ISSN: 2277-9655
(I2OR), Publication Impact Factor: 3.785

IJESRT

INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY

REDUCING SMARTPHONE ENERGY CONSUMPTION WITH TASK OFFLOADING

Shaikh Nisar Salim, Prof. Raut S.Y.
Computer Engineering Department, PREC Loni Ahmednagar, Maharashtra, India

ABSTRACT

Assignment offloading from cell phones to the cloud is a promising system to improve the processing capacity of cell phones and draw out their battery life. Then again, assignment offloading presents a correspondence cost for those gadgets. In this manner, the thought of the correspondence expense is essential for the adequacy of errand offloading. To make undertaking offloading valuable, one of the difficulties is to evaluate the vitality devoured in correspondence exercises of assignment offloading. Precise vitality estimation models will empower these gadgets to settle on the right choices with reference to regardless of whether to perform errand offloading, taking into account the vitality expense of the correspondence exercises. Basically, if the offloading procedure expends less vitality than preparing the errand on the gadget itself, then the undertaking is offloaded to the cloud. To outline a vitality mindful offloading methodology, we create vitality models of the WLAN, third-era, and fourth-era interfaces of cell phones. These models make cell phones able to do precisely assessing the vitality expense of undertaking offloading. We accept the models by directing a broad arrangement of investigations on five cell phones from distinctive merchants. The trial results demonstrate that our estimation models precisely assess the vitality required to offload errands.

KEYWORDS: Portable figuring, distributed computing, cell phones, offloading choice, vitality sparing, WLAN vitality, 3G vitality, 4G vitality, vitality estimation.

INTRODUCTION

Cell phones have novel limitations, to be specific constrained battery vitality, handling ability, and memory limit. In the course of the most recent couple of years, quick advances in semiconductor advances have reduced some of those requirements. In any case, the restricted battery vitality requirement has not been agreeably tended to. As per Moore's law, the quantity of transistors on an incorporated circuit duplicates at regular intervals. Conversely, battery limit increments just by 5% consistently [1]. This suggests the hole between vitality request and supply develops by 4% every year [2], [3]. In the late years, the issue has gotten to be common among cell phone clients, while the cell phones are turning out to be progressively prevalent on account of their abilities and functionalities. With intense working frameworks (e.g., Windows Mobile, Android, Apple iOS, BlackBerry, and Symbian), cell phones can run advance applications that are practically like desktop PC applications. Each cell phone application performs a progression of errands, with every undertaking executing particular calculations on a given information.

The need to lessen the vitality utilization of cell phones has been drawing in endeavors from numerous specialists [4] , [6]. Numerous strategies and methods have been proposed in writing. Keen batteries, force booking, proficient working frameworks and applications, productive graphical client interfaces, vitality mindful correspondence conventions, and errand offloading are all illustrations of these philosophies and strategies [7]. Undertaking offloading is a promising method to decrease vitality utilization in cell phones; extraordinarily, with the development of fast broadband remote Internet access. That is on the grounds that fast systems expand the association accessibility to the registering assets behind the Internet. Utilizing the offloading procedure, cell phones can offload their overwhelming errands to remote machines and spare their vitality of executing the assignment locally [8], [9].

In the period of Cloud Computing (CC), the vitality imperative on cell phones can be backed off offloading substantial assignments from cell phones to the cloud [10], [11]. The cell phone can spare vitality by offloading overwhelming assignments to the cloud, and afterward the cloud executes the undertakings and furnishes the cell phone with the outcomes. For instance, a cell phone can transfer a video _le to a cloud and demand to encode the _le into a coveted organizing the cell phone capacity with less vitality utilization than doing the encoding on the gadget. Errand offloading will get to be basic for the Information and Communication Technology (ICT) sooner rather than later in light of the fact that CC will be an overwhelming administrator for versatile processing [12], [13]. Portable information stockpiling and information handling will occur on the cloud, and a promising approach to have this sort of ICT structure is to utilize offloading procedures [14], [15].

