Enabling Mobile Cloud Wide Spread through an Evolutionary Market-based Approach

Cristian Chilipirea, Andreea-Cristian Petre, Ciprian Dobre, Florin Pop

***Abstract*—Mobile Clouds are an ongoing research topic that has yet to become ubiquitous as the now popular cloud paradigm. This is because of a number of issues with mobile clouds that still need to be addressed such as: Incentives, Security, Privacy, Context, Data Management, Usability, Cost benefits. Out of these issues the most important one that needs to be addressed is the issue of incentives, without which mobile clouds cannot gain enough users for the concept to be useful. Unlike public, company owned cloud systems, in mobile clouds the amount of resources or processing power is directly dependent on mobile cloud users that are in the proximity of the individual that requires extra resources. With an increase in the number of mobile cloud users willing to share resources or willing to use the service offered by others, comes an increase in the likeliness that enough mobile cloud enabled devices will be available. In this paper we study incentives for mobile cloud systems and consider as a solution an evolutionary market-based approach to create these incentives. Creating a market for these systems is particularly difficult because of the large number of individuals that need to be involved and their high mobility.**

***Index Terms*—market, mobile cloud, incentives, quality of experience, multimedia.**

I. Introduction

CLOUD computing [3] is now ubiquitous. There are a large number of companies that offer public clouds (such as Amazon EC2 [1] or Microsoft Azure [26]), and numerous open source solutions enable the construction of privately-owned clouds (such as OpenStack [27] or Nimbus [35]) as well as the possibility to integrate them with public clouds, creating hybrid clouds. They have become popular because of the elasticity and ease of use that bring with them important cost reductions for small and medium companies. Elasticity offers users the ability to change the number of used resource depending on real time needs of the application, providing fast, efficient and financially fair access to these resources.

However clouds, in their current form, have some limitations. The geographical distance between the user and the Cloud provider can bring latency delays that are not always acceptable. In specific scenarios, such as Smart Cities or Internet-of-Thing applications relying on large amounts of sensory data, large amounts of data, sometimes redundant, need to be moved between the user and the cloud [7], aggregating data from distributed sources [33]. Moving part of the Cloud storage closer to the user, in local caches, can bring benefits such as faster access to information, multiple stage data filtering, etc. [4][28]. This can also solve the problem of data access for cases where users have limited Internet access (or the networking costs for accessing the cloud can be prohibitive).

Mobile devices are now commonly used, as they support day-by-day activities with access to information anywhere, anytime. But they still have a number of unsolved issues such as: limited energy, low connectivity, heterogeneity of capabilities [31]. These issues limit the possible applications that can be run on the mobile device. Applications can potentially demand high resource availability and responsiveness, even beyond the capability of the available mobile device. Applications that require a large amount of processing, such as natural language processing or image processing, are still not feasible when only considering the limited resources of a single mobile device. Today, such issues are solved by extending the individual capacity of any single mobile device, into the Cloud.

The Mobile Cloud brings a potential solution for the limitations of current Cloud systems and the need of mobile devices for additional resources [14]. Mobile cloud computing is an emerging cloud service model, pushed by the ubiquity of mobile devices and the increase of performance and capabilities of such devices. With Mobile cloud computing, among other paradigms, mobile devices work together and share resources, extending their individual capabilities [14].

We identified 3 possible uses for mobile clouds that could directly benefit from wide spread of mobile clouds and the use of incentives:

A. Boosting entertainment and multimedia quality of experience by accessing multiple computing resources.

Current mobile devices are not capable of offering the entertainment performance characteristics of a desktop pc, a laptop or a gaming console: there is not enough RAM, the processor is slower and the graphics card is not powerful enough.

A person using its mobile device when in transit, going to work or to a different city wants to have the same quality of experience for multimedia and entertainment as the one on his desktop or console. Because he does not have enough resources he could access the compute resources found in the devices of his fellow passengers, which don't need to use the mobile devices to have such high quality entertainment and just use them for example to check e-mail.

In this scenario the individual that expects to have high quality of experience on his mobile device could be willing to pay the individuals that are not using the full computing

C. Chilipirea, A.-C. Petre, C. Dobre, and F. Pop are with University Politehnica of Bucharest, Faculty of Automatic Control and Computers, 313, Splaiul Independentei, 060042, Bucharest, Romania E-mail: cristian.chilipirea@cs.pub.ro, andreea.petre@cti.pub.ro, ciprian.dobre@cs.pub.ro, florin.pop@cs.pub.ro.

Manuscript received December 7, 2014; revised December 27, 2014.

capabilities to gain access to these capabilities. More computing power could translate in better rendering, better artificial intelligence, better path finding, increasing the number of rendered objects and others. Here the system we propose could automatically provide the transactions, distribute the resources and make sure that everyone benefits.

