GEARING RESOURCE-POOR MOBILE DEVICES WITH POWERFUL CLOUDS: ARCHITECTURES, CHALLENGES, AND APPLICATIONS

FANGMING LIU, PENG SHU, HAI JIN, LINJIE DING, AND JIE YU, HUAZHONG UNIVERSITY OF SCIENCE AND TECHNOLOGY
DI NIU, UNIVERSITY OF ALBERTA
BO LI, THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY

ABSTRACT

Mobile cloud computing, with its promise to meet the urgent need for richer applications and services of resource-constrained mobile devices, is emerging as a new computing paradigm and has recently attracted significant attention. However, there is no clear definition and no well defined scope for mobile cloud computing due to commercial hype, and diverse ways of combining cloud computing and mobile applications. This article makes the first attempt to present a survey of mobile cloud computing from the perspective of its intended usages. Specifically, we introduce three common mobile cloud architectures and classify comprehensive existing work into two fundamental categories: computation offloading and capability extending. Considering the energy bottleneck and user context of mobile devices, we discuss the research challenges and opportunities of introducing cloud computing to assist mobile devices, including energy-efficient interactions, virtual machine migration overhead, privacy, and security. Moreover, we demonstrate three real-world applications enabled by mobile cloud computing, in order to stimulate further discussion and development of this emerging field.

INTRODUCTION

With the explosive growth of mobile devices in recent years, there is a shift of user preferences from traditional cell phones and laptops to smartphones and tablets. Advances in the portability and capability of mobile devices, together with widespread third/fourth generation (3G/4G) Long Term Evolution (LTE) networks and WiFi access, have brought rich mobile application experiences to end users. Undoubtedly, the demand for ubiquitous access to a wealth of media content and services will continue to increase, as indicated in a report by Cisco: traffic from mobile devices is anticipated to account for 60 percent of the total global IP traffic by 2016.

However, the resource-constrained nature of mobile devices, especially the limited battery life, has been a stumbling block to further improvements of mobile applications and services. According to the 2012 U.S. Wireless Smartphone Customer Satisfaction Study, battery life is the least satisfying aspect of smart phones, and is one of the few attributes that have a declined satisfaction score among consumers, rating an average score of 6.7 out of 10 in 2012, down from 6.9 in 2011. While new smart phones with faster CPUs, larger storage, and bigger screens are launched every day, and the bandwidth of wireless networks has increased from kilobits per second to megabits per second in just a few years, the development of batteries has lagged far behind the development of other components (e.g., processors, storage, networks, and displays) in mobile devices. In fact, faster CPUs and larger displays consume more battery energy: among users of 4G-enabled smart phones, satisfaction with battery performance is rated as 6.1, which is much lower than the 6.7 battery satisfaction score of 3G users. Moreover, despite the fast development of hardware technology, it is still difficult to support computation-intensive applications (e.g., image processing, augmented reality) on mobile devices, hindering developers from bringing richer experiences and complex applications to mobile users.

The paradigm of mobile cloud computing (MCC) is introduced to resolve the conflicts mentioned above, in which the cloud serves as a powerful complement to resource-constrained mobile devices. MCC is a model for elastic augmentation of mobile device capabilities via ubiquitous wireless access to cloud storage and computing resources, with context-aware dynamic adaption to changes in the operating environment. Rather than conducting all computational and data operations locally, MCC takes advan-

