An SMDP-Based Service Model for Interdomain Resource Allocation in Mobile Cloud Networks

Hongbin Liang, *Student Member, IEEE*, Lin X. Cai, *Member, IEEE*, Dijiang Huang, *Senior Member, IEEE*, Xuemin (Sherman) Shen, *Fellow, IEEE*, and Daiyuan Peng, *Member, IEEE*

***Abstract*—Mobile cloud computing is a promising technique that shifts the data and computing service modules from individual devices to a geographically distributed cloud service architecture. A general mobile cloud computing system is comprised of multiple cloud domains, and each domain manages a portion of the cloud system resources, such as the Central Processing Unit, memory and storage, etc. How to efficiently manage the cloud resources across multiple cloud domains is critical for providing continuous mobile cloud services. In this paper, we propose a service decision making system for interdomain service transfer to balance the computation loads among multiple cloud domains. Our system focuses on maximizing the rewards for both the cloud system and the users by minimizing the number of service rejections that degrade the user satisfaction level significantly. To this end, we formulate the service request decision making process as a semi-Markov decision process. The optimal service transfer decisions are obtained by jointly considering the system incomes and expenses. Extensive simulation results show that the proposed decision making system can significantly improve the system rewards and decrease service disruptions compared with the greedy approach.**

***Index Terms*—Blocking probability, mobile cloud computing service domain, semi-Markov decision process (SMDP), system rewards.**

I. Introduction

CLOUD computing is a promising platform to assist mobile devices in computing and communication. In cloud computing, data and computing modules are located at remote devices in a resource-on-demand and a pay-as-you-go manner [1]. Mobile cloud has become a service model that allows mobile devices to utilize the resource from the cloud without complex hardware and software implementations at the device side [2]–[5]. Due to the mobility of mobile users, location-based (or geo-based) cloud resource provisioning is required to reduce the end-to-end communication delay. As the result, the mobile cloud system should consist of multiple cloud service domains (i.e., partitioned by geographic locations). One cloud service domain usually provides cloud services to local mobile devices that are connected through local base stations or Internet access points. Although the resources of the mobile cloud are considered as "infinite" compared with those in a single mobile device, the available resources in one cloud service domain are usually limited. Therefore, the service transitions between different mobile cloud domains play a critical role in improving the overall cloud resource utilization and quality of experience (QoE) [6] for mobile users (e.g., less response time).

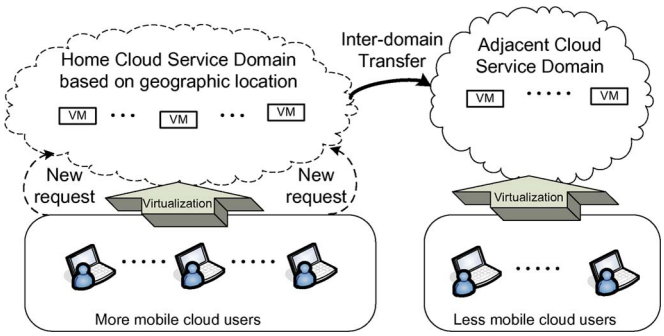

Fig. 1. Example of mobile cloud computing.

Fig. 1 shows our mobile cloud service model, which is a geographically distributed mobile cloud system that is currently developed by MobiCloud [7]. The mobile cloud service model follows the mobile cloud service framework of [2], where a virtual machine (VM) is the minimal portion of the cloud resource that can be allocated to a cloud service. When a mobile user sends a service request to the mobile cloud system, a cloud service domain (i.e., confined by a geographic location) that is close to the mobile users' location is selected. After connecting to the mobile cloud, one or multiple VMs are dedicatedly assigned to each mobile device that is located in any mobile cloud service domain. We denote the connecting mobile cloud service domain as the home cloud domain, which is geographically close to the mobile device's location.

In MobiCloud, the elastic mobile cloud service model is defined as follows: To initiate the mobile cloud service, a mobile user first initiates a request to its home cloud domain according

Manuscript received November 28, 2011; revised February 6, 2012; accepted March 19, 2012. Date of publication April 14, 2012; date of current version June 12, 2012. This work was supported by the Natural Sciences and Engineering Research Council of Canada and the National Basic Research Program of China under Grant 2012CB316100 and Grant 2011CB302902. The review of this paper was coordinated by Dr. P. Lin.

