Journal of Network and Computer Applications 59 (2016) 46–54

Contents lists available at ScienceDirect

Journal of Network and Computer Applications

journal homepage: www.elsevier.com/locate/jnca

Dynamic energy-aware cloudlet-based mobile cloud computing model for green computing

Keke Gai [a], Meikang Qiu [a,*], Hui Zhao [b], Lixin Tao [a], Ziliang Zong [c]

[a] Department of Computer Science, Pace University, NY, USA
[b] Software School, Henan University, Kaifeng, Henan 475000, China
[c] Department of Computer Science, Texas State University, TX, USA

ARTICLE INFO

Available online 11 June 2015

Keywords:
Mobile cloud computing
Green computing
Dynamic program
Energy-aware
Cloudlets

ABSTRACT

Employing *mobile cloud computing* (MCC) to enable mobile users to acquire benefits of cloud computing by an environmental friendly method is an efficient strategy for meeting current industrial demands. However, the restrictions of wireless bandwidth and device capacity have brought various obstacles, such as extra energy waste and latency delay, when deploying MCC. Addressing this issue, we propose a *dynamic energy-aware cloudlet-based mobile cloud computing model* (DECM) focusing on solving the additional energy consumptions during the wireless communications by leveraging *dynamic cloudlets* (DCL)-based model. In this paper, we examine our model by a simulation of practical scenario and provide solid results for the evaluations. The main contributions of this paper are twofold. First, this paper is the first exploration in solving energy waste problems within the dynamic networking environment. Second, the proposed model provides future research with a guideline and theoretical supports.

1. Introduction

Mobile cloud computing (MCC) is an emergence of multiple Internet-based technologies development, which enables mobile users to acquire benefits of cloud computing and achieve green computing by using their mobile devices (Sabharwal et al., 2013; Bonino et al., 2013). The technology mainly derives from three hemispheres, including mobile computing, mobile Internet, and cloud computing. Combing the advantages of multiple techniques allows users to offload data processing and storage to the cloud-based servers (Huang et al., 2011; Kumar and Lu, 2010). However, behind the benefits of adopting this approach, the implementations of MCC are still facing a few challenges that limit its performance, such as energy over consumptions while the wireless communications are weak (Guan et al., 2011). Keeping searching wireless signals can dry out the power of mobile devices, which may cause unexpected energy waste (Han et al., 2011).

In this paper, we propose an advanced dynamic model, *dynamic energy-aware cloudlet-based mobile cloud computing model* (DECM), which uses *cloudlets* technique to assign, manage, and optimize the cloud-based infrastructure usages and services for achieving green computing. This model uses dynamic programming to assist cloudlets cloud computing resources within a changing operational environment. The intention of DECM matches practical demands of mobile industry because various elements can have major influences on the cloud services quality. For example, mobile users who are using map services highly rely on the speed of wireless communications while the mobile devices are rapidly moving. Nevertheless, unstable and inefficient wireless connections usually shorten the battery life.

Many researchers and scholars have done various achievements in energy-aware mobile cloud computing in previous research. The research is diverse in different perspectives (Gupta and Roy, 2013). Zhu and his team (Yang et al., 2014) developed a real-time tasks oriented virtualized cloud computing system that was designed to achieve energy-aware scheduling in their recent works. The proposed solution (Yang et al., 2014) intends to integrate various energy-aware scheduling algorithms by employing a rolling-horizon optimization policy. However, this approach did not consider mobility usage and the similar research focusing on energy-aware cloud computing systems has been accomplished by other scholars (Mezmaz et al., 2011; Beloglazov and Buyya, 2010; Berl et al., 2010).

Furthermore, as one of the core techniques in cloud computing, virtual machine (VM) is considered an efficient approach for building up cloud-based datacenter to achieve green computing

---

* Corresponding author.
E-mail address: mqiu@pace.edu (M. Qiu).

http://dx.doi.org/10.1016/j.jnca.2015.05.016
1084-8045/© 2015 Elsevier Ltd. All rights reserved.

(Aksanli et al., 2012). Nevertheless, VM is only a service representation approach that does not bring much technical innovations, even though VM has been broadly applied in deploying green *Information Technology* (IT) industry, such as green data processing, storage, and transmissions (Baliga et al., 2011; Lee and Zomaya, 2012; Xiao et al., 2013).

Based on the industrial needs, our advanced deployment model, DECM, offers a unique mechanism to avoid the energy waste when users are suffering a complicated and unstable networking environment. The model is a type of web service that focuses on efficient communications between user devices and cloud servers. Figure 1 represents a conceptual model of DECM. Three main components of DECM include mobile device, cloudlets with dynamic searching, and cloud computing. The relationship between cloudlet and dynamic searching is that the cloudlets provide an operating platform in which the dynamic search is executed.

The procedure of the service delivery is followed by the directions of the arrows. Mobile devices send the service requests to the closest cloudlets before the requests reach the cloud servers. The cloudlets allocate the cloud servers for better service performance determined by a group of constraints, such as nearby server locations or networking stability. Dynamic programming is applied in cloudlets for adapting to the constant changing context. Selecting the best solution directed by the dynamic-based cloudlets is the core component of DECM, which is expected to avoid energy waste when switching cloud servers or wireless networks.

We develop a motivational example explaining the fundamental methods of adopting DECM. The implementation of the example is a simulation followed by the proposed model. Figure 2 illustrates the fundamental concepts of DECM. Compare with classic web services, business logics are refereed by cloudlets that use dynamic programming to search efficient cloud services. This difference is a core session in our model, which is optimizing the utilizations of cloud resources in mobile cloud. Detailed information of fundamental concepts for DECM is given in the following section.

The main contributions of this paper are twofold.

1. This research is the first attempt on the functionality of cloudlets in order to achieve energy-aware performances in the dynamic networking environment.
2. The results of this research provide theoretical supports and explorations. The model may be migrated and applied in multiple domains. The model may be able to be migrated and applied in multiple industries, which requires further research for identifying and proving.

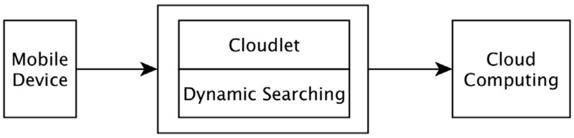

Fig. 1. A high level conceptual model of DECM.

The remainder of this paper is follows. We provide explanations about DECM in Section 2. Following the description of the model, an example is given in Section 3 in order to demonstrate the implementation of DECM in practice. Section 4 proposes and evaluates DECM algorithm. The experimental results are represented in Section 5. The conclusions are given in Section 6.

2. Concepts and model

2.1. Green computing

The principle of DECM can enable green computing because the model is designed for mainly reducing energy consumptions, which matches one of the characteristics of green computing as an energy-aware feature (Beloglazov et al., 2012; Mukherjee et al., 2014). The approaches of achieving green computing are from software and hardware to management, policy, and legal issues. Our model focuses on technical side that leverages a few cloud-related techniques, such as VM, wireless networks, and dynamic programming.