Undertaking offloading is a basic system in light of the fact that now and again it expands the vitality utilization of cell phones. To represent this, if a cell phone needs to perform an undertaking calculation where errand information exists on the cell phone, there are two situations: either execute the assignment locally (S1), or offload the assignment to the cloud (S2). Expect that the cell phone expends vitality equivalent to E(S1) when the undertaking is executed locally. Also, accept that the cell phone expends vitality equivalent to E(S2) when the errand is offloaded, which includes transferring of undertaking information and downloading of assignment results to and from the cloud, separately. For this situation, offloading is just valuable if E(S2) < E(S1).

Keeping in mind the end goal to make the offloading advantageous, the vitality expense of offloading for a given assignment ought to be assessed to contrast it and the vitality expense of executing the undertaking locally. From a cell phone perspective, the vitality expended amid assignment offloading is predominantly brought on by the systems administration exercises. The center of this study is creating vitality models to gauge the expense of errand offloading brought about by the systems administration exercises. Exceptionally, we display the vitality cost at the application level considering every one of the subtle elements of the system stack (i.e., Transmission Control Protocol (TCP), Media Access Control (MAC), and Physical layer (PHY)). Assessing the vitality devoured because of nearby errand execution is past the extent of this study as we study it in a dynamic work, and the writing shows other valuable methodologies that can be embraced [16], [17].

This study broadens our past work on examination of the practicality of undertaking offloading to regardless of whether a cell phone can spare vitality by offloading assignments to the cloud [18]. We led countless on mainstream cell phones and genuine mists with four distinctive offloading situations. The outcomes uncovered the capability of assignment offloading to the cloud and the advantage of offloading to the cloud as far as vitality sparing. The cell phone can spare vitality somewhere around 30% and 70% by offloading substantial errands to the cloud [18].

Given that an undertaking includes execution of particular code on given information, we have four conceivable offloading situations, as recorded in Table 1 and clarified in what takes after.

TABLE 1. Offloading scenarios from the viewpoint of Smartphone.

Scenario	Data	Execution	Networking Activities
S1	local	local	none
S2	local	cloud	upload task data and download task results
S3	cloud	local	download task data
S4	cloud	cloud	download task results

S1: In this situation, the info information is accessible locally on the cell phone and undertaking execution happens on the cell phone also. This is the ordinary situation where no offloading happens. We utilize this situation as a kind of perspective case for correlation reason.

S2: The second situation is the place the undertaking execution happens on the cloud however the assignment information exists locally on the cell phone. In this situation, the cell phone needs to transfer the errand information to the cloud and after that download the assignment results.

S3: The third situation is the place the undertaking execution is performed locally on the cell phone,

however the assignment information exists on the cloud. In this situation, the cell phone needs to download the errand information and perform the undertaking execution locally.

S4: In this situation, the information is accessible on the cloud and errand execution happens in the cloud too. Subsequently, the cell phone simply needs to download the assignment results.

In this work, we create and approve numerical models for the vitality that cell phones expend amid system exercises for errand offloading. We consider in our models the most widely recognized system interfaces: WLAN and 3G/4G. We lead probes prevalent cell phones (i.e., HTC Nexus One, LG Nexus 4, Samsung Galaxy S3, BlackBerry Z10, and Samsung Galaxy Note 3) to accept our vitality models. The trial results uncover that our vitality estimation models can assess the vitality precisely.

In this paper, we make the accompanying commitments:
1) We acquaint models with assessment the vitality devoured in a cell phone to perform assignment offloading:
a) File downloading utilizing WLAN and 3G/4G system interfaces; and
b) File transferring utilizing WLAN and 3G/4G system interfaces.
2) We created models so that give a precise estimation to the aggregate vitality expended for undertaking offloading by just taking the measure of information that the cell phone would exchange for assignment offloading as a data.
3) We accept the vitality models by method for execution and estimation. In these examinations, we measure the real vitality expended in the cell phones for each of the previously stated system exercises.

