Directory-based incentive management services for ad-hoc mobile clouds

Abdullah Yousafzai[1, 2,*], Victor Chang[3,] Abdullah Gani[1, 2, **], Rafidah Md Noor[1]

[1]Faculty of Computer Science and Information Technology, University of Malaya (UM), 50603 Kuala Lumpur, Malaysia

[2]Center for Mobile Cloud Computing Research, University of Malaya (UM), 50603 Kuala Lumpur, Malaysia

[3]International Business School Suzhou, Xi'an Jiaotong Liverppol University, Suzhou, China

[*]Correspondence Email: abdullahyousafzai@siswa.um.edu.my, Tel: +6073967786

[**]Correspondence Email: abdullah@um.edu.my, Tel: +60129156105

Abstract

Mobile cloud computing is envisioned as a promising approach to augment the computational capabilities of mobile devices for emerging resource-intensive mobile applications. This augmentation is generally achieved through the capabilities of stationary resources in cloud data centers. However, these resources are mostly not free and sometimes not available. Mobile devices are becoming powerful day by day and can form a self-organizing mobile ad-hoc network of nearby devices and offer their resources as on-demand services to available nodes in the network. In the ad-hoc mobile cloud, devices can move after consuming or providing services to one another. During this process, the problem of incentives arises for a node to provide service to another device (or other devices) in the network, which ultimately decreases the motivation of the mobile device to form an ad-hoc mobile cloud. To solve this problem, we propose a directory-based architecture that keeps track of the retribution and reward valuations (in terms of energy saved and consumed) for devices even after they move from one ad-hoc environment to another. From simulation results, we infer that this framework increases the motivation for mobile devices to form a self-organizing proximate mobile cloud network and to share their resources in the network.

Keywords*: Ad-Hoc Mobile Cloud; Mobile Cloud Computing; Incentives; Directory Service; Applications and Business Prospects.*

1. Introduction

Smart mobile devices are gaining enormous popularity, and new smart mobile applications, such as natural language processing, face recognition, interactive gaming, and augmented reality, are rising and attracting great attention (Cohen, 2008; Soyata, Muraleedharan, Funai, Kwon, & Heinzelman, 2012). These smart mobile applications are typically resource-intensive and demand thorough computation and high energy. Owing to the physical size constraint, mobile devices generally have limited computational resources and scarce battery life. Thus, the tension between resource-intensive applications and resource-constrained mobile devices poses a significant challenge for the future mobile platform development (Cuervo et al., 2010).

Mobile cloud computing (MCC) is envisioned as a promising approach to address such a challenge by offloading the extensive computation via wireless access (cell network, Wi-Fi, or Bluetooth) to the resource-rich cloud infrastructure to extend the device battery lifetime and to improve user response time. In MCC, the ecosystem of computational offloading contains different resources and may use any of them depending on availability and scheduling decisions. These resources are presented in a generic MCC diagram shown in Figure 1. The most common resources available and used are public cloud service providers (CSPs such as Amazon EC2 and Google Compute Cloud), which provide on-demand services (software, platform, and infrastructure) to the mobile applications using a pay-as-you-go model. The second type of resource is the cloudlet (Satyanarayanan, Bahl, Caceres, & Davies, 2009), a nearby (local) Internet-enabled rich-computing infrastructure that is connected to mobile devices through wireless access. The third type of resource is the local proximate mobile cloud, which is based on the formation of an ad-hoc network of devices within the vicinity to collectively serve one another by either Wi-Fi or Bluetooth network interfaces.
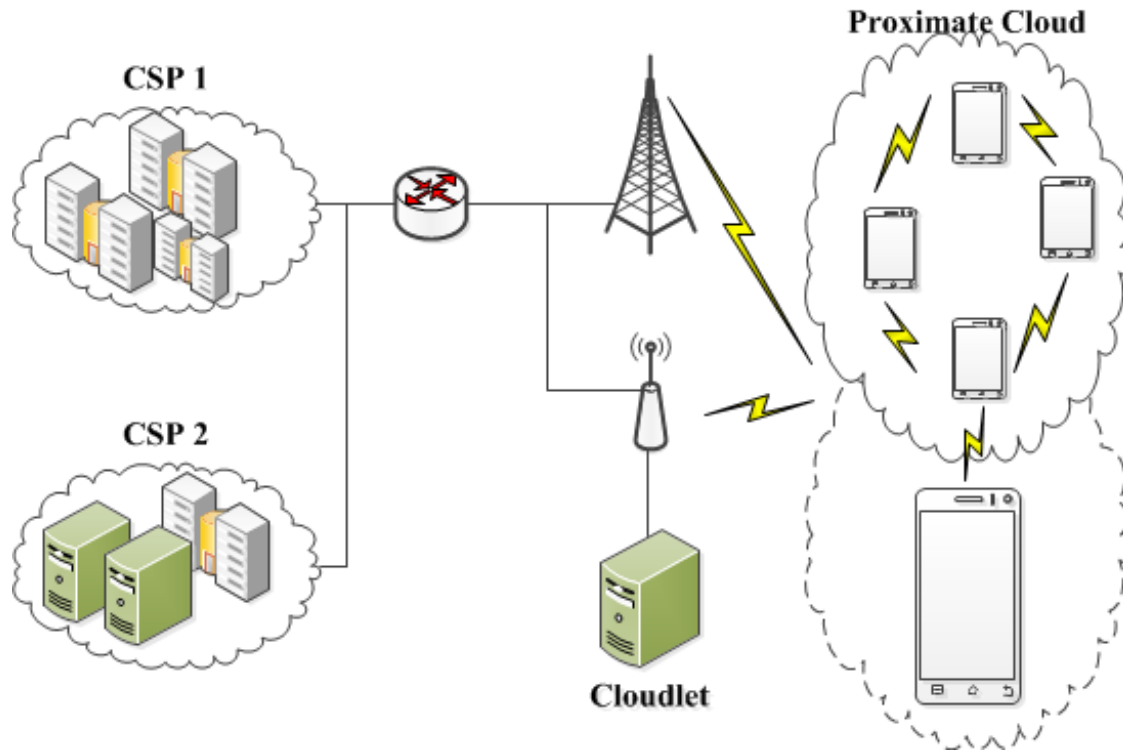
Figure 1. Generic View of Mobile Cloud Environment