Using our proposed schema can reduce the energy consumption in wireless communications. The system reliability is also considered in the proposed schema, which aims to guarantee the performance of the system. Saving the energy costs of the communications can achieve green computing. The goal of performing green computing is to apply DECM to reduce energy consumptions on mass mobile devices without weakening the performance of cloud services.

2.2. Mobile cloud computing

MCC is the conceptual architecture that combines three technologies, including mobile Internet, mobile computing, and cloud computing, to enable mobile users to offload data processing and storage onto clouds via wireless networks and mobile devices (Dinh et al., 2013; Kumar and Lu, 2010; Guan et al., 2011; Song and Su, 2011). The motivation of applying MCC is gaining benefits of cloud computing technologies by leveraging mobile techniques. The dynamic networking environment results in more complicated service deployments and implementations, comparing with basic cloud computing.

The first basic part of MCC is *mobile Internet*, also known as *wireless networks*, that is a series of computing networks for achieving wireless communications by connecting network nodes and using wireless protocols. A variety of mobile Internet techniques are available for different demands (Chan et al., 2011). Main techniques of wireless networks include *wireless personal area networks* (WPAN), *wireless local area networks* (WLAN), *wireless metropolitan area networks* (WMAN), *wireless wide area networks* (WWAN), and cellular networks. Among these techniques, the cellular network is an approach of deploying wireless communications, such as 3G and *long-term evolution* (LTE).

Moreover, supported by mobile Internet, *mobile computing* (Satyanarayanan, 2011) is a group of techniques for executing wireless communications on both software and hardware sides, such as mobile

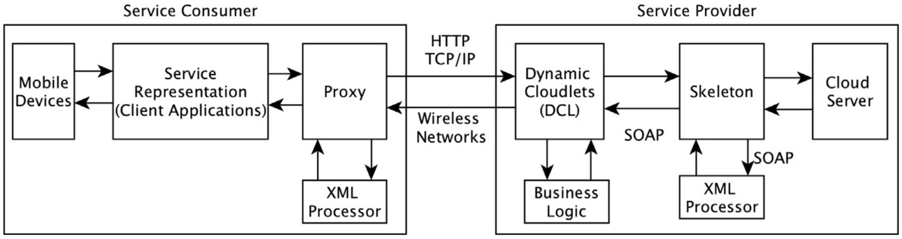

Fig. 2. Fundamental concepts for DECM.

devices, protocols, and networking infrastructure. The concept includes most current mobile devices, such as portable computers, smartphones, and tablets. One of the restrictions of using mobile computing is that most devices still highly rely on the battery life. A high performance usually conflicts with the durations of battery power and less efficient workload generates energy waste that is non-environmental friendly.

Furthermore, cloud computing is a core component of MCC because it provides service deployment framework and mechanisms. The concept of cloud computing is leveraging Internet-based technologies to perform computer resources sharing and mask technical complexities so that cloud customers can access computing services according to their real-time demands without technical obstacles (Gai and Li, 2012). In MCC, both data processing and data storage can be migrated onto the cloud-based servers, which is an energy-aware approach for mobile devices while the wireless communications are stable and efficient (Miettinen and Nurminen, 2010; Ma et al., 2012).

2.3. Cloudlets with dynamic programming

Term *cloudlet* refers to a layer connecting mobile devices and cloud servers in MCC, which plays a mediator role focusing on the business logic. The concept of the cloudlet addresses a trusted party that can be either a computer or a cluster of digital infrastructure with software that provides cloud users with a rapid response and specific customized functionalities (Verbelen et al., 2012; Fesehaye et al., 2012; Soyata et al., 2012). It is a self-management mechanism that is used to strengthen communications between mobile devices and cloud servers by reducing latency. For example, a cloudlet can be utilized to transfer communication information to mobile nodes by wireless networks (Fesehaye et al., 2012).

In DECM, cloudlets are deployed with dynamic programming and we conceptualize it as a *dynamic cloudlet* (DCL). Mobile cloud users send out the service requests through the *virtual machine* (VM) attached to the client applications and the requests will efficiently reach the nearest cloudlet. The access method to the DCLs resembles the access points to the public Wi-Fi. Before the service requests are passed to the cloud, most calculations concerning the resource allocations are done in DCLs according to the business rules. Typically, DCLs are designed to complete at least three missions, which include the following:

1. Leveraging dynamic programming to determine and predict physical machines on the cloud for current and future use.
2. Calculations of simple applications and quickly respond to end-users.
3. Predict whether users should switch to other cloudlets and enable a real-time suggestion for cloudlets switching.

The predictions are based on the services content and other attached information to the service requests. Cloudlets apply dynamic programming to determine which cloud servers they need to connect with. Energy saving is the main purpose of DECM so that all parameters related to predications and determinations are defined for measuring energy costs.

Moreover, according to the various service demands, there are two types of DCL deployments. The deployments are defined from the perspective of cloud services, as follows:

- *Private dynamic cloudlets*: This deployment supports one single cloud service provider for the purpose of increasing service performance and reducing energy consumptions. For example, private dynamic cloudlets can be employed for *global position system* (GPS) services that is provided by one service provider.
- *Public dynamic cloudlets*: Public DCLs stand for a public-purpose access to the DCL-based cloud computing services, which may support multiple service providers. We give an example that a dynamic searching services can be delivered by deploying public DCLs since the services maybe supported by multiple service providers who present various data mining offerings.

2.4. Web service

Web service states and defines the fundamental of DECM. The concept of web service is a set of techniques for executing machine-to-machine communications via wireless networks. As shown in Fig. 2, service representation is given by the client applications, which can be supported by *application programming interface* (API). *Proxy* receives the service requests who claims the service delivery capability. *XML (extensible markup language) processors* provide web services with a mediative service that integrates different computing languages and ensures that machines can understand each other by universally being translated into a mediatorial language.

Once the service request is received by DCLs, business logic is applied to the requests and the further determinations will be made by dynamic programming. *Simple object access protocol* (SOAP) supports the structured information exchanges, aligning with other web protocols, such as *hypertext transfer protocol* (HTTP). The same process of translating computing languages occurred at the XML processors in order to communicate with various cloud servers. The service responses are sent back from the physical servers on the cloud through the CDLs, proxy, and service representation layer, and finally back to mobile devices.

The following section represents a motivational example using DECM in a practical scenario.

3. Motivational example

We exhibit an example in this paper in order to further clarify the description of DECM. The scenario is that mobile users are using an app that is a cloud-based real-time acquaintances alarm system while users travel between two places. The system offers the service that lets mobile users know whether their acquaintances are nearby who are using the same function of the app. In a traditional way, mobile devices communicate with cloud severs on wireless networks directly. This approach keeps the communications in the box containing the specific cloud servers and mobile devices until the signals or connections become weak or disconnected.

Compare with the traditional manner, DECM offers a dynamic wireless communication that uses CDLs to dynamically select and determine the most nearby and efficient cloud servers based on the business logic. Figure 3 represents a conceptual work flow describing the method of DECM operations.

