Energy Consumption Optimization through Pre-scheduled Opportunistic Offloading in Wireless Devices

Constandinos X. Mavromoustakis
Department of Computer Science
University of Nicosia
46 Makedonitissas Avenue, P.O.Box 24005,
1700 Nicosia, Cyprus
mavromoustakis.c@unic.ac.cy

George Mastorakis, Stelios Papadakis
Department of Business Administration
Technological Educational Institute of Crete
Lakonia, Agios Nikolaos, 72100, Crete, Greece
gmastorakis@staff.teicrete.gr, spap@staff.teicrete.gr

Andreas Andreou
Faculty of Computer Science and Technology
University of Cambridge
William Gates Building, 15 JJ Thomson Ave,
Cambridge CB3 0FD, UK
aa773@cam.ac.uk

Athina Bourdena, Dimitris Stratakis
Department of Informatics Engineering
Technological Educational Institute of Crete
Estavromenos, Heraklion, 71500, Crete, Greece
bourdena@pasiphae.eu, dstrat@ie.teicrete.gr

Abstract—**The current research on mobile cloud computing systems and mechanisms has identified several challenges that have to be addressed for permitting execution on remote terminals/servers. A mobile cloud computing service provision has to be based on a framework that will ensure the effective execution of applications under an energy-efficient approach. In this context, this paper elaborates on the assessment of a framework that exploits a cooperative process-execution offloading scheme, pointing at offering energy conservation. The proposed approach utilizes a dynamic scheduling process to ensure that no discontinuous execution will happen on mobile devices. In addition, this paper elaborates on a partial offloading algorithm for an energy-efficient failure-aware allocation of the resources, by considering temporal execution-oriented metrics for the evaluation of the performance. The proposed scheme is analytically assessed through event driven simulation tests, towards verifying the effectiveness of the anticipated offloading approach, in terms of the energy consumption of the mobile devices and the quality of the degree offered.**

Keywords- ***Mobile cloud; offloading methodology; temporal execution-oriented metrics; opportunistic cloud reliability; dynamic resource migration.***

I. INTRODUCTION

The increasing number of the mobile devices (e.g. smart phones, tablets) introduces a new ground of research in the field of mobile wireless communications. This development has additionally been powered by a large number of applications that can be exploited in each mobile device, making the need for a dependable and high execution mobile computing processing environment. The proposed approach adopts an offloading process to partially outsource the resources that are required on a server rack or on an alternative mobile device with redundant resources. This offloading mechanism is adopted as a part of the application start-up, towards minimizing the GPU/CPU efforts, as well as the energy that is consumed by the mobile device, running out of resources.

Mobile cloud computing services have to be provided according to a synchronous mode [1], while several metrics in terms of the mobile devices, as well as the availability of the offloading by other terminals or serves have to be also taken into account [1]. The current cloud computing models are considered as ‘low offered’ throughput models [2], [3], while sometimes they offer low Quality of Service (QoS) or Quality of Experience (QoE) to the end-recipients (i.e., mobile devices users). The restrictions regarding the processing power, as well as the bounded capacity availability of mobile devices, aggravate the execution and harmfully affect the consistency offered to the mobile user, due to capacity-oriented failures and intermittent execution. If the available resources (processing and/or memory-oriented) are restricted, the mobile devices could exploit a cloud network infrastructure [1], towards supporting precise execution, via a resource/task migration mechanism. Such a mechanism has not yet been investigated to guarantee that satisfactory processing power is available, towards executing the application of a mobile device.

In this framework, this paper elaborates on a dynamic scheduling scheme, towards enabling offloading of the resources from a mobile device to another mobile device. The proposed scheme reduces the resources utilization of the mobile devices (GPU, CPU, RAM, battery consumption), by supporting at the same time, the extensibility of their lifetimes. The proposed approach in this paper is adopted to minimize the computational load of mobile devices, towards extending the lifetime of their battery. In addition, this approach takes into account a partitionable parallel processing wireless system, where the resources are partitioned and handled by a subsystem [1], evaluating the resource offloading process. Finally, a certain algorithm is proposed for the offloading process, towards dynamically defining the optimal resource manipulation in an energy-efficient approach.

