Research on Mobile Cloud Computing

Gaochao Xu, Yan Ding, Liang Hu, Xiaodong Fu, Jia Zhao, Hao Yan, Jianfeng Chu

College of Computer Science and Technology, Jilin University, Changchun, China,
xugc@jlu.edu.cn, dingyan11@mails.jlu.edu.cn, hul@jlu.edu.cn, fuxd@jlu.edu.cn, zhaiyj049@sina.com, chujf@jlu.edu.cn

College of Software, Jilin University, Changchun, China, dickyan19850410@163.com

Corresponding author: Jia Zhao

Abstract

Mobile cloud computing introduces the cloud computing technology to the mobile Internet. It can greatly promote the development and application of cloud computing, and can promote the development and application of the Internet technology to a higher level. Mobile cloud computing system consists of mobile devices, wireless networks and cloud data centers. It combines cloud computing technology and mobile Internet technology. With the combination of cloud computing and mobile Internet, mobile cloud computing causes some new research issues belonging to the field of mobile cloud computing. This paper introduces some research directions of mobile cloud computing such as application partition, computation offloading, resource management and context-aware services as well as future research trends in brief.

Keywords: Mobile Cloud Computing, Application Partition, Computation Offloading, Context-aware Services, Resource Management

1. Introduction

1.1. Cloud computing and mobile cloud computing

Cloud computing is the development of parallel computing, distributed computing and grid computing. It distributes the computation tasks into a resource pool composed by great amounts of computing resources, and makes users on-demand obtain computing power, storage space and information service. Compared to cluster computing and grid computing, cloud computing is a commercial computing model which is more suitable for providing services to users. "Cloud" is large amounts of virtual computing resources which have abilities to maintain and manage themselves in the system. Generally, these resources include computing servers, storage servers and bandwidth resources etc. Cloud computing system pools the computing resources into a resource pool and achieves their automatic management by using specialized software. Users can dynamically utilize and reserve some of the resources to support varieties of applications. It benefits improving efficiency, reducing costs and makes users more focus on their own business. The development of cloud computing is not limited to PC. With the vigorous development of the mobile Internet, the cloud computing services based on mobile devices such as mobile phones, tablet PC etc. have emerged and been widely used for the field of information sharing, mobile learning, e-commerce, home monitoring and mobile health etc. Mobile cloud computing refers to applying cloud computing technology into mobile Internet. It has become a promising research hotspot with application prospects in the field of information technology. The benefits of mobile cloud computing contain extending battery lifetime, improving data storage capacity and processing power as well as improving reliability etc.

1.2. The general architecture of mobile cloud computing

It is the organization and structure of mobile cloud computing systems that the architecture of mobile cloud computing refers to. Generally, some researchers would like to take advantage of cloud computing technology to enhance capability of mobile devices. Also, some researchers want to use...
cloud computing to run mobile application instead of mobile devices. Therefore, mobile cloud computing architecture can contain two types: agent-client and collaborated.

1.2.1. Agent-client type

In this type, cloud computing data centers provide mobile devices with all resource management to help mobile devices overcome some inherent limitations such as processing power, data storage and energy consumption. Each of mobile devices has an agent-client in cloud data centers. Mobile devices can contact with other entities outside its domain by communicating with its agent-client. For instance, Mahadev Satyanarayanan et al. in [1] present a mobile computing architecture, in which a mobile user exploits virtual machine (VM) technology to rapidly instantiate customized service software on a nearby cloudlet, and then uses that service over a wireless LAN. The mobile device typically functions as a thin client with respect to the service. A cloudlet is a trusted, resource-rich computer or cluster of computers that is well-connected to the Internet and is available for use by nearby mobile devices.

1.2.2. Collaborated type

Collaborated type is different from agent-client type. In this type, the servers in cloud data centers may only provide mobile devices with the function of controlling and scheduling for collaboration among the mobile devices, each of which is regarded as a part of cloud. It’s the remainder resources of mobile devices that the mobile cloud computing utilizes. For instance, in [2] the authors propose a Hyrax platform derived from Hadoop that supports cloud computing on Android smartphones. Hyrax allows client applications to conveniently utilize data and execute computing jobs on networks of smartphones and heterogeneous networks of phones and servers. By scaling with the number of devices and tolerating node departure, Hyrax allows applications to use distributed resources abstractly, oblivious to the physical nature of the cloud.