1536-1284/13/$25.00 © 2013 IEEE

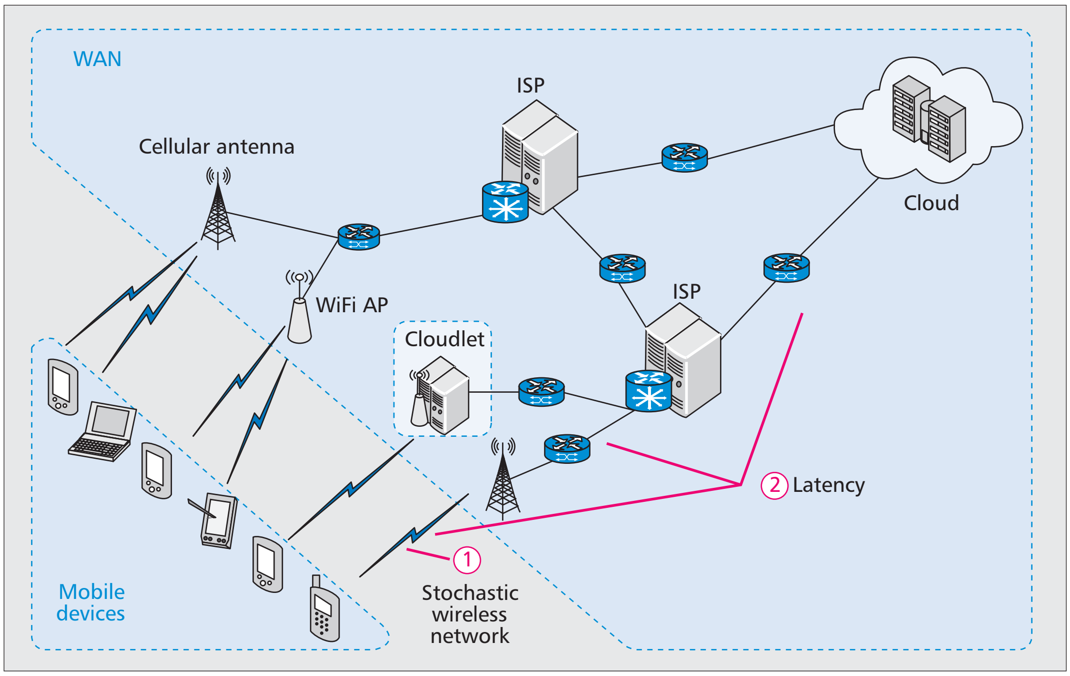

Figure 1. *Centralized cloud and cloudlet architecture.*

Rather than running applications locally and directly requesting data from content providers, a mobile device can offload parts of their workload to the cloud, taking advantage of the abundant cloud resources to help gather, store, and process data for the mobile device.

tage of the abundant resources in cloud platforms to gather, store, and process data for mobile devices. Many popular mobile applications have actually employed cloud computing to provide enhanced services. For example, Apple's iCloud enables users to store and synchronize their data (photos, videos, contacts, etc.) in the cloud. Mobile web browsers like Amazon Silk (Dolphin, Opera, etc.) exploit the computation and network resources in Amazon Elastic Compute Cloud (EC2) to compress web pages, making it faster and cheaper to download these pages to mobile devices. More innovative cloud-based mobile applications like healthcare monitoring and massively multiplayer online (MMO) mobile games are also under development.

Cloud computing and mobile applications have been the leading technology trends in recent years. It is not surprising that MCC, combining the two, would create a thrilling field and attract attention from both industry and academia. But what is MCC? Does it really bring a novel and useful computing paradigm, or is it just commercial hype? In what ways can mobile devices benefit from the cloud? And what are the biggest challenges of using cloud computing to augment and enhance mobile applications?

To answer the above questions, the rest of this article is organized as follows. First, we start with an overview of the existing architectures of MCC. Second, we analyze how to exploit cloud resources to benefit mobile devices, classifying existing usage models into two categories: computation offloading and capability extending. Third, we unveil the challenges in gearing mobile devices with the cloud. Finally, we highlight three representative killer applications driven by MCC before concluding this article.

Architectures and Visions

We first provide an overview of the architectures underlying MCC, which define the way in which mobile devices are connected to and interact with the cloud. Specifically, we discuss the traditional centralized cloud, as well as the recently proposed cloudlet and peer-based ad hoc mobile cloud scenarios, followed by our visions for future integrated mobile cloud architectures.

Centralized Cloud

We refer to the architecture where mobile devices obtain services from a traditional data center as a centralized cloud [1]. In this architecture, the cloud resource is placed in a remote centralized cloud infrastructure, as illustrated in Fig. 1. To access data center resources, mobile devices usually access the backbone network via WiFi access points (APs) or 3G/4G cellular networks. The cloud here acts as an agent between the original content providers and mobile devices. Rather than running applications locally and directly requesting data from content providers, a mobile device can offload parts of their workload to the cloud, taking advantage of the abundant cloud resources to help gather, store, and process data for the mobile device. Recently, many solutions based on this architecture have emerged to enhance mobile experiences, on which we further elaborate in Table 2.

Cloudlet

The access to the centralized cloud incurs long latency due to wide area network (WAN) delays, as illustrated in Fig. 1. However, real-time mobile applications (e.g., online games, speech recognition, Facetime) have strict requirements on response time. Users of these applications may have unsatisfactory experiences due to delayed communication between the mobile device and cloud.

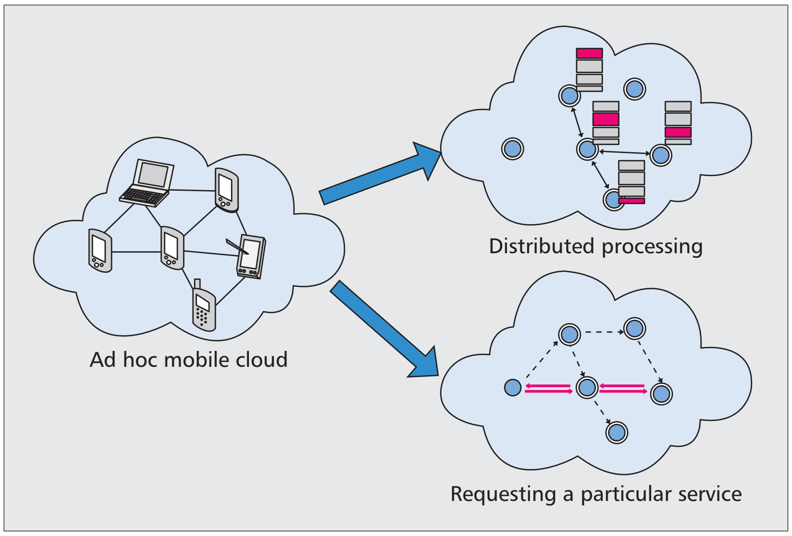

Figure 2. *Ad hoc mobile cloud and its two working modes.*

Cloudlet [2] has emerged as a promising architecture to reduce such communication delay. A cloudlet is a resource-rich server that has Internet access and is well connected to mobile devices via a high-speed local area network (LAN), as highlighted in Fig. 1. Smooth interaction can easily be achieved because of the low latency in one-hop LANs. This paradigm is similar to that of WiFi hotspots nowadays, where a mobile device gains high-speed Internet access whenever a WiFi hotspot is nearby. However, many practical issues remain to be addressed, including the sparseness of cloudlet deployment, the business model, and cloudlet reliability. Although there is a long way to go before the cloudlet can be deployed, we believe that it has inspired a new MCC paradigm.