In this context, the sections of this paper are based on the following structure. The related work and the research motivation are described in Section II. This section particularly focuses on the current research approaches, as well as on the resource offloading/migration scheduling policies. Section III then elaborates on the proposed offloading scheme and the associated mechanisms to reduce the energy consumption, maximizing the lifetime of the mobile devices. The proposed approach is based on the available resources of the mobile devices, the temporal terminals characteristics and the server-based parameters, along with the communication-oriented diversities. This approach establishes and maintains an efficient resources manipulation in the mobile devices, under an energy-efficient manner. Section IV presents the results that were obtained, by conducting simulation experiments, towards evaluating the performance of the proposed mechanism, by focusing on the behavioral characteristics of the scheme, along with the system response, as well as the energy consumption achieved. Finally, Section V summarizes the research findings of this paper and discusses the potential future directions for further experimentation and research.

II. RELATED WORK AND RESEARCH MOTIVATION

It is definitely valid that over the last years, a number of research efforts have been dedicated to device-to-device or Machine-to-Machine communication networks, ranging from physical layer communications to communication-level networking challenges. Since mobile devices are able to exchange resources through mobile networks, they generate large amounts of data [2]. The QoS [3] offered by these devices where on-the-move services are taking place is aggravated significantly by energy-hungry applications (e.g., video services, interactive gaming, etc.). In this context, the explicit lifetime of such devices has to be extended, especially when energy-hungry applications are exploited. In addition, efficient resource management has to be achieved within the context of the cloud computing paradigm and effective allocation issues of the processor power, the memory capacity resources, as well as the network bandwidth have to be considered. The resource management mechanisms have to allocate the resources of the mobile devices, on a cloud-based infrastructure, towards migrating some of their resources on the cloud servers [4]. The mobile devices have also to operate under specific QoS requirements, as defined by the users and the applications characteristics. The resource management process at the cloud scale involves a rich set of resource and task management schemes, which are able to effectively manage the QoS provisioning, by preserving the efficiency of the total system. However, the energy efficiency issue is one of the greatest challenge for this optimization problem, along with the offered scalability in the framework of the performance evaluation and measurement. In this context, different dynamic resource allocation policies have been explored in [5], towards elaborating on the enhancement of the application execution performance and the efficient utilization of the resources. Other research approaches associated with the performance of dynamic resource allocation policies, had led to the development of a computing framework [6] that takes into account the countable and measureable parameters affecting the task allocation.

In addition, authors in [7] address this problem, by using the CloneCloud approach [8] of a smart and efficient architecture. This architecture is exploited for the seamless use of the ambient computation to augment mobile device applications, off-load the right portion of their execution onto device clones and operate in a computational cloud. Authors in [8] statically partition service tasks and resources between the client and the server portions. The service is then reassembled on the mobile device at a later stage. The research approach in [8] is based on a cloud-augmented execution, by exploiting a cloned VM image as a powerful virtual device. This approach has many weaknesses, since it considers the resources of each cloud rack, depending on the expected workload and execution conditions (CPU speed, network performance). In addition, a computation offloading scheme is proposed in [9] that is exploited in cloud computing infrastructures to minimize the energy consumption of a mobile device, enabling the execution of certain/specified and under constrains applications. Energy consumption issues have also been investigated in [10], towards supporting computation offloading in a combination of 3G and Wi-Fi network infrastructures. However, such evaluations do not maximize the benefits of offloading, as they are considered as high latency offloading processes and require low amount of information to be offloaded. Cloud computing is currently impaired by the latency that is experienced during the data offloading through the wireless network infrastructure. Towards this direction, authors in [10] and [1], elaborate on issues, where the mobile devices exploit delay sensitive services. However, the variability of this delay in turn impairs the QoS/QoE of the mobile users.

Furthermore, authors in [11] elaborate on the resource processing poverty for ‘energy-hungry’ applications that request large amount of processing resources to run on a mobile device. Authors in [12] propose a resource manipulation scheme as a solution that is based on the failure rates of cloud servers in a large-scales datacenters. However, such criteria do not include the communications diversities of the servers during the communication process with the mobile users’ claims. This approach also does not take into account the available processing resources, the utilization of the device memory, the remaining energy and the available capacity with the communication of each of the device with the closest –in terms of latency- cloud terminal.

Within this context, this paper proposes an offloading resource mechanism, which is used in collaboration with an energy-efficient model. The scheme exploits an offloading methodology in order to guarantee that no intermittent execution will occur on mobile devices, whereas the application explicit runtime will meet the required deadlines to fulfil the QoS requirements. This paper also elaborates on the development of an offloading scenario, in which the scheduling policy for guaranteeing the efficiency in the execution of mobile users' tasks/applications can be achieved in an energy-efficient manner. The proposed framework is thoroughly evaluated through event driven simulation experiments, in order to define the efficiency of the proposed offloading policy, in contrast to the energy consumption of the wireless devices, as well as for the reliability degree offered.