B.Accessing unavailable sensing resources.

Not all mobile devices have the same sets of sensors, for examples some do not have a GPS module. Even in the case where different mobile devices have the same sensors, the quality of the individual sensors can differ dramatically, in the case of two devices that are one next to the other, one can have good GPS signal and identify the location with an error of under a meter, while the other can't detect the satellite signal.

For this scenario Mobile Clouds can be used for sharing the sensing resources, the GPS is only one example, in this scenario we can consider a larger number of sensors, as long as we don't bring privacy concerns, for instance sharing the sensing data of the microphone should not be acceptable. Here the user of the device without the sensor or with a faulty sensor can pay the user with the working sensor for access to the data read by the sensor. It is out of the scope of the paper the guarantee that the data is correct, we only consider how to enable such transactions in a way that it would incentives both users.

C. Information or data access

Mobile Clouds could be used to more efficiently distribute data. In the current Internet architecture the client-server paradigm dominates information transfers, mostly in the form of web pages. Because the client-server paradigm does not scale well, cache nodes and network delivery services have been introduce to compensate, by providing multiple locations from where the data can be downloaded. Caches and network delivery services usually have compute nodes inside Internet service provider clusters. These clusters are rare, only a few of them for each city and because of this the physical distance and the network distance from a user to an ISP cluster is much larger than the distance between two mobile devices' users that share a room or a transportation method.

Instead of downloading the data from a server or from a delivery service, a mobile device user can get the data from another mobile device user that accessed the same information. This method can bring a transfer speed increase for the case of large files and can even offer a financial boost to the buyer if the user that needs the data only has access to a paid cellular data plan.

The main contributions of this paper:

- Present the possible use of incentives to permit the wide spread of mobile clouds.
- Analyze incentives based on an evolutionary market-based approach; the form of incentives is generic and can be represented by monetary exchanges or point systems.
- Give details on the behavior of our solution on an extremely dynamic environment as we have in the case of a Mobile Cloud.
- Present experimental results from a number of traces that show how our solution will behave in a real life scenario and what to expect from implying this kind of incentives to the Mobile Cloud systems.

The rest of the paper is organized as follows. In Section II we present related work, together with details on mobile computing and evolutionary based-market solutions. Section III presents our contribution, with details on how our solution works on an extremely mobile environment that comes with the Mobile Cloud systems. In Section II, we present experimental results obtained in realistic simulations. Section V concludes the paper.

II. Related Work

Our work is based on results shown by Lewis et al. (2010) [22]. The authors propose an algorithm for managing resource allocation in decentralized computational systems, considering an evolutionary market-based approach [22]. We used a similar evolutionary algorithm such that the seller determines an optimal price at which to offer resources. In the original work, the algorithm is tested and compared using multiple types of rational buyer behaviors, and authors show that the price estimation converges regardless of the buyer behavior.

In our work, *we apply Lewis' solution for distributed systems to the particular case of mobile cloud systems, and determine an optimal asking price for resources*. To achieve this, we apply the solution in a highly mobile system with intermittent connections and identifying the effects of such a system on the convergence of the asking price.

begin
  $p_min \longleftarrow 0$;
  $p_max \longleftarrow 500$;
  $alpha \longleftarrow 0.1$;
  $n \longleftarrow 10$;
  $Pop \longleftarrow \{p_i from[p_min, p_max]\}$ randomly such that $|Pop| = n$;
 first
  $askingPrice =$ one value from Pop, until all are used;
 then
  $o \longleftarrow$ new mutated individual based on Pop;
  select random x1, x2, x3, x4 from Pop;
  $x \longleftarrow min(fitness[x1, x2, x3, x4])$;
  $Pop \longleftarrow Pop + o - x$;
  $askingPrice \longleftarrow o$;
end

Algorithm 1: Service Provider Algorithm (based on [22]).

According to the algorithm (see Algorithm 1), a service provider creates a population Pop of prices he wants to process a job. The price population is initialized with random values and when inquired by a service requester, a different node, a value from the initial population is provided. When all values are used a new value is created through mutations based on the current Pop. This new value replaces the least profitable value from Pop, the one with the lowest fitness, the price times the amount requested.

Market-based solutions have been tested in various settings. Hua, Zhuo, and Panwar (2013) demonstrate auction based incentives for femtocell access, in cases where femtocells are used for network access and usage needs to be limited to provide the highest quality to all its users [17]. A similar approach is taken in [21] to control a communication network. We previously demonstrated as well a market-based solution in OpenMobs [8], where broadband Internet connections are being shared dynamically between multiple users, and incentives are given by monetary exchange and reduction in the Internet cost.