H. Liang is with the School of Information Science and Technology, Southwest Jiaotong University, Chengdu, Sichuan 610031, China, and also with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON N2L 3G1, Canada (e-mail: hbliang@bbcr.uwaterloo.ca).

L. X. Cai is with the Department of Electrical Engineering, Princeton University, Princeton, NJ 08544 USA (e-mail: lincai@princeton.edu).

D. Huang is with the School of Computing Informatics and Decision Systems Engineering, Arizona State University, Tempe, AZ 85281-8809 USA (e-mail: dijiang@asu.edu).

X. Shen is with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON N2L 3G1 Canada (e-mail: xshen@bbcr.uwaterloo.ca).

D. Peng is with the School of Information Science and Technology, Southwest Jiaotong University, Chengdu, Sichuan 610031, China (e-mail: dypeng@swjtu.edu.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TVT.2012.2194748

to its geographical locations; if the mobile cloud service request is new in the home cloud domain, the resource management controller of the home cloud domain decides whether the service request should be accepted or transferred based on the available system resource; when a request is accepted, a VM is or multiple VMs are allocated to the requesting mobile device for cloud related operations; if the available resources of the home cloud domain is not sufficient, a transfer decision will be made, and then the service request will be transferred to an adjacent cloud service domain.

In this paper, we study cloud resource allocation in a multidomain mobile cloud system that has the following properties: 1) Both the arrivals and departures of mobile cloud services follow Poisson distribution; 2) the available resource of the cloud is time varying; and 3) current resource decision may have a big impact on the future decision. In a multidomain cloud system, the overall system performance degrades if the mobile cloud system does not consider the relationship between present and future in terms of the resource allocation decisions and outcomes [8]. To construct a comprehensive resource allocation model for geo-based mobile cloud computing, we present a decision support system for resource management with considering cloud system resource, profit gain, and mobile users' QoE. The objective of this paper is to maximize the overall rewards of the cloud system and mobile users. In our presented model, both the arrivals and departures of mobile application services are random and bring the state changes of the cloud's resource. According to the definition of semi-Markov decision process (SMDP) [8], [9], the decision epoch of SMDP can be chosen at the point when any random event occurs. Thus, we first analyze the system rewards within a cloud domain considering interdomain resource transfer based on a SMDP model. The presented resource allocation decision model is to obtain the optimal resource allocation among mobile cloud service domains. We show that the presented solution can not only improve the cloud system resource utilization but also achieve better QoE for mobile users. To verify the performance of our proposed model, we perform a simulation-based study by comparing the performance of our model with the greedy algorithm [10], where greedy algorithm always allocates as much resource as possible to the mobile service requests. Our extensive simulation results show that the service rejection probability with interdomain service transfer is decreased by 20% compared with the greedy approach.

The remainder of this paper is organized as follows. The related work is presented in Section II. The system model is described in Section III. An SMDP model is developed in Section IV. Based on the SMDP model, we derive the dropping probabilities in Section V, followed by performance analysis in Section VI. Finally, concluding remarks and future work are given in Section VII.

II. Related Work

Recent research on cloud computing has been focused on mobile devices of cloud computing [11], which enables running applications between resource-constrained devices and Internet-based clouds. Moreover, resource-constrained mobile devices can outsource computation/communication/storage intensive operations to the mobile cloud. CloneCloud [5] focused on execution augmentation with less consideration on user preference or device status. Elastic applications for mobile devices via cloud computing were studied in [12]. Oberheide *et al.* [13] presented a framework that outsources the antivirus services from mobile devices to a cloud. Goyal and Carter proposed a secure cyber foraging mechanism for resource-constrained devices [14]. In [2], Huang *et al.* presented a mobile cloud computing model that allows the mobile device related operations residing either on mobile devices or dedicated VMs in the cloud. The problem of ensuring the integrity of data storage in cloud computing is studied in [15] and [16]. Although resource management in wireless networks has been extensively studied [17]–[24], it is not well studied in mobile cloud computing. In [25], an economic cloud computing model is presented to decide how to manage the computing tasks with a given configuration of the cloud system, i.e., the computing tasks can be migrated between the mobile devices and the cloud servers.