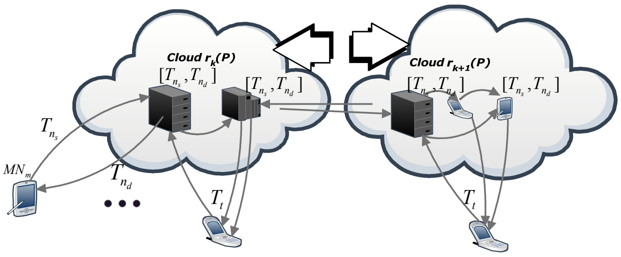

Figure 1. Cloud configuration and offloading process in order to achieve the best effort processing on-device power.

III. PRE-SCHEDULED OPPORTUNISTIC OFFLOADING IN WIRELESS DEVICES

A. Pre-scheduled offloading mechanism

Due to the heterogeneity in the hardware of both mobile devices and the servers on the cloud that the resources will be potentially (based on the proposed scheme) offloaded, the proposed framework encompasses the execution environment volatility and considers the cloud servers' response time, in order to a-priori compare them and select the appropriate server, according to the best fit-case. More specifically, this work considers the network-oriented parameters for bandwidth provisioning to achieve an acceptable resource offloading downtime δ (e.g., $\delta \leq 1.6\,s$ as the experimental process validates in [1]). To this end, from the network perspective, the modelled parameters can be expressed, for an offloading process *O for an* executable resource task O_{a_j}, as a 5-tuple given by:

$O_{a_j}(MN)$ = <n$_s$, T_{n_s}, T_{n_d}, BW, T_t >, for the a_j executable task, where n$_s$ is the devices or cloud terminals that the a_j from MN device will be offloaded, T_{n_s} is the source location best effort access time, T_{n_d} is the destination device or cloud location best effort access time from the source, BW is the required connection bandwidth and T_t is the connection holding duration for the a_j executable resource task.

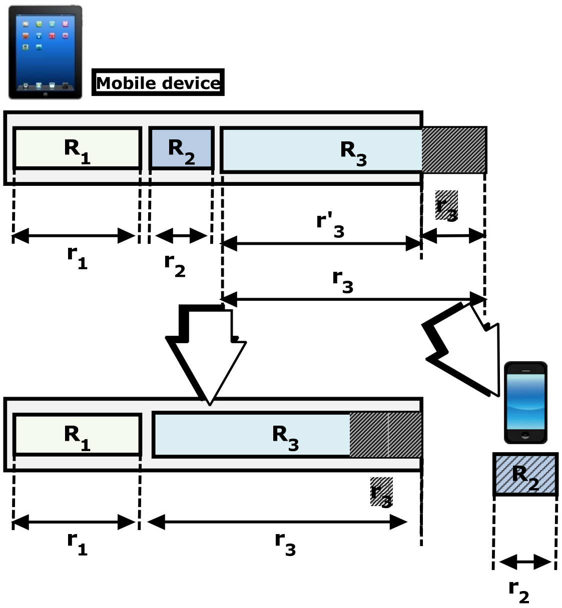

Figure 2. Resource partitioning onto mobile device.

In essence, the work done in [1] considers the resource transfer time, by taking into account the volume of traffic that should be transferred to the destination-source node. The total data volume that will be transferred, if the request meets the BW criteria can be provided by BW × T_t. In this work, the typical values ranges that were utilized in our experimental processes were $1MBps \leq BW \leq 20MBps$ and $T_t = 2s + t_x, s = T_{n_s} + T_{n_d}$, where t_x is the time to process x-partitionable parts that are processed during the offloading process. Every executable resource task may have *x-partitions*, which in this work are considered as t_x partitioning parts/tasks where $1 \leq t_x \leq z*P$, where z is the number of different devices that the resource can be offloaded. Therefore, the number of tasks per executable resource task is limited to the number of terminals in the system. An executable resource can be shared and partitioned to x_1, x_2, ... x_n, while it can be simultaneously processed with r sequential partitions, where $0 \leq r < z*P$, if and only if the following relation holds:

$$r + \sum_{i=1}^{n} p(x_i) \leq z*P \qquad (1)$$

where $p(x)$ represents the number of cloud terminals (mobile and statically located) that are needed to host the a_j. The scheduling strategy that was used is based on the Largest First Served/ Shortest Sequential Resource First Served and the service notations of [1] with a-priori knowledge of the $[T_{n_s}, T_{n_d}]$ service durations, as shown in Fig. 1. Every device with executable resource tasks and limited resources (memory) to execute, has to consider the offloading mechanism. In addition, Fig. 2 shows the dynamic offloading scheme, by considering executable resource tasks (t_x partitions) that will be offloaded according to the scheme proposed in the next section to conserve energy. In this scheme, partitionable tasks are offloaded onto other devices or cloud terminals based on the evaluation mechanisms shown in the next section, aiming at conserving energy on each mobile device, while running energy draining applications.