As displayed in Fig. 3, mobile devices, defined as a *mobile user layer*, communicate with *cloudlets layer* by searching the nearest CDLs while mobile users are accessing the cloud services. *Cloudlet A* that is assumed the nearest cloudlet at the beginning of the service does the calculations based on the business logic and start predicting other potential cloudlet services according to the service contents, such as multiple locations and dynamic service requests. Once *Cloudlet A* confirms that *Cloudlet B* can offer a better service performance due to the constraints, such as geographic distance and networking conditions, the connections will switch to *Cloudlet B* so that mobile users can obtain greater services by communication *Cloudlet B*.

Moreover, as we mentioned in Section 2, CDLs are responsible for searching the cloud servers that are nearest or can offer better service performance. The communications between *cloudlets layer* and cloud computing layer deliver the service requests and responses and responses in a dynamic manner. The cloud server that physically provides services is selected and determined by CDLs. Figure 4 illustrates an example of the DECM implementation.

Depicted in Fig. 4, the *starting point* describes the position where the service requests being sent. The multiple positions of *mobile devices* identify the locations passed by users during the service delivery process. Positions are connected by the solid arrow lines, which exhibits the shift routes of users. Solid color dots stand for cloudlets that have potentials of providing services by connecting with mobile devices. Numerous cloudlets nearby the mobile devices imply a dynamic surrounding environment. Broken lines attached to cloudlets indicate the potential wireless communications of cloud-based service implementations between mobile devices and cloudlets, which will be determined by the dynamic program operated by DECM.

4. Algorithm

4.1. DECM algorithm

Table 1 lists a group of predefined elements used for the simulations and formulations of the solution algorithm in this paper.

Definition 1. ECM: *energy cost minimization problem during the wireless communications by adopting cloudlets.* Given a set of cloudlets coded by x, $\forall x \in N$ and $\forall x \geq 1$, N=natural numbers.

The total cost, $C_{Total}(T, x)$, consists of two components. The first component addresses the energy consumption among mobile users, cloudlets, and cloud servers specifically occurred on one route, defined as a $E_i(t, x)$. A *route* is a preference selection of CDLs implementations.

Moreover, considering the service performances, the other side is $P_i(t, x)$ that defines a performance level localized on one route x for one specific operating time unit t. This aspect is for avoiding the reduction of service quality while selecting an energy-aware route. Our goal is to generate the minimum energy consumption under a specific timing constraint that is often considered a workload planning time. Therefore, the ECM problem can be formulated into the following equation:

$$Minimize\ C_{Total}(T, x) \quad (\forall T \in R^{+};\ \ \forall x \in N) \tag{1}$$

The total energy consumption is a sum of the products of energy cost for each model and performance percentage levels. Eq. (2) represents the method of generating total energy consumptions. For the meanings of the symbols in the equation refer to Table 1.

$$C_{Total}(T, x) = \sum_{x=1}^{n} f(P_i(t, x), E_i(t, x)) \quad (\forall i \in N;\ \ \forall x \in N;\ \ \forall t \in R^{+}) \tag{2}$$

Using a dynamic programing approach on DCLs aims to select the most efficient communication between mobile devices and cloud servers. On the basis of Eq. (2), we generate and propose our algorithm that is given in the following sections.

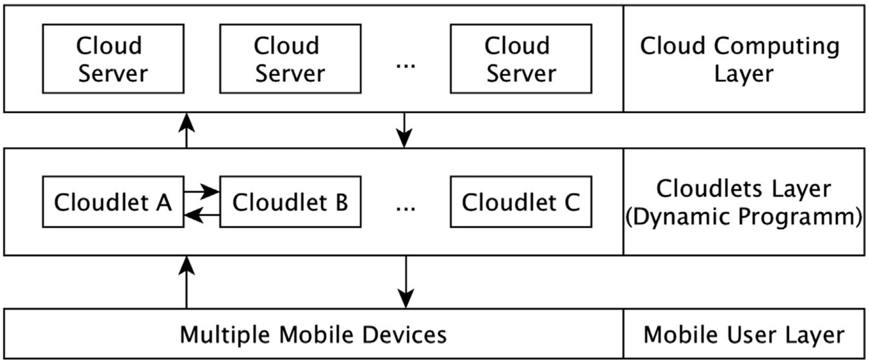

Fig. 3. Conceptual workflow of DECM for motivating example.

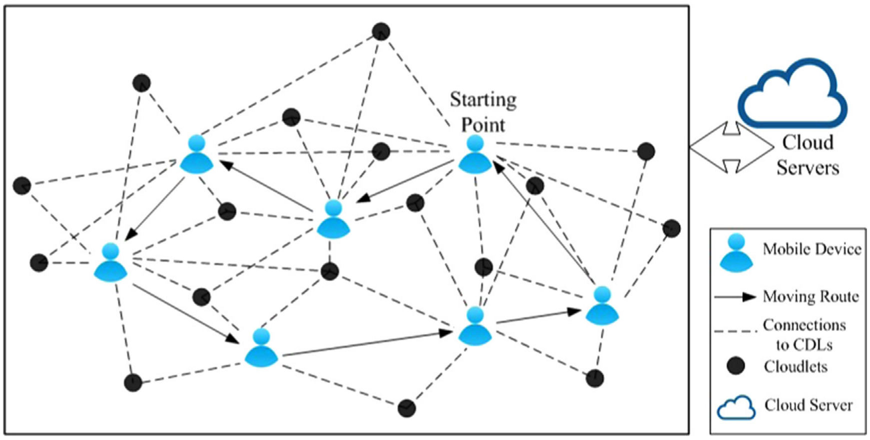

Fig. 4. An example of cloudlet-based wireless communications implemented by DECM.

Table 1
Main notations used in this paper and the definitions.

Notations	Definitions
x	Cloudlet node code, referring to the operation manners aligning with the cloudlet nodes
i	Deciding which method route will be used, $\forall i \in N$
M_i	(*Method route*) means which cloudlet route will be used, $\forall i \in N$
t	Specific latency or timing cost for each node
T	(*Execution time unit*) referring to the time length of delivering services under requests
$P_i(t,x)$	(*Service performance*) represents the level of service performance at the time node t and cloudlet node x, eg. 0.8 explains 80 percent of full performance
$E_i(t,x)$	(*Energy consumption*) is the energy cost during the wireless communications at the time node t and cloudlet node x
©	An operation that calculates $(P, E)=(P_i, E_i)©(P_j, E_j)$; $P=P_i \times P_j$; $E=E_i+E_j$
$CDL_{i,t}(M_i(x))$	Attributes of one CDL with two variables, i and t, and the corresponding $M_i(x)$
$CostCDL$	Performance and energy consumption of each cloudlet
$C_{Total}(T,x)$	(*Total cost consumption*) is an outcome of energy total consumptions considering service performances within a specific scheduling demands
$f(P_i(t,x), E_i(t,x))$	Total cost function with two variables $P_i(t,x)$, $E_i(t,x)$