LITRATURE SURVEY

Early years

Devices that combined telephony and computing were first conceptualized by Tesla 1909 and Theodore Paraskevakos in 1971 and patented in 1974, and were offered for sale beginning in 1993. Paraskevakos was the first to introduce the concepts of intelligence, data processingand visual display screens into telephones. In 1971, while he was working with Boeing in Huntsville, Alabama, demonstrated a transmitter andreceiver that provided additional ways to communicate with remote equipment, however it did not yet have general purpose PDA applications in a wireless device typical of smartphones. They were installed at Peoples' Telephone Company in Leesburg, Alabama and were demonstrated to several telephone companies. The original and historic working models are still in the possession of Paraskevakos.

Forerunner

The first mobile phone to incorporate PDA features was an IBM prototype developed in 1992 and demonstrated that year at the COMDEXcomputer industry trade show. The prototype demonstrated PDA features as well as other visionary apps like maps, stocks and news incorporated with a cellular phone. A refined version of the product was marketed to consumers in 1994 by BellSouth under the name Simon Personal Communicator. The Simon was the first cellular device that can be properly referred to as a "smartphone", although it was not called that in 1994. In addition to its ability to make and receive cellular phone calls, Simon was able to send and receive faxes and emails and included several other apps like address book, calendar, appointment scheduler, calculator, world time clock, and note pad through its touch screen display. Simon is the first smartphone to be incorporated with the features of a PDA. The term "smart phone" first appeared in print in 1995, for describing AT&T's "PhoneWriter Communicator" as a "smart phone".

PDAs

In the late 1990s, many mobile phone users carried a separate dedicated PDA device, running early versions of operating systems such as Palm OS,BlackBerry OS or Windows CE/Pocket PC.[1] These operating systems would later evolve into mobile operating systems. In March 1996, Hewlett-Packard released the OmniGo 700LX, which was a modified 200LX PDA that supported a Nokia 2110-compatible phone and had integrated software built in ROM to support it. The device featured a 640x200 resolution CGA compatible 4-shade gray-scale LCD screen and could be used to make and receive calls, text messages, emails and faxes. It was also 100% DOS 5.0 compatible, allowing it to run thousands of existing software titles including early versions of Windows.

In August 1996, Nokia released the Nokia 9000 Communicator which combined a PDA based on the GEOS V3.0 operating system from Geoworks with a digital cellular phone based on the Nokia 2110. The two devices were fixed together via a hinge in what became known as a clamshell design. When opened, the display was on the inside top surface and with a physical QWERTY keyboard on the bottom. The personal organizer provided e-mail, calendar, address book, calculator and notebook with text-based web browsing, and the ability to send and receive faxes. When the personal organizer was closed, it could be used as a digital cellular phone.

In June 1999, Qualcomm released a "CDMA Digital PCS Smartphone" with integrated Palm PDA and Internet connectivity, known as the "pdQ Smartphone".

In early 2000, the Ericsson R380 was released by Ericsson Mobile Communications, and was the first device marketed as a "smartphone". It combined the functions of a mobile phone and a PDA, supported limited web browsing with a resistive touchscreen utilizing a stylus.

In early 2001, Palm, Inc. introduced the Kyocera 6035, which combined a PDA with a mobile phone and operated on Verizon. It also supported limited web browsing.

Smartphones before Android, iOS and BlackBerry, typically ran on Symbian, which was originally developed by Psion. It was the world's most widely used smartphone operating system until the last quarter of 2010.

