# A proposed Model of Smartphone Energy-Aware for Mobile Learning

Mohamed A. Mohamed, Ibrahim Y. Abdel-Baset

**Abstract**— Energy management efficiency is paramount in modern smartphones. The diverse range of wireless interfaces, sensors, and the increasing popularity of power-hungry applications that take advantage of these resources can reduce the battery life of mobile handhelds to few hours of operation. In this paper, we explore the problem of supporting efficient usage of smartphone battery for the mobile learning. We propose and implement hybrid online/offline model based architecture to extend the battery life of smartphone. Using knowledge of the relevance and current network/system condition, we present scheduling algorithms that determine which M-Learning contents are delivered to the devices. Our experimental results indicate the proposed model is energy efficient, which saves battery life of the mobile device by 28.6% for Wi-Fi and 66.7% for 3G connections.

**Index Terms**— Mobile learning, M-learning Application, Offline Access, Smartphone, Energy consumption.

## 1 Introduction

TODAY, the more and more rapid development of the information and communications technology (ICT) contributes to the increasing abilities of the mobile devices (cell phones, Smartphones, PDAs, etc.) and wireless communications, which are the main parts of mobile learning (M-Learning). Mobile devices offer an attractive environment for learning; however, they have inherent technological constrains that need to be taken into account such as the device capabilities (screen size, battery life, and software constrains), and the network connection (availability, bandwidth, and the communication cost) [1]. On the other hand, for the management of such type of education. The most of the mobile applications that introduced to the market place are highly consuming energy, energy efficiency of the mobile application is an important concern for energy restricted embedded M-Learning system [2].

Applications operated on mobile devices have generally been designed for either online or offline use. Both types of mobile applications tend to use the same form of browser to interact with a user. Online applications enjoy continual access to an enterprise server. Offline applications, in contrast, operate with minimal or no contact with an enterprise server [3].

More specifically, an online mobile application can access data on the enterprise serve whenever needed; thereby possibly obviating any need to store data locally. Also, an online mobile application generally suffers from unpredictable latency. When the online application transmits a request to the server, the response time depends upon the level of usage of the mobile device's wireless network in addition to any congestion on the server. Further, usage of the online application consumes battery power of mobile device result from continual access to an enterprise server (e.g., central database server). Finally, the online application may be geographically limited, depending on the extent of the wireless network.

In contrast to an online application, an offline mobile application does not enjoy continual access to data maintained by the enterprise server. Although the offline application may always be usable, it will not always have fresh data, and it can only access data that were copied to it. Because of the intrequent rate of data synchronization, the offline mobile applications are not suitable for use with data that are highly dynamic. In addition, an offline mobile application is often required to maintain a transaction log of all data changes made by the application in order to facilitate synchronization.

Further, mobile client applications that attempt to provide significant functionality to users tend to require robust software and/or hardware configurations. In fact, these type applications can drain 30% or 40% of a mobile device's battery [4]. In one embodiment, an intelligent client agent and a method for using the client agent to operate a hybrid online/offline client application are provided.

In this paper, we provide a proposed model to operate a hybrid online/offline for M-Learning applications based on three stages; decision, scheduling, and caching/storage. The proposed model aims to minimize the power consumption of mobile battery, and decreasing on network traffic. Two approaches are taken to achieve our aims; first approach, scheduling algorithm to control of decision M-Learning application for receiving contents. While the other approach, caching technique which presents an appropriate policy for storing receiving data. We scope and focus on first approach (i.e. scheduling algorithm).

The rest of this paper is structured as follows: Next section present contribution of this paper. Section 3 presents the proposed model as an addition for M-Learning system structure. Section IV shows the structure of the proposed model. While section 4 displays scheduling algorithm and mathematical model. Section 5 experimental setup and the followed section evaluate the procedures considering empirical measurements and investigating various communication impacts the performance and energy efficiency. Finally, Section 6 concludes the paper identifying future research directions.

## 2 M-learning Systems

According to MoLeNET, M-Learning can be broadly defined as 'the exploitation of ubiquitous handheld technologies, together with wireless and mobile phone networks, to facilitate, support, enhance and extend the reach of teaching and learning [5]. The three components composing a mobile learning

International Journal of Scientific & Engineering Research, Volume 8, Issue 4, April-2017
ISSN 2229-5518

environment are: mobile devices, wireless technology and beneficiaries [6]. Orr et al. [7] provided a study that focuses on technical as well as pedagogical issues for instructional technology. It reviews the affordance and constrains of M-Learning and addresses how it is being deployed as a supplement to electronic learning (E-Learning) or in addition to a traditional classroom. Sha et al. [8] presented a conceptual study of how theories and methodology of self-regulated learning an active area in conceptual educational psychology are inherently suited to address the issues originating from the defining characteristics of M-Learning: enabling studded-centered, personal, and ubiquitous learning. Cox et al. [9] reviews the wide range of technological and educational research changes that have taken place over the last 40 years, the affordances these provide, and the consequent implications for research methods and issues regarding investigating the impact of formal and informal learning. However, most of the previous reviews have focused on the introduction and summary of M-Learning, emphasizing studies on how education and learning should be developed, but do not provide a means to the current reality of mobile devices provide a direct and new philosophy for teaching and learning methodologies since it can it can provide rich, fast, dynamic, and robust applications [10].

In this paper, we envision a technical solution for M-Learning systems, in which scheduling/cache algorithm in the proposed model employs to help mobile devices to optimally fetch M-Learning contents and efficiently storing. The goal of the proposed model architecture is to minimize battery power consumption and saving bandwidth. The main contributions of this paper are:

1. Designing a proposed Schedule/Cache model to fetch learning content from online learning networks for delivering and caching the contents efficiently to mobile users.
2. Developing mechanism control for optimizing data transfer to mobile devices based on a proposed model that minimizes the power consumption of mobile battery and decreasing on network traffic for mobile devices.
3. Evaluation for the proposed model comparing mobile application with using proposed model and without based on system that executes completely at mobile devices using same simulation parameter software/hardware. The results from the simulations confirm the correctness and efficiency of our proposed model (Section 6).