Table 2
Algorithm of dynamic cloudlets-based module for determining cloud servers.

```
Input: $P_i(t,x)$, $E_i(t,x)$, $T_i(x)$, $M_i(x)$, and $Nd(x)$
Output: Minimum energy consumptions within a specific timing period

for $x \leftarrow 1$ to $Nd(x)$
  for $m \leftarrow 1$ to $M_i(x)$
    for $t_T \leftarrow$ to $T_i(x)$
      $P_m(t_T,x) \leftarrow \sum_{t=1}^{t_T} P_m(t,x)$
      $E_m(t_T,x) \leftarrow E_m(t_T,x)$
      do the comparison and cancel the pair performing worse
    end for
  end for
end for
for $t_T \leftarrow 1$ to $T_i(x)$
/* calculate every CostCDL from $Nd(1)$ to $Nd(x)$
  do the comparison and cancel the pair performing worse */
end for
return all $CostCDL[CDL_{i,t}(M_x)]$
```

4.2. Recursive formulation

The crucial component of dynamic programming algorithm is to generate the recursive formulation, which drills down the large size problems into smaller sized sub-problems to make the whole problem solvable. Selecting the best solution to each sub-problem is one of the principles of dynamic programming, which is applicable for complicated problems, such as polynomial time problems (Qiu et al., 2014). In this paper, we propose an approach of using dynamic programs to optimize operating manners of DCLs.

First, aligning with Eq. (2), we propose an efficient schema that enables a dynamic switch between cloudlets that can have better service offerings. According to the matrix shown in Eq. (3), the minimum energy consumption with quality services can derive from the recursive formulation. The performance of each cloudlet is symbolized by *CostCDL*:

$$\begin{cases} EnergyCloudlet(CDL_{i,t}(M_1)) = CostCDL[CDL_{i,t}(M_1)] \\ EnergyCloudlet(CDL_{i,t}(M_2)) = CostCDL[CDL_{i,t}(M_2)] \\ \vdots \\ EnergyCloudlet(CDL_{i,t}(M_n)) = CostCDL[CDL_{i,t}(M_n)] \end{cases} \tag{3}$$

Second, with cognitions of the recursive formulation, we define an energy unit cost function addressing each DCL as *EnergyCloudlet* ($P_i(t,x)$, $E_i(t,x)$), which represents the total energy costs with the service performance at each cloudlet. The total cost function is formulated in $f(P_i(t,x), E_i(t,x))$, which is given in Eq. (4). In this case, we do not consider the energy cost that occurred while switching cloudlets because of its small amount of energy consumptions:

$$f(P_i(t,x), E_i(t,x)) = \begin{cases} \sum_{x=1}^{n} EnergyCloudlet_{min}(CDL_{i,t}(M_i(x))) \\ \prod_{x=1}^{n} P_i(t,x) \\ \forall i \in N; \quad \forall x \in N; \quad \forall t \in R^+; \quad 0 < t < T \end{cases} \tag{4}$$

Theorem. *By applying the recursive function, each element attached to the matrix CostCDL[*$CDL_{i,t}(M_x)$*] is the minimum energy cost for selecting cloudlets by examining the energy cost,* $E_i(t,x)$*, and service performance,* $P_i(t,x)$.

Proof. The assumption of the execution scenario is that the mobile device connects to the cloud server with one route M_1 at the beginning of the service. The scenario considers a common situation for starting a service. The following routes have multiple options, $M_1, M_2, \ldots, M_n$, and the routes may have different values for variables. For example, route M_2 may have a longer latency at the first time-unit than M_1. At the starting point, the existence of $P_1(1,1)$ and $E_1(1,1)$ represent that $\exists i=1$, $t=1$, and $x=1$, which are inputs to ECM problem.

Moving forward to the next time unit, a decision will be made depending on the result of the comparison among all routes, $\{P_1(2,1),\ E_1(2,1)\ ©\ P_2(2,1),\ E_2(2,1)\acute{A}\ldots P_n(2,1),\ E_n(2,1)\}$. The route (s) with a better performance will be selected. The resolution will be made for each time unit by using the same method and finally it reaches a minimum total energy cost with quality services in a defined timing period, when $t=t_T$, $x=x_T$, and valued $\{(P_1(t_T,x_T),\ E_1(t_T,x_T))\}\ ©\ \{(P_2(t_T,x_T),\ E_2(t_T,x_T))\}\ \acute{A}\ldots\{(P_n(t_T,x_T),\ E_n(t_T,x_T))\}$. The final result will be a set of $\{CDL_{i,T}(M_i(x))\}$ that may consist of only one or multiple paths. Associating with restrictions and considered parameters, our theorem is a proved energy-aware approach for the implementations of energy-aware solutions. The next section describes our main algorithm. Table 2 the dynamic programming algorithm we used for our proposed mechanism that derives from Eq. (4).□

5. Experiment and results

We evaluate our algorithm by using the scenario provided in Section 3. The evaluation is based on a simulation that is a mathematic deduction by comparing DECM approach with the traditional cloud computing method. The experiment can simulate both private and public dynamic cloudlet executions. The workload has been examined on the simulator named DECM-Sim that is developed by our lab. The simulator is designed to calculate input data and generate the results with using the dynamic programming. A few parameters are defined for the comparison

Table 3
Energy consumptions and performances between different cloudlet-based modules.

Nd	M_1			M_2			Nd	M_1			M_2		
	T_1	P_1	E_1	T_2	P_2	E_2		T_1	P_1	E_1	T_2	P_2	E_2
0	1	0.8	9	2	0.7	4	4	2	0.7	8	4	0.9	5
	2	0.2	9	3	0.3	4		5	0.3	8	5	0.1	5
1	1	0.9	7	2	0.8	2	5	1	0.8	7	2	0.9	3
	2	0.1	7	5	0.2	2		3	0.2	7	4	0.1	3
2	1	0.9	10	2	0.9	5	2	2	0.9	6	4	0.8	2
	3	0.1	10	4	0.1	5		3	0.1	6	6	0.2	2
3	2	0.8	8	5	0.9	3	7	1	0.7	8	3	0.9	4
	3	0.2	8	7	0.1	3		3	0.3	8	5	0.1	4

Table 4
Minimum total costs with computed performance capacities under various timing constraints for a DECM.