Emerging cloud services (Sharma, Chang, Tim, Wong, & Gadia, 2016) provided by public clouds are not free, and the cloudlet is not always available. To augment the capabilities of smartphones, they still can form a proximate mobile cloud environment of nearby mobile devices. The problem with the devices in a proximate mobile cloud environment is their mobility, which raises questions about how to provide incentives to devices that are sharing their resources. This incentive problem also decreases the motivation for devices to join a self-organizing proximate mobile cloud environment and to provide and consume services to and from other devices. To solve the retribution and reward problem for devices in a proximate mobile cloud environment, we propose a directory-based framework that mitigates the issue of incentives to the nodes in an ad-hoc environment, even after their movement from one environment to another. The assurance of the incentives to devices motivates the devices to join a self-organizing proximate mobile cloud environment.

In this article, we interchangeably use the terms proximate mobile cloud and mobile ad-hoc cloud because they represent the same concept. To the best of our knowledge, this is the first attempt to solve the incentive problem in an ad-hoc cloud environment using a third-party settlement. The remainder of this article is divided into six sections. Section 2 presents the motivation for this research based on a brief summary of related works on the incentive mechanisms for an ad-hoc mobile cloud environment. Section 3 describes the problem. Section 4 provides the details of the functional components of the frameworks along with their interaction and flow of communication. Section 5 presents the results obtained b simulating the scenario presented in this paper and provides a discussion of the data collected using experiments. Section 6 indicates the application and business prospects that can be envisioned through the adaptability and tailoring of the proposed framework. Finally, Section 7 concludes this paper as well as presents the limitations and possible future research directions using the proposed framework.

2. Related Work

a) MCC

Research on MCC is centered on augmenting the computational and storage capabilities of mobile devices using the services offered by remote computing infrastructure (Karthik, 2010). A dynamic computational offloading algorithm for a mobile cloud environment was proposed by (Huang, Wang, & Niyato, 2012) to assist mobile device users in determining which part of a mobile application to be executed remotely. (Barbera, Kosta, Mei, & Stefa, 2013) evaluated the energy cost and bandwidth of computational offloading in MCC. (Zhang, Wen, & Wu, 2013) explored the minimum-energy task-scheduling problem by employing a Markovian chain model. (Chen, 2015) designed a decentralized computational offloading game for decision making, in which multiple mobile users were multiplexed over a common infrastructure. These preceding works have assumed the case of augmentation of mobile devices through public cloud providers, which may not be true for some mobile users and mobile environments. (Zhou, Dastjerdi, Calheiros, Srirama, & Buyya, 2015) envisioned a mobile cloud environment that comprises multiple cloud resources,

such as remote public clouds, cloudlet, and mobile ad-hoc cloud, to provide adaptive and context-aware mobile cloud services.

Some of the recent research has utilized nearby smart mobile devices to form a mobile cloud that allows avoiding a connection to remote cloud (and public cloud) while upholding the main benefits of computational offloading (Al Noor, Hasan, & Haque, 2014; Fernando, Loke, & Rahayu, 2011; Y. J. Li, Sun, & Wang, 2014; Penner, Johnson, Slyke, Guirguis, & Gu, 2014).

(Huerta-Canepa & Lee, 2010) proposed a framework in which multiple mobile devices situated nearby could form a cloud and cooperatively execute the workload. (Fernando et al., 2011) investigated the viability of a mobile cloud framework to utilize the resources of local devices to address the resource-sparseness problem. Transient clouds, a collaborative computing platform proposed by (Penner et al., 2014), enable nearby smart mobile devices to form an ad-hoc network and to provide resources as a service. (Y. J. Li et al., 2014) proposed resource-discovery schemes in proximity and allocated tasks to discovered resources periodically.

b) Incentive Mechanism:

Although the MCC frameworks cited are promising, the deficiency of proper incentive mechanisms will fail to motivate users to share their resources in an ad-hoc mobile cloud environment (Tang, He, & Li, 2016). Limited research has been commenced to address the problem of designing an incentive mechanism for mobile cloud (Al Noor et al., 2014). Compared with the traditional public CSP, the billing/incentive mechanism for a mobile device in a mobile ad-hoc cloud is more difficult because of the decentralized nature of administration and working. Considerable works on the incentive mechanism for mobile ad-hoc networks, such as (Crowcroft, Gibbens, Kelly, & Ostring, 2004; Gobel & Krzesinski, 2008; Z. Li & Shen, 2012; Lu, Lin, Zhu, Shen, & Preiss, 2010), have been conducted. However, a subtle difference exists between a mobile ad-hoc cloud and a mobile ad-hoc network, i.e., an ad-hoc network normally shares network resources to relay network traffic in relevant situations, whereas a mobile ad-hoc cloud shares

computational resources to save energy and, in some cases, to improve response time. Consequently, the incentives schemes (Crowcroft et al., 2004; Gobel & Krzesinski, 2008; Z. Li & Shen, 2012; Lu et al., 2010) for ad-hoc networks are unsuitable for ad-hoc clouds. (Miluzzo, Cáceres, & Chen, 2012) analyzed possible incentive mechanisms in which mobile devices were connected in a master–slave fashion in a proximate ad-hoc mobile cloud environment. (Al Noor et al., 2014) proposed CellCloud and developed a reputation-based incentive model based on economic theory to compensate mobile device users for the provisioning of their resources to other devices. (Tang et al., 2016) proposed and analyzed a double-sided bidding mechanism and considered the competition on resource demanding and supplying sides in the mobile cloud.

However, the existing works on incentive management in a mobile cloud environment do not focus on the mobility and movement of devices after successful/unsuccessful offloading transactions. Another problem with the existing incentive approaches in the mobile cloud is the difficulty in billing mobile devices. The difficulty is due to the constraint and nature of the ad-hoc mobile cloud, such as changing connections, carriers, and devices.

3. *Problem Description*

In this section, we introduce a system model for a proximate mobile cloud environment. We consider a set of collocated mobile device users, and some of them have a computationally intensive task to be completed to save their battery consumption. A wireless access base station exists, through which the mobile device users are connected with one another or the devices are connected with one another using peer-to-peer Bluetooth connections. Similar to many previous studies in MCC (Barbarossa, Sardellitti, & Di Lorenzo, 2013; Wen, Zhang, & Luo, 2012) and mobile networking (S. L. Wu, Tseng, Lin, & Sheu, 2002; Y. N. Wu, Chou, & Kung, 2005), we consider a quasi-static scenario to enable a tractable analysis and to obtain useful insights. In the scenario, the set of mobile device users N remains unchanged during a computation offloading transaction, but it may change across different transactions. This scenario is presented in Figure 2.
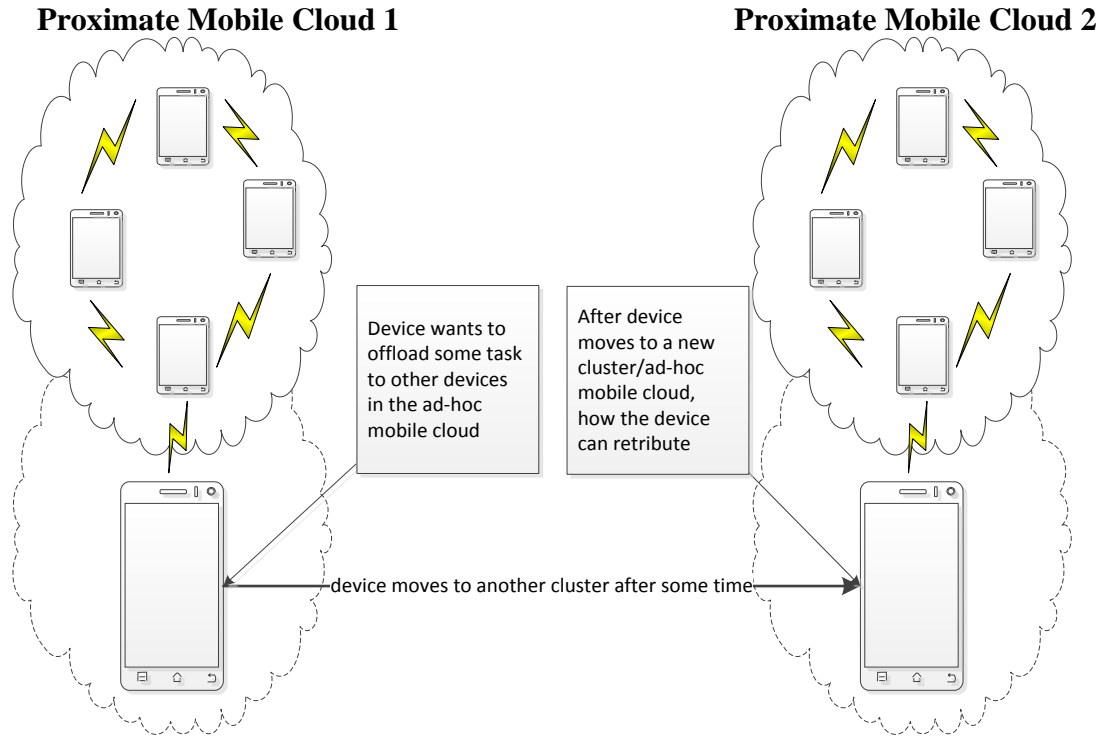
Figure 2. Abstraction of reward/retribution in case a device moves from one location to another.