B. Energy-consumption model using temporal capacity measurements

Wireless nodes are error-prone with limited battery power, vulnerable and uncertain reliability, hosting energy-hungry applications. Therefore a challenging aspect for these devices is to design a resource offloading scheme that will significantly minimize the energy consumption and at the same time will enable the reliability in the smooth execution of any resource. As the consumed power varies with traffic and depends on the variations of the signal characteristics, as well as on the traffic-aware measurements [13], it is desirable to minimize the amount of power consumption, according to the resources that cannot 'run' on the mobile device or devices. To this end, the proposed scheme in this paper makes a progress beyond the current state-of-the-art, by elaborating on the association of the measurements of the partitionable tasks for two distinct cases: when resources can run on the devices. In this case, towards achieving energy conservation, the resources may be offloaded to a cloud or any other peer-neighboring device (so that the device that needs to run may potentially conserve energy); and the case that the device or devices cannot run the resources (as in Fig. 2) as the processing and memory requirements cannot support this execution. Thus, the measurable energy consumption can be evaluated according to the:

$$E_{r(a_j)} = E_c(a_j) \cdot \frac{C}{S_{a_j}} \tag{2}$$

In (2), $E_{r(a_j)}$ is the energy consumption, the parameter C is the number of instructions that can be processed within T_t, parameter S_{a_j} is the processing time of the server and $E_c(a_j)$ is the relative energy consumption expressed as:

$$E_c(r_i) = \frac{Cost_{c(r_i)}}{S_{c(r_i)}} \cdot W_c \tag{3}$$

where S_c is the server processing instruction speed for the computation resources, $Cost_c$ the resources processing instruction cost for the computation resources W_c energy consumption of the device or server in mW.

Each mobile device examines, if all neighboring 2-hops devices (via lookup table) can provide information about their offloading capabilities without affecting their energy status (thus without draining their energy to run other devices resources). In addition, the closest cloud rack is considered, if the relations exposed in (4) and (5) are not satisfied. Hence, for the neighboring devices *N* within *2-hops vicinity coverage* (based on the maximum signal strength and data rate model [1]) should stand:

$$\frac{Cost_{c(r_i)}}{S_{c(r_i)}} \cdot W_c \Big|^{r_i} > \frac{Cost_{c(r_i)}}{S_{c(r_i)}} \cdot W_c \Big|^{1,2..N} \tag{4}$$

$$W_{r_i} > W_c \forall 1,2,3,...N \text{ devices} \tag{5}$$

The energy consumption of each device should satisfy the (4)-(5) for each of the resources (executable processes) running onto the device MN_{m-1} hosting the r_i. Otherwise, the r_i with the maximum energy consumption is running in a partitionable manner to minimize the energy consumed by other peer-devices. These actions are shown in the steps of the proposed algorithm in table I.

TABLE I. DYNAMIC RESOURCE-BASED OFFLOADING SCHEME

```
Inputs: $MN_m$, Location($[T_{n_s}, T_{n_d}]$), resources
    $r_{1,} r_2, r_3, ..r_i \forall MN_m$
for all Cloud devices that stands $r_{1,} r_2, r_3, ..r_i$ find the $r_i$
    that can be offloaded to run onto another device
for all $MN_{m-1}$ do{
Estimate the $N_i$ //(as derived in ( 4))
  if ( $N_i$ is above a threshold){
    while ($T_t$== TRUE) {
      while ($1 \le t_x \le z*P$)
        search for $MN_{m-1}$ device that satisfies
            $\frac{Cost_{c(r_i)}}{S_{c(r_i)}} \cdot W_c \Big|^{r_i} > \frac{Cost_{c(r_i)}}{S_{c(r_i)}} \cdot W_c \Big|^{1,2..N}$, $W_{r_i} > W_c \forall 1,2,3,...N$
    offload ( $r_i, MN_{k(i)}$) //to $MN_{k(i)}$ to execute resource (i)
      end while
    end while ( $C_{a_j} = \frac{T_k^j}{\sum_k T_{a_j}(r)} \forall \min(E_c(r_i)) \in 1,2,..N$ )
  end for
end if
end for
```