Ad Hoc Mobile Cloud

The above architectures assume persistent connectivity to either the cloud or cloudlet, which is not always available or affordable (e.g., when the mobile device is roaming). A third architecture regards a crowd of mobile devices as an "ad hoc mobile cloud," where the neighboring mobile devices are pooled together for resource sharing. As shown in Fig. 2, in this paradigm, a task from a mobile device can be either processed in a distributed and collaborative fashion on all the mobile devices or handled by a particular mobile device that acts as a server. The feasibility of this architecture has been demonstrated in Virtual Cloud Provider [3], a peer-based MapReduce framework for mobile devices. It organizes nearby mobile devices pursuing the same task into a virtual cloud to distribute computation tasks.

Visions for the Future Architecture

First, we envision that future MCC will leverage a combination of different architectures, with traditional data-center-based solutions on one end of the spectrum and the ad hoc mobile cloud on the other end. An ad hoc mobile cloud is a backup solution when a group of users in collection are unable to access WiFi or cellular networks (e.g., passengers in the subway). Even with good network access, the ad hoc mobile cloud may be desirable for highly parallel tasks due to its crowd-based nature. Cloudlets are placed between mobile users and data centers, and are preferable for highly interactive and delay-sensitive applications (e.g., computer vision, augmented reality). Cloudlets can be deployed along with wireless APs and be charged for in a "pay-as-you-go" fashion similar to traditional cloud computing.

Second, the mobile cloud framework should support strong fault tolerance and transparent migration across different architectures to achieve seamless integration. A device may lose its connection to the centralized cloud during a service due to interrupted network connectivity. In an ad hoc mobile cloud, peers may unexpectedly quit the ad hoc hot zone. Under such situations, the system should recover from the fault as quickly as possible, and the current computing state should be suspended, migrated to the accessible cloud providers, and resumed; otherwise, the related devices will have to spend extra time on native execution. Such seamless integration and switching between different architectures require an open interface standard for heterogeneous cloud platforms and mobile devices.

Last but not least, mobile devices should play the role of not only independent consumers, but also interdependent contributors. Besides being resource providers in the ad hoc mobile cloud, mobile devices can be service participants. Various sensors (e.g., cameras, GPS, microphones) have made the mobile device a natural collector of context information. For example, mobile devices in vehicles can report their local traffic conditions to the centralized cloud. With traffic information collected from all over the city, the cloud can predict traffic jams or suggest routes for drivers. Moreover, a mobile device can serve as a resource relay; for example, nodes with stable Internet access in an ad hoc group could share the network to other mobile devices with poor connections, keeping the ad hoc mobile cloud connected to the Internet.

The seamless integration of heterogeneous cloud platforms and mobile devices, together with the proper utilization of each mobile inhabitant, will finally make mobile cloud computing a strong ecosystem. With the mobility and context awareness of mobile devices, such an ecosystem could *better connect the cyber world with the physical world*, eventually forming a foundation of human-centric computing, exemplified by augmented reality, healthcare, and personal assistance, as demonstrated later.

Intended Uses

Based on the above representative architectures, we proceed to discuss how mobile devices can leverage cloud computing for various uses. Considering the bottleneck of battery life and the demand for richer services in mobile devices, existing work can be categorized into two classes: computation offloading and capability extending, along with a detailed taxonomy summarized in Tables 1 and 2, respectively.

	Approaches	Blocking	Optimization	Offloading unit	Concurrency	Concept
Computation Offloading	MAUI	✓	Battery	Method	X	Runtime environment duplication
	CloneCloud	✓	Battery and performance	Thread	X	
	COMET	X	Battery and performance	Multithreads	X	
	ThinkAir	✓	Battery	Method	✓	
	mCloud	X	Performance	Component	✓	Component-based application
	Weblet	X	Battery	Component	✓	

Table 1. *Comparison of computation offloading models.*

	Approaches	Feature	Improved functions	Network traffic	Major resource provided by cloud
Capabilities Extending	Silk	Web browser	Loading speed, data volume	Medium	Computation
	Voice Search/Siri	Natural language UI	Speech recognition, semantic analysis	Medium	Computation
	iCloud/Dropbox	Extended storage	Storage capacity, file sharing	High	Storage
	CloudTorrent	File downloading	Download speed	High	Network
	Reflex	Social interaction	Spontaneous and global interaction	Low	Network
	MobiCloud	Network optimization	Informed and secure routing/communication	Low	Computation

Table 2. *Comparison of approaches in capability extension.*

Computation Offloading

To overcome resource constraints on mobile devices, a general idea is to offload parts of resource-intensive tasks to the cloud. Since execution in the cloud is considerably faster than that on mobile devices, it is worth shipping code and data to the cloud and back to prolong the battery life and speed up the application. This offloading framework is illustrated in Fig. 3. Despite diverse technologies to realize the runtime environment in the cloud (Table 1), the major differences between offloading techniques lie in the offloading unit and partitioning strategies.