In this paper, we analyze the feasibility to enable a Mobile Cloud [14], with mobile resources being shared between several geographically co-located users – in addition to resources that are inside a single location such as a remote server or a public cloud. This is different from the vision of mobile cloud as a system, where the mobile devices are seen as only an interface to access data from a public cloud [11].

The possibility of using mobile devices to do intensive computation has been proved by Dou et al. [12], where they showed that a paradigm such as MapReduce, a method for simplified data processing on large clusters introduced by Dean et al. [10], can be implemented over a network of mobile phones in a scalable and efficient manner.

Mobile Clouds have now a large number of applications such as healthcare presented, as in case of the MoCAsH system [16], or collaboration, like in case of traffic light detection [2]. Another use could be the construction of a mobile multimedia database [20] that would permit access to multimedia content in a fast and efficient manner. A generic Mobile Cloud framework that permits code execution on different mobile devices is presented in [23].

The issue of security in Mobile Clouds has also been thoroughly researched [19]. A framework for secure data processing in these systems is presented in [18], and a survey on mobile cloud computing security is presented in [19].

A large amount of work similar to mobile clouds is done on the concept of offloading and obtaining energy efficiency by doing so [24][6]. Offloading means moving data from mobile devices to the cloud or from the physical infrastructure to a network made by the mobile devices, so that it uses available resources to make the entire network more efficient [25]. The feasibility of offloading in mobile computing with respect to bandwidth and energy costs is studied in [5] [37] [36]. The economics of mobile data offloading are studied by in [15]. The performance evaluation of an offloading framework has been presented by Constandinos et al. (2014), in [8].

Quality of Experience is a novel feature that aims to measure how the users relate to an application; it is best quantified by the ability to receive service and the waiting time for the service to complete [34]. Quality of Experience or QoE is critical for providers as it could affect their revenue. In our case QoE directly affects the number of users that are willing to try a mobile cloud and with this the availability of such a service. A model to evaluate Quality of Experience for online service hosted on Clouds is presented in [30].

Increasing the Quality of Experience in Mobile Clouds has been tried in Cloud2Bubble [9] where they try to offer a unified framework that eases development and brings context to the application. In contrary we believe that the most important impact on Mobile Clouds and the QoE in such systems is given purely by the number of users, which will directly increase availability of such services.

III. Simulation Environment

To enable mobile clouds, we choose as incentives financial transactions between mobile users (i.e., users carrying a portable device). Here we define “buyers” as mobile devices of individuals that are willing to pay to have access to a resource such as a sensor, or to processing time on a different mobile CPU. In contrast, a “seller” is a mobile device of an individual that is willing to share his resources to gain a financial reward. The financial reward can be actual money exchange or a point system for a specific application or existing service. If it is modeled as a point system, we can also imagine the points being used when the seller wants to become a buyer of resources, with a limited number of pre-existing points in the market.

To be able to have a market, we need to achieve fairness. For this, we considered as starting point the method being proposed by Lewis et al. (2010) [22]. The authors propose a market mechanism that requires no coordinating node or complex negotiation. The interaction between self-interested agents is captured through the use of competitive coevolution.

To be able to adapt Lewis’s method, we made several changes to the original proposal. First of all, the evolutionary market algorithm presented for the sellers require that all individuals participate in the market transaction at the same time during a number of iterations.

In our scenario, the nodes that represent buyers and sellers are mobile device carried around by individuals. These mobile devices do not always have Internet access, and due to the movement made by the owners, some devices can encounter others at different time intervals. Usually, only a small number of devices are in wireless communication range of each other. We make here the assumption that devices can connect through a mesh network, which can be used for communication. If this is not possible, devices can only communicate when they have an Internet connection. The advantage is that more devices can communicate at the same time. The disadvantage is the costs of having an Internet connection, the high latency and low bandwidth that can be used between two different devices that could otherwise connect directly. Here, we assume only the scenario where mobile devices can connect through a mesh network and they choose to only request resources from other device that are in Wi-Fi range.

Because of our specific scenario, it is impossible for all devices to communicate at the same time and have iterations in which they all set and correct the price. Connections between individuals are sparse, and very small groups of buyers and sellers connect at the same time. Instead of having steps at which all individuals need to participate, we let all sellers adapt their price individually at their own pace, depending on the connections they make. Each time they are queried for a resource they offer a price and adapt according to the response they receive. This leads to a case where sellers with extremely few connections, few encounters with buyers do not manage to have their asking price converge at all. A small number of connections also means the seller won't be able to make his resources available to enough buyers and will not be able to make a significant profit.