2. The main research directions of mobile cloud computing

2.1. Application partition

In mobile cloud computing system, each of mobile applications needs be partitioned into several computation modules or components, the complicated computations of which should be performed on a server of cloud data center generally. Application partition should consider resource and energy consumption as well as data dependency.

2.1.1. Calling the cloud: enabling mobile phones as interfaces to cloud applications

To optimally partition an application between a mobile phone and a server, Ioana Giurgiu et al. in [3] approach the problem in two steps. First, they abstract an application’s behavior as a data flow graph of several inter-connected software modules. Modules encapsulate small functional units supplied by the application developer. Each module provides a set of services, and modules are connected through the corresponding service dependencies. Through an offline application profiling, modules and service dependencies are characterized in terms of their resource consumption (data exchange, memory cost, code size), thus providing the knowledge base for the optimization process. Given this graph as shown in Figure 1, in the second step, a partitioning algorithm finds the optimal cut that maximizes (or minimizes) a given objective function. The objective function expresses a user’s goal such as to minimize the interaction latency or the data traffic. Moreover, the optimization also takes into account a mobile phone’s resource constraints such as memory and network resources available. They propose two types of partitioning algorithms: ALL and K-step. They look at the problem both as a static and dynamic optimization. In the first case, the best partitioning is computed offline by considering different types of mobile phones and network conditions. In the second case, the partitioning is computed on-the-fly, when a phone connects to the server and specifies its resources and requirements. ALL fits the first scenario, while K-step the second one. The approach does not require new infrastructures as it uses existing software for module management that can support the actual
distributed deployment of an application between a phone and a server. However, to find out an optimal portioning, the approach needs large amounts of computation and to occupy lots of space. From the view of providers, it is a waste of limited resource.

![Example of application’s consumption graph](image)

**Figure 1.** Example of application’s consumption graph

### 2.1.2. From Augmented Reality to Augmented Computing: a Look at Cloud-Mobile Convergence

Based on the fact that there has been considerable number of virtual and augmented reality applications designed and developed for mobile device, however the state-of-the-art systems are commonly confined by several limitations. Xun Luo et al. in [4] propose the “Cloud-Mobile Convergence for Virtual Reality (CMCVR)” concept. In CMCVR, to take advantage of the better load balancing inherent in by-region task partitioning, and lower the computation overhead as much as possible by exploiting the characteristics of by-scale task portioning. A hybrid approach is presented. The task portioning is a two-stage process. The first stage breaks down the workload with the by-scale strategy. Large workloads at high scale levels are further portioned in the second stage which uses the by-region strategy. The partitioning process is completed when an optimized overall system performance is achieved. For the large scale applications, the approach can partition it quickly but after several rounds of partitioning, the partition isn’t accurate enough during every round.

### 2.1.3. Dynamically Partitioning Applications between Weak Devices and Clouds

The applications in mobile cloud computing systems run diverse workloads under diverse device platforms, networks and clouds. Traditionally these applications are statically partitioned between weak devices and clouds, thus may be significantly inefficient in heterogeneous environments and workloads. Byung-Gon and Petros in [5] introduce the notion of dynamic partitioning of applications between weak devices and clouds and argue that this is the key to addressing heterogeneity problems. The authors found that currently these applications running on weak devices are structured such that they are statically partitioned between the weak device and a server running in the cloud. Two representative application partitions are as follows. First, most of an application’s processing is done at the server(s) in the cloud, and the end-user device runs simple tasks such as UIs, acting like a thin client. However, partitioning applications statically does not provide optimal user experience as more and more applications are used in diverse environments and inputs. That is, there is no single partitioning that fits all due to environment heterogeneity (device, network, and cloud) and workload. Furthermore, there are many partitioning choices. Their vision is a system that can seamlessly adapt to different environments and workloads by dynamically instantiating what partitioning to use between weak devices and clouds. They point out different sources of heterogeneity and discuss how dynamic partitioning can address these heterogeneity problems. They then formalize the dynamic partitioning problem, and sketch how to construct a system that supports dynamic partitioning. The proposed approach employs static analysis and dynamic instantiating mechanism. According to specific environment and goal, it can on-demand instantiate some specific partition. However, its computation is a bit complicated and it is hard to model the problem.
2.2. Computation offloading

Sending computation to another machine is not a new idea. The currently popular client-server computing model enables mobile users to launch Web browsers, search the Internet and shop online. What distinguishes cloud computing from the existing model is the adoption of virtualization. Instead of service providers managing programs running on servers, virtualization allows cloud vendors to run arbitrary applications from different customers on virtual machines. Cloud vendors thus provide computing cycles, and users can use these cycles to reduce the amounts of computation on mobile systems and save energy. Thus, cloud computing can save energy for mobile users through computation offloading. The problem is the choice of offloading percent and methods [6,7].