Offloading Existing Applications — The first class of approaches attempts to extract offloading-friendly parts of codes from existing applications. MAUI [4] proposes a model to enable fine-grained code offloading to the cloud. To leverage MAUI, developers need to manually annotate which methods could be offloaded. A runtime profiler then predicts the energy usage of the method invocation according to current status information, and decides whether a method should be executed natively or remotely in order to minimize energy consumption.

Although the fine-grained method-level offloading strategy used in MAUI may maximize energy savings in mobile devices, it takes a considerable amount of time for programmers to annotate *offloadable* methods. To avoid manual annotation, CloneCloud [5] automatically decides what could be offloaded by utilizing a static code analyzer to mark possible migrate/merge points in the program bytecode. It then utilizes dynamic profiling to determine the optimal partitioning under different computation and network environments.

Whereas MAUI and CloneCloud can only offload one method/thread at a time and are troubled by locking issues, COMET [6] overcomes these limitations by relying on distributed shared memory (DSM) systems and virtual machine (VM) synchronization techniques to enable multithreaded offloading. A field-level granularity is used to manage memory consistency in order to reduce the frequency of required communication between the mobile device and cloud.

ThinkAir [7] also tries to extend MAUI and CloneCloud by further exploiting the auto-scaling feature offered by the cloud. For example, when the resource depletion (e.g., out of memory) of a clone VM is detected, instead of propagating the exception back, the cloud will allocate more VMs for the task. Moreover, if the cloud detects that the computation is parallelizable, it will automatically launch multiple VMs to execute the job in parallel.

New Development Models — While the above methods all focus on offloading *existing* applications, offloading can also be taken into account during the initial development stage of *new* applications. Mobile applications usually consist of vari-

Unlike a mobile device that has limited network connectivity, a cloud provider is typically connected to multi-carriers and ISPs with high-speed links, and can access Internet resources easily. Therefore, the cloud also promises to be a powerful agent in network connectivity improvement.

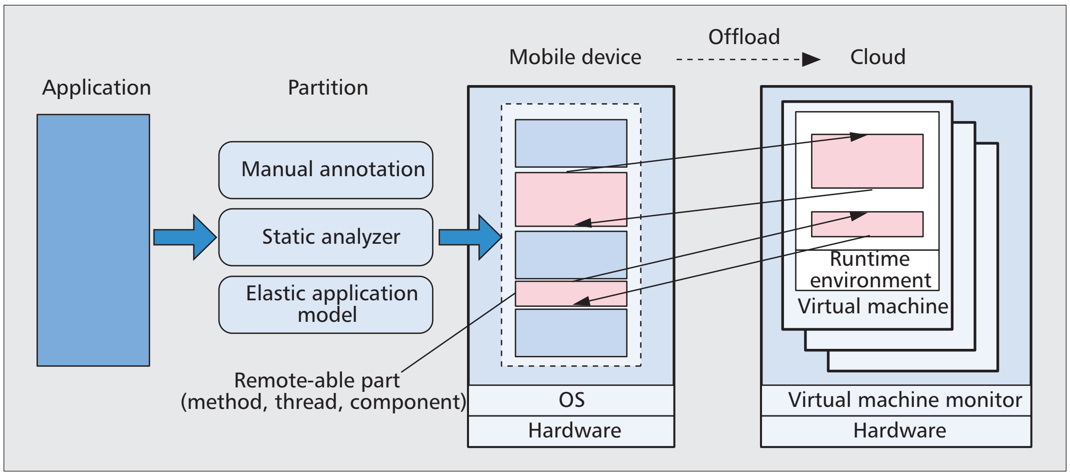

Figure 3. *The abstract procedure of computation offloading.*

ous components serving different high-level functionalities, many of which are reusable (e.g., image processing, face recognition). In a new model, offloadable components are predefined in the development phase, and are shipped between mobile devices and the cloud in an on-demand manner. For example, MCloud [8] introduces a composition approach that treats an application as a combination of reusable components. Component developers can publish to the market their developed components, based on which other developers can easily build a new application.

Weblet [9] adopts a similar design, in which an application is composed of *weblets*. A weblet is a platform-independent entity that presents a stateless and HTTP-based web service interface. The framework dynamically decides whether a weblet should run on the mobile device or in the cloud, based on attributes such as power consumption, monetary cost, and performance. Furthermore, a weblet in the cloud can replicate itself in the forms of pool and shadow. Pool handles multiple tasks in parallel, while shadow provides fault tolerance and latency control.

Capability Extending

Besides saving energy, cloud computing also enables enhanced mobile experiences that were previously impossible on resource-constrained mobile devices. Many commercial mobile applications use the cloud to bring about rich features, as illustrated in Table 2. They usually employ a client-server framework that consists of two parts, which run on the mobile device and the cloud, respectively. Essentially, cloud computing helps extend the capabilities of mobile devices in three aspects: computation, storage, and networking.