We chose to simulate only the bargain–hunter behavior for the buyers [22]. The bargain hunters are buyers that choose to buy resources at the smallest price available. Because of this, the behavior of the bargain hunter is the most possibly aggressive one – this types of buyers try to make the price on the market go to 0, forcing the sellers to make no profit. If buyers manage to do this and have the price converge to 0, the incentives for using a mobile cloud as a seller disappear and with this the availability of such resources.

To simulate real-life processing, we did not split the processing between two or more sellers. We do this because most processing jobs cannot be split at all. The ones that can be split can be split before the price is being negotiated, and the price can then be negotiate for the smallest undividable processing job.

We make the assumptions that all jobs are equal in size, and that processing time to be able to better analyze the price dynamics in the market. In real life we can assume the existence of a standard processing job that takes x seconds to process. Let us assume x equals 10, and all jobs are of equal length or multiple length of this one. In this way we can compare two different loads and have a global price for job processing. The same method can be applied to any type of resource, for instance when requesting sensor data, a single read from the sensor can be used as a baseline and any requests can be equal to this or its multiples.

Having jobs equal in size and small does bring another advantage. Because of the volatility of the connection between the buyer and the seller, if the job takes a large time frame to be accomplished, the seller can be out of range of the buyer by the time it can return the results. On the other hand if jobs are small it makes it more likely that the seller will be in range when the results can be returned to the buyer.

To test this method, we considered a set of scenarios consisting of randomly generated data and existing real life traces. We list all the scenarios on which we simulated the presented solution, in the followings.

A. Random dataset

This is a simulated trace based on a random uniform distribution. Devices are placed around a square area and they all get a random location and a random direction of movement along with speed. When a device reaches a border of the square area it goes back in the opposite direction. Steps in the simulation take 60 seconds and this is when encounters are measured. An encounter is represented by two or more devices that are within 100 m (Wi-Fi range) of each other.

qIn the simulation we placed 30 of these devices and build a trace that lasts for 30 days.

Unlike real life traces the distribution of encounters (or detections) is very uniform and most devices have a similar number of contacts by the end of the trace. This makes it a good theoretical model of how the algorithms could behave in an ideal scenario and what the limitations of the system are.

A similar solution to the one we used to generate this trace can be identified in the work of Petz, Enderle, and Julien (2009), in their delay tolerant network framework for the evaluation of mobility models [29] .

B. UPB2012 dataset

To create this trace a number of 65 students from University Politehnica of Bucharest had a tracing application installed on their smartphones. The students did their normal activities for a period of 62 days, using their phones as they normally would.

The application recorded different sets of data such as battery status and most importantly for us Bluetooth or Wi-Fi contact as well as social information using Facebook.

The trace is ideal for modeling applications over a set of individuals that have a recurring schedule every day, forming similar connection patterns at all times. This is true for most individuals who work by a fixed schedule and in a fixed group.

C. StAndrews dataset

This is another publicaly-available mobility trace [32], with data collected at the University of St. Andrews trace. The trace was generated using 25 mobile devices distributed among participants, here 22 were undergraduate student, 3 were postgraduate students and 2 were staff members.

This study lasted from 15th of February until 29th of April 2008; period in which the members in the study carried around at all times the mobile devices. The devices themselves then recorded when they were in proximity of one another.

We should notice here that the devices were with their owners even during nighttime and that most owners lived in the student housing, in signal range from one another, thus many detections occur even during the nighttime.

D. Cambridge-Infocom dataset

The trace of Infocom is different from most others because of the limited time frame and specific scenario [32]. The trace is constructed using small devices called iMotes by participants at the IEEE Infocom conference in Grand Hyatt Miami, 2005. These devices measured Bluetooth sightings of each other. The devices were carried around by 41 students attending the student workshop.

Because of the measurements being taken at a conference the trace had a very large number of detections but very short time frame of only 3 days.

E. Cambridge-Infocom06 dataset

This trace is a continuation of the Cambridge Infocom trace [32]. It does the same measurements using the same devices but it increases the number of devices. This time 98 devices are used to take the measurements. The conference at which the measurements are taken is IEEE Infocom 2006.

Dataset	Duration (days)	No. devices	No. contacts
Reality	246	97	2,367,235
UPB2012	62	65	111,630
StAndrews	74	25	41,804
Infocom05	3	41	332,717
Infocom06	3	98	4,217,901
Random	30	30	38,280

TABLE I: Characteristics of the experimental data sets.

F. Reality dataset

This is the largest real life trace based on individual carrying mobile devices we could identify [13]. It consists of 100 individual devices in the form of smartphones that are preinstalled with software that measures Bluetooth contacts at a 5 minute interval.

