–8201. [CrossRef]
5. Xiao, L.; Wan, X.; Lu, X.; Zhang, Y.; Wu, D. IoT security techniques based on machine learning: How do IoT devices use AI to enhance security? *IEEE Signal Process. Mag.* **2018**, *35*, 41–49. [CrossRef]
6. Vaiyapuri, T.; Parvathy, S.V.; Manikandan, V.; Krishnaraj, N.; Gupta, D.; Shankar, K. A Novel Hybrid Optimization for Cluster-Based Routing Protocol in Information-Centric Wireless Sensor Networks for IoT Based Mobile Edge Computing. *Wirel. Pers. Commun.* **2021**, 1–24. [CrossRef]
7. Kumar, D.; Shen, K.; Case, B.; Garg, D.; Alperovich, G.; Kuznetsov, D.; Gupta, R.; Durumeric, Z. All things considered: An analysis of IoT devices on home networks. In Proceedings of the 28th USENIX Security Symposium (USENIX Security 19), Santa Clara, CA, USA, 14–16 August 2019; pp. 1169–1185.
8. Meidan, Y.; Bohadana, M.; Shabtai, A.; Ochoa, M.; Tippenhauer, N.O.; Guarnizo, J.D.; Elovici, Y. Detection of unauthorized IoT devices using machine learning techniques. *arXiv* **2017**, *arXiv:1709.04647*.
9. Liu, S.; Yao, S.; Huang, Y.; Liu, D.; Shao, H.; Zhao, Y.; Li, J.; Wang, T.; Wang, R.; Yang, C.; et al. Handling Missing Sensors in Topology-Aware IoT Applications with Gated Graph Neural Network. In *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*; ACM: New York, NY, USA, 2020; Volume 4, pp. 1–31.
10. Al-Fatlawi, A.H.; Fatlawi, H.K.; Ling, S.H. Recognition physical activities with optimal number of wearable sensors using data mining algorithms and deep belief network. In Proceedings of the 2017 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), Jeju Island, Korea, 11–15 July 2017; pp. 2871–2874.
11. Ali, A.; Haseeb, M. Radio frequency identification (RFID) technology as a strategic tool towards higher performance of supply chain operations in textile and apparel industry of Malaysia. *Uncertain Supply Chain. Manag.* **2019**, *7*, 215–226. [CrossRef]

12. Ratnadewi, R.; Adhie, R.P.; Hutama, Y.; Ahmar, A.S.; Setiawan, M. Implementation cryptography data encryption standard (DES) and triple data encryption standard (3DES) method in communication system based near field communication (NFC). *J. Phys. Conf. Ser.* **2018**, *954*, 012009. [CrossRef]
13. Fong, T. Wireless sensor networks. In *Internet of Things and Data Analytics Handbook*; Wiley: Hoboken, NJ, USA, 2017; pp. 197–213.
14. Varghese, B.; Buyya, R. Next generation cloud computing: New trends and research directions. *Future Gener. Comput. Syst.* **2018**, *79*, 849–861. [CrossRef]
15. Malik, S.U.; Akram, H.; Gill, S.S.; Pervaiz, H.; Malik, H. EFFORT: Energy efficient framework for offload communication in mobile cloud computing. *Softw. Pract. Exp.* **2021**, *51*, 1896–1909. [CrossRef]
16. Liang, W.; Huang, W.; Long, J.; Zhang, K.; Li, K.-C.; Zhang, D. Deep reinforcement learning for resource protection and real-time detection in IoT environment. *IEEE Internet Things J.* **2020**, *7*, 6392–6401. [CrossRef]
17. Bhowmik, A.; De, D. mTrust: Call Behavioral Trust Predictive Analytics Using Unsupervised Learning in Mobile Cloud Computing. *Wirel. Pers. Commun.* **2021**, *117*, 483–501. [CrossRef]
18. Elazhary, H. Applications Internet of Things (IoT), mobile cloud, cloudlet, mobile IoT, IoT cloud, fog, mobile edge, and edge emerging computing paradigms: Disambiguation and research directions. *J. Netw. Comput. Appl.* **2019**, *128*, 105–140. [CrossRef]
19. Dinh, T.; Kim, Y.; Lee, H.J.S. A location-based interactive model of internet of things and cloud (IoT-Cloud) for mobile cloud computing applications. *Sensors* **2017**, *17*, 489. [CrossRef] [PubMed]
20. Yuan, J.; Li, X. A reliable and lightweight trust computing mechanism for IoT edge devices based on multi-source feedback information fusion. *IEEE Access* **2018**, *6*, 23626–23638. [CrossRef]
21. Kalkan, K.; Rasmussen, K. TruSD: Trust framework for service discovery among IoT devices. *Comput. Netw.* **2020**, *178*, 107318. [CrossRef]
22. Mollah, M.B.; Azad, M.A.K.; Vasilakos, A. Security and privacy challenges in mobile cloud computing: Survey and way ahead. *J. Netw. Comput. Appl.* **2017**, *84*, 38–54. [CrossRef]