Computational Capability — Many applications nowadays support speech recognition (e.g., Google Voice Search, Apple Siri). The acoustic models for recognition and high-quality speech synthesis must be trained with millions of voice samples in thousands of dialects. This computation-intensive task is infeasible on a mobile device and should be delegated to the cloud. "Only there can you reach the scale, the enormous volume of interactions required to create a speech system capable of rivaling human understanding," said Larry Heck, Microsoft's chief scientist in the Speech group and former vice president of Nuance.

Storage — Another class of popular mobile cloud applications aims to extend storage, in which user data are stored in the cloud and synchronized to pervasive devices. Dropbox, a popular cloud storage service, uses Amazon's S3 storage system as its back-end, and secures user data with AES-256 encryption. To minimize synchronization cost, it utilizes binary-delta encoding to upload the changes made to a file rather than the entire file. Apple's iCloud is another storage service that enables users to store or access data like applications, music, and contacts in the cloud through multiple devices, including iOS-based devices and PCs. It uses both Microsoft Azure and Amazon Web Services as its service hosts. Although there is steady growth in mobile storage capacity, the ever increasing appetite of users for high-resolution videos and images promises the increasing popularity of cloud storage.

Networking — The popularity of mobile devices places higher requirements on network availability and stability. Unlike a mobile device that has limited network connectivity, a cloud provider is typically connected to multiple carriers and Internet service providers (ISPs) with high-speed links, and can access Internet resources easily. Therefore, the cloud also promises to be a powerful agent in network connectivity improvement.

For instance, CloudTorrent [10] employs the cloud as an agent between content providers and mobile devices, which builds a virtual link for mobile devices to access Internet resources. It lets the cloud run BitTorrent file sharing applications and send downloaded files to mobile devices to achieve fast and energy-efficient file downloading. Reflex [1] proposes a flexible and reusable system framework for social network application development, where worldwide users could spontaneously be connected together for conferencing and gaming via the cloud. It exploits well connected data centers across dif-

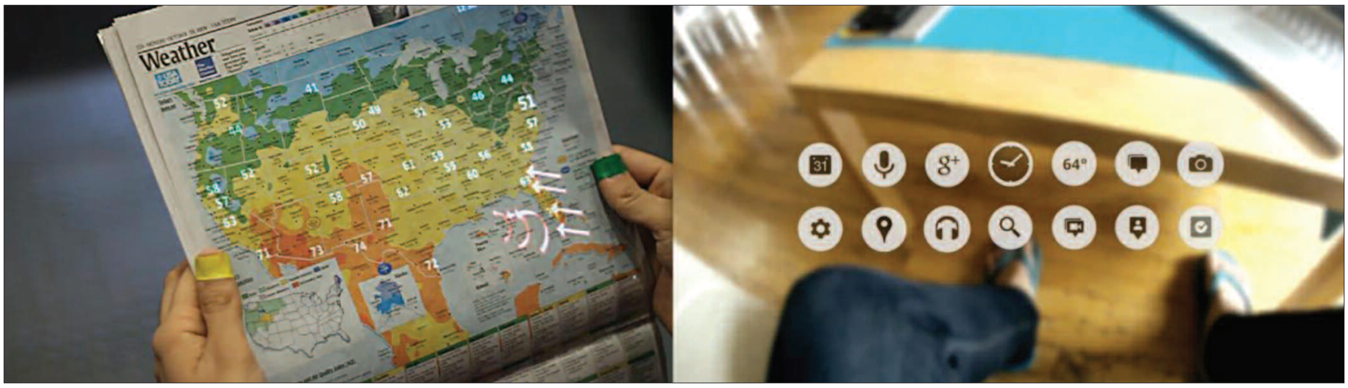

Figure 4. *SixthSense[1] (left) and Project Glass[2] (right).*

ferent regions of the same cloud provider to support real-time interactive sessions with low latency. MobiCloud [11] builds up a secure service-oriented mobile cloud framework to assist communications in mobile ad hoc networks (MANETs). It employs the cloud to create a virtualized environment where physical devices are mirrored to so-called extended semi-shadow images (ESSIs), creating a virtualized MANET routing layer that can guide routing among real mobile devices.

Challenges and Opportunities

We further identify four critical challenges and opportunities in MCC, with respect to the stochastic characteristic of wireless networks, virtual machine (VM) migration overhead, privacy, and security.

Energy-Efficient Interactions

Wireless networks are stochastic in nature: not only the availability and network capacity (e.g., bandwidth, signal strength) of APs vary from place to place, but the downlink and uplink bandwidth also fluctuates due to weather, building shields, mobility, flash crowds, and so on, as shown in Fig. 1. This stochastic characteristic may incur unpredictable energy consumption in communications between mobile devices and the cloud: measurement studies [12] show that the energy consumption for transmitting a fixed amount of data is inversely proportional to the available bandwidth. This implies that transmitting data in bad connectivity could consume considerably more energy than doing so in good connectivity.

Energy saving from computation offloading is not guaranteed on mobile devices if the evoked data transfers via wireless networks consume an unpredictable amount of energy. Offloading can save energy only if heavy computation is needed and a relatively small amount of data has to be transferred. Energy efficiency can be substantially improved if the cloud stores the data required for computation, reducing data transmission overhead. Admission control and bandwidth allocation mechanisms in cellular base stations and APs may guarantee network connectivity to a certain extent, but cannot eliminate the stochastic nature of wireless links. An alternative approach is to dynamically adjust application partitioning between the cloud and mobile devices according to network conditions [4, 5], although it is challenging to quickly and accurately estimate the network connectivity with low overhead.