The 100 devices were carried by students, faculty staff of the MIT Media Laboratory, a part of them are students to an adjacent faculty.

The size of the data is also given by the large time frame in which the trace was constructed, namely 9 months.

During these 9 months the students were very active and they moved through different places in the town area. Connections are rarer than in all other traces but this is compensated by the large time frame in which they were detected.

Using this large variety of traces we can safely assume that we covered most scenario in which individuals come in contact with each other, be it in a crowded environment with a large number of connections such as the Infocom traces or in a day to day scenario such as in the UPB2012, Reality and St. Andrews traces. We also use the Random generated trace to show the system behavior at the possible ideal limits.

To summarize the data, we present in Table I the number of days in which the trace measurements were taken or generated; Number of devices that were used; and number of contacts. We show here the number of contacts as the sum of the counters that keep track of connections from all of the devices. A connection can take place between two or more devices. If a connection starts with two devices and a third device joins at a later time, these are considered two individual connections, one between two devices and one between three.

In our simulation we randomly choose which devices are buyers and which devices are sellers. Because we randomly choose for each device we expect to have a 50-50% split between the two groups.

IV. Experimental Results

For a stable economic system and a viable market method, the asking price needs to converge (as also shown in [22]). The price convergence is most visible on the generated Random connection trace.

As we can see in Figure 1 the asking price reaches maximum convergence after about 120 steps. The markers represent the average price and the line represent the span of asking prices from global smallest to global highest. A step represents a connection between a seller and a number of buyers. The number of steps is taken for each seller individually, but in this trace, being generated has all sellers with an almost equal number of connections so they all reach step 120 at about the

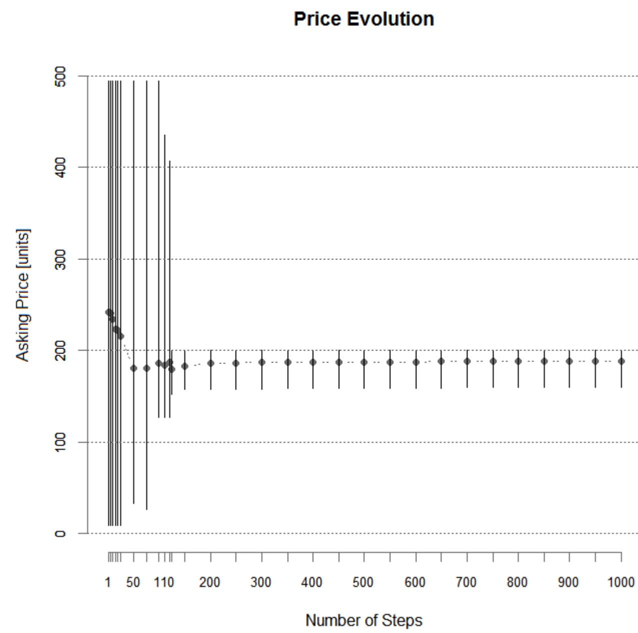

Fig. 1: Asking Price evolution for random generated trace.

same time. The convergence is made visible by the reduction in line size, with the maximum and minimum getting closer to the markers, representing the average value. We note here that the buyers are set with a default maximum price they are willing to offer to acquire the resource to 200. We have observed the same effects regardless of what value this maximum price takes: the sellers will converge near it. Full convergence in this scenario can't be achieved because of the random mutation added at each generation of asking prices.

For the price to be considered converged all the individuals in the asking price population of a seller need to have values between 180 and 220.

We tested the same algorithm over all our traces and we obtained similar results. For the results in Figure 2 we used the same maximum willing to pay value, of 200, for all the buyers and we observed how the seller's asking price converged. Unlike the generated random trace, in this traces the number of connections each seller has varies. Some sellers have a lot of connections and manage to have their price converge and manage to generate a lot of profit while others have very few connections, and because of this their asking price does not manage to converge.

In Figure 2 we show how depending on the features of the trace the number of devices that manage to have their price converge varies.

In grey we have the devices that have been randomly set to be buyers. In red and green we have the sellers. For simplicity a device can be either a seller or a buyer, meaning that the bars represent the total number of devices for each trace.