Mass adoption

In 1999, the Japanese firm NTT DoCoMo released the first smartphones to achieve mass adoption within a country. These phones ran on i-mode, which provided data transmission speeds up to 9.6 kbit/s. Unlike future generations of wireless services, NTT DoCoMo's i-mode used cHTML, a language which restricted some aspects of traditional HTML in favor of increasing data speed for the devices. Limited functionality, small screens and limited bandwidth allowed for phones to use the slower data speeds available.[19]

The rise of i-mode helped NTT DoCoMo accumulate an estimated 40 million subscribers by the end of 2001. It was also ranked first in market capitalization in Japan and second globally. This power would wane in the face of the rise of 3G and new phones with advanced wireless network capabilities.

Outside of Japan smartphones were still rare until the introduction of the Danger Hiptop in 2002, which saw moderate success in the US as the T-Mobile Sidekick. Later, in the mid-2000s, devices based on Microsoft's Windows Mobile started to gain popularity among business users in the U.S. The BlackBerry later gained mass adoption in the U.S., and American users popularized the term "CrackBerry" in 2006 due to its addictive nature. The company first released its GSM BlackBerry 6210, BlackBerry 6220, and BlackBerry 6230 devices in 2003.

Symbian was the most popular smartphone OS in Europe during the middle to late 2000s. Initially, Nokia's Symbian devices were focused on business, similar to Windows Mobile and BlackBerry devices at the time. From 2006 onwards, Nokia started producing entertainment-focused smartphones, popularized by the Nseries. In Asia, with the exception of Japan, the trend was similar to that of Europe.

iOS and Android

In 2007, Apple Inc. introduced the iPhone, one of the first smartphones to use a multi touch interface. The iPhone was notable for its use of a large touchscreen for direct finger input as its main means of interaction, instead of a stylus, keyboard, or keypad typical for smartphones at the time.

2008 saw the release of the first phone to use Android called the HTC Dream (also known as the T-Mobile G1). Android is an open-source platform founded by Andy Rubinand now owned by Google. Although Android's adoption was relatively slow at first, it started to gain widespread popularity in 2010, and now dominates the market.

These new platforms led to the decline of earlier ones. Microsoft, for instance, started a new OS from scratch, called Windows Phone. Nokia abandoned Symbian and partnered with MS to use Windows Phone on its smartphones. Windows Phone then became the third-most-popular OS. Palm's web OS was bought by Hewlett-Packard and later sold to LG Electronics for use on LG smart TVs. BlackBerry Limited, formerly known as Research In Motion, also made a new platform based on QNX, BlackBerry 10.

The capacitive touchscreen also had a knock-on effect on smartphone form factors. Before 2007 it was

ISSN: 2277-9655
(I2OR), Publication Impact Factor: 3.785

common for devices to have a physical numeric keypad or physical QWERTY keyboard in either a candy bar or sliding form factor. However, by 2010, there were no top-selling smartphones with physical keypads.

SYSTEM MODEL

Our framework comprises of two noteworthy parts, cell phones (i.e., client gear, UE) and Cloud Computing (CC), both connected to the Internet, as delineated in Fig. 1. The cell phones are associated with the Internet through a WLAN access point or a cell information system base station (3G/4G). These cell phones give all of portable processing functionalities to the end clients by means of distinctive applications. Then again, the CC part comprises of cloud server farm and cloud supplier, which are available through the Internet. The cloud gives the end clients (e.g., cell phone clients) with the greater part of the CC functionalities that are required for versatile registering.

In the offloading system, cell phones get to the cloud through the Internet. Accordingly, offloading is considered as a Network Related Application (NRA). Toward the start of examining NRA, system interfaces (i.e., 3G/4G and WLAN) ought to be considered in light of the fact that each of these interfaces has its own particular attributes, for example, bolstered information rate. Therefore, every system interface expends unequal measure of vitality. What's more, the Internet conventions, specifically, the Hypertext Transfer Protocol (HTTP) and the File Transfer Protocol (FTP) should be considered. The system interfaces and conventions are the main considerations that influence the vitality expenses of errand offloading.