2.2.1. Augmented smartphone application through clone cloud execution

Smartphones enable a new, rich user experience in pervasive computing, but their hardware is still very limited in terms of computation, memory, and energy reserves, thus limiting potential applications. In [8,9], the author propose a novel architecture that addresses these challenges via seamlessly—but partially—offloading execution from the smartphone to a computational infrastructure hosting a cloud of smartphone clones. They outline new augmented execution opportunities for smartphones enabled by their CloneCloud architecture. The system provides a way to boost a smartphone application by utilizing heterogeneous computing platforms through cloning and computation transformation. For doing so, the system (semi-)automatically transforms a single-machine execution (e.g., smartphone computation) into a distributed execution (e.g., smartphone plus cloud computation) in which the resource-intensive part of the execution is run in powerful clones. An additional benefit of cloning is that if the smartphone is lost or destroyed, the clone can be used as a backup. They achieve this by combining whole-system replication through incremental checkpointing, (semi)-automatic partitioning and invocation of augmented execution, and coordination of computation between the primary (phone) and the clone. Figure 2 illustrates the system model of the approach. Their main idea is: copy the set of data and applications from the smartphone onto the clone cloud and execute some functionality modules on the clones VM in terms of the checkpoingings set in advanced. From the view of mobile users, they need not to worry about computing power, energy consumption and storage space. Then the system reintegrates the results back into the smartphone. The authors attempt to categorize the types of augmentation they envision. They summarize there are five types of augmentation. Each of them uses special method to offloading. Every smart-phone can have multiple clones, which are assumed to be more powerful smart-phones.

![Figure 2. System model. The system transforms a single-machine execution (smartphone computation) into a distributed execution (smartphone and cloud computation) (semi-)automatically.](image)

2.2.2. MAUI: Making Smartphones Last Longer with Code Offload

Eduardo Cuervo et al. in [10,11] present MAUI, a system that enables fine-grained energy-aware offload of mobile code to the infrastructure as shown in Figure 3. MAUI’s goal is to maximize the benefits of code offload for today’s smartphone devices. They present a high-level overview of MAUI’s components on a mobile device and in the infrastructure in order to understand how they all integrate into one platform for developing mobile applications. MAUI uses the benefits of managed code to reduce the burden on programmers to deal with program partitioning while maximizing the
potential for energy savings through fine-grained code offload while minimizing the changes required to applications.

2.3. Context-aware mobile cloud services

For the service providers, how to maximize the revenues is the most important issue. To achieve this goal, mobile cloud computing providers consider not only how to achieve better resource management but also how to mine the potential needs of mobile users to achieve their more profits while fulfilling mobile users’ satisfaction better and making mobile users reach a higher user experience. The information, which is used for deducing mobile users’ potential needs and need to be collected from users and devices, is called context. So we can image what context-aware mobile cloud services are. Many research want to utilize the local contexts (e.g., data types, network status, device environments, and user preferences) to enhance the quality of service. Context leads to advent of many different mobile applications. Generally, we can classify contexts into two types: spatial contexts and social contexts.

2.3.1. Spatial contexts service

Pelín Angin et al. in \[12,13\] propose a mobile-cloud collaborative approach for context-aware navigation by exploiting the computational power of resources made available by Cloud Computing providers as well as the wealth of location-specific resources available on the Internet. They propose an extensible system architecture that minimizes reliance on infrastructure, thus allowing for wide usability. They present a traffic light detector that they developed as an initial application component of the proposed system. Patrick Stuedi et al. in \[14\] present WhereStore, a location-based data store for Smartphones interacting with the cloud. The key property of WhereStore is that it uses the phone's location history to determine what data to replicate locally. Their key insight is that a person's past is a good indication of his future locations and hence his future information needs. The advantage of the kind of approach is that it decreases the overall data access latency and also reduces the probability of data becoming unavailable in periods of no connectivity. Furthermore, it can also potentially reduce the overall energy consumption on the phone.