We notice how the best results are given by the Infocom traces, with Infocom06 having 100% of their seller's asking prices converge. This is because the traces consist of a large number of devices with an extremely large number of con-

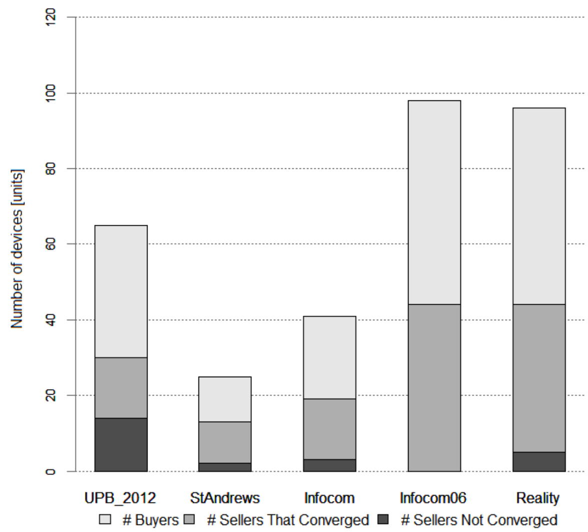

Fig. 2: Distribution of buyers/seller and the number of sellers that converge and that do not.

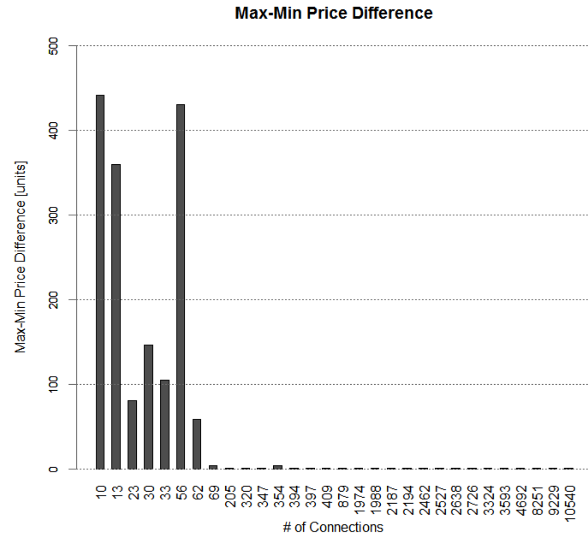

Fig. 3: UPB 2012 Price Convergence.

nections, devices move a lot and meet each other a lot during the Infocom conference, giving the sellers a better chance of discovering the maximum price the buyers are willing to pay.

The worst results are given by the UPB trace, here some devices are carried by students that are extremely active, they go to all classes, they meet a lot of other students and as such other devices, but there are also a lot of students which have a lower attendance or forget to start their tracing application. These students carry devices that cannot make as many connections and cannot converge to the asking price values.

In Figure 3 we present the UPB2012 trace in more detail. In this trace the largest percentage of devices have not managed to have their asking price converge. The results presented indicate the asking price status based on the number of connections the device has managed to make at the end of the simulation. Here we only show the sellers because these are the relevant ones for the presented method, the buyers do not have an asking price.

Here we see the difference between the biggest and the smallest asking price values. This shows that only 7 of the devices did not have their asking price converge. All the other have really small differences between the biggest (max) and the smallest (min) asking price. However in Figure 2 we show that almost half the devices did not converge to the correct value, the maximum value the buyers are willing to offer. This means that there are a part of devices, 7 to be more precise, that manage to converge their asking price values but converged to the wrong value.

V. Conclusion

Mobile Clouds are still a growing research topic and no applications have yet managed to capture the attention of the general population towards this novel paradigm.

These systems can be compared to grids and how they opened cooperation between different computing clusters and different groups of individuals.

In this paper, we present how to enable mobile clouds, both in the form of mobile cloud computing and the form of resource sharing such as sensors, by using incentives to motivate the use of Mobile Cloud systems.

To provide proper incentives for the users we propose a market based approach.

We take an evolutionary market method originally presented by Lewis et al. [22], and we apply this method to a Mobile Cloud.

We simulate a Mobile Cloud by using existing mobility traces and generating workloads, jobs that need to be run on different resources. The jobs can take the form of resource request, such as data from a sensor or task processing.

We prove that by applying Lewis's method to the Mobile Cloud we obtain a market that manages to have fair prices for both the buyers and the sellers. Because of the existence of a market, sellers are incentivized to use the Mobile Cloud because of financial gains and buyers are incentivized to use it to gain resources they would normally not have access to.

We did not model the needs of the buyers to extract the value they are putting on a job request, as in the maximum value they are willing to pay. Instead we set this value as a maximum default for all the buyers and test with multiple values set. However we present a discussion on what this needs could be and how can be supported by a Mobile Cloud ecosystem.

As future work we can imagine the implementation of our simulated solution in a real world environment where individuals can gain access to resources and gain benefits from sharing their own resources. We offered examples of possible implementations and use cases in the previous chapter.

Next we need to consider the security of such transactions and the privacy of individuals.