T	(P, E)	(P, E)	(P, E)	(P, E)	(P, E)
11	(0.183, 54.0)				
12	(0.206, 50.0)	(0.229, 54.0)	(0.162, 49.0)		
13	(0.258, 50.0)	(0.183, 45.0)	(0.286, 54.0)	(0.142, 44.0)	(0.229, 49.0)
14	(0.265, 46.0)	(0.327, 54.0)	(0.321, 50.0)	(0.16, 40.0)	(0.258, 45.0)
	(0.286, 49.0)	(0.203, 44.0)			
15	(0.368, 50.0)	(0.331, 46.0)	(0.236, 41.0)	(0.409, 54.0)	(0.321, 45.0)
	(0.229, 40.0)	(0.254, 44.0)			
16	(0.339, 43.0)	(0.459, 50.0)	(0.413, 46.0)	(0.206, 36.0)	(0.368, 45.0)
	(0.331, 41.0)	(0.286, 40.0)			
17	(0.51, 50.0)	(0.473, 47.0)	(0.426, 43.0)	(0.303, 38.0)	(0.459, 45.0)
	(0.413, 41.0)	(0.327, 40.0)	(0.294, 36.0)	(0.525, 51.0)	(0.18, 35.0)
18	(0.183, 32.0)	(0.459, 41.0)	(0.59, 47.0)	(0.531, 43.0)	(0.265, 33.0)
	(0.473, 42.0)	(0.426, 38.0)	(0.368, 36.0)		
⋮	⋮	⋮	⋮	⋮	⋮
32	(0.9, 29.0)	(0.729, 24.0)	(0.81, 28.0)	(**1.0, 37.0**)	
33	(0.729, 24.0)	(**1.0, 37.0**)	(0.9, 28.0)		
34	(0.81, 24.0)	(0.9, 28.0)	(**1.0, 36.0**)		
35	(0.81, 24.0)	(**1.0, 33.0**)	(0.9, 28.0)		
36	(0.9, 24.0)	(**1.0, 32.0**)			
37	(0.9, 28.0)	(**1.0, 32.0**)			
38	(0.9, 28.0)	(**1.0, 32.0**)			
39	(**1.0, 28.0**)				

and assessment purposes, including timing units (T_x), wireless communication performances (P_x), energy consumptions (E_x), mobility routes (M_x), and switching nodes among different routes (n). Data collection is followed by the simulation research methodology.

The configurations of the experiment are defined as follows. Data collection is followed by the simulation research methodology communication standard for our experiment. LTE advance is an advanced standard for deploying LTE, which enables multiple mobile devices to be compatible with various LTE networks. The performance is measured by the wireless communication latency time, which are represented by the variable (P_x). The minimum latency time for each route is defined as a full level performance that is marked as 1.0. The tested mobile devices are smart phones with Android 4.4.2 and up operating system, which support at least 300 Mbps max speed, e.g. HTC One (M8), Samsung Galaxy Note 3, and Samsung Galaxy Note 4.

A few method routes are simulated in the experiment and the workflow method is given in Fig. 3. Two different experiments with 2 and 4 method routes are applied for examining our algorithm and the feasibility of the model. We assume that there are 8 nodes in the simulation of 2-method-route. This experimental configuration simulates the scenario that having few method routes in which a complicated availability of the cloudlet nodes is given. For 4-method-route scenario, we produce four nodes tested in the second experiment. This configuration simulates the scenario with a few method routes that consists of less cloudlet nodes settings.

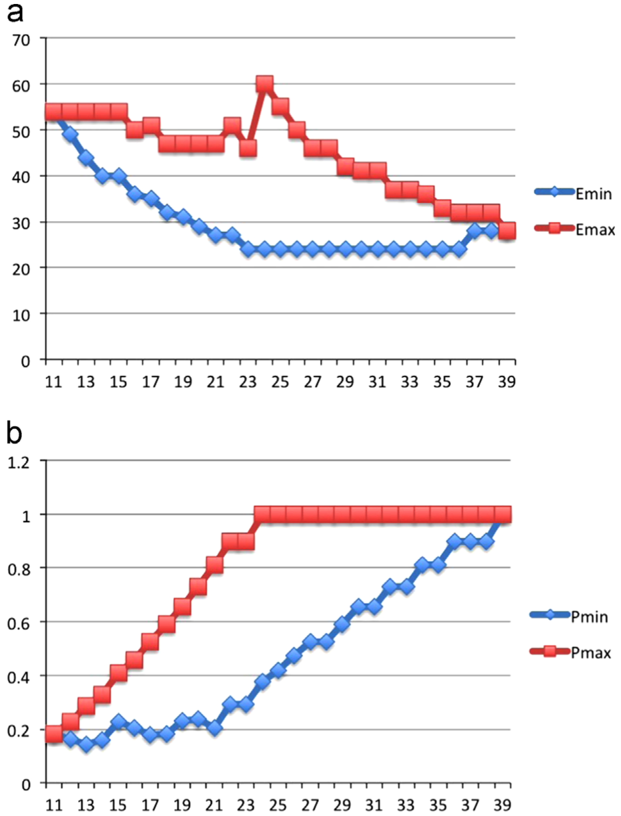

Fig. 5. Experimental results. (a) The result of DECM approach focusing on energy-aware performance (the line marked as Emin refers to the most energy-aware path and the line named Emax is the most costly path produced by adopting DECM). 98 (b) The result of DECM approach focusing on energy-aware performance (the line 99 marked as Pmax refers to the highest level of wireless communications path and the line named Pmin is the path providing lowest level of quality).

The purpose of the experiment 1 is to examine whether DECM is applicable for an operating interval associated with specific timing constraints, to examine the relationships between two main variables, P_x and E_x, and to evaluate whether there is an optimized solution at each timing constraint when two-route is deployed. Meanwhile, the second experiment is designed for testing the operability of DECM in a complicated CDLs-connected environment. Four routes provide various data in order to simulate a complex networking surrounding.

The parameters in the experiment examining 2 method routes are represented in Table 3 that explicates a variety of cloudlet nodes with diverse energy consumptions and performances based on our assumptions. Nodes represent the switches between cloud servers, which are symbolized as *Nd*. Addressing the *ECM* problem defined in Section 4.1, we present a minimum total energy cost with the performance capacities under various timing constraints for a DECM in Table 4 by using the data from Table 3.

We use the data in Table 3 to evaluate our DECM algorithm and gain the experimental results. The issue represented in Table 4 is to illustrate optimized selections of switching CDLs under different timing constraints, which are examined by two parameters, including energy consumption level (E) and performance capacity percent level (P). Under this configuration, the latency of the mobile environment is involved in the parameter P. For example, under timing constraint 32, there are four optimal options (0.9,