23. Wang, Z.; McNally, R.; Lenihan, H. The role of social capital and culture on social decision-making constraints: A multilevel investigation. *Eur. Manag. J.* **2019**, *37*, 222–232. [CrossRef]
24. Noor, T.H.; Zeadally, S.; Alfazi, A.; Sheng, Q.Z. Mobile cloud computing: Challenges and future research directions. *J. Netw. Comput. Appl.* **2018**, *115*, 70–85. [CrossRef]
25. Liu, Y.; Zhang, Y.; Ling, J.; Liu, Z. Secure and fine-grained access control on e-healthcare records in mobile cloud computing. *Future Gener. Comput. Syst.* **2018**, *78*, 1020–1026. [CrossRef]
26. Sundararaj, V. Optimal task assignment in mobile cloud computing by queue based ant-bee algorithm. *Wirel. Pers. Commun.* **2019**, *104*, 173–197. [CrossRef]
27. Almusaylim, Z.A.; Jhanjhi, N.Z. Comprehensive review: Privacy protection of user in location-aware services of mobile cloud computing. *Wirel. Pers. Commun.* **2020**, *111*, 541–564. [CrossRef]
28. Walia, A.S. Security Vulnerability in Mobile Cloud Computing (MCC). *Int. J. Adv. Res. Comput. Sci.* **2017**, *8*, 277–281.
29. Tawalbeh, L.A.; Ababneh, F.; Jararweh, Y.; AlDosari, F. Trust delegation-based secure mobile cloud computing framework. *Int. J. Inf. Comput. Secur.* **2017**, *9*, 36–48. [CrossRef]
30. Surridge, M.; Correndo, G.; Meacham, K.; Papay, J.; Phillips, S.C.; Wiegand, S.; Wilkinson, T. Trust Modelling in 5G mobile networks. In Proceedings of the 2018 Workshop on Security in Softwarized Networks: Prospects and Challenges, Budapest, Hungary, 24 August 2018; pp. 14–19.
31. Li, W.; Cao, J.; Hu, K.; Xu, J.; Buyya, R. A trust-based agent learning model for service composition in mobile cloud computing environments. *IEEE Access* **2019**, *7*, 34207–34226. [CrossRef]
32. Chen, R.; Guo, J.; Wang, D.-C.; Tsai, J.J.P.; Al-Hamadi, H.; You, I. Trust-based service management for mobile cloud IoT systems. *IEEE Trans. Netw. Serv. Manag.* **2018**, *16*, 246–263. [CrossRef]
33. Stergiou, C.; Psannis, K.E.; Kim, B.-G.; Gupta, B. Secure integration of IoT and cloud computing. *Future Gener. Comput. Syst.* **2018**, *78*, 964–975. [CrossRef]
34. Subramanian, N.; Jeyaraj, A. Recent security challenges in cloud computing. *Comput. Electr. Eng.* **2018**, *71*, 28–42. [CrossRef]
35. Sunyaev, A. Cloud computing. In *Internet Computing*; Springer: Berlin/Heidelberg, Germany, 2020; pp. 195–236.
36. De Donno, M.; Tange, K.; Dragoni, N. Foundations and evolution of modern computing paradigms: Cloud, iot, edge, and fog. *IEEE Access* **2019**, *7*, 150936–150948. [CrossRef]
37. Fulmer, A.; Dirks, K. *Multilevel Trust: A Theoretical and Practical Imperative*; Taylor & Francis: Abingdon, UK, 2018.
38. Mozumder, N.A. A multilevel trust-based model of ethical public leadership. *J. Bus. Ethics* **2018**, *153*, 167–184. [CrossRef]
39. Rezvani, A.; Khosravi, P.; Ashkanasy, N.M. Examining the interdependencies among emotional intelligence, trust, and performance in infrastructure projects: A multilevel study. *Int. J. Proj. Manag.* **2018**, *36*, 1034–1046. [CrossRef]
40. Li, J.J.; Kim, W.G.; Zhao, X.R. Multilevel model of management support and casino employee turnover intention. *Tour. Manag.* **2017**, *59*, 193–204. [CrossRef]
41. Friend, S.B.; Johnson, J.S.; Sohi, R.S. Propensity to trust salespeople: A contingent multilevel-multisource examination. *J. Bus. Res.* **2018**, *83*, 1–9. [CrossRef]
42. Costa, A.C.; Fulmer, C.A.; Anderson, N.R. Trust in work teams: An integrative review, multilevel model, and future directions. *J. Organ. Behav.* **2018**, *39*, 169–184. [CrossRef]

43. Yu, Y.; Hao, J.-X.; Dong, X.-Y.; Khalifa, M. A multilevel model for effects of social capital and knowledge sharing in knowledge-intensive work teams. *Int. J. Inf. Manag.* **2013**, *33*, 780–790. [CrossRef]
44. Chong, S.-K.; Abawajy, J.; Hamid, I.R.A.; Ahmad, M. A multilevel trust management framework for service oriented environment. *Procedia Soc. Behav. Sci.* **2014**, *129*, 396–405. [CrossRef]
45. Dinh, H.T.; Lee, C.; Niyato, D.; Wang, P. A survey of mobile cloud computing: Architecture, applications, and approaches. *Wirel. Commun. Mob. Comput.* **2013**, *13*, 1587–1611. [CrossRef]
46. Gupta, P.; Gupta, S. Mobile cloud computing: The future of cloud. *Int. J. Adv. Res. Electr. Electron. Instrum. Eng.* **2012**, *1*, 134–145.