The resource allocation will take place, towards responding to the performance requirements as in [1]. A significant measure in the system is the availability of memory and the processing power of the mobile cloud devices, as well as the server-based terminals. The processing power metric is designed and used to measure the processing losses for the terminals that the r_i will be offloaded, as in (6), where a_j is an application and T_k^j is the number of terminals in forming the cloud (mobile and static) rack that are hosting application a_j and $T_{a_j}(r)$ is the number of mobile terminals hosting process of the application across all different cloud-terminals (racks).

$$C_{a_j} = \frac{T_k^j}{\sum_k T_{a_j}(r)} \forall \min(E_c(r_i)) \in 1,2,..N \qquad (6)$$

The Eq. (6) shows that if there is minimal loss in the capacity utilization i.e., $C_{a_j} \cong 1$ then the sequence of racks $T_{a_j}(r)$ are optimally utilized. The latter is shown through the conducted simulation experiments in the next section. The dynamic resource migration algorithm is shown in Table I with the basic steps for obtaining an efficient execution for a partitionable resource that cannot be handled by the existing cloud rack and therefore the migration policy is used to ensure that it will be continuing the execution. The continuation is based on the migrated policy of the partitionable processes that are split, in order to be handled by other cloud rack terminals and thus omit any potential failures. The entire scheme is shown in Table I, with all the primary steps for offloading the resources onto either MN_{m-1} neighbouring nodes or to server racks (as in [1]) based on the delay and resources temporal criteria.

IV. PERFORMANCE EVALUATION ANALYSIS, EXPERIMENTAL RESULTS AND DISCUSSION

The mobility model used in this work is based on the probabilistic Fraction Brownian Motion (FBM) adopted in [13], where nodes are moving, according to certain probabilities, location and time. Towards implementing such scenario, a common look-up application service for resource execution offloading is set onto each one of the mobile nodes MN_m. Topology of a 'grid' based network [1] was modeled, where each node can directly communicate with other nodes. For the simulation of the proposed scenario, the varying parameters described in the previous section were exploited, by using a two-dimensional network, consisting of nodes that vary between 10-150 (i.e., terminal mobile nodes) located in measured area, as well as five cloud terminals statically located on a rack. All measurements were performed using WLAN (Wi-Fi based technology specifications) varying with different 802.11X specifications. During simulation the transfer durations are pre-estimated or estimated, according to the relay path between the source (node to offload resources) and the destination (node to host the executable resources).

In this direction, Fig. 3 shows the number of requests (total over failed) with the number of mobile devices participating in the evaluated area. It is important to mark that when the dynamic offloading scheme takes place the total failed requests among nodes are significantly reduced, particularly when the nodes number is small. Towards examining the impact of the different capacities, several sets of experiments were conducted, using the presented resource offloading scheme. Large memory resources are executable resources/processes that are between 500 MBytes -1 GBytes, whereas small resources are executable resources/processes that host capacity between the range of 10-400 MBytes.

In addition, the partitionable task offloading on different mobile devices, on which the proposed offloading procedure takes place, is shown in Fig. 4, in contrast to the Throughput response of the system. Throughput response is presented with the number of the 'in-service' terminals on the cloud racks with different processing power and speed characteristics, as shown in Table II. Fig. 4 indicates that for large memory -required- resources and when the mobile terminals have no remaining memory to process these resources, throughput dramatically drops. Furthermore, the Service Time with the number of racks is shown in Fig. 5. The Service Time is greater for large files that are not migrated in partitionable parts to the other terminals on the cloud racks.

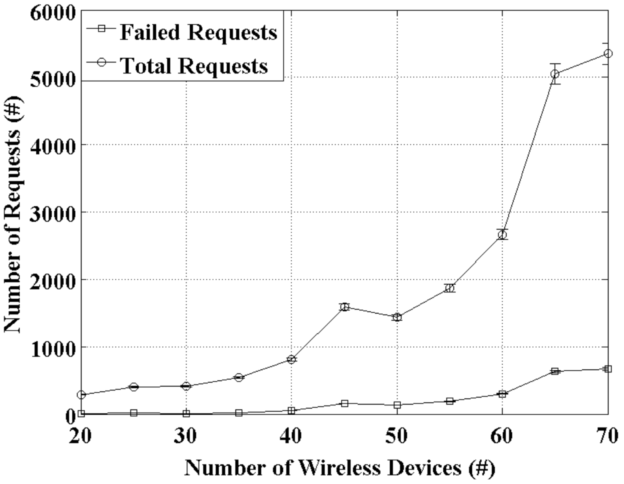

Figure 3. Number of requests with the number of mobile devices participating in the evaluated area.