Acknowledgment

The research presented in this paper is supported by national project MobiWay, Project PN-II-PT-PCCA-2013-4-0321. We would like to thank reviewers for their constructive comments and valuable insights.

References

[1] Amazon. Amazon EC2. http://aws.amazon.com/ec2/, 2014. Accessed: November 25th, 2014.

[2] Pelin Angin, Bharat Bhargava, and Sumi Helal. A mobile-cloud collaborative traffic lights detector for blind navigation. In *Mobile Data Management (MDM), 2010 Eleventh International Conference on*, pages 396–401. IEEE, 2010.

[3] Michael Armbrust, Armando Fox, Rean Griffith, Anthony D Joseph, Randy Katz, Andy Konwinski, Gunho Lee, David Patterson, Ariel Rabkin, Ion Stoica, et al. A view of cloud computing. *Communications of the ACM*, 53(4):50–58, 2010.

[4] Siamak Azodolmolky, Philipp Wieder, and Ramin Yahyapour. Cloud computing networking: challenges and opportunities for innovations. *Communications Magazine, IEEE*, 51(7):54–62, 2013.

[5] Marco Valerio Barbera, Sokol Kosta, Alessandro Mei, and Julinda Stefa. To offload or not to offload? the bandwidth and energy costs of mobile cloud computing. In *INFOCOM, 2013 Proceedings IEEE*, pages 1285–1293. IEEE, 2013.

[6] Athina Bourdena, , George Kormentzas, Evangelos Pallis, , and Muneer Bani Yassein. A resource intensive traffic-aware scheme using energy-aware routing in cognitive radio networks. *Future Generation Computer Systems*, 39:16–28, 2014.

[7] Elena Burceanu, Ciprian Dobre, and Valentin Cristea. Adaptive distributed data storage for context-aware applications. *Journal of Telecommunications and Information Technology*, 2013.

[8] Nicolae-Valentin Ciobanu, Dragos-George Comaneci, Ciprian Dobre, Constandinos X. Mavromoustakis, and George Mastorakis. Openmobs: Mobile broadband internet connection sharing. In *Mobile Networks and Management (MONAMI 2014), 6th International Conference on*. IEEE, 2014.

[9] Pedro Maurício Costa, Jeremy Pitt, João Falcão e Cunha, and Teresa Galvão. Cloud2bubble: enhancing quality of experience in mobile cloud computing settings. In *Proceedings of the third ACM workshop on Mobile cloud computing and services*, pages 45–52. ACM, 2012.

[10] Jeffrey Dean and Sanjay Ghemawat. Mapreduce: simplified data processing on large clusters. *Communications of the ACM*, 51(1):107–113, 2008.

[11] Hoang T Dinh, Chonho Lee, Dusit Niyato, and Ping Wang. A survey of mobile cloud computing: architecture, applications, and approaches. *Wireless communications and mobile computing*, 13(18):1587–1611, 2013.

[12] Adam Dou, Vana Kalogeraki, Dimitrios Gunopulos, Taneli Mielikainen, and Ville H Tuulos. Misco: a mapreduce framework for mobile systems. In *Proceedings of the 3rd international conference on pervasive technologies related to assistive environments*, page 32. ACM, 2010.

[13] Nathan Eagle and Alex Pentland. Reality mining: sensing complex social systems. *Personal and ubiquitous computing*, 10(4):255–268, 2006.

[14] Niroshinie Fernando, Seng W Loke, and Wenny Rahayu. Mobile cloud computing: A survey. *Future Generation Computer Systems*, 29(1):84–106, 2013.

[15] Lin Gao, George Iosifidis, Jianwei Huang, and Leandros Tassiulas. Economics of mobile data offloading. In *Computer Communications Workshops (INFOCOM WKSHPS), 2013 IEEE Conference on*, pages 351–356. IEEE, 2013.

[16] Doan B Hoang and Lingfeng Chen. Mobile cloud for assistive healthcare (mocash). In *Services Computing Conference (APSCC), 2010 IEEE Asia-Pacific*, pages 325–332. IEEE, 2010.

[17] Sha Hua, Xuejun Zhuo, and Shivendra S Panwar. A truthful auction based incentive framework for femtocell access. In *Wireless Communications and Networking Conference (WCNC), 2013 IEEE*, pages 2271–2276. IEEE, 2013.

[18] Dijiang Huang, Zhibin Zhou, Le Xu, Tianyi Xing, and Yunji Zhong. Secure data processing framework for mobile cloud computing. In *Computer Communications Workshops (INFOCOM WKSHPS), 2011 IEEE Conference on*, pages 614–618. IEEE, 2011.