We introduce a broad assessment of the vitality expenses of an arrangement of cell phones with an extensive number of trials. We tentatively assess the vitality cost on cell phones when the offloading strategy is utilized over distinctive system interfaces and Internet conventions. We led our analyses

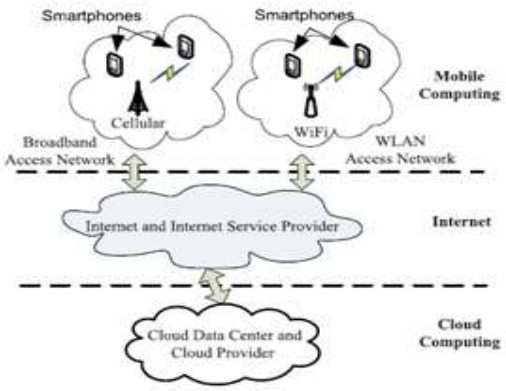


Fig1: The System Model

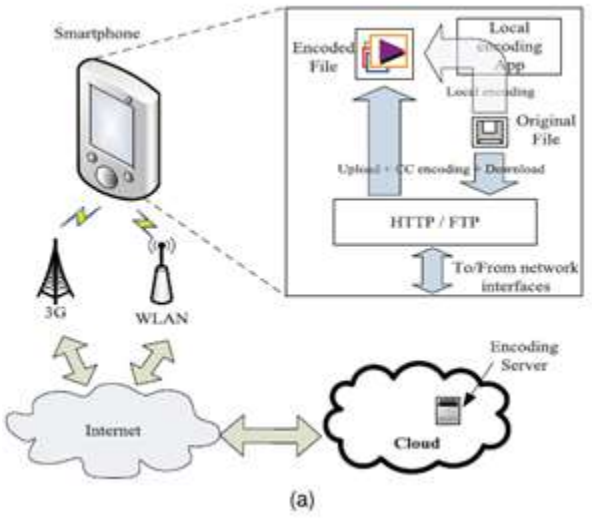
in two wide test situations identified with the area of the undertaking information as delineated in Fig. 2. In Fig. 2(a), the errand information is accessible on the cell phone itself while in Fig. 2(b) the undertaking information is accessible in the cloud. There are four situations identified with the area of the assignment information as takes after. The first situation compares to S1, where there is a neighborhood assignment execution and the errand information exists on the cell phone, as appeared by ``Local encoding" bolt in Fig. 2(a). The second situation compares to S2, where transferring the assignment information, doing the undertaking calculation (encoding) by the cloud, and downloading the errand result is introduced by the ``Upload C CC encoding C Download" bolt in Fig. 2(a). The third situation relates to S3, where there is a nearby assignment execution and the errand information is downloaded from the cloud, as appeared by the ``Download C Local encoding" bolt in Fig. 2(b). The fourth situation relates to S4, where the undertaking information exists in the cloud and the assignment executed on the cloud, and the undertaking result is just downloaded, as displayed by the ``CC encoding C Download" bolt in Fig. 2(b). For transferring and downloading _les to and from the cloud, we consider the vitality ramifications of: (i) utilizing the HTTP and FTP conventions at the application level; and (ii) utilizing the 3G and WLAN interchanges at the remote interface level. Utilizing Fig. 2(a), we directed the examinations to assess the vitality expense of performing file encoding locally on a cell phone, and the vitality expense of performing the same operation in the cloud remotely. Essentially, utilizing Fig. 2(b), we directed the examinations to assess the vitality expense of downloading an encoded

ISSN: 2277-9655
(I2OR), Publication Impact Factor: 3.785

file, and the vitality expense of downloading the file and performing encoding on the cell phone.

Thusly, we performed the examinations with 13 cases for Fig. 2 as recorded in Table 2. A piece of our outcomes is appeared in Fig. 3. The outcomes uncover that the FTP convention is a vitality efficient application convention. In this way, we consider in this work the FTP convention utilizing of both the 3G/4G and WLAN networks. In the accompanying, we create numerical models for the vitality devoured in cell phones. Specifically, we create four vitality models that give cell phones the capacity to gauge the vitality devoured for offloading any given undertaking.