TABLE II. CLOUD RACK TERMINALS CHARACTERISTICS.

Device #	CPU (GHz)	RAM (GB)	Core No.	Hard Disk (GB)	Cache (MB)	Core Speed (GHz)	Upload Speed (Mbits/sec)
1	2.1	8	Intel Duo	600	2	5	0.6
2	2.3	16	Quad 6600	500	2	5	2.6
3	2.1	4	i5	400	2	3	2.6
4	4.0	16	i5	1000	2	5	2.6
5	2.1	32	i7	600	2	3	4.6
6	2.3	16	i5	500	2	5	2.6
7	2.1	4	Quad 6600	400	2	3	1.5
8	4.0	16	i5	1000	2	5	2.6

On the other hand, the packet drop ratio of the proposed scheme for different mobility variations and without mobility over time is shown in Fig. 6. It is important to emphasize to the fact that the proposed scheme scales well in the presence of FBM and even better when the FBM with distance broadcasting is applied. Fig. 7 presents the average lifetime for both active and idle time with the number of the mobile devices, participating in the evaluated area. The overall energy consumption for each mobile device for three different schemes in the evaluated area (for the most energy draining resources) is shown in Fig. 8. The proposed scheme shows that it outperforms the scheme proposed in [1], as well as the scheme in [7] for the Wi-Fi/WLAN connectivity configuration.

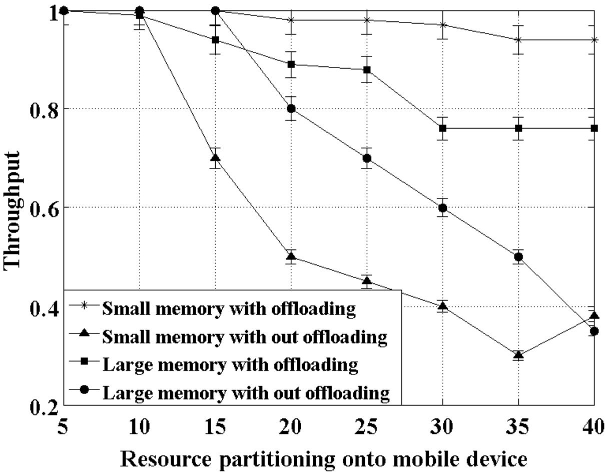

Figure 4. Throughput response with the mean number of executable resources that are partitioned per mobile device.

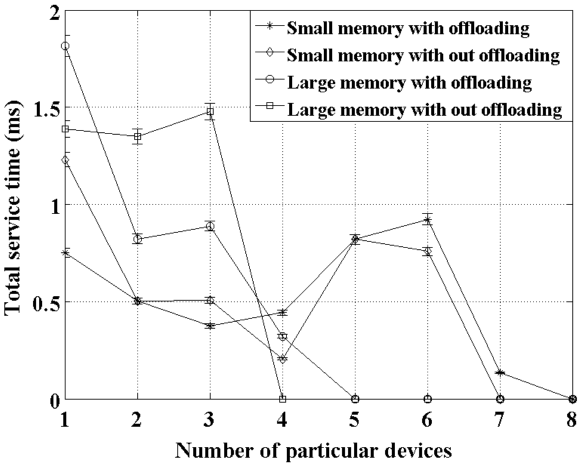

Figure 5. Number of requests with the number of mobile devices participating in the evaluated area.

When resources are offloaded, a critical parameter is the execution time, while nodes are moving from one location to another. In Fig. 9, the execution time during simulation for mobile nodes with different mobility patterns is evaluated, for GSM/GPRS, Wi-Fi/WLAN and for communication within a certain Wi-Fi/WLAN to another Wi-Fi/WLAN remotely hosted. The latter scenario -from a Wi-Fi/WLAN to another Wi-Fi/WLAN- shows to exhibit significant reduction, in terms of the execution time duration, whereas it hosts the minimum execution time through the FBM with distance broadcast mobility pattern. This is due to the access/propagation technology that is used, playing a major role and enabling faster connection, as well as higher transfer rates.