47. Liu, Q.; Jian, X.; Hu, J.; Zhao, H.; Zhang, S. An optimized solution for mobile environment using mobile cloud computing. In Proceedings of the 2009 5th International Conference on Wireless Communications, Networking and Mobile Computing, Beijing, China, 24–26 September 2009; pp. 1–5.
48. Guo, S.; Liu, J.; Yang, Y.; Xiao, B.; Li, Z. Energy-efficient dynamic computation offloading and cooperative task scheduling in mobile cloud computing. *IEEE Trans. Mob. Comput.* **2018**, *18*, 319–333. [CrossRef]
49. Tang, C.; Xiao, S.; Wei, X.; Hao, M.; Chen, W. Energy efficient and deadline satisfied task scheduling in mobile cloud computing. In Proceedings of the 2018 IEEE International Conference on Big Data and Smart Computing (BigComp), Shanghai, China, 15–17 January 2018; pp. 198–205.
50. Lee, J.; Gil, J. Adaptive fault-tolerant scheduling strategies for mobile cloud computing. *J. Supercomput.* **2019**, *75*, 4472–4488. [CrossRef]
51. Raju, D.N.; Saritha, V. Architecture for fault tolerance in mobile cloud computing using disease resistance approach. *Int. J. Commun. Netw. Inf. Secur.* **2016**, *8*, 112.
52. Abd, S.K.; Al-Haddad, S.A.R.; Hashim, F.; Abdullah, A.B.; Yussof, S. Energy-aware fault tolerant task offloading of mobile cloud computing. In Proceedings of the 2017 5th IEEE International Conference on Mobile Cloud Computing, Services, and Engineering (MobileCloud), San Francisco, CA, USA, 6–8 April 2017; pp. 161–164.
53. Park, J.; Yu, H.; Kim, H.; Lee, E. Dynamic group-based fault tolerance technique for reliable resource management in mobile cloud computing. *Concurr. Comput. Pract. Exp.* **2016**, *28*, 2756–2769. [CrossRef]
54. Al-Sayed, M.M.; Khattab, S.; Omara, F.A. Prediction mechanisms for monitoring state of cloud resources using Markov chain model. *J. Parallel Distrib. Comput.* **2016**, *96*, 163–171. [CrossRef]
55. Keshanchi, B.; Souri, A.; Navimipour, N.J. An improved genetic algorithm for task scheduling in the cloud environments using the priority queues: Formal verification, simulation, and statistical testing. *J. Syst. Softw.* **2017**, *124*, 1–21. [CrossRef]
56. Peng, H.; Wen, W.-S.; Tseng, M.-L.; Li, L.-L. Joint optimization method for task scheduling time and energy consumption in mobile cloud computing environment. *Appl. Soft Comput.* **2019**, *80*, 534–545. [CrossRef]
57. Tang, C.; Hao, M.; Wei, X.; Chen, W. Energy-aware task scheduling in mobile cloud computing. *Distrib. Parallel Databases* **2018**, *36*, 529–553. [CrossRef]
58. Lin, X.; Wang, Y.; Xie, Q.; Pedram, M. Energy and performance-aware task scheduling in a mobile cloud computing environment. In Proceedings of the 2014 IEEE 7th International Conference on Cloud Computing, Anchorage, AK, USA, 27 June–2 July 2014; pp. 192–199.
59. Guo, S.; Xiao, B.; Yang, Y.; Yang, Y. Energy-efficient dynamic offloading and resource scheduling in mobile cloud computing. In Proceedings of the IEEE INFOCOM 2016—The 35th Annual IEEE International Conference on Computer Communications, San Francisco, CA, USA, 10–14 April 2016; pp. 1–9.
60. Wei, X.; Fan, J.; Lu, Z.; Ding, K. Application scheduling in mobile cloud computing with load balancing. *J. Appl. Math.* **2013**, *2013*, 409539. [CrossRef]