[19] Abdul Nasir Khan, ML Mat Kiah, Samee U Khan, and Sajjad A Madani. Towards secure mobile cloud computing: A survey. *Future Generation Computer Systems*, 29(5):1278–1299, 2013.

[20] Amreen Khan and Kamlkant Ahirwar. Mobile cloud computing as a future of mobile multimedia database. *International Journal of Computer Science and Communication*, 2(1):219–221, 2011.

[21] Kazuhiro Kuwabara, Toru Ishida, Yoshiyasu Nishibe, and Tatsuya Suda. An equilibratory market-based approach for distributed resource allocation and its applications to communication network control. *Market-based control: A paradigm for distributed resource allocation*, pages 53–73, 1996.

[22] Peter R Lewis, Paul Marrow, and Xin Yao. Resource allocation in decentralised computational systems: an evolutionary market-based approach. *Autonomous Agents and Multi-Agent Systems*, 21(2):143–171, 2010.

[23] Radu-Corneliu Marin and Ciprian Dobre. Reaching for the clouds: contextually enhancing smartphones for energy efficiency. In *Proceedings of the 2nd ACM workshop on High performance mobile opportunistic systems*, pages 31–38. ACM, 2013.

[24] Constandinos X Mavromoustakis, Athina Bourdena, George Mastorakis, , and Georgios Kormentzas. An energy-aware scheme for efficient spectrum utilization in a 5g mobile cognitive radio network architecture. *Telecommunication Systems*, pages 1–13, 2014.

[25] Constandinos X Mavromoustakis, George Mastorakis, Athina Bourdena, Evangelos Pallis, Georgios Kormentzas, and Joel J. P. C. Rodrigues. Context-oriented opportunistic cloud offload processing for energy conservation in wireless devices. In *Proceedings of Second International Workshop on Cloud Computing Systems, Networks, and Applications (CCSNA), IEEE Globecom 2014 – Cloud Computing Systems, Networks, and Applications (Globecom 2014 Workshops)*, 2014.

[26] Microsoft. Microsoft Azure. http://azure.microsoft.com/en-us/, 2014. Accessed: November 25th, 2014.

[27] OpenStack. OpenStack Foundation. http://www.openstack.org/, 2014. Accessed: November 25th, 2014.

[28] Francesco Palmieri. Scalable service discovery in ubiquitous and pervasive computing architectures: A percolation-driven approach. *Future Generation Computer Systems*, 29(3):693–703, 2013.

[29] Agoston Petz, Justin Enderle, and Christine Julien. A framework for evaluating dtn mobility models. In *Proceedings of the 2nd International Conference on Simulation Tools and Techniques*, page 94. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2009.

[30] Haiyang Qian, Deep Medhi, and T Trivedi. A hierarchical model to evaluate quality of experience of online services hosted by cloud computing. In *Integrated Network Management (IM), 2011 IFIP/IEEE International Symposium on*, pages 105–112. IEEE, 2011.

[31] Mahadev Satyanarayanan. Fundamental challenges in mobile computing. In *Proceedings of the fifteenth annual ACM symposium on Principles of distributed computing*, pages 1–7. ACM, 1996.

[32] James Scott, Richard Gass, Jon Crowcroft, Pan Hui, Christophe Diot, and Augustin Chaintreau. Crawdad data set cambridge/haggle (v. 2006-01-31). *CRAWDAD wireless network data archive*, 2006.

[33] Vlad Nicolae Serbanescu, Florin Pop, Valentin Cristea, and Gabriel Antoniu. Architecture of distributed data aggregation service. In *28th IEEE International Conference on Advanced Information Networking and Applications, AINA 2014, Victoria, BC, Canada, May 13-16, 2014*, pages 727–734, 2014.

[34] G. Skourletopoulos, R. Bahsoon, , George Mastorakis, and Evangelos Pallis. Predicting and quantifying the technical debt in cloud software engineering. In *Proceedings of the 19th IEEE International Workshop on Computer-Aided Modeling Analysis and Design of Communication Links and Networks (IEEE CAMAD 2014)*, 2014.

[35] University of Chicago. Nimbus Project. http://www.nimbusproject.org/, 2014. Accessed: November 25th, 2014.

[36] Xiaomin Zhu, Chuan He, Kenli Li, and Xiao Qin. Adaptive energy-efficient scheduling for real-time tasks on dvs-enabled heterogeneous clusters. *J. Parallel Distrib. Comput.*, 72(6):751–763, June 2012.

[37] Xiaomin Zhu, L.T. Yang, Huangke Chen, Ji Wang, Shu Yin, and Xiaocheng Liu. Real-time tasks oriented energy-aware scheduling in virtualized clouds. *Cloud Computing, IEEE Transactions on*, 2(2):168–180, April 2014.