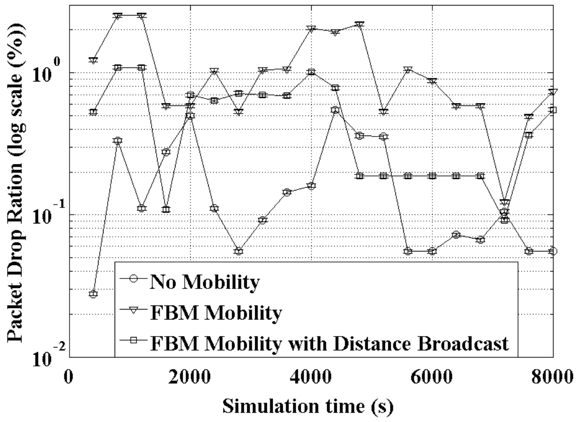

Figure 6. Packet drop ratio of the proposed scheme for different mobility variations and no-mobility model over time.

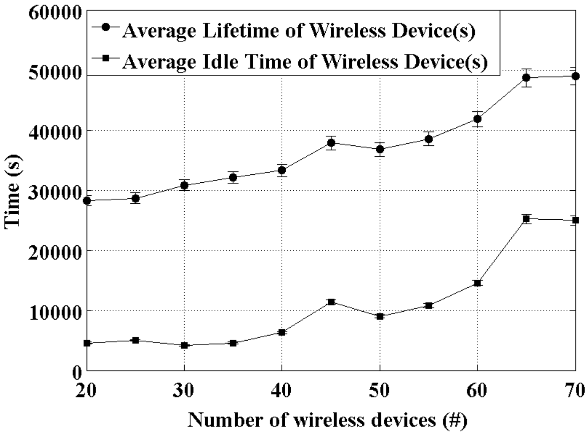

Figure 7. Average lifetime for both active and idle time with the number of mobile devices participating in the evaluated area.

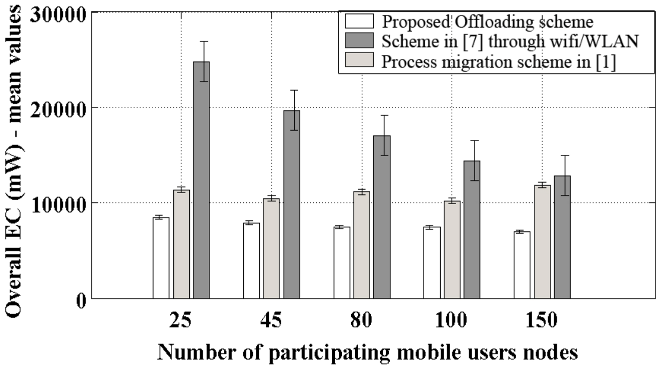

Figure 8. Overall energy consumption for each mobile device for three different schemes in the evaluated area (evaluated for the most energy draining resources).

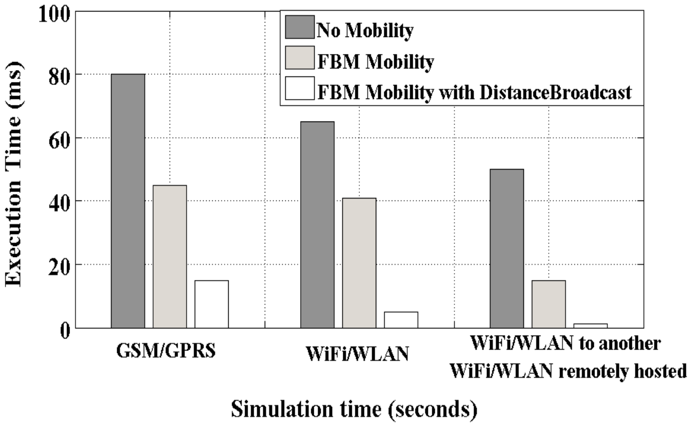

Figure 9. Execution time during simulation for nodes with different mobility patterns for three different schemes of communication.

V. CONCLUSIONS

This paper proposes a novel task outsourcing mechanism comprising of an executable resource offloading scheme. In the proposed scheme partitionable resources can be offloaded, in order to be executed according to their limited service, the transfer time, as well as the allowed execution (round-trip) duration, during the communication with the cloud terminal. The proposed scheme targets the minimization of the energy consumption and the maximization of the lifetime of each mobile device based on the available resources. The proposed offloading scheme is thoroughly evaluated through simulation experiments to validate the efficiency of the offloading policy, in contrast to the energy consumption of the mobile devices, as well as for the reliability degree offered. Future directions in our on-going research encompass the improvement of an opportunistically formed mobile cloud, which will allow delay-sensitive resources to be offloaded using the mobile peer-to-peer (MP2P) technology.

ACKNOWLEDGMENT

The work presented in this paper is co-funded by the European Union, Eurostars Programme, under the project 8111, DELTA "Network-Aware Delivery Clouds for User Centric Media Events".