Multiple factors contribute to the non-uniform nature of the problem at: (i) Mobile users have different/personalized viewing needs and preferences, (ii) Learning contents are diverse in size and importance, and (iii) Network conditions may be highly dynamic due to fading, shadowing, interference, and congestion on wireless/wired links.

## 3 Modified System Architecture

The M-Learning system aims to provide learning services to its users anywhere and anytime. Hence, it is expected to bear the specific characteristic of the mobile devices mobility. This system should be able to provide particular functions encompassing the principles and theories of M-Learning. In the following subsections, we will describe the functional layers of adopted M-Learning system and proposed scheduling model. Fig. 1 describes the adopted framework of M-Learning system [11]. It is a four tier M-Learning system comprising of Data layer, function logic layer, presentation layer, and user layer. The presentation layer would be present in the client devices, whereas the function logic layer and data layer are server side layers

### 3.1 Data Layer

The data layer is responsible for providing data sources to the applications. The data layer of adopted M-Learning system consists of learning resource database, teaching database, outdoor assignment database, system database, educational blogs database, examination & training database, help & support database, and games database [12].

### 3.2 Function Logic Layer

The most important part of M-Learning system is the function logic layer. All the features of the system are placed in this layer. It deals with the application requests of the presentation layer and does logical estimation of the results. The data are then invoked by the system from the related databases; provided the logical estimation is legal. The data layer deals with the invoked data and then returns the results to the presentation layer. The function logic layer of the M-Learning system consists of the two sets of modules: M-Learning modules and administrative support services modules [13].

### 3.3 Presentation Layer

The presentation layer acts as an interactive interface between users and the system. It is implemented on the client devices and its responsibility is to deal with the dialog between users and system. The client devices are primarily categorized as intelligent mobile phones (smartphones), PDAs, and tablet PCs. Each of these comprised of their respective hardware and software [14]. We will add the proposed schedule model into the presentation layers as a modification of adopted framework of M-Learning system, which will be describe in details in the next section.

### 3.4 User Layer

The M-Learning system is utilized by three types of users: students, faculty members, and administrators. Separate user interfaces for students and faculty members are provided by the presentation layer [15].


- *Mohammed A. Mohammed is Professor and Head of Department of electronics and communication engineering at faculty of engineering in Mansoura University, Egypt, PH-00201007206064.*
  *E-mail: mazim12@yahoo.com .*
- *Ibrahim Y. Abdel-Baset is currently pursuing Ph.D. degree program in electronics and communication engineering in Mansoura University, Assistant Lecturer at Nile Higher Institute of Engineering and Technology, Egypt, PH-00201006685403.*
  *E-mail: ibrahim_yasser@mans.edu.eg, ibrahimyasser14@gmail.com .*

International Journal of Scientific & Engineering Research, Volume 8, Issue 4, April-2017
ISSN 2229-5518

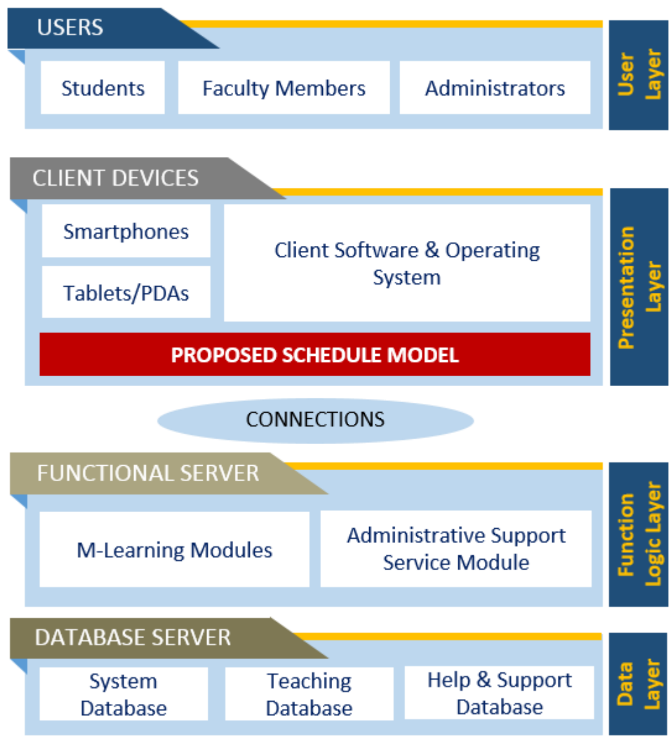


Fig. 1. Structure of modified M-Learning system architecture.

## 4 Structure of The Proposed Model

A straight forward approach to enabling online M-Learning access is to program mobile devices to periodically check new updates and download relevant updates. It is expensive both in terms of the wireless network availability (accessibility of Wi-Fi access points, unpredictable data rates in 3G networks) as battery-powered with stringent energy budgets that are easily depleted from constant connectivity and interaction with Wi-Fi/3G networks, which is caused primarily by the limitations in the current APIs of M-Learning providers. In order to login M-Learning contents, the mobile device has to set up connections all time. When the user wants to browse the educational content, it may be downloaded without any constrains (i.e., content compatibility with the mobile device, data size, battery level, bandwidth, etc.). In some cases, mobile devices even have to pull/retrieve multiple data items (e.g., text, audio, video), and analyze the data after downloading. Executing these processes on mobile devices periodically is taxing on resource-constrained mobile devices. A better approach is to employ offline/online application as a client-based for decreasing network traffic and minimizing the power consumption for mobile battery.

Proposed approach consists of a system of three stages that decision, scheduling, and downloading. In this model as shown in Fig. 2, the key coordination is Metadata files for the required M-Learning content. The decision service checks the required content throw Metadata file in Contents-ID list if it found before, connecting M-Learning application as offline and obtain the content from application storage. If the required content was not found before, connecting as online. Hence, we turn on the scheduling stage, receiving Metadata file for required content through relevance evaluator and pass the evaluation result of the scheduling algorithm which make actual delivery decisions by controlling the next stage where receiving the required content or not. In the case of receipt data, we turn on the downloading stage in which the content downloading through the caching policy algorithm.