Specialized hardware-based solutions for high availability (HA) are expensive and may require changes on the applications [26]. Software-based solutions for HA provide virtualized execution environment (VM) for applications and fast recovery mechanisms when physical hosts become unavailable [27], [28]. A game theory-based resource allocation model to allocate cloud resources according to the users' QoS requirements is proposed in [29]. The other mobile cloud computing solutions are limited and solely focused on the enhancement of the individual mobile device's capability. To the best of our knowledge, none of the previous works addressed how to construct a mobile cloud computing system reward model for resource allocation considering the whole rewards of both cloud systems and mobile users and how to select a cloud domain to allocate system resource through interdomain service transfers.

III. System Model

In this section, we present our proposed mobile cloud resource management model for choosing the optimal adjacent cloud domains. We first describe the optimal algorithm of our presented mobile cloud resource management model. Then, the system states of our proposed model and the actions of each state are described. Finally, we describe the system reward model, which is critical for connecting home cloud domain to decide whether the mobile service should be accepted, rejected, or transferred to an adjacent cloud domain. Before making a decision, home cloud domain needs to obtain the entire system reward for each action (i.e., accept, reject, and transfer) and makes the optimal decision for our reward model.

A. System Description

As shown in Fig. 1, we consider a mobile cloud system that is composed by multiple service domains. Suppose there are K-VM resource available in one cloud domain, and a service occupies c VMs, where $c \in \{1, 2, \ldots C\}$, $C \leq K$. In each service domain, there are two types of service requests, namely,

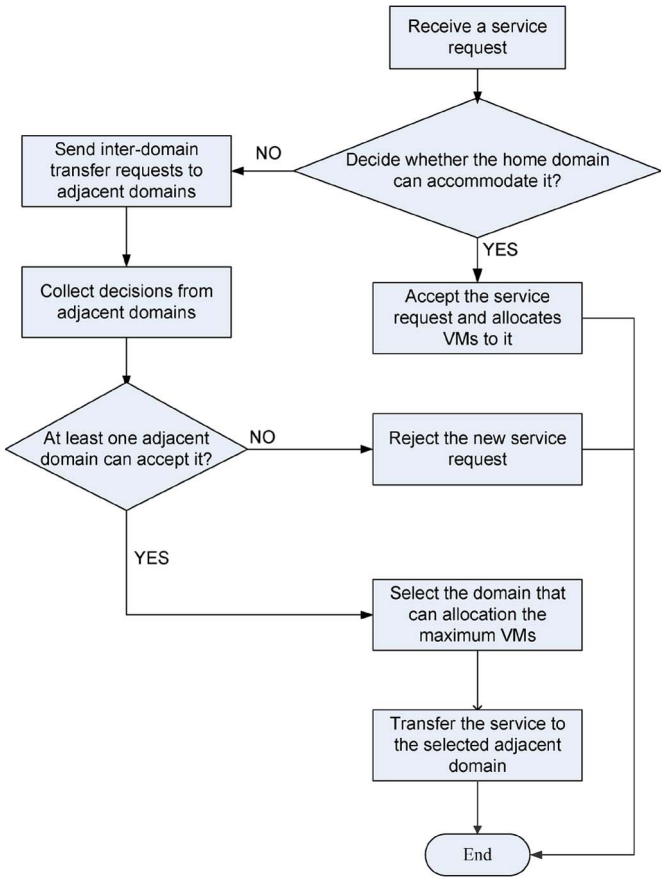

Fig. 2. System algorithm of proposed dynamic selection adjacent mobile cloud domain model.

TABLE I
NOTATIONS

K	The number of maximal VMs in one cloud domain
C	The maximal number of VMs that the cloud domain can allocate to a mobile application service
c	The actual number of VMs that the cloud domain can allocate to a mobile application service
λ_n	The arrival rate of new service
λ_t	The arrival rate of inter-domain service
μ	The departure rate of the finished service which occupies one VM
i	The index of the adjacent cloud domain
s_c	The number of served services that have been allocated c VMs
S	The system state aggregation of one cloud domain
e	The event aggregation of one cloud domain
A_n	The arrival of a new service
A_t	The arrival of an inter-domain service
F_c	The departure of a finished service
E_d	The income of admitting a new service
E_t	The payment to transfer an inter-domain service to an adjacent cloud domain
δ_s	The transition time of the inter-domain service between cloud domains
δ_d	The transition time of the new service between mobile device and cloud domain
β	The price per unit time
U_d	The energy expense of a mobile device to run a mobile application service
θ_d	The time of a mobile device to run a mobile application service
α	The continuous-time discounting factor