From Figure 2, we consider mobile device M that aims to offload some of its tasks to nearby available mobile device(s). In this regard, mobile device M starts negotiating with other device(s). Device D is assumed to confirm to execute some tasks of device M. By agreement, device D utilizes its computational and energy resources to execute the tasks offloaded by device M. Device M then moves from its current location to another location. In this scenario, two problems arise, namely, (i) how device D will acquire benefit/reward for sharing its computational resources and for its energy depletion caused by the execution for other devices, and (ii) how device M will pay the retribution in the form of computational resources, energy, or monetary bills. The case of monetary bills is out of the question and is not within the scope of this study because providing monetary billing procedures for such self-organizing systems is difficult. However, the reward and retribution of mobile devices can be solved by involving a centralized entity that

keeps track of the debts and credits of the energy valuations spent for other and saved by executing on other devices. Through this method, a device can keep its balance level and use and provide services according to the valuations it has on the central directory.

4. Proposed Framework

In a static environment where mobile devices do not move, reward and retribution can be performed efficiently using a barter form of agreement for cooperating with one another to maximize the net utility. However, while the devices move with one another, they cannot form a barter agreement. To solve this issue, we design a conceptual framework based on an agent-oriented methodology. A bird's-eye view of the framework is presented in Figure 3. The proposed framework introduces the concept of a third-party transitive barter system in which a device generates compensation from other devices. We assume that the network traffic from the mobile devices to the directory is free of monetary cost. Proper security and trust mechanisms are needed to facilitate the adoption of this framework. We consider that the devices are trusted devices in which any third-party computation is executed within well-defined and secure environments (Cox & Chen, 2007; Miluzzo et al., 2012). Any device-to-device and device-to-directory interactions should be regulated by off-the-shelf authentication and authorization techniques (Chang, Ramachandran, Yao, Kuo, & Li, 2016).

A device that aims to take advantage of the framework will be registered on the directory. A newly registered device obtains a premium capital of served energy. The initial energy capital is given, so that the newly registered devices can actually start working in the framework lifecycle. In the next subsection, we present some details on the structure and locality of the directory used for record keeping of the energy valuations of the devices in question.
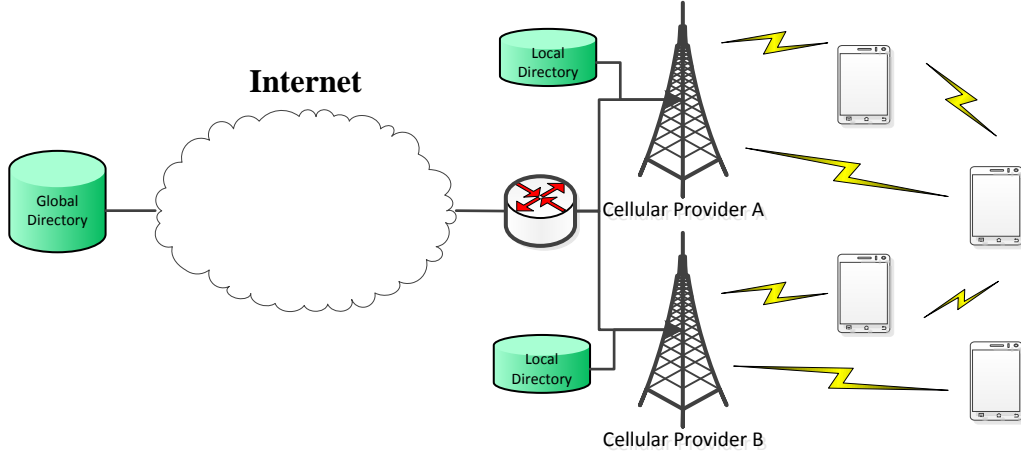
Figure 3. Bird's-eye view of proposed framework.

a) Global and Local Directories

A global directory will be available through the Internet to all computing devices that are registered with the directory, and the content of this directory will be updated by the member mobile devices at specific intervals. Thus, the latest information is available for every device registered in the directory. The global directory database contains a table with fields [$udid$, $current\_location$, $energy\_served$, $current\_willingness$, $specs$, $current\_battery$, $current\_no\_of\_user$].

The description of each of these fields is as follows:

1. $udid$: unique device ID.

2. $current\_location$: current location in terms of latitude and longitude or any other localization primitive of a mobile device that will be used as a key to form a local proximate cloud environment and an offloading device list.

3. $energy\_served$: the energy a device consumes while serving and performing computation for other devices; every time a device consumes E energy units for other devices, the field is incremented with E energy units. If a device utilized some E energy units of other device, this

field will be decremented by E energy units. Once a new device is registered, a premium capital is added to this field, such that the device can utilize other devices.

4. *current_willingness* : this Boolean field represents whether a device at current time is willing to entertain other nearby devices by executing some of their tasks.

5. *specs* : the specifications (MIPS, memory, storage, and OS version) of a mobile device.

6. *current_battery* : the current battery level of a device.

7. *current_number_of_user* : indicates how many users a device is currently serving.

A local directory cache can be installed at each cellular provider infrastructure to improve the response time for directory lookup.