In this paper, we assume that content selection, is done by individual mobile user, so as to consolidate all parameters, where software and hardware. Caching one of the most successful solutions for improving the performance of client-based web system where: (i) decreasing user perceived latency, (ii) reducing network bandwidth usage, (iii) reducing loads on the origin servers, and (iv) saving clients' battery power. More caching policies for efficiently contents storing in M-Learning content to achieve these goals and relieve the mobile device from the associated computation and communication overhead (As demonstrated in this document, the numbering for section) illustrates the software modules of the proposed system. The crux of the system resides in two key modules that are described below: (i) user-content relevance evaluator, which evaluator, which evaluates the M-Learning content being watched by a user before receiving through Metadata file relative to profiler setting and (ii) scheduling algorithm, which performs the actual delivery of contents to mobile users such that the benefits of fetching contents is maximized under the resource constrains.

### 4.1 1st Layer: User Interface Layer (UI)

In the information technology, the user interface is everything designed into an information device with which a human being may interact including display screen, keyboard, mouse, light pen, the appearance of a desktop, illuminated characters, help messages, and how an application program or a Web site invites interaction and responds to it [17]. UI layer consists of the following three components: (i) Authenticator, required for accessing the user's M-Learning application. Achieved by obtaining the access token on behalf of the user via the login service [18]. (ii) Content Displayer, the app displays the user's M-Learning through the Content Displayer component. It also provides interfaces for the user to interact with the contents [19]. (iii) Configurator, the app also provides interfaces to Change system settings for prefetching conditions (e.g. Network and battery). This is achieved through the Configurator component [20].

### 4.2 2nd Layer: Service Layer

Service layer works as the core of M-Learning application. It systematically detects required contents from the offline cache in the mobile user and in the case of non-existence, intelligently fetched contents in order to provide seamless access under intermittent internet access. It consists of the following components:

1. Decision Stage, which consist of: (i) Usage Metric Selector, in which switch the M-learning application either offline or online based on content-ID list. If the requested content exists, switches as offline. If not, switches as online. (ii) Offline Pre-Fetcher, the component decides

International Journal of Scientific & Engineering Research, Volume 8, Issue 4, April-2017
ISSN 2229-5518

on which contents to prefetch and when to prefetch. This is achieved by carefully taking into account the variations in the sizes. Downloaded contents are saved in content storage and displayed to the user when requested [21]. (iii) Online Fetcher, if a user requests a content that hasn't been prefetched, the component downloads the content on the fly. This requires the Internet connection at the request time, which resembles the behavior of traditional applications [22].

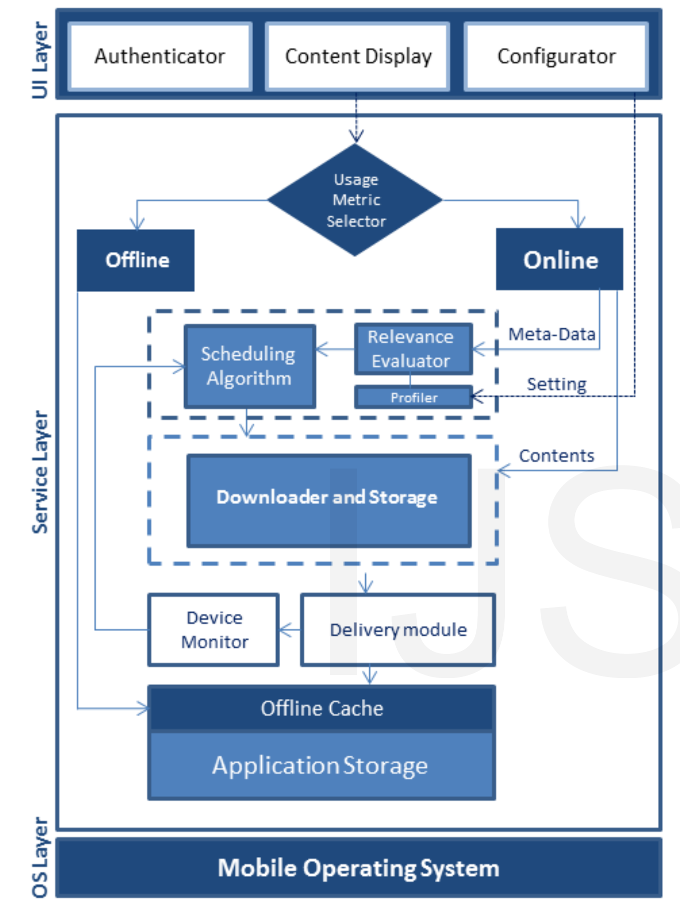


Fig. 2. Structure of the Proposed Model

2. Scheduling Stage, which consist of: (i) Profiler, monitors network battery conditions, network type, signal strength, and battery level. Monitors the storage and size of total downloads to avoid filling up the storage space (ii) Relevance Evaluator, there are two steps in our relevance evaluation scheme: (a) Metadata feature construction and (b) viewing profiler setting. In the first step, it constructs requested content features from Metadata (i.e. Content type and content size). The second step employs a logistic regression classifier to compute expectation battery power consumption and evaluate related to profiler setting and passing the result to scheduling algorithm [23]. (iii) Scheduling Algorithm, the second key component is the Scheduling Algorithm module that executes on the mobile device. The challenge here is to utilize output (relevance) from the relevance evaluator module to efficiently schedule the delivery of contents from function logic layer to the mobile device based on the available resources at the mobile device, e.g., network bandwidth and energy level. Receiving required contents from functional servers for a user are downloaded via the Downloader and Storage module [24]. Decisions are made by the scheduling algorithm either receiving or not based on network conditions and battery levels at the mobile device. These system conditions are captured via a Device Monitor at the client and the Delivery module implements the content transfer from download module (i.e. Online cache) to the application storage (i.e. Offline cache). The details of the scheduling algorithm are presented in the next section.
3. Caching/Storage Stage, Download and Storage module: Provides intelligent fetching, the cache size of the mobile device was limited. Caching technique is one of the most efficiently storing for receiving contents [25]. We will present in the next work; multiple caching techniques and select the best for using in our proposed module based on performance metrics such as Hit Ratio (HR), Byte hit ratio ($B_{HR}$), Bandwidth Saved (BS), and average download time ($T_{AD}$).