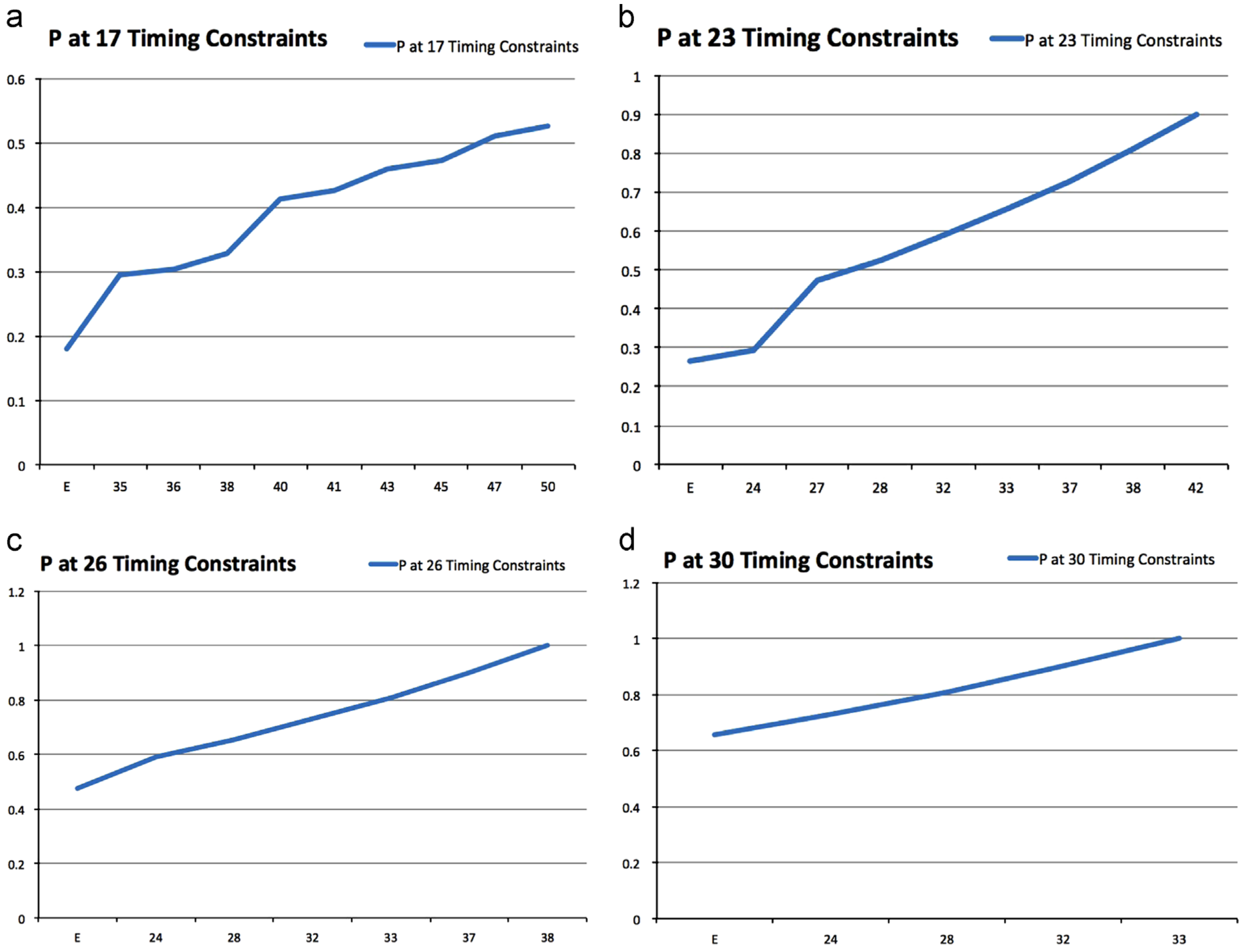

Fig. 6. Experimental results for two-route at different timing constraints. (a) The result of DECM approach stating wireless communication quality level with the corresponding energy consumption at 17 timing constraints). (b) The result of DECM approach stating wireless communication quality level with the corresponding energy consumption at 23 timing constraints). (c) The result of DECM approach stating wireless communication quality level with the corresponding energy consumption at 26 timing constraints. (d) The result of DECM approach stating wireless communication quality level with the corresponding energy consumption at 30 timing constraints.

Table 5
Energy consumptions and performances for four modules.

Nodes	M_1			M_2			M_3			M_4		
	T_1	P_1	E_1	T_2	P_2	E_2	T_3	P_3	E_3	T_4	P_4	E_4
0	1	0.8	9	2	0.7	4	1	0.6	6	3	0.9	9
	2	0.2	9	3	0.3	4	3	0.4	6	4	0.1	9
1	1	0.9	7	2	0.8	2	2	0.4	5	1	0.5	7
	2	0.1	7	5	0.2	2	3	0.6	5	2	0.5	7
2	1	0.9	10	2	0.9	5	1	0.2	5	2	0.3	4
	3	0.1	10	4	0.1	5	2	0.8	5	3	0.7	4
3	2	0.8	8	3	0.7	7	1	0.7	5	4	0.9	2
	3	0.2	8	4	0.3	7	4	0.3	5	5	0.1	2

29.0), (0.729, 24.0), (0.81, 28.0), and (**1.0**, **37.0**). The option (0.9, 29.0) means the value of P is 0.9 and the value of E is 29.0. $P\,E$. Compare with another option (1.0, 37.0), the option (0.9, 29.0) has a lower performance capacity with a lower energy consumption. The determination can be made by the priority of the user settings.

Deriving from Eq. (4), we calculate $E = \sum CostCDL$. T refers to various timing constraints, $\{T \in N | 11 \leq T \leq 39\}$. The bolded data

Table 6
Minimum total costs with computed performance capacities under various timing constraints for a DECM.

T	(P, E)	(P, E)	(P, E)	(P, E)	(P, E)
4	(0.454, 31.0)	(0.101, 26.0)	(0.34, 28.0)	(0.076, 23.0)	
5	(0.504, 26.0) (0.567, 31.0)	(0.378, 23.0)	(0.113, 22.0)	(0.09, 21.0)	(0.067, 18.0)
6	(0.648, 34.0) (0.63, 26.0)	(0.504, 25.0) (0.078, 16.0)	(0.448, 21.0)	(0.336, 18.0)	(0.101, 17.0)
7	(0.72, 29.0) (0.118, 15.0)	(0.81, 34.0) (0.63, 21.0)	(0.448, 20.0)	(0.7, 26.0)	(0.392, 16.0)
8	(0.729, 28.0) (0.7, 21.0)	(0.9, 29.0) (0.56, 16.0)	(0.72, 24.0)	(0.392, 15.0)	(0.63, 20.0)
9	(0.576, 18.0) (0.9, 24.0)	(0.13, 14.0) (0.56, 15.0)	(0.81, 23.0) (0.7, 19.0)	(0.101, 13.0)	(**1.0, 29.0**)
10	(0.576, 17.0) (**1.0, 24.0**)	(0.504, 13.0) (0.56, 15.0)	(0.151, 12.0)	(0.9, 21.0)	(0.81, 18.0)
11	(0.9, 18.0) (0.72, 13.0)	(0.504, 12.0)	(0.81, 17.0)	(**1.0, 21.0**)	(0.8, 16.0)
12	(**1.0, 18.0**)	(0.8, 13.0)	(0.72, 12.0)	(0.9, 16.0)	
13	(0.8, 12.0)	(**1.0, 16.0**)	(0.9, 15.0)		
14	(**1.0, 15.0**)	(0.8, 12.0)	(0.9, 13.0)		
15	(**1.0, 13.0**)	(0.9, 12.0)			
16	(**1.0, 12.0**)				
17	(**1.0, 13.0**)				
18	(**1.0, 18.0**)				