Energy-efficient communications are also critical when exploiting the cloud to extend the capabilities of mobile devices. Frequent transmissions in bad connectivity will overly consume energy, making the extended capabilities unattractive, as battery life is always the top concern of mobile users. The latest practical solution, called eTime [13], is to adaptively seize the timing opportunity when network connectivity is good to prefetch frequently used data while deferring delay-tolerant data. However, this approach is mainly suitable for prefetch-friendly or delay-tolerant applications, such as social networking services (SNSs) and cloud storage.

VM Migration Overhead

Elastic resource scaling is one of the most important advantages of cloud computing. Enabled by the virtualization technology, cloud resources can be provisioned as needed to complement mobile applications [7]; that is, a VM instance can be launched when user demands arise, and shut down after the task terminates. However, the dynamic VM provisioning and time-varying resource demand of each VM instance may lead to underutilization on the underlying physical servers. To enhance utilization and energy efficiency, cloud operators need to periodically migrate and consolidate VM instances across physical servers and even set some servers to power saving mode if they are not used.

The above VM migration mechanisms may impact the performance of mobile cloud applications: not only does the migration itself incur suspended execution time, but the resource competition between different VMs hosted on the same physical server may also delay task completion. A possible remedy is that since the applications are processed at both the cloud and mobile sides, migration can be conducted when the current task is executed on the mobile side. Moreover, we can carefully group the tasks that require different types of resources on a server. For example, a CPU-intensive image processing task and a network-intensive video streaming task can be placed on the same physical server to alleviate resource contention, eventually

---

[1] *"The thrilling potential of SixthSense technology": http://on.ted.com/bDed*

[2] *"Project Glass: One day…": http://www.youtube.com/watch?v=9c6W4CCU9M4*

Since mobile devices are usually personal items, privacy must be considered when leveraging the cloud to store and process their confidential data remotely. Many mobile users are concerned with advertisement pushes, through which their private information like hobbies and locations may be unconsciously collected and illegally spread.

improving front-end quality of experience of mobile cloud applications.

On the other hand, VM migration mechanisms can actually be leveraged to improve task performance of mobile cloud applications; for example, a VM instance can be migrated from a busy server to a lightly loaded one to speed up execution. VM instances can also be adaptively migrated across geographically distributed data centers, as the mobile user moves, to reduce the interaction latency between the mobile device and cloud platform. The trade-off between migration overhead and performance in mobile cloud computing should be carefully tuned and balanced under different scenarios.

Privacy

Since mobile devices are usually personal items, privacy must be considered when leveraging the cloud to store and process their confidential data remotely. Many mobile users are concerned with advertisement pushes, through which their private information like hobbies and locations may be unconsciously collected and illegally spread.

Huang *et al.* propose a secure data processing framework for mobile cloud computing [14], in which critical data are protected by the unique encryption key generated from the user's trusted authority and stored in an area isolated from the public domain. Even when storage is breached in the cloud, unauthorized parties including the cloud vendor cannot obtain the private data. But such encryption cannot handle scenarios in which the cloud needs to operate on data (e.g., spelling checks).

Another particular privacy issue for mobile users is the leakage of personal location information in location-based services. To address the problem, a method called "location cloaking" makes user location data slightly imprecise before submitting them to the cloud [15]. But the imprecise data sometimes cannot provide relevant or satisfactory results to users in certain applications. Therefore, location cloaking should be adaptively tuned to balance the trade-off between privacy and result accuracy [15].

Security

There are several aspects of mobile cloud security, including antivirus, authentication, data protection, and digital rights management. Security vulnerability can cause serious problems, including property damage, cloud vendor economic loss, and user distrust. There are many instances of malware attempting to steal personal information or intercept mobile transactions. Since mobile devices are resource-constrained, locally executed antivirus software can hardly protect them from threats efficiently. A current solution is to offload the threat detection functionality to the cloud. Nevertheless, since a pure cloud antivirus relies on cloud resources, it is difficult to deal with malware that can block the device's Internet connection.

Besides, authentication is critical for access to sensitive information such as bank accounts and confidential files. With constrained text input on mobile devices, users tend to use simple passwords, making mobile applications more vulnerable to authentication threats. To solve this issue, [16] builds up an authorization platform where users are identified by their habits (e.g., calling patterns, location information, and web access). The platform routinely records user behavior information. When a server receives an authorization request, it redirects the request to an authorization engine, which uses the aggregated behavior information and an authorization policy to decide whether to accept the request or not.

Applications

Augmented Reality

A new class of mobile applications, augmented reality (AR), has started to draw users' attention. Wearable mobile devices, like gestural interface *SixthSense* and Google's head-mounted display *Project Glass*, aim to blur the boundary between the cyber world and real world. For example, as shown in Fig. 4, SixthSense can project augmented live news on a real-world newspaper; Google Glass can overlay wearers' vision with map directions, calendar reminders, text messages, and so on. Augmented reality is also incorporated into mobile games, where virtual objects are projected into the real world so that users can interact with them. Nevertheless, algorithms in augmented reality are mostly resource- and computation-intensive, posing challenges to resource-poor mobile devices.