### 4.3 3rd Layer: Operating System Layer (OS)

A mobile operating system (Mobile OS) is a softer platform on top of which other programs called application programs, can run on mobile devices such as personal digital assistants (PDA), tables, cellular phones, smart phone and so on [16]. Over the years, the Mobile OS design has experienced a three-phase evolution: from the PC-based operating system to an embedded operating system to the current smartphone-oriented operating system in the past decade. Throughout the process, Mobile OS architecture has gone from complex to simple to something in-between. There are many mobile operating systems. The followings demonstrate in Table. (1) are the most important ones: Java ME Platform, Palm OS, Symbian OS, Linux OS, Windows Mobile OS, Blackberry OS, iPhone OS, and Google Android Platform [26].

## 5 Scheduling Algorithm

In this section, fetching of M-Learning contents to mobile device controlled by the scheduling algorithm problem which formulates to receive content optimally [27].

### 5.1 Parameter Tree

Scheduling is based on four parameters content type (e.g. text, audio, video, or compound), battery power level, device type (e.g. smartphone, cellular phone), and communication type (e.g. Wi-Fi, 3G). Metadata play an active role which contains the requested content information before receiving such as content size and content type, and other parameters can get from client device [28].

International Journal of Scientific & Engineering Research, Volume 8, Issue 4, April-2017
ISSN 2229-5518

TABLE 1
COMPARISON OF MAJOR MOBILE OPERATING SYSTEMS

| | Android | iOS | Windows Phone | Black-berry | Web | Symbian |
|---|---|---|---|---|---|---|
| Deployed Software Develop-ment | Java, C/C++ Python, Lua | Objective C, C/C++ | NET Languages (C#, VB.NET, F#, etc.), VB, C++, JScript | Java | JavaS-cript | C, C++/Qt, Java ME, Python, Ruby, Flash Lite |
| Market Size | Very High | High | Medium | Low | Very Low | Very low |
| SDK Platform | Windows XP, Vista and 7; Linux, Mac OS X | Mac OS X Snow Leopard 10.6.4 | Windows Vista & 7 | 32-bit Windows XP, Vista and 7 | OS X, Ub-untu, Win-dows | Windows XP Profes-sional SP2; Vista & 7 for some SDKs |
| Openness to Application Developers | Very High | Very Low | Medium | Low | Very High | High |
| OS Family | Linux | Darwin | Windows CE 7 Windows NT 8 | QNX | Linux | RTOS |
| Supported CPU Architecture | ARM, MIPS, x86 | ARM | ARM | ARM | ARM | ARM |
| Application Store | Android Market | iPhone App Store | Windows Market Place | Black-berry App World | Palm App Cata-log | Nokia Ovi Store |
| Future Prospect | Very High | High | Medium | Low | Low | Low |

*iOS = iPhone operating system, VB = Visual Basic, ARM = Advanced RISC (reduced instruction set computing) Machine; OS = Operating System, SDK = Software Development Kit , App = Application.*

## 5.2 System Models and Problem Formulation

The goal is to determine a schedule with which the parameters should be subject to examination from a set of constrain. The time envisioned in the schedule to be a set of time slots $\{t_1, t_2, t_3, \ldots\}$.

### 5.2.1 Content Parameter

#### Content Arrival.

Another substance c required by a client of u is gotten its Metadata which have content size by the client before it can be delivered to the user [84].

$$\lambda(t) = \sum\nolimits_{c \in a(t)} s_c \tag{1}$$

where λ(*t*) is the aggregate size of all substance for the client brought by the client P in time slot t, *a*(*t*) is an arrangement of substance touching base to the client in t, and *Sc* is the size of content. Assume that λ(t) is limited and constrained by $\lambda_{max}$, i.e., $\lambda(t) \le \lambda_{max}$ ∀ t. The time averaged fetched data amount is defined as:

$$\overline{\lambda} = \lim_{t\to\infty} \frac{1}{t} \sum\nolimits_{\tau=0}^{t-1} E\{\lambda(\tau)\} \tag{2}$$

#### Content Departure.

Every time opening, the arranging estimation picks a subset of the motivated substance to pass on to the user.

$$\mu_{(t)} = \sum\nolimits_{c \in r(t)} S_c \tag{3}$$

where $\mu_{(t)}$ is the total amount of data, and $r_{(t)}$ is a set of contents scheduled to be delivered in time slot t.

$$\overline{\mu} = \lim_{t\to\infty} \frac{1}{t} \sum_{\tau=0}^{t-1} E\{\mu(\tau)\} \tag{4}$$

where $\overline{\mu}$ is, time arrived at the midpoint of information sum conveyed to the client.

#### Content Dropping

The client drops some fetched contents due to low viewing probability or large size (hence, high energy consumption).

$$\chi_{(t)} = \sum\nolimits_{c \in d(t)} S_c \tag{5}$$

where $\chi_{(t)}$ is, the total dropped data amount in time slot, and d(*t*) is a set of contents scheduled to be dropped in time slot t

$$\overline{\chi} = \lim_{t\to\infty} \frac{1}{t} \sum_{\tau=0}^{t-1} E\{\chi(\tau)\} \tag{6}$$

where $\overline{x}$ is the time averaged data amount dropped from the fetched content list

### 5.2.2 Energy Parameters

#### Consumption Energy

For every content download, the consumed energy is deducted from the energy budget. The total energy consumption in time slot t defines us:

$$\rho_{(t)} = \sum\nolimits_{c \in d(t)} \rho_c(t) \tag{7}$$

$\rho_c$ is the energy amount used for receiving content c in time slot t.
The time averaged energy consumption amount is written as:

$$\overline{\rho} = \lim_{t\to\infty} \frac{1}{t} \sum_{\tau=0}^{t-1} E\{\rho(\tau)\} \tag{8}$$