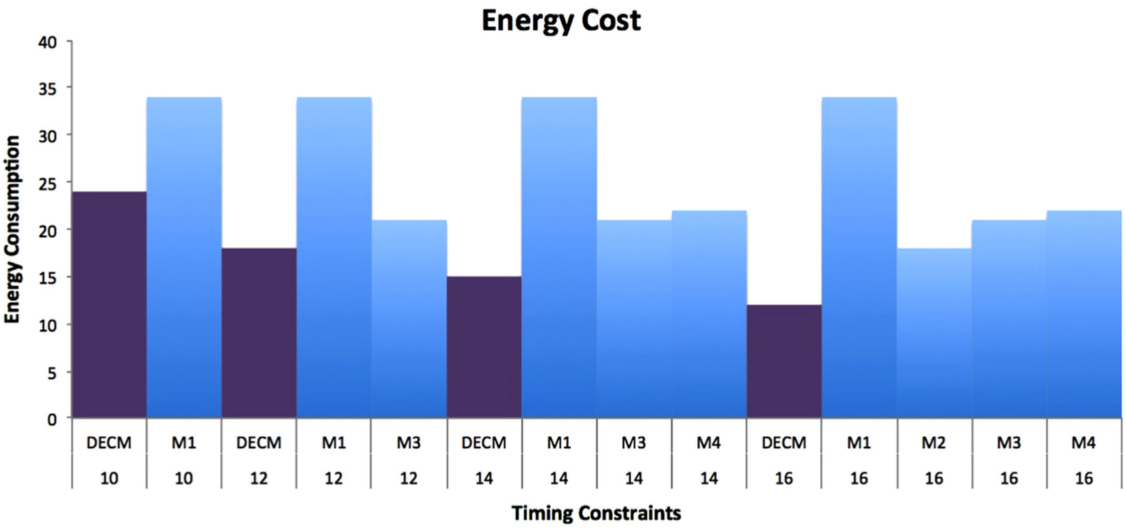

Fig. 7. Conceptual workflow of DECM for motivating example.

represent a full-level service status with the corresponding energy-saving efficiency, e.g. (1.0, 32.0).

There are a few solution options with different energy cost performances and service quality levels when the timing constraints are different. For instance, at the timing constraint *12*, there are three approaches, $(P, E)=\{(0.206, 50.0), (0.229, 54.0), (0.162, 49.0)\}$. Adding more constraints can assist mobile users to further determine the outcomes depending on the demands.

Figure 5 illustrates two results obtained from our experiment that are displayed in two graphics, Fig. 5(a) and (b). These graphics represent the implementations of DECM from both energy saving and communication perspectives.

Figure 5(a) exhibits the DECM approach focusing on energy-aware performance. *X*-axis represents a range of timing constraints that can accomplish the task, from 11 to 39, and the *Y*-axis shows the energy consumptions. Emin refers to the minimum energy cost and Emax means the maximum energy consumption. According to Fig. 5(a), the task can be completed in the interval between Emin and Emax lines. The figure depicts that using DECM can save more energy by defining a longer timing constraint.

Distinguishing from Fig. 5(a), (b) describes the scenario examining the service quality (P) by two boundaries, *maximum performance* (Pmax) and *minimum performance* (Pmin). Both Pmax and Pmin have a positive relationship with timing constraints, T. According to the representation of Fig. 5(a) and (b), energy consumptions, E have a negative relationship with communication performances, P.

Moreover, using DECM offers various outcomes at different defined timing constraints, T. Figure 6 provides four scenarios defined by the varied timing constraints, namely 17, 23, 26, and 30, which are given by four sub-figures, including Fig. 6(a)–(d). Each figure represents a trend of wireless communication performances (P) aligning with the energy consumptions (E). All outcomes from these figures illustrate that P has a positive relationship with the amount of E and there is always an optimized solution at each timing constraint. The solution is associated with the requirements of P and E.

Dissimilar to the two-route module, we further made an experiment for the four routes module and the parameters data are given in Table 5. Displayed in the table, this experiment defines four routes, namely M_1, M_2, M_3, and M_4, and four nodes.

By using DECM-Sim, we generate a set of solutions to each route. Table 6 represents optimized selections of switching CDLs under different timing constraints determined by P and E. The shortest timing constraints are 4 timing units and the longest constraints are 18 timing units.

One of the experimental results are depicted in Fig. 7. The figure represents the energy consumptions for different routes with full service performances at four timing constraints, namely 10, 12, 14, and 16. At 10 timing constraints, only DECM approach and M_1 can accomplish all nodes with full service quality, $P=1.0$. Similarly, DECM approach, M_1, M_3, and M_4 can finish all the nodes at timing constraint 14. According to the demonstration of the figure, DECM performs better in saving energy than all other method routes when delivering the full service performance.

In summary, our experimental results have provided a solid evidence showing that the proposed DECM is efficient in saving energy as well as higher-level wireless service quality (Verbelen et al., 2012). Many previous research focused on the improvement of cloud service quality throughout different fields, from architectural to operating levels (Soyata et al., 2012; Satyanarayanan et al., 2009). Jararweh et al. (2014) proposed a scalable cloudlet-based approach that aimed to reduce energy consumptions and latency by scaling up and down the service. This approach was efficient under a specific condition that implies the users can accept a scalable service delivery manner during the usages. Some other research had other focuses, such as using cloudlets for efficient data allocation (Quwaider and Jararweh, 2015), resource allocation (Bohez et al., 2015), or admission control policy (Hoang et al., 2012). Nonetheless, none of them had the same target as ours. We intended and explored to leverage use dynamic programming cloudlets to gain energy-aware MCC approach offering a high-performance service quality. Our approach has been verified as a productive method for solving the cloudlet-based service deployment in a dynamic circumstance.

6. Conclusions

This paper proposed a novel approach of using dynamic cloudlet-based MCC model, DECM, to gain the benefits of green computing. The experimental results proved that our approach was an effective mechanism that could enable mobile users to address green IT within a dynamic complicated wireless environment. Future research includes two aspects. The first focus is to examine whether DECM is applicable in multiple industries with different service requirements. Another aspect is to build up structured connections among CDLs to strengthen communications between cloud servers and mobile devices.

References

Aksanli B, Venkatesh J, Rosing T. Using datacenter simulation to evaluate green energy integration. Computer 2012;45(9) 0056–64, doi:http://doi.ieeecomputersociety.org/10.1109/MC.2012.249.

Baliga J, Ayre RW, Hinton K, Tucker R. Green cloud computing: balancing energy in processing, storage, and transport. Proc IEEE 2011;99(1):149–67. http://dx.doi.org/10.1109/JPROC.2010.2060451.

Beloglazov A, Buyya R. Energy efficient resource management in virtualized cloud data centers. In: Proceedings of the 10th IEEE/ACM international conference on cluster, cloud and grid computing. IEEE Computer Society; Melbourne, Australia, 2010. p. 826–31.

Beloglazov A, Abawajy J, Buyya R. Energy-aware resource allocation heuristics for efficient management of data centers for cloud computing. Future Gener Comput Syst 2012;28(5):755–68. http://dx.doi.org/10.1016/j.future.2011.04.017.

Berl A, Gelenbe E, Girolamo MD, Giuliani G, Meer HD, Dang MQ, et al. Energy-efficient cloud computing. Comput J 2010;53(7):1045–51. http://dx.doi.org/10.1093/comjnl/bxp080.