b) Components Needed by Framework on Mobile Devices

A mobile device can be either in a role of offloading service consumer or provider. Besides the role of the mobile device, all the components/agents presented in Figure 4 should be available on the mobile device registered with the directory. As stated, the framework is based on an agent-oriented methodology. Agents work atomically and interact with one another to improve the total welfare of every mobile device. How different agents in and across different mobile devices interact with one another in an offloading transaction is also presented in Figure 4. In this figure, the arrows are labeled with the number and description of the step involved in an offloading transaction between two devices.
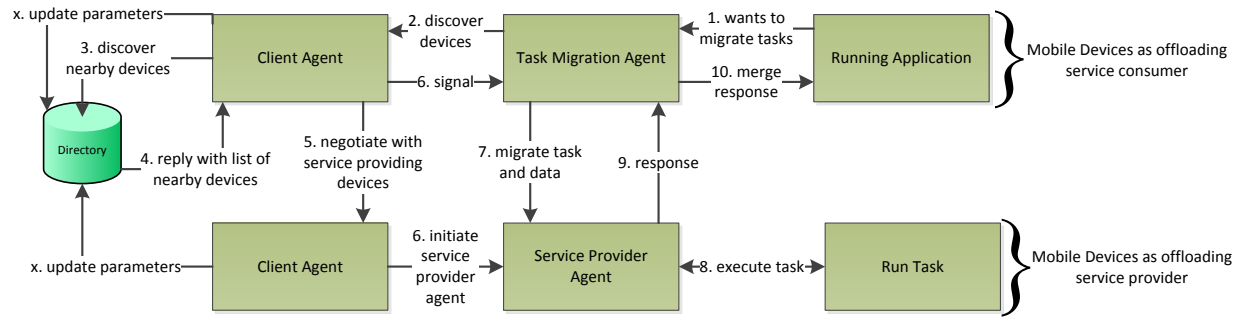
Figure 4. Interaction of agents in the framework

c) Task Migration Agent

When an application aims to migrate/offload some of its executing tasks to other nearby devices, the task migration agent signals the "client agent" to discover nearby mobile devices. The task migration agent is implemented as an add-on to the application runtime environment in Android. Whenever a method call is encountered with offload-able annotations, the task migration agent starts its actions for offloading the execution of the annotated method to a nearby device. To initiate the offloading process, the task migration agent checks whether an established connection with any service-providing agent exists. If no such connection exists, the task migration agent sends a discovery signal to the client agent and waits for a response from the client agent. The response signal from the client agent contains the connection information of the offloading service provider device. Using the established connection, the task migration agent sends the task and its associated data (if any) to the service-providing device for remote execution and waits for its response. If the task can be divided into subtasks, parallel processing can also be performed.

d) Client Agent

The client agent is an application that will be deployed on all mobile devices, which are a part of the proximate mobile cloud environment and will execute in background mode. The agent is also responsible

for communicating with the directory and negotiating with possible service-providing and service-consuming nearby devices. The agent periodically and asynchronously updates the directory with the parameters stored in the directory. The parameters to be gathered are the current location (latitude and longitude), CPU and memory utilizations, battery status, bandwidth, and service parameters, i.e., service consumption and providing time and energy. Besides the communication with the directory, the functionality of the client agent varies depending on the role of the mobile device. This functionality is explained in the following subsection.

e) Functionality of Client Agent on Offloading Service-consuming Device

Upon receiving a discover request from the task migration agent, the client agent sends a discovery query to the directory. The directory responds to this signal by sending a list of available nearby devices based on the device current location. The device information sent by the directory contains knowledge about the device specification, current battery level, and any other related information. Such information from devices will be used to select the best device to start negotiating with. The decision of the best can be simply done using any decision formulation (Karthik, 2010).

f) Functionality of Client Agent on Offloading Service-providing Device

On the service-providing device, the client agent is responsible for negotiating with the consuming device and can reject the proposal of the consuming device based on the perception available in the directory. The perception (how much energy the requesting device has served for other devices) is necessary for the reward and retribution of mobile devices for providing and consuming computational services to and from other devices. In an emergency situation, the perception is not considered and the proposal is accepted immediately.

g) Service Provider Agent

Service provider agent, once initiated, becomes the end-point where the offloaded method execution occurs. After the service method execution is completed, the client agent of both devices updates the directory about the service time and energy valuations. The service time and energy consumption are updated, so that the valuation can be incremented or decremented from the device record to perform the reward and retribution.

5. *Discussion*

To verify the effectiveness and feasibility of the proposed framework, we construct the simulation environment in such a way that after every successful offloading transaction, one or both of the devices move/disconnect from each other and appear in another cluster/ad-hoc mobile network setting. In the experiment, we conduct consecutive offloading transactions to show the variation in the energy valuation at the directory for each device. The initial capital of $energy\_served$ to all devices at the start of the experiment is given as 1000000. This value is based on the setting we used and from the value of fitness function of the decision formulation.