2.3.2. Social contexts service

Eric Jung et al. in \[15\] propose to exploit the potential of smart phones in proximity cooperatively, using their resources to reduce the demand on the cellular infrastructure, through a decision framework called RACE (Resource Aware Collaborative Execution). RACE enables the use of other mobile devices in the promixity as mobile data relays. RACE is a Markov Decision Process (MDP) optimization framework that takes user profiles and user preferences to determine the degree of collaboration. Both centralized and decentralized policies are developed and validated through simulation using real mobile usage traces. Lan Zhang et al. in \[16\] design and construct a multi-hop networking system named MoNet based on WiFi, and on top of which they designed and implemented WiFace, a privacy-aware geosocial networking service. For the situation without any infrastructure, they designed a distributed content sharing protocol which can significantly shorten the relay path,
reduce conflicts and improve data persistence and availability. A role strategy is designed to encourage users to collaborate in the network. Furthermore, a key management and an authorization mechanism are developed to prevent some attacks and protect privacy. Dejan Kovachev et al. in [17] propose Mobile Community Cloud Platform (MCCP) as a cloud computing system that can leverage the full potential of mobile community growth. An analysis of the core requirements of common mobile communities is provided before they present the design of their cloud computing architecture that supports building and evolving of mobile communities.

2.4. Resource management

Mobile cloud computing combines wireless access service and cloud computing to improve the performance of mobile applications. Apart from partition and offloading [18], resource management has emerged to be an important issue. Resource management for mobile cloud computing must take into account not only the radio resource for wireless access and the computing resource for data processing, but also all kind of system resources. For mobile cloud computing systems, the mechanisms of resource management from all angles should be explored to achieve the maximum profits of providers while fulfilling the maximum satisfaction of mobile users.

2.4.1. Resource Allocation for Security Services in Mobile Cloud Computing

Hongbin Liang et al. in [19] propose a Security Service Admission Model (SSAM) based on Semi-Markov Decision Process to model the system reward for the cloud provider. They first define system states by a tuple represented by the numbers of cloud users and their associated security service categories, and current event type (i.e., arrival or departure). They then derive the system steady-state probability and service request blocking probability by using the proposed SSAM. The approach provides strong security protection while achieving resource management for the maximum revenue.

2.4.2. Access Schemes for Mobile Cloud Computing

In [20], Andreas Klein et al. propose an architecture to provide an intelligent network access strategy for mobile users to meet the application requirements. The authors propose Context Management Architecture (CMA) which is responsible for acquiring, processing, managing, and delivering context information. Finally, the authors present a context-aware radio network simulator (CORAS) that is able to model context availability, accuracy, and delay, thus enabling an evaluation of the impact of different levels of context relevance, confidence, and quality on simulation results.

2.4.3. Game Theoretic Modeling of Cooperation among Service Providers in Mobile Cloud Computing Environments

Dusit Niyato et al. in [18] present a game theoretic modeling of cooperation among service providers in mobile cloud computing environments. The authors consider a mobile cloud computing environment in which the service providers can form a coalition to create a resource pool to support the mobile applications. First, an admission control mechanism is used to provide services of mobile applications to the users given the available long-term reserved resources in a pool. An optimization formulation is introduced to obtain the optimal decision of admission control. Then, for a given coalition of service providers, the revenue obtained from utilizing the resource pool has to be shared among the service providers. A coalitional game model is developed for sharing the revenue. In addition, since the service providers can decide on short-term capacity expansion of the resource pool, a game model is introduced to obtain the optimal strategies of service providers on capacity expansion such that their profits are maximized. The approach has achieve the maximum profits, while making the sharing of revenue more reasonable among the providers.
Mobile cloud computing is the combination of cloud computing and mobile Internet. It aims to take advantage of cloud computing technologies for storage and processing of data on mobile devices, leading to reducing their limitations and enhancing the capability of mobile devices. In this paper, first we introduce the cloud computing and mobile cloud computing. After that, we introduce the architecture of mobile cloud computing in brief. And then we introduce the recent research about several core issues of mobile cloud computing which have been challenging the development. The issues contain mobile applications partition, computation offloading, context-aware mobile services and resource management. At present, most research works contribute to the development of mobile cloud computing by mainly dealing with directions as mentioned in previous section. However, we know that there are still some issues such as low bandwidth, network access management, quality of service, pricing, standard interface and service convergence etc. needing to be addressed in the future.

4. References


[16] Lan Zhang, Xuan Ding, Zhiguo Wan, Ming Gu and Xiang-Yang Li, “WiFace: a secure geosocial networking system using WiFi-based multihop MANET”, In Proceeding(s) of the 1st ACM Workshop on Mobile Cloud Computing & Services: Social Networks and Beyond, 2010.