Bohez S, Verbelen T, Simoens P, Dhoedt B. Discrete-event simulation for efficient and stable resource allocation in collaborative mobile cloudlets. Simul Model Pract Theory 2015;50:109–29. http://dx.doi.org/10.1016/j.simpat.2014.05.006.

Bonino D, Russis LD, Corno F, Ferrero G. JEERP: energy aware enterprise resource planning. IT Prof 2013;50–6. doi:http://dx.doi.org/10.1109/MITP.2013.22.

Chan HA, Yokota H, Xie J, Seite P, Liu D. Distributed and dynamic mobility management in mobile internet: current approaches and issues. J Commun 2011;6(1):4–15. http://dx.doi.org/10.4304/jcm.6.1.4-15.

Dinh HT, Lee C, Niyato D, Wang P. A survey of mobile cloud computing: architecture, applications, and approaches. Wirel Commun Mob Comput 2013;13(18):1587–611. http://dx.doi.org/10.1002/wcm.1203.

Fesehaye D, Gao Y, Nahrstedt K, Wang G. Impact of cloudlets on interactive mobile cloud applications. In: IEEE 16th international enterprise distributed object computing conference (EDOC). IEEE; Beijing, China, 2012. p. 123–32. doi:http://dx.doi.org/10.1109/EDOC.2012.23.

Gai K, Li S. Towards cloud computing: a literature review on cloud computing and its development trends. In: 4th international conference on multimedia information networking and security (MINES). IEEE; Nanjing, China, 2012. p. 142–6. doi:http://dx.doi.org/10.1109/MINES.2012.240.

Guan L, Ke X, Song M, Song J. A survey of research on mobile cloud computing. In: Proceedings of the 10th IEEE/ACIS international conference on computer and information science. IEEE Computer Society; Sanya, China, 2011. p. 387–92. doi: http://doi.ieeecomputersociety.org/10.1109/ICIS.2011.67.

Gupta S, Roy K. Comparison of different energy minimization techniques in wireless sensor network. Energy 2013;75(18), . http://dx.doi.org/10.5120/13348-0576.

Han B, Hui P, Srinivasan A. Mobile data offloading in metropolitan area networks. ACM SIGMOBILE Mob Comput Commun Rev 2011;14(4):28–30. http://dx.doi.org/10.1145/1942268.1942279.

Hoang DT, Niyato D, Wang P. Optimal admission control policy for mobile cloud computing hotspot with cloudlet. In: IEEE wireless communications and networking conference (WCNC). IEEE; Shanghai, China, 2012. p. 3145–9. doi: http://dx.doi.org/10.1109/WCNC.2012.6214347.

Huang D. Mobile cloud computing. IEEE COMSOC Multimed Commun Tech Comm (MMTC) E-Lett 2011;6(10):27–31.

Jararweha Y, Tawalbehb L, Ababneha F, Khreishahc A, Dosarib F. Scalable cloudlet-based mobile computing model. Procedia Comput Sci 2014;34:434–41. http://dx.doi.org/10.1016/j.procs.2014.07.051.

Kumar K, Lu Y. Cloud computing for mobile users: can offloading computation save energy?. Computer 2010;43(4):51–6. http://dx.doi.org/10.1109/MC.2010.98.

Lee YC, Zomaya AY. Energy efficient utilization of resources in cloud computing systems. J Supercomput 2012;60(2):268–80. http://dx.doi.org/10.1007/s11227-010-0421-3.

Ma X, Cui Y, Stojmenovic I. Energy efficiency on location based applications in mobile cloud computing: a survey. Procedia Comput Sci 2012;10:577–84. http://dx.doi.org/10.1016/j.procs.2012.06.074.

Mezmaza M, Melabb N, Kessacib Y, Leec Y, Talbib E, Zomayac. A parallel bi-objective hybrid metaheuristic for energy-aware scheduling for cloud computing systems. J Parallel Distrib Comput 2011;71(11):1497–508. http://dx.doi.org/10.1016/j.jpdc.2011.04.007.

Miettinen AP, Nurminen JK. Energy efficiency of mobile clients in cloud computing. In: Proceedings of the 2nd USENIX conference on hot topics in cloud computing. USENIX Association; Boston, MA, USA, 2010. p. 4.

Mukherjee A, Gupta P, De D. Mobile cloud computing based energy efficient offloading strategies for femtocell network. In: Applications and innovations in mobile computing (AIMoC). IEEE; Kolkata, India , 2014. p. 28–35. doi:http://dx.doi.org/10.1109/AIMOC.2014.6785515.

Qiu M, Chen Z, Ming Z, Qin X, Niu J. Energy-aware data allocation with hybrid memory for mobile cloud systems. IEEE Syst J 2014:1–10. doi:http://dx.doi.org/10.1109/JSYST.2014.2345733.

Quwaider M, Jararweh Y. Cloudlet-based efficient data collection in wireless body area networks. Simul Model Pract Theory 2015;50:57–71. http://dx.doi.org/10.1016/j.simpat.2014.06.015.

Sabharwal M, Agrawal A, Metri G. Enabling green IT through energy-aware software. IT Prof 2013:19–27. doi:http://doi.ieeecomputersociety.org/10.1109/MITP.2012.104.

Satyanarayanan M. Mobile computing: the next decade. ACM SIGMOBILE Mob Comput Commun Rev 2011;15(2):2–10. http://dx.doi.org/10.1145/2016598.2016600.

Satyanarayanan M, Bahl P, Caceres R, Davies N. The case for vm-based cloudlets in mobile computing. IEEE Pervasive Comput 2009;8(4):14–23. http://dx.doi.org/10.1109/MPRV.2009.82.

Song W, Su X. Review of mobile cloud computing. In: IEEE 3rd international conference on communication software and networks (ICCSN). IEEE; Xi'an, China, 2011. p. 1–4. doi:http://dx.doi.org/10.1109/ICCSN.2011.6014374.

Soyata T, Muraleedharan R, Funai C, Kwon M, Heinzelman W. Cloud-vision: Real-time face recognition using a mobile-cloudlet-cloud acceleration architecture. In: IEEE symposium on computers and communications (ISCC). IEEE; Cappadocia, Turkey, 2012. p. 059–66. doi:http://dx.doi.org/10.1109/ISCC.2012.6249269.

Verbelen T, Simoens P, Turck FD, Dhoedt B. Cloudlets: bringing the cloud to the mobile user. In: Proceedings of the 3rd ACM workshop on mobile cloud computing and services. ACM; Low Wood Bay, UK, 2012. p. 29–36. doi:http://dx.doi.org/10.1145/2307849.2307858.

Xiao Z, Song W, Chen Q. Dynamic resource allocation using virtual machines for cloud computing environment. IEEE Trans Parallel Distrib Syst 2013;24(6):1107–17. http://dx.doi.org/10.1109/TPDS.2012.283.

Yang L, Zhu X, Chen H, Wang J, Yin S, Liu X. Real-time tasks oriented energy-aware scheduling in virtualized clouds. IEEE Trans Cloud Comput 2014;2(2):168–80. http://dx.doi.org/10.1109/TCC.2014.2310452.