These applications can integrate the power of the cloud to handle complex processing of augmented reality tasks. Specifically, data streams of the sensors on a mobile device can be directed to the cloud for processing, and the processed data streams are then redirected back to the device. It should be noted that AR applications demand low latency to provide a life-like experience. In this sense, apart from exploiting cloud resources, a mobile device can also offload data processing to a nearby cloudlet or ad hoc mobile cloud as elaborated earlier to avoid unpredictable multihop network latencies.

Remote Healthcare

With the increasing popularity of mobile devices, all industries are changing their business and products to adapt to mobile technology. Healthcare [17] is one of the leading areas where smart phones, tablets, and sensors are deployed on a large scale. Through remote communication, doctors and nurses can get a real-time picture of patients' conditions and perform the corresponding treatments. For example, Rehabcare Group Inc., which runs hospitals and medical facilities throughout the United States, has built cloud-based apps on SalesForece.com to improve healthcare experiences, such as paperless patient preadmission and screening, and remote post-operative monitoring.

A body area network is another example of applying mobile technology in healthcare. Projects like CodeBlue and CareNet deploy small wearable sensors on elderly people, monitoring their physical conditions (e.g., temperature, heart rate). These systems need to employ cloud computing to simulate and analyze the massive data collected. When emergencies like myocardial infarction and falling down are detected, hospitals aree informed immediately to provide the necessary help.

Web Applications

Being cross-platform in nature, web applications prevail nowadays, although their performance cannot be compared to native applications. With the development of mobile browsers, rich web applications implemented with HTML5 and JavaScript can now function properly on mobile devices. Applications like web games utilize advanced techniques like canvas, 3D transform, and complex program logic in JavaScript to realize dazzling visual effects and an interactive user experience. However, graphic rendering and JavaScript interpretation are energy-consuming. These applications can offload parts of the processing to the cloud, liberating the mobile device from computation-intensive tasks. As a demonstration, COMET [6] has experimented offloading JavaScript to the cloud and achieved an acceleration rate of up to 8.5×. Amazon also released the Silk browser for its Kindle Fire tablet. Silk uses a split architecture, where some browser subsystems are invoked on Amazon EC2. Upon requesting a web page, Silk dynamically partitions processing into a local part and a remote part, based on attributes like the resource size and page complexity. We believe that future web browsers may integrate the power of cloud computing to provide better web experiences.

Conclusions

In this article, we describe the state of the art of mobile cloud computing, from the brewing offloading technology to commercialized mobile cloud applications. We unveil the major challenges in mobile cloud computing and discuss potential solutions. Although mobile devices geared toward cloud computing will undoubtedly change technology trends as well as our daily lives, many practical problems remain to be resolved to structure a full-fledged mobile cloud system.

Although mobile devices geared toward cloud computing will undoubtedly change the technology trends as well as our daily lives, many practical problems remain to be resolved to structure a full-fledged mobile cloud system.

Acknowledgment

The corresponding author is Hai Jin. The research was supported by a grant from the National Natural Science Foundation of China (NSFC) under grant No. 61133006.

References

[1] Z. Liu *et al.*, "Socialize Spontaneously with Mobile Applications," *Proc. IEEE INFOCOM*, 2012.
[2] M. Satyanarayanan *et al.*, "The Case for VM-based Cloudlets in Mobile Computing," *IEEE Pervasive Computing*, vol. 8, no. 4, 2009, pp. 14–23.
[3] G. Huerta-Canepa *et al.*, "A Virtual Cloud Computing Provider for Mobile Devices," *Proc. ACM Wksp.Mobile Cloud Computing & Services: Social Networks and Beyond*, 2010.
[4] E. Cuervo *et al.*, "Maui: Making Smartphones Last Longer with Code Offload," *Proc. ACM MobiSys*, 2010.
[5] B. Chun *et al.*, "Clonecloud: Elastic Execution Between Mobile Device and Cloud," *Proc. ACM EuroSys*, 2011.
[6] M.S. Gordon *et al.*, "COMET: Code Offload by Migrating Execution Transparently," *Proc. USENIX OSDI*, 2012.
[7] S. Kosta *et al.*, "Thinkair: Dynamic Resource Allocation and Parallel Execution in the Cloud for Mobile Code Offloading," *Proc. IEEE INFOCOM*, 2012.
[8] V. March *et al.*, "mCloud: Towards a New Paradigm of Rich Mobile Applications," *Proc. MobiWis*, 2011.
[9] X. Zhang *et al.*, "Towards an Elastic Application Model for Augmenting the Computing Capabilities of Mobile Devices with Cloud Computing," *Springer Mobile Networks & Applications*, vol. 16, no. 3, 2011, pp. 270–84.
[10] I. Kelenyi *et al.*, "CloudTorrent — Energy-Efficient BitTorrent Content Sharing for Mobile Devices via Cloud Services," *Proc. IEEE CCNC*, 2010.
[11] D. Huang *et al.*, "MobiCloud: Building Secure Cloud Framework for Mobile Computing and Communication," *Proc. IEEE SOSE*, 2010.
[12] N. Balasubramanian *et al.*, "Energy Consumption in Mobile Phones: a Measurement Study and Implications for Network Applications," *Proc. ACM IMC*, 2009.
[13] P. Shu *et al.*, "eTime: Energy-Efficient Transmission between Cloud and Mobile Devices," *Proc. IEEE INFOCOM*, 2013.
[14] D. Huang *et al.*, "Secure Data Processing Framework for Mobile Cloud Computing," *Proc. IEEE INFOCOM Wksp. 2011*.
[15] R. Cheng *et al.*, "Preserving User Location Privacy in Mobile Data Management Infrastructures," *Proc. Wksp. Privacy Enhancing Technologies*, 2006.
[16] R. Chow *et al.*, "Authentication in the Clouds: a Framework and Its Application to Mobile Users," *Proc. ACM Wksp. Cloud Computing Security Wksp.*, 2010.
[17] M. Chen *et al.*, "A 2G-RFID Based E-healthcare System," *IEEE Wireless Commun.*, vol. 17, no. 1, 2010, pp. 37–43.