REFERENCES

[1] P. Mousicou, C. X. Mavromoustakis, A. Bourdena, G. Mastorakis, and E. Pallis, "Performance evaluation of Dynamic Cloud Resource Migration based on Temporal and Capacity-aware policy for Efficient Resource sharing, accepted to MSWiM, The 16th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems, November 3-8 2013, Barcelona, Spain, pp. 59-66.

[2] S. Abolfazli, Z. Sanaei, E. Ahmed, and A. Gani, R. Buyya, "Cloud-Based Augmentation for Mobile Devices: Motivation, Taxonomies", and Open Challenges. IEEE Communications Surveys and Tutorials 16(1), 2014, pp. 337-368.

[3] C. Dimitriou, C.X. Mavromoustakis, G. Mastorakis, and E. Pallis, "On the performance response of delay-bounded energy-aware bandwidth allocation scheme in wireless networks", in Proceedings of the IEEE/ICC 2013 International Workshop on Immersive & Interactive Multimedia Communications over the Future Internet, organized in conjunction with IEEE International Communications Conference (ICC 2013), 9-13 June 2013, Budapest, Hungary, pp.641-646.

[4] M. A. Salehi, B. Javadi, and R. Buyya, "Resource provisioning based on preempting virtual machines in distributed systems". Concurrency and Computation: Practice and Experience 26(2), 2014, pp. 412-433.

[5] C. X. Mavromoustakis, C. Dimitriou, G. Mastorakis, E. Pallis, "Real-Time Performance Evaluation of F-BTD scheme for optimized QoS Energy Conservation in Wireless Devices", in Proceedings of the GlobeCom 2013 2013, pp. 1156-1161.

[6] D. Warneke and O. Kao, "Exploiting Dynamic Resource Allocation for Efficient parallel data processing in the cloud". IEEE Transactions on Parallel and Distributed Systems, VOL. 22, No. 6, June 2011, pp.985-997.

[7] B. Chun, S. Ihm, P. Maniatis, M. Naik, and A. Patti. "Clonecloud: Elastic execution between mobile device and cloud". Proceedings of the sixth conference on Computer systems of EuroSys, pp.301-314, 2011.

[8] B. G. Chun and P. Maniatis, "Augmented smartphone applications through clone cloud execution". In HotOS, 2009.

[9] M. V. Barbera, S. Kosta, A. Mei, and J. Stefa, "To Offload or Not to Offload? The Bandwidth and Energy Costs of Mobile Cloud Computing", Proc. of INFOCOM 2013, April 2013, Turin, Italy.

[10] E. Cuervo, A. Balasubramanian, D. Cho, A. Wolman, S. Saroiu, R. Chandra, and P. Bahl, "MAUI: making smartphones last longer with code offload," Proc. of the ACM International Conference on Mobile Systems, Applications, and Services, pp.49-62, San Francisco, CA, USA, 15-18 June 2010, pp. 647-652.

[11] M. Satyanarayanan, P. Bahl, R. Caceres, and N. Davies, "The case for vm-based cloudlets in mobile computing". Pervasive Computing, 8(4), 2009, pp. 14-23.

[12] V. K.Vishwanath and N. Nagappan, "Characterizing cloud computing hardware reliability". Proceedings of the 1st ACM Symposium on Cloud Computing, June, pp. 193-204, 2010.

[13] C. X. Mavromoustakis, C. D. Dimitriou and G. Mastorakis, "On the Real-Time Evaluation of Two-Level BTD Scheme for Energy Conservation in the Presence of Delay Sensitive Transmissions and Intermittent Connectivity in Wireless Devices", International Journal On Advances in Networks and Services, Vol. 6 number 3&4 2013, pp. 148-162.

[14] K. Papanikolaou and C. X. Mavromoustakis, "Resource and scheduling management in Cloud Computing Application Paradigm", appears in book "Cloud Computing: Methods and Practical Approaches", Eds. Prof. Zaigham Mahmood, published in Methods and Practical Approaches Series: Computer Communications and Networks by Springer International Publishing, May 13, 2013, ISBN 978-1-4471-5106-7, pp. 107-132.

[15] A. Bourdena, C. X. Mavromoustakis, G. Kormantzas, E. Pallis, G. Mastorakis, M. B. Yassein, "A Resource Intensive Traffic-Aware Scheme using Energy-efficient Routing in Cognitive Radio Networks", Future Generation Computer Systems Journal, Elsevier; to appear during the year 2014.