new service requests initiated in the home cloud domain and interdomain transfer service request from/to adjacent cloud domains. We assume that the arrival rates of both new service requests and interdomain transfer service requests follow Poisson distributions with mean λ_n and λ_t, respectively. For different services, the service time follows an exponential distribution with different mean rates. Let μ denote the computation rate of one VM for the requested service task. However, when a request is admitted, computing resources (i.e., VMs) will be allocated. Thus, the occupation time of c VMs in the cloud is $1/(c\mu)$.

The decision-making procedure in a multidomain cloud system is shown in Fig. 2. When a new mobile cloud service request arrives, the controller of the home cloud domain evaluates the expected system gain and the expected system expenses, including the cost of occupying VMs during the computation period, the communication cost between the cloud and mobile devices, and the power consumption of the mobile devices, to decide whether to accept, reject, or transfer the request to adjacent cloud domains. If the home domain cannot accommodate the mobile service request, then it needs to evaluate if the service request can be successfully transferred to another cloud service domain. Hence, the home domain cloud sends an interdomain transferring request to its geographically adjacent domains. Each adjacent domain will then return a decision by considering the expected computing resource occupation in its own domain, the extra communication overheads between the mobile device and the cloud, and the involved transfer fee, if there is one. If there is no adjacent domain that can accept the service request, the home domain should only reject the mobile service request. Then, the mobile device has to run the service task on the device itself, the restricted power and computing resources of which may result in a low QoE. While if multidomains can accept the service request, the home cloud domain needs to decide which adjacent cloud domain the home service request can be accepted based on the feedback collected from its adjacent cloud domains. For instance, a home cloud domain may select a neighboring cloud domain that can allocate the maximum number of VMs to the requested service

$$\tilde{i} = arg \max_i \{c_i\} \tag{1}$$

where i denotes the ith adjacent cloud domain, $\tilde{i}$ denotes the optimal adjacent cloud domain that the home cloud domain selects, and c_i is the number of VMs that the ith adjacent cloud domain can allocate to the interdomain transfer from the home cloud domain.

In Table I, we highlight the notations used in this paper.

Detailed formulation of the system rewards and the action model will be described in the following sections.

B. System States

The number of service requests that have been allocated c VMs is denoted as s_c. The total number of occupied VMs in a

cloud domain is $\sum_{c=1}^{C}(s_c * c)$, where C is the maximum number of VMs that can be allocated to a service request. A_n and A_t are the arrival of a new service request and an interdomain transfer service request, respectively. When a service completes and leaves the cloud system, the occupied VMs will be released, and the available VMs in a cloud domain need to be updated. We use F_c to denote a departure of a service with $1 \leq c \leq C$ VMs. An event in the event set $\mathbf{e}$ of a cloud computing system can be described as $e \in \mathbf{e} = \{A_n, A_t, F_1, F_2, \ldots, F_C\}$.

The system state S of a cloud domain can be characterized by the current services with different numbers of VMs and an event in the system, which could be either an arrival or a departure, i.e.,

$$S = \{s | s = \langle s_1, s_2, \ldots, s_C, e\rangle = \langle \overline{s}, e\rangle\} \tag{2}$$

where $\overline{s} = \langle s_1, s_2, \ldots, s_C\rangle$, and $\sum_{c=1}^{C}(s_c * c) \leq K$.

C. Actions

Upon receiving a request, three actions can be chosen from the action set, accept with c VMs, reject, and transfer, which can be denoted as $a(s) = c$, $c \in \{1, 2, \ldots C\}$, $a(s) = 0$, and $a(s) = -1$, respectively. When a service completes and departs the cloud domain, no other action is required except the available VMs in the cloud should be updated, which is denoted as $a(s) = -2$. Thus, the action space $Act_s = -2, -1, 0, 1, 2, \ldots, C$, and the action set $a(s)$ is

$$a(s) = \begin{cases} \{-1, 0, 1, \ldots C\}, & e \in \{A_n, A_t\} \\ -2, & e \in \{F_1, F_2, \ldots, F_C\}. \end{cases} \tag{3}$$

D. Reward Model

Based on the system state and the corresponding action, the overall system reward of a cloud network, denoted by $r(s, a)$, can be evaluated as

$$r(s, a) = w(s, a) - g(s, a) \tag{4}$$

where $s = \langle \overline{s}, e\rangle$, $e \in \{A_n, A_t, F_1, F_2, \ldots, F_C\}$, $w(s, a)$ is the lump sum income of the cloud system by making a decision/action a when event e occurs in state s, and $g(s, a)$ is the expected system cost.