After each offloading transaction, each device updates the results at the central directory for their retribution and reward energy valuations. At this point, the devices can provide the directory with false information to increase their valuations. However, the directory should crosscheck such valuations by matching the data provided by both devices. After one successful offloading transaction, the updated results of the retribution and reward subtracted and added from the initial capital for both devices involved are presented in Figure 5. In this figure, device 1 is the device that offloads its task, and device 2 is the device to which the task from device 1 is offloaded. According to the result presented in Figure 5, the initial capital for both devices is the same. However, after the offloading transaction and applying the retribution and reward mechanism, the capital values change for both devices.
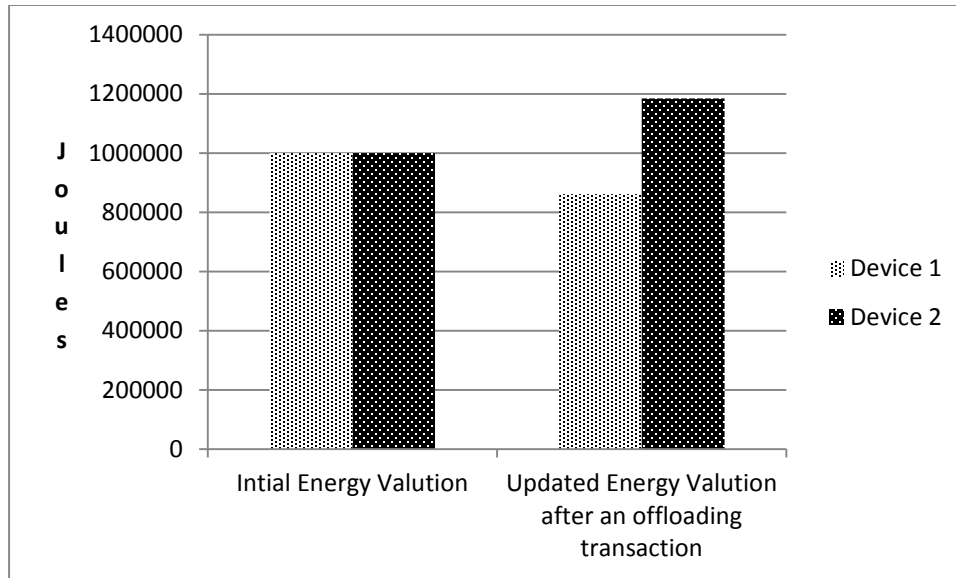
Figure 5 Change in capital value after one successful offloading transaction

On the other hand, if device 1 does not serve the requests of other devices and is only rational toward saving its energy, then after some successful offloading transactions, the device energy capital at the directory becomes zero or less than zero, which results in no cooperation from other devices. This phenomenon is presented in Figure 6, i.e., the energy valuation at the directory for device 1 at the start of every offloading transaction. From the start of the ninth transaction, the valuation is already in negative. Thus, other devices may reject its offloading request based on its rational behavior.
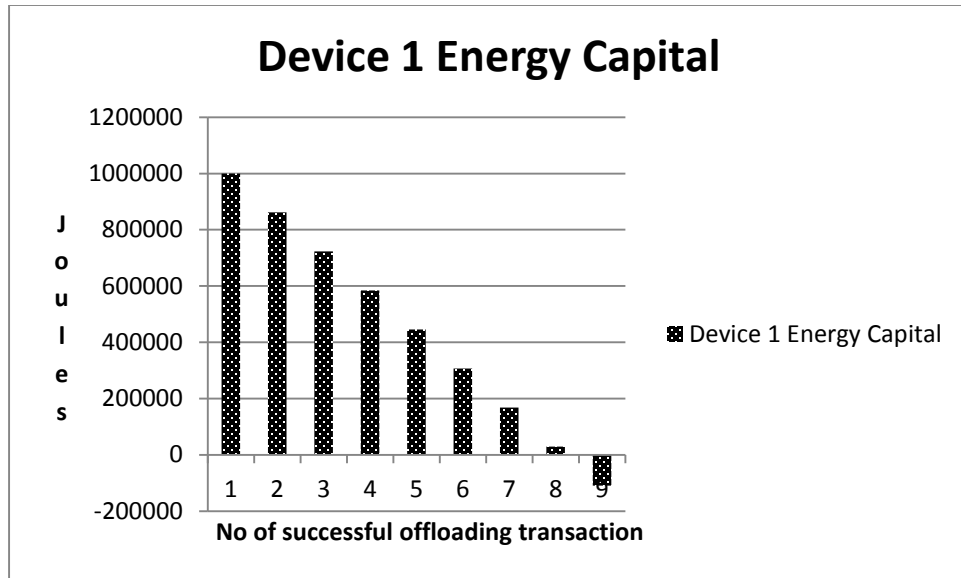
Figure 6 Changes in energy valuation after completion of each offloading transaction at directory for device 1 while only using other devices and not providing services to others

However, if the device is also serving other devices along with using them, then the balance is maintained. This phenomenon is presented in Figure 7. Even after 20 transactions, the valuation is not zero. From this behavior, we infer that the framework may help increase the motivation of mobile devices to form an ad-hoc mobile cloud.
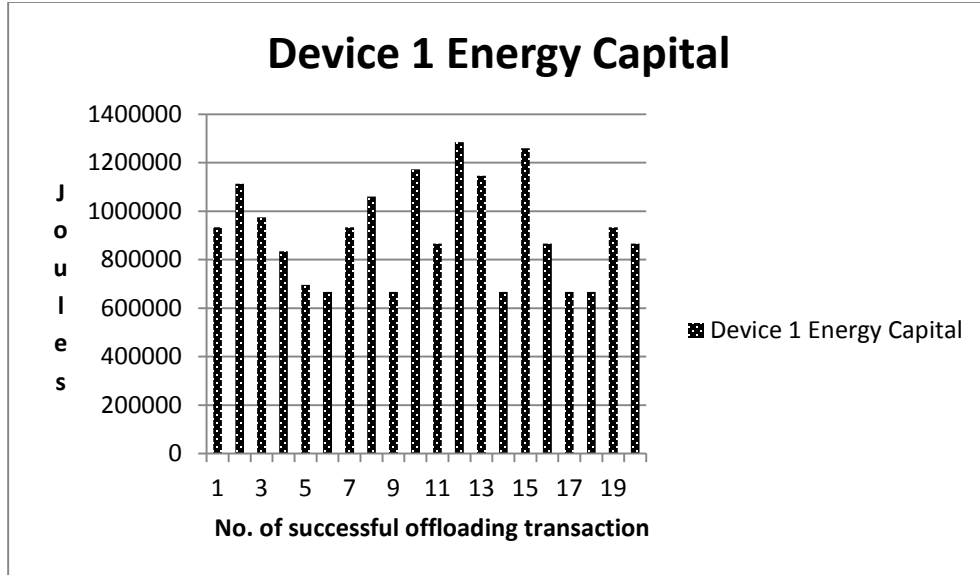
Figure 7 Changes in energy valuation after completion of each offloading transaction at directory for device 1 while using and providing services to other devices