#### Charged Energy

The energy budget may increase if the phone is charged. In time slot t, if the phone is charged and the current energy level is larger than et. The energy added to the budget is

$$\varepsilon_{(t)} = \omega_{(t)} e_u \tag{9}$$

where $\omega_{(t)}$ is a charged energy in the time slot t, is a percentage of the charged energy

$$\overline{\varepsilon} = \lim_{t\to\infty} \frac{1}{t} \sum_{\tau=0}^{t-1} E\{\varepsilon(\tau)\} \tag{10}$$

where $\overline{\varepsilon}$ is the time averaged energy added to the budget

International Journal of Scientific & Engineering Research, Volume 8, Issue 4, April-2017
ISSN 2229-5518

### 5.3 Scheduling Mathematical Model

Receiving data is subject to a set of constrain. Through the next presentation shows that algorithm ensures that bandwidth is enough for data transfer, and energy budget at any time is not more than a threshold. We know that energy budget may increase when the device is charged.
Ensures that bandwidth is enough for data transfer at any time slot.

$$\sum_{c\in r(t);r(t)\in f(t)} S_c \le \theta(t) \qquad (11)$$

Where θ(t) is the mobile device's downlink bandwidth at t.
Dropping some contents to stabilize the system from Eq. (12).

$$\bar{\lambda} \le \bar{\mu} + \bar{\chi} \qquad (12)$$

Ensure that the long-term energy budget is enough to perform scheduled transfers.

$$\bar{\rho} \le \bar{\varepsilon} \qquad (13)$$

Ensure that energy is enough for data receipt from the functional server (note that, ($e(t)$-$e_t$) is the difference of the current energy level with the power threshold.

$$\rho_c(t) \le \min(e(t) - e_t\, p(t)) \qquad (14)$$

Where e(t) is the current energy level, et is the power threshold, and P(t) is the remaining amount in energy budget.

### 5.4 Proposed Model Description

The scheduling algorithm selects a set of contents r(t) to deliver to the mobile device and a set of contents d(t) to drop from the client based on the following formulation in Table (2).

TABLE 2
MODEL DESCRIPTION OF SCHEDULING ALGORITHM

| | |
|---|---|
| $\lambda(t) = \sum_{c\epsilon a(t)} S_c$ | (a) |
| $\bar{\lambda} \le \bar{\mu} + \bar{x}$ | (b) |
| $\bar{\rho} \le \bar{\varepsilon}$ | (c) |
| $\rho_c(t) \le \min(e(t) - e_t\, ,\mathrm{P(t)})$ | (d) |

*$S_c$ = The size of content, λ(t)= The total size of all contents for the user fetched by the client P in time slot t, $\bar{\lambda}$ = The time averaged fetched data amount; $\bar{\mu}$ = The time averaged data amount delivered to the user, $\bar{\chi}$ = The time averaged data amount dropped from the fetched content list, $\bar{\varepsilon}$ = The time averaged energy added to the budget, $\bar{\rho}$ = The time averaged energy consumption amount, $\rho_c$ = The energy amount used for receiving content c in time slot t., e(t) = The current energy level, $e_t$ = The power threshold.*

The objective function in Eq. (a) ensures that bandwidth is enough for data transfer at any time slot. The algorithm drops some contents to stabilize the system as shown in constraint in Eq. (b). Constraint Eq.(c) ensures that long term energy budget for data receipt from the functional server. The first constraint guarantees that the device's downlink data rate in the time slot is enough to deliver the selected contents, and the second constrain ensures the energy is sufficient for the mobile device to receive the contents.

## 6. Experimental Setup

In a simulation scenario, we have created 500 files with random size between 500KB and 5GB on the server side. We have made 10,000 random requests on files. Following subsections, A, B and C discuss more simulation setup setting.

### 6.1 Software Simulator

To evaluate the proposed model, we performed a simulation of remote file accesses using Python programming which is an object-oriented and high-level programming language with dynamic semantics. Its high-level built in data structures, combined with dynamic typing and dynamic binding. We install the Python on an external Ubuntu machine which is an open-source operating system (OS) based on the Debian GNU/Linux distribution. We use VMware Workstation as a test-and-development environment that allows systems administrators to create and run virtual machines (VMs) directly on a desktop.

### 6.2 Generated Test Database

We create a simulation database which collecting between different types of database advantages based on learning management system (LMS) and using a Metadata generator to create Metadata files which have the information about every file such as File-ID, File-Size, Author-ID, Content-Type, and etc. Metadata size doesn't exceed 1KB. The simulation setting listed in Table (3).

TABLE 3
DEFAULT SYSTEM PARAMETER SETTINGS

| Description | Setting |
|---|---|
| Processor | Intel Dual Core 2.2 GHz |
| Installed memory (RAM) | CPU with 4 GB RAM |
| Number of files | 500 |
| File popularity | Gaussian Random |
| Number of requests | 10,000 |
| File size | 500KB ~ 5GB |
| File bit rate | 80 Mbps |

*GHz = Gigahertz, CPU = Central Processing Unit, GB= Gigabyte, Mbps = Megabits per second*

International Journal of Scientific & Engineering Research, Volume 8, Issue 4, April-2017
ISSN 2229-5518

### 6.3 Methodology

In order to evaluate proposed model, the conducted experiments for each of the aforementioned parameters will be used at full battery level. The adopted measurement methodology can be divided into three phases, while in the fourth phase the experimental results will be analyzed. In the first phase, relevant data for the proposed model based on Wi-Fi and 3G communication technologies (i.e. percentage of available battery, amount of transferred data and elapsed time) will be collected. In the second phase, the experimental results using M-Learning application of the proposed model will be compared with the traditional application (i.e. without proposed model). Based on the performed measurements in the previous phase, in the third phase, a recommendation of using one of the two types of the communication technologies based on the lowest energy consumption for device battery will be presented. Finally, in the fourth phase, the obtained results of the proposed system will be compared with the results of previous studies. Furthermore, the measurements study used of Balasubramanian et al. [29] showed in Fig. 3 that data transfer of 50 KB when using 3G needs 12.5 J, while when using Wi-Fi, the same data transfer consumes 7.6 J of energy.