Biographies

Fangming Liu [M] (fangminghk@gmail.com) is currently an associate professor in the School of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan, China; and he is the CHUTIAN Scholar of Hubei Province, China. Since 2012 he has also been invited as a member of StarTrack Visiting Young Faculty at Microsoft Research Asia (MSRA), Beijing. He received his B.Engr. degree in 2005 from Department of Computer Science and Technology, Tsinghua University, Beijing, China; and his Ph.D. degree in computer science and engineering from the Hong Kong University of Science and Technology in 2011. From 2009 to 2010, he was a visiting scholar at the Department of Electrical and Computer Engineering, University of Toronto, Canada. He was the recipient of two Best Paper Awards from IEEE GLOBECOM 2011 and IEEE CloudCom 2012, respectively. His research interests include cloud computing and data center networking, mobile cloud, green computing and communications, software-defined networking and virtualization technology, large-scale Internet content distribution, and video streaming systems. He is a member of ACM, as well as a member of the China Computer Federation (CCF) Internet Technical Committee. He has been a Guest Editor for *IEEE Network*, and served as TPC for IEEE INFOCOM 2013 and 2014 and GLOBECOM 2012 and 2013.

Peng Shu is currently a Master's student in the School of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan, China. His research interests focus on cloud computing and wireless mobile applications.

Hai Jin [SM] is a Cheung Kung Scholars Chair Professor of computer science and engineering at the Huazhong University of Science and Technology (HUST), China. He is now dean of the School of Computer Science and Technology at HUST. He received his Ph.D. degree in computer engineering from HUST in 1994. In 1996, he was awarded a German Academic Exchange Service fellowship to visit the Technical University of Chemnitz in Germany. He worked at the University of Hong Kong between 1998 and 2000, and as a visiting scholar at the University of Southern California between 1999 and 2000. He was awarded the Excellent Youth Award from the National Science Foundation of China in 2001. He is the chief scientist of ChinaGrid, the largest grid computing project in China, and chief scientist of the National 973 Basic Research Program Project of Virtualization Technology of Computing Systems. He is a member of the ACM. His research interests include computer architecture, virtualization technology, cluster computing and grid computing, peer-to-peer computing, network storage, and network security.

Linjie Ding is currently a research intern in the School of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan, China. His research interests focus on cloud computing and wireless mobile applications.

JIE YU is currently a research intern in the School of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan, China. Her research interests focus on cloud computing and wireless mobile applications.

DI NIU is currently an assistant professor in the Department of Electrical and Computer Engineering, University of Alberta, Canada. He received his B.Engr. degree from the Department of Electronics and Communication Engineering, Sun Yat-sen (Zhongshan) University, Guangzhou, Guangdong, China, in 2005, and his M.A.Sc. and Ph.D. degrees from the Department of Electrical and Computer Engineering, University of Toronto, Ontario, Canada, in 2009 and 2012, respectively. His research interests include cloud computing, large-scale data analytics, parallel and distributed optimization, network economics, statistical and machine learning for Internet applications, time series analysis, stochastic modeling, multimedia transmission, and network coding.

BO LI [F] is a professor in the Department of Computer Science and Engineering, Hong Kong University of Science and Technology. He received his B.Engr. degree in computer science from Tsinghua University and his Ph.D. degree in the electrical and computer engineering from the University of Massachusetts at Amherst. He was with IBM Networking System, Research Triangle Park, North Carolina, between 1993 and 1996. He was an adjunct researcher at Microsoft Research Asia (MSRA) (1999–2006). He was with Microsoft Advanced Technology Center (ATC) in the summers of 2007 and 2008. He has made original contributions on Internet proxy placement, capacity provisioning in wireless networks, routing in WDM optical networks, and Internet video streaming. He is best known for a series of works on a system called Coolstreaming (Google entries over 1,000,000 in 2008 and Google scholar citations over 800), which attracted millions of downloads and was credited as the first large-scale peer-to-peer live video streaming system in the world. His recent work on the peer-assisted online hosting system FS2You (2007–2009) (Google entries 800,000 in 2009) has also attracted millions of downloads worldwide.