6. *Applications and Business Prospects*

The proposed framework can open another venue of income for mobile network operators (MNOs) by provisioning directory services through a subscription-based mechanism bounded by a time frame. MNOs can further federate the directories with one another so that true ubiquitous ad-hoc mobile cloud formation and incentive service can be envisioned. Another deployment strategy from business perspectives is for edge network operators that can utilize it according to their business policy.

The framework can also increase the accessibility of mobile services and increase the revenue of mobile service providers by reducing the cost of computation on the server side through opportunistic local offloading aided by the directory services. The framework can also be utilized by federated cloud infrastructures based on the energy valuation and third-party settlement concept presented with this framework. Implementation and tailoring of the framework is expected to reduce the number of monetary transactions and their management among cloud providers.

Many other applications and business opportunities revolving around the concept of the proposed framework can be utilized. Another example is in analytics and data science (Hashem et al., 2016; Larson & Chang, 2016) based on mobile crowd sensing, in which the directory services can be improved by crediting the energy valuation in the directory for an individual device. Big-data applications (Newman, Chang, Walters, & Wills, 2016; Yaqoob et al., 2016) can employ a tailored version of the proposed agent-oriented framework following the rules and incentive unit of the environment.

7. *Conclusion and Future Research Directions*

In this paper, we present a framework to improve the motivation of mobile devices in forming a mobile ad-hoc cloud. The framework utilizes the principle of retribution and reward for services based on third-party settlements. The experimental results show that mobile device users may be motivated to become part of the ad-hoc formation by maintaining a balance between service consumption and provision.

However, one of the obvious drawbacks of the proposed framework is that when the device energy valuation at the directory is zero, the device cannot consume service from other devices. Another possible direction of investigation is the effect of the mobile application provided through the communication of agents with one another and the directory on the quality of service/quality of experience. The amount of energy consumed by the agents themselves should also be determined. Furthermore, the proposed ad-hoc mobile cloud formation architecture must be adopted in systematic ways, such as that presented by (Chang, 2015) to widen their acceptability.

References

Al Noor, S., Hasan, R., & Haque, M. M. (2014, 2014). *Cellcloud: A novel cost effective formation of mobile cloud based on bidding incentives*.

Barbarossa, S., Sardellitti, S., & Di Lorenzo, P. (2013). Joint allocation of computation and communication resources in multiuser mobile cloud computing. *2013 Ieee 14th Workshop on Signal Processing Advances in Wireless Communications (Spawc)*, 26-30.

Barbera, M. V., Kosta, S., Mei, A., & Stefa, J. (2013). To Offload or Not to Offload? The Bandwidth and Energy Costs of Mobile Cloud Computing. *2013 Proceedings Ieee Infocom*, 1285-1293.

Chang, V. (2015). *A proposed cloud computing business framework*: Nova Science Publisher.

Chang, V., Ramachandran, M., Yao, Y., Kuo, Y.-H., & Li, C.-S. (2016). A resiliency framework for an enterprise cloud. *International Journal of Information Management, 36*(1), 155-166. doi: http://dx.doi.org/10.1016/j.ijinfomgt.2015.09.008

Chen, X. (2015). Decentralized Computation Offloading Game for Mobile Cloud Computing. *Ieee Transactions on Parallel and Distributed Systems, 26*(4), 974-983. doi: 10.1109/Tpds.2014.2316834

Cohen, J. (2008). *Embedded speech recognition applications in mobile phones: Status, trends, and challenges.* Paper presented at the Acoustics, Speech and Signal Processing, 2008. ICASSP 2008. IEEE International Conference on.

Cox, L. P., & Chen, P. A. (2007). Pocket hypervisors: Opportunities and challenges. *Eighth Ieee Workshop on Mobile Computing Systems and Applications, Proceedings*, 46-50.

Crowcroft, J., Gibbens, R., Kelly, F., & Ostring, S. (2004). Modelling incentives for collaboration in mobile ad hoc networks. *Performance Evaluation, 57*(4), 427-439. doi: 10.1016/j.peva.2004.03.003

Cuervo, E., Balasubramanian, A., Cho, D.-k., Wolman, A., Saroiu, S., Chandra, R., & Bahl, P. (2010). *MAUI: making smartphones last longer with code offload.* Paper presented at the Proceedings of the 8th international conference on Mobile systems, applications, and services.

Fernando, N., Loke, S. W., & Rahayu, W. (2011, 2011). *Dynamic mobile cloud computing: Ad hoc and opportunistic job sharing*.

Gobel, J., & Krzesinski, A. E. (2008). Modelling Incentives and Protocols for Collaboration in Mobile Ad Hoc Networks. *Mswim'08: Proceedings of the Eleventh Acm International Conference on Modeling, Analysis, and Simulation of Wireless and Mobile Systems*, 78-85.