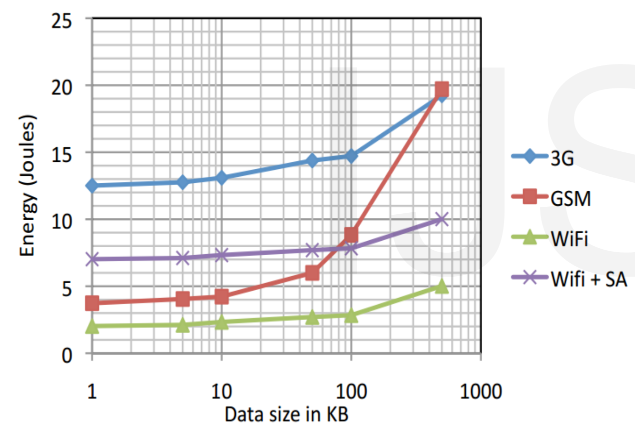


Fig. 3. Wi-Fi versus 3G versus GSM measurements: Average energy consumed for downloading data of different sizes against the inter-transfer time

## 7. Simulation Result

The performance of the proposed model was evaluated via the performance of scheduling algorithm under varying parameters. In this section, the results of the simulated scenarios will be presented. The experimental results in these experiments showed that using the proposed model achieves up to 28.6% improvement in saving battery power level compared to without using proposed model using Wi-Fi connection. On the other hand, using the proposed model in case of 3G connection achieves up to 66.7% improvement with respect to the traditional system. It has been observed that the battery level (battery power consumption) related to the number of battery life hours based on Wi-Fi and 3G communication technologies has been greatly improved with the use of the proposed model.

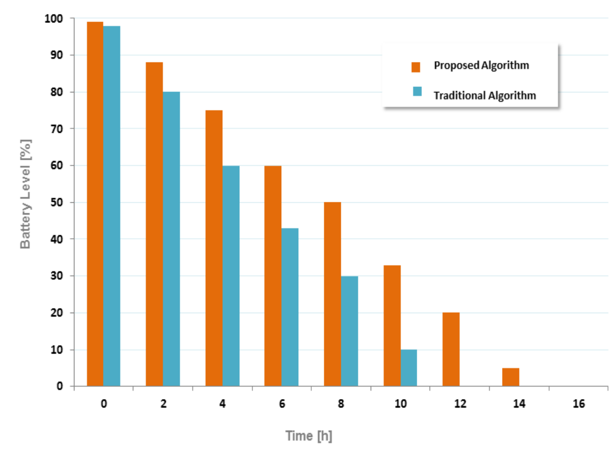


Fig. 4. Energy Consumption of the proposed and traditional Algorithms based on Wi-Fi Connection

Fig. 4 shows the energy consumption compared to the elapsed time when using Wi-Fi communication technology. As shown, data transfer when using the proposed model consumes significantly less energy with respect to the traditional algorithm. The battery life approximately four hours longer when using the proposed model continuously than the traditional system. If we compared the performance of the two systems (the proposed and traditional) after six hours about mid time of the minimum time, it was concluded that the power difference of batteries levels approximately equal 17%.

Next, comparing the energy consumption of both the proposed system and the traditional system based on 3G Connection are shown in Fig. 5. In a similar procedure followed in the experiment based on 3G communication, the experimental results of the monitored power level continuous nearly to twelve hours for the proposed system, while for the traditional system, the battery life does not exceed four hours. The power difference of batteries levels approximately 45%, if we compare after two hours continuously.

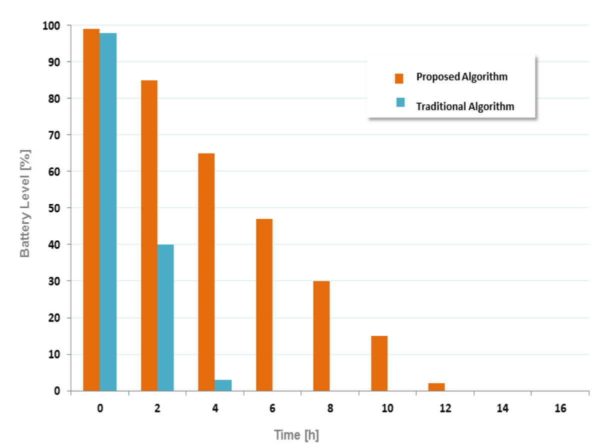


Fig. 5. Energy Consumption of the proposed and traditional Algorithms based on 3G Connection

International Journal of Scientific & Engineering Research, Volume 8, Issue 4, April-2017
ISSN 2229-5518

The simulation results of the proposed system implemented using 3G and Wi-Fi technologies, shown in Fig. 6: (i) the energy consumption of the system based on Wi-Fi decreases with respect to the system implemented using 3G connection. The battery life of Wi-Fi increases by two hours compared with 3G. By comparing after six hours, the difference of batteries levels is nearly about 12% improvement of using Wi-Fi connection relative to using 3G connection.