Since the vitality expense of errand offloading begins from assignment information exchanging (i.e., transferring and downloading), there are four instances of errand information exchanging in the event that we consider the two sorts of cell phone systems. For a given errand, a cell phone needs two sorts of data: the system sort to pick the relating vitality model, and the measure of undertaking information that would be exchanged. By this data, the cell phone correctly ascertains the vitality cost for offloading the given undertaking, and after that it can settle on the offloading choice taking into account the ascertained expense. Besides, we tentatively accept the created models by actualizing an arrangement of tests for every model. We set up our trials as per our framework model and measure the real vitality devoured by a cell phone.



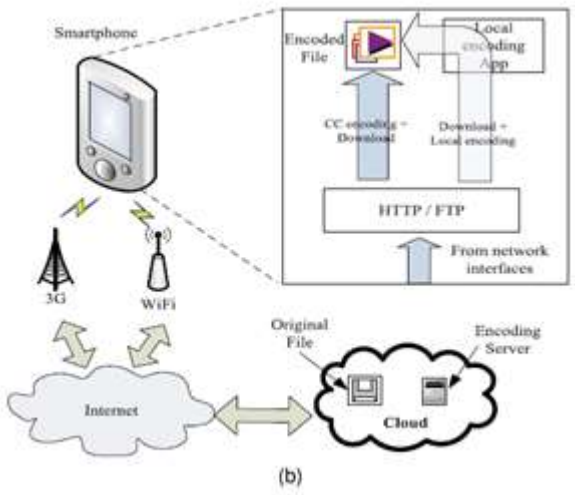


FIGURE 2. Task offloading scenarios. (a) Encoding scenarios where the task data (Original file) exists on the smartphone. (b) Encoding scenarios where the task data (Original file) exists on the CC.

CONCLUSIONS

Augmenting the capacities of cell phones is conceivable by undertaking offloading to the cloud. In any case, assessing the vitality expended in errand offloading is pivotal to making undertaking offloading Beneficial, which happens just when the vitality devoured in the offloading procedure is not exactly the vitality expended without it. In this manner, the significant test in assignment offloading is to assess precisely the vitality devoured amid the system exercises of errand offloading. In this work, we created scientific models to evaluate this vitality utilization. We considered the subtle elements of the system stack from lower systems administration layers up to high layers. The proposed vitality models of WLAN, 3G, and 4G interfaces permit cell phones to settle on right offloading choices. Besides, our models help for undertaking offloading as well as opens new entryway for vitality arrangements that require foreseeing the vitality utilization. We tentatively approved those models by directing an arrangement of trials on an arrangement of cell phones and measuring the vitality expended amid assignment offloading. The test results uncover that our vitality estimation models can evaluate vitality cost with suffcient exactness. The models simply need to know the measure of exchanged information and some framework parameters, and they can give great estimations of vitality cost.