Hashem, I. A. T., Chang, V., Anuar, N. B., Adewole, K., Yaqoob, I., Gani, A., . . . Chiroma, H. (2016). The role of big data in smart city. *International Journal of Information Management, 36*(5), 748-758. doi: http://dx.doi.org/10.1016/j.ijinfomgt.2016.05.002

Huang, D., Wang, P., & Niyato, D. (2012). A Dynamic Offloading Algorithm for Mobile Computing. *Ieee Transactions on Wireless Communications, 11*(6), 1991-1995. doi: 10.1109/Twc.2012.041912.110912

Huerta-Canepa, G., & Lee, D. (2010). *A virtual cloud computing provider for mobile devices*. Paper presented at the Proceedings of the 1st ACM Workshop on Mobile Cloud Computing & Services: Social Networks and Beyond, San Francisco, California.

Karthik, K. (2010). Cloud Computing for Mobile Users: Can Offloading Computation Save Energy?, *43,* 51-56.

Larson, D., & Chang, V. (2016). A review and future direction of agile, business intelligence, analytics and data science. *International Journal of Information Management, 36*(5), 700-710. doi: http://dx.doi.org/10.1016/j.ijinfomgt.2016.04.013

Li, Y. J., Sun, L., & Wang, W. Y. (2014). Exploring Device-to-Device Communication for Mobile Cloud Computing. *2014 Ieee International Conference on Communications (Icc)*, 2239-2244.

Li, Z., & Shen, H. Y. (2012). Game-Theoretic Analysis of Cooperation Incentive Strategies in Mobile Ad Hoc Networks. *Ieee Transactions on Mobile Computing, 11*(8), 1287-1303. doi: 10.1109/Tmc.2011.151

Lu, R. X., Lin, X. D., Zhu, H. J., Shen, X. M., & Preiss, B. (2010). Pi: A Practical Incentive Protocol for Delay Tolerant Networks. *Ieee Transactions on Wireless Communications, 9*(4), 1483-1493. doi: 10.1109/Twc.2010.04.090557

Miluzzo, E., Cáceres, R., & Chen, Y.-F. (2012). *Vision: mClouds-computing on clouds of mobile devices.* Paper presented at the Proceedings of the third ACM workshop on Mobile cloud computing and services.

Newman, R., Chang, V., Walters, R. J., & Wills, G. B. (2016). Model and experimental development for Business Data Science. *International Journal of Information Management, 36*(4), 607-617. doi: http://dx.doi.org/10.1016/j.ijinfomgt.2016.04.004

Penner, T., Johnson, A., Slyke, B. V., Guirguis, M., & Gu, Q. (2014, 8-12 Dec. 2014). *Transient clouds: Assignment and collaborative execution of tasks on mobile devices.* Paper presented at the 2014 IEEE Global Communications Conference.

Satyanarayanan, M., Bahl, P., Caceres, R., & Davies, N. (2009). The Case for VM-Based Cloudlets in Mobile Computing. *Ieee Pervasive Computing, 8*(4), 14-23.

Sharma, S., Chang, V., Tim, U. S., Wong, J., & Gadia, S. (2016). Cloud-based emerging services systems. *International Journal of Information Management.* doi: http://dx.doi.org/10.1016/j.ijinfomgt.2016.03.006

Soyata, T., Muraleedharan, R., Funai, C., Kwon, M., & Heinzelman, W. (2012). *Cloud-vision: Real-time face recognition using a mobile-cloudlet-cloud acceleration architecture.* Paper presented at the Computers and Communications (ISCC), 2012 IEEE Symposium on.

Tang, L., He, S., & Li, Q. (2016). Double-sided Bidding Mechanism for Resource Sharing in Mobile Cloud. *IEEE Transactions on Vehicular Technology, PP*(99), 1-1. doi: 10.1109/TVT.2016.2565505

Wen, Y. G., Zhang, W. W., & Luo, H. Y. (2012). Energy-Optimal Mobile Application Execution: Taming Resource-Poor Mobile Devices with Cloud Clones. *2012 Proceedings Ieee Infocom*, 2716-2720.

Wu, S. L., Tseng, Y. C., Lin, C. Y., & Sheu, J. P. (2002). A multi-channel MAC protocol with power control for multi-hop mobile ad hoc networks. *Computer Journal, 45*(1), 101-110. doi: DOI 10.1093/comjnl/45.1.101

Wu, Y. N., Chou, P. A., & Kung, S. Y. (2005). Minimum-energy multicast in mobile ad hoc networks using network coding. *Ieee Transactions on Communications, 53*(11), 1906-1918. doi: 10.1109/Tcomm.2005.857148

Yaqoob, I., Chang, V., Gani, A., Mokhtar, S., Hashem, I. A. T., Ahmed, E., . . . Khan, S. U. (2016). Information fusion in social big data: Foundations, state-of-the-art, applications,

challenges, and future research directions. *International Journal of Information Management*. doi: http://dx.doi.org/10.1016/j.ijinfomgt.2016.04.014

Zhang, W. W., Wen, Y. G., & Wu, D. O. (2013). Energy-efficient Scheduling Policy for Collaborative Execution in Mobile Cloud Computing. *2013 Proceedings Ieee Infocom*, 190-194.

Zhou, B., Dastjerdi, A. V., Calheiros, R., Srirama, S., & Buyya, R. (2015). mCloud: A Context-aware Offloading Framework for Heterogeneous Mobile Cloud. *IEEE Transactions on Services Computing, PP*(99), 1-1. doi: 10.1109/TSC.2015.2511002