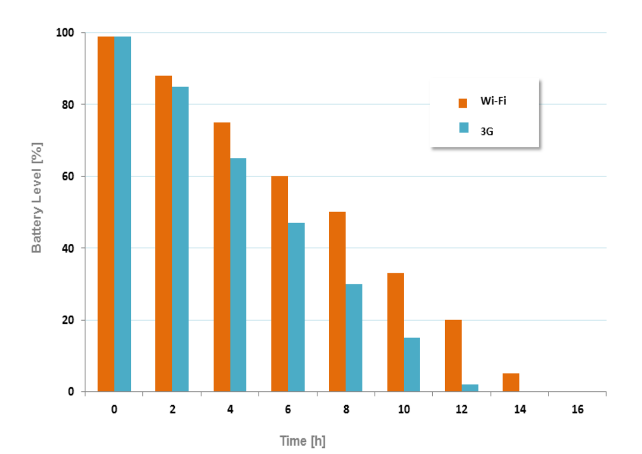


Fig. 6. The Proposed Model Energy Consumption based on Wi-Fi and 3G Connections

Fig. 7, presented a comparison of the batteries levels based on Wi-Fi communication technology, related to battery life measured in hours of the downloaded contents for the proposed system and two of the most famous M-Learning systems [16,30] It was observed that the proposed system achieves the best battery life of mobile device.

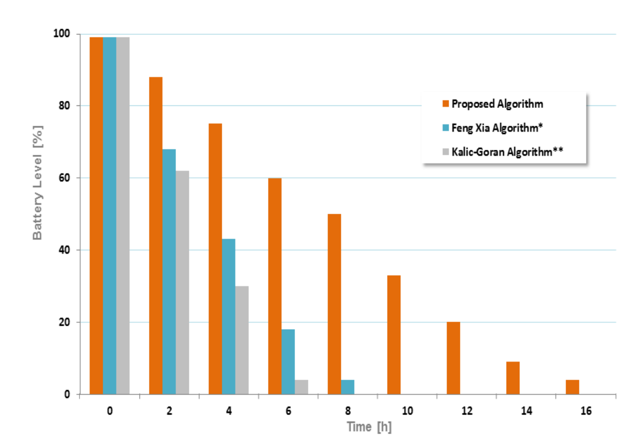


Fig. 7. Energy Consumption for different Algorithms based on Wi-Fi Connection

As shown, the proposed system achieved sixteen hours compared with Feng et al. [16] which does not exceed ten hours, while Goran et al. [30] does not exceed eight hours. If the three system (proposed, Feng et al. [16], and Goran et al. [30]) were compared we after six hours, the power consumption of Goran et al. [30] battery was about 96%, while Feng et al. [16] battery consumed 82%, on the other hand the proposed system achieved about 40%. After ten hours Feng and Goran et al. [30] approximately consumed totally battery power, proposed model sill have about 33% of battery life. Using proposed model continuous till about sixteen hours, which result based on downloading data size of M-Learning contents.

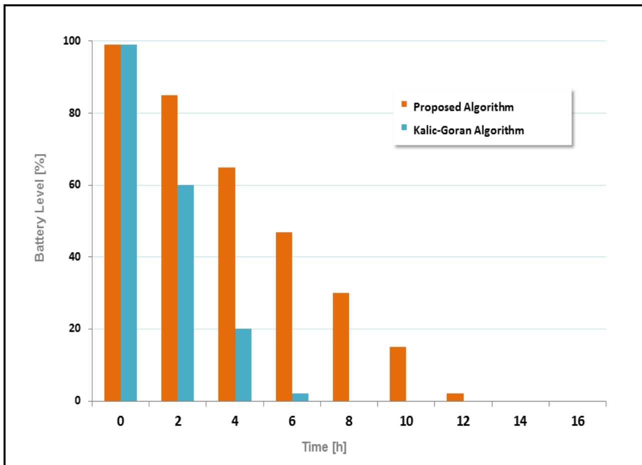


Fig. 8. Energy Consumption for different Algorithms based on 3G Connection

As shown in Fig. 8, energy consumption compared to the battery life (hours) between batteries levels of the proposed system and Goran et al. [30] when using 3G communication technology for data download. Goran et al. [30] resulted in a totally power consumption after eight hours, while the proposed system consumed only about 70% of the rated power. The output power of battery using the proposed system continuous till about fourteen hours before died (i.e. totally power consumption). The amount of battery power level for same transferred data when using Wi-Fi and 3G is different, it is important to note that battery saving when using Wi-Fi longer than when using 3G communication technology.

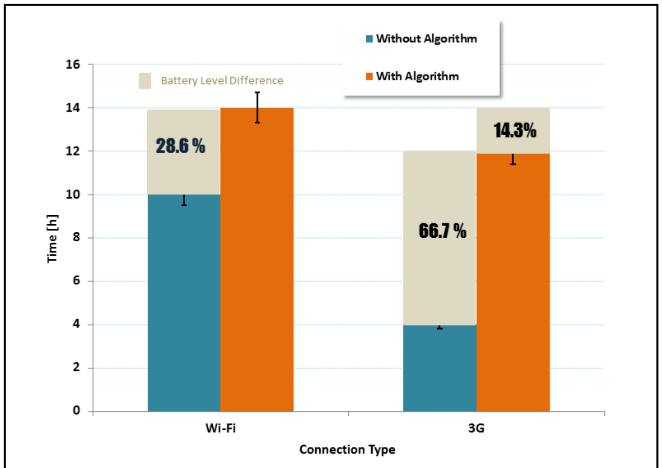


Fig. 9. Experimental results summary of proposed model based on Wi-Fi and 3G Communication technologies.

International Journal of Scientific & Engineering Research, Volume 8, Issue 4, April-2017
ISSN 2229-5518

In summary, as represented in Fig. 9, through experiments and simulation studies, we have shown that:

1. Proposed Model improves M-Learning mobile device battery life up to 28.6% based on Wi-Fi communication technology.
2. Using 3G connection, the proposed model achieves 66.7% as improvement of mobile device battery life comparing without using the proposed model.
3. Using same size can save battery life up 14.3% over using 3G connection.

## 8. Concluding Remarks

Nowadays, the smartphone owner population is growing. Multi-functionality, portability, and connectivity are opening doors for teaching and learning. Network connection and device battery life are the most challenges of using smartphones for mobile learning. Our aim in this paper is to be presented developing an application for M-Learning based on minimizing the power consumption of mobile battery, and decreasing on network traffic.