REFERENCES

1. A. P. Miettinen and J. K. Nurminen, ``Energy ef_ciency of mobile clients in cloud computing," in *Proc. 2nd USENIX Conf. Hot Topics Cloud Comput. (HotCloud)*, 2010, p. 4.
2. J. A. Paradiso and T. Starner, ``Energy scavenging for mobile and wirelesselectronics," *IEEE Pervasive Comput.*, vol. 4, no. 1, pp. 18_27, Jan. 2005.
3. S. Robinson, ``Cellphone energy gap: Desperately seeking solutions," Strategy Analytics, Chicago, IL, USA, Tech. Rep., Mar. 2009.
4. A. Kansal and F. Zhao, ``Fine-grained energy pro_ling for power-aware application design," *ACM SIGMETRICS Perform. Eval. Rev.*, vol. 36, no. 2, pp. 26_31, Aug. 2008.
5. N. Vallina-Rodriguez, P. Hui, J. Crowcroft, and A. Rice, ``Exhausting battery statistics: Understanding the energy demands on mobile handsets," in *Proc. 2nd ACM SIGCOMM Workshop Netw., Syst., Appl. Mobile Handhelds (MobiHeld)*, 2010, pp. 9_14.
6. G. P. Perrucci, F. H. P. Fitzek, and J. Widmer, ``Survey on energy consumption entities on the smartphone platform," in *Proc. IEEE 73rd Veh. Technol. Conf.*, May 2011, pp. 1_6.
7. K. Naik, ``A survey of software based energy saving methodologies for handheld wireless communication devices," Dept. Electron. Commun. Eng., Univ. Waterloo, Waterloo, ON, Canada, Tech. Rep. 2010-13, 2010.
8. X. Ma, Y. Zhao, L. Zhang, H. Wang, and L. Peng, ``When mobile terminals meet the cloud: Computation of_oading as the bridge," *IEEE Netw.*, vol. 27, no. 5, pp. 28_33, Sep./Oct. 2013.
9. W. Zhang, Y. Wen, J. Wu, and H. Li, ``Toward a uni_ed elastic computing platform for smartphones with cloud support," *IEEE Netw.*, vol. 27, no. 5, pp. 34_40, Sep./Oct. 2013
10. K. Kumar and Y.-H. Lu, ``Cloud computing for mobile users: Can of_oading computation save energy?" *Computer*, vol. 43, no. 4, pp. 51_56, Apr. 2010.
11. M. Altamimi and K. Naik, ``The concept of a mobile cloud computing to reduce energy cost of smartphones and ICT systems," in *Proc. 1st Int. Conf. Inf. Commun. Technol. Fight Against GlobalWarming (ICT-GLOW)*, Aug. 2011, pp. 79_86.
12. S. Perez. (Aug. 4, 2009). *Why Cloud Computing is the Future of Mobile*. [Online]. Available: http://readwrite.com/2009/08/04/why_cloud_computing_is_the_future_of_mobile
13. A. Manjunatha, A. Ranabahu, A. Sheth, and K. Thirunarayan, ``Power of clouds in your pocket: An ef_cient approach for cloud mobile hybrid application development," in *Proc. IEEE 2nd Int. Conf. Cloud Comput. Technol. Sci. (CloudCom)*, Nov./Dec. 2010, pp. 496_503.
14. J. Kim, ``Architectural patterns for service-based mobile applications," in *Proc. IEEE Int. Conf. Service-Oriented Comput. Appl. (SOCA)*, Dec. 2010, pp. 1_4.
15. X. Zhang, A. Kunjithapatham, S. Jeong, and S. Gibbs, ``Towards an elastic application model for augmenting the computing capabilities of mobile devices with cloud computing," *Mobile Netw. Appl.*, vol. 16, no. 3, pp. 270_284, Jun. 2011.
16. P. Bellasi,W. Fornaciari, and D. Siorpaes, ``Predictive models for multimedia applications power consumption based on use-case and OS level analysis," in *Proc. Design, Autom. Test Eur. Conf. Exhibit. (DATE)*, Apr. 2009, pp. 1446_1451.
17. S. Hao, D. Li, W. G. J. Halfond, and R. Govindan, ``Estimating Android applications' CPU energy usage via bytecode pro_ling," in *Proc. 1st Int. Workshop Green Sustain. Softw. (GREENS)*, Jun. 2012, pp. 1_7.
18. M. Altamimi, R. Palit, K. Naik, and A. Nayak, ``Energy-as-a- Service (EaaS): On the ef_cacy of multimedia cloud computing to save smartphone energy," in *Proc. IEEE 5th Int. Conf. Cloud Comput. (CLOUD)*, Jun. 2012, pp. 764_771.