Towards this end, we present a propose hybrid online/offline model architecture that can efficiently receive and store the M-Learning contents; and schedule the content downloads to the mobile device based on content relevance and current system conditions. We then conducted experiments on our test bed and extensive Python simulation to evaluate our proposed approach and system. Our evaluation results show the value of the proposed model architecture and the associated algorithms that control the transfer of the contents within this system that using proposed model saving battery level 28.6% and 66.7% based on Wi-Fi and 3G, respectively. Caching technique approach to efficiently storing for receiving contents, and policies of delivery module for intelligent transfer content from online cache to offline cache (i.e. Application storage); Structured approaches to address these issues with meaningful, practical solutions are key topics addressed in our future research. Outperform the baseline approaches.

## References

[1] M. Alzaabi, B. J., and Zemerly M. J., "Web-based Architecture for Mobile Learning." International Journal for Infonomics (IJI), Vol.3, No.1, pp: 207-216, 2010.

[2] C. K., Keong,, et al. "Towards the Use of Software Product Metrics as an Indicator for Measuring Mobile Applications Power Consumption." World Academy of Science, Engineering and Technology, International Journal of Computer, Electrical, Automation, Control and Information Engineering, Vol.9, No.11, pp: 1785-1794, 2015.

[3] https://mobilereach.com/blog/should-your-mobile-application-operate-online-offline-or-both/

[4] S. Jha., "Poorly written apps can sap 30 to 40% of a phone's juice," CEO, Motorola Mobility, Bank of America Merrill Lynch 2011Technology Conference, 2011.

[5] http://www.molenet.org.uk/about.html

[6] Berking, Peter, et al. "Mobile learning: Not just another delivery method."Proceedings of the 2012 Interservice/Industry Training, Simulation, and Education Conference. 2012.

[7] G. Orr, "A review of literature in mobile learning: Affordances and constraints." Wireless, Mobile and Ubiquitous Technologies in Education (WMUTE), 2010 6th IEEE International Conference on. IEEE, pp: 107-111, 2010.

[8] Sha, L., Looi, C. K., Chen, W., Zhang, and B. H, "Understanding mobile learning from the perspective of self-regulated learning," Journal of Computer Assisted Learning, Vol.28, No.4, pp: 366-378, 2012.

[9] Cox, and M.J. "Formal to informal learning with IT: research challenges and issues for e-learning," Journal of computer assisted learning Vol.29, No.1, pp: 85-105, 2013.

[10] S. She, S. Sivapalan, I. Warren, "Hermes: A Tool for Testing Mobile Device Applications," Australian Softw. Eng. Conf., pp:121–130, 2009.

[11] T. Altameem, "Contextual mobile learning system for Saudi Arabian universities," International Journal of Computer Applications, Vol.21, No.4, pp: 21-26, 2011.

[12] D. Keegan, "Mobile learning: the next generation of learning." Distance Education International, pp: 137-143, 2005.

[13] Jin, Yi. "Research of one mobile learning system," Wireless Networks and Information Systems, 2009. WNIS'09. International Conference on. IEEE, pp:162-165, 2009.

[14] Obe Olumide, O., S. Fakuade, and T. Ogundele, "Development of a Yoruba Bible Native Application for Smartphones," Development, Vol.95, No.8, 2014.

[15] S. Kim, M. Clif, and P. Kerry, "An introduction to current trends and benefits of mobile wireless technology use in higher education,"AACE journal Vol.14, No.1, pp: 77-100, 2006.

[16] Xia, Feng, et al. "The power of smartphones." Multimedia Systems, Vol.21, No.1, pp: 87-101, 2015.

[17] J. Hudson, "User interface," U.S. Patent Application No. 10/560,403, 2004.

[18] J. French, and W. Jone, "System and method for authentication of network users." U.S. Patent No. 6,496,936, 2002.

[19] M. Ausems, and Q. Matthew, "User interface for handheld communication device," U.S. Patent Application, No. 09/900,316, 2001.

[20] Miller, S. P., and B. W. Lampson. "Distributed configuration profile for computing system," U.S. Patent No. 5,475,819, 1995.

[21] F. A. Marco, J. Gallud, V. M. Penichet, and Winckler, "A model-based approach for supporting offline interaction with web sites resilient to interruptions," International Conference on Web Engineering, Springer International Publishing, pp:156-171, 2013.

[22] Z.E. Britton, and et al., "Method and apparatus for asymmetric internet traffic monitoring by third parties using monitoring implements," U.S. Patent No. 7,953,851, 2011.

[23] M. Billi, and et al., "A unified methodology for the evaluation of accessibility and usability of mobile applications," Universal Access in the Information Society, Vol.9, No.4, pp: 337-356, 2010.

[24] M. J. GUY, "System for tracking activity and delivery of advertising over a file network," U.S. Patent Application, No 10/400,889, 2003.

[25] F. Lin, and Y. Weiguo, "Operating system battle in the ecosystem of smartphone industry," Information Engineering and Electronic Commerce, IEEC'09. International Symposium on. IEEE, pp:617-621, 2009.

[26] J. Janniello, and W. Christopher "Method and apparatus for broadcast delivery of content to a client-side cache based on user preferences," U.S. Patent Application No. 09/929,555, 2001.

[27] O. K. Shoukry, and B. F. Magda, "Evolutionary scheduling for mobile content pre-fetching," Theory and Practice of Natural Computing. Springer Berlin Heidelberg, pp:228-239, 2013.

[28] S. Y. Sim, and et al., "Method and system for managing distributed content and related metadata," U.S. Patent No. 7,272,613, 2007.

[29] N. Balasubramanian, B. Aruna, and V. Arun, "Energy consumption in mobile phones: a measurement study and implications for network applica-

International Journal of Scientific & Engineering Research, Volume 8, Issue 4, April-2017
ISSN 2229-5518

tions," Proceedings of the 9th ACM SIGCOMM conference on Internet measurement conference. ACM, pp:280-293, 2009.

[30] G. Kalic, B. Iva, and K. Mario, "Energy consumption in android phones when using wireless communication technologies," MIPRO, Proceedings of the 35th International Convention. IEEE, pp:754-759, 2012.