The lump sum income $w(s, a)$ is computed as

$$w(s, a) = \begin{cases} 0, & a(s) = -2, \\ & e \in \{F_1, F_2, \ldots, F_C\} \\ E_d - E_t - \delta_s\beta - \delta_d\beta, & a(s) = -1, e = A_n \\ -\delta_s\beta - \delta_d\beta, & a(s) = -1, e = A_t \\ -U_d - \theta_d\beta, & a(s) = 0, e = A_n \\ 0, & a(s) = 0, e = A_t \\ E_d - \delta_d\beta - \frac{\beta}{c\mu}, & a(s) = c, e = A_n \\ E_t - \frac{\beta}{c\mu}, & a(s) = c, e = A_t. \end{cases} \tag{5}$$

A system will not gain income when a service completes and leaves the system, and we have $w(s, a) = 0$ for $a(s) = -2$, and $e \in \{F_1, F_2, \ldots, F_C\}$. When a new service request is admitted into the system, the income of E_d is earned, and in the meantime, the admitted service will use c VMs in the cloud domain, which involves $\delta_d\beta$ transition expense and $\beta/c\mu$ resource occupation expense. Here, the transition expense is the cost to transfer the computing task from the mobile device to the cloud, and the resource expense is the cost of VMs being occupied during the service time of the request, where δ_d denotes the time consumed on transmitting the new service request from the mobile device to the cloud through wireless, and β denotes the price per unit time, which has the same measurement unit as the income.

For an interdomain transfer request, the home cloud domain pays E_t income to an adjacent domain, and thus, the expected reward of the new cloud is E_t minus the resource expense $E_t - \beta/c\mu$ for $a(s) = c$ and $e = A_t$. If the new service request is rejected, the computing task has to be run at the mobile device, which causes energy expense U_d and the resource expense $\theta_d\beta$, where θ_d is the service time of a mobile device, and $\theta_d \gg 1/\mu$ due to the limited computation capability of a mobile device. There is no income to reject a transfer request as the income has been calculated in the home cloud domain. When a new request is transferred to an adjacent domain, the home cloud domain earns E_d and pays E_t to the transferred adjacent domain. Similarly, there is a transition expense $\delta_d\beta$ to transfer the computations from mobile device to the cloud. In addition, the migration between different cloud domains also involves an extra communication expense, which is denoted by $\delta_s\beta$, where δ_s denotes the time consumed when transferring the service request between different cloud domains. Therefore, $w(s, a) = E_d - E_t - \delta_s\beta - \delta_d\beta$, where $a(s) = -1$, and $e = A_n$. However, the reward model is different when an interdomain transfer request is transferred to an adjacent domain by the home domain. In this case, the home domain obtains E_t from the adjacent domain that transfers the new request to the home domain but pays E_t to the transferred adjacent domain for the interdomain transfer request. Thus, the earning of the home cloud domain is 0. The costs $\delta_s\beta$ and $\delta_d\beta$ are the same as that of a new request as well. Then, $w(s, a) = -\delta_s\beta - \delta_d\beta$ when $a(s) = -1$ and $e = A_t$.

The expected system cost $g(s, a)$ is given by (4), i.e.,

$$g(s, a) = \tau(s, a)o(s, a), a(s) \in Act_s \tag{6}$$

where $\tau(s, a)$ is the expected service time when the system transfers from the current state s to the next state when decision a is made; $o(s, a)$ is the cost rate of the service time, and it is determined by the number of occupied VMs

$$o(s, a) = \sum_{c=1}^{C}(s_c * c). \tag{7}$$

IV. Semi-Markov Decision Process-Based Mobile Computing Model

In this section, we develop an SMDP-based mobile computing model to analyze the performance of a cloud network. Our main objective is to make optimal decisions at decision epochs, i.e., when a service request arrives (i.e., A_n or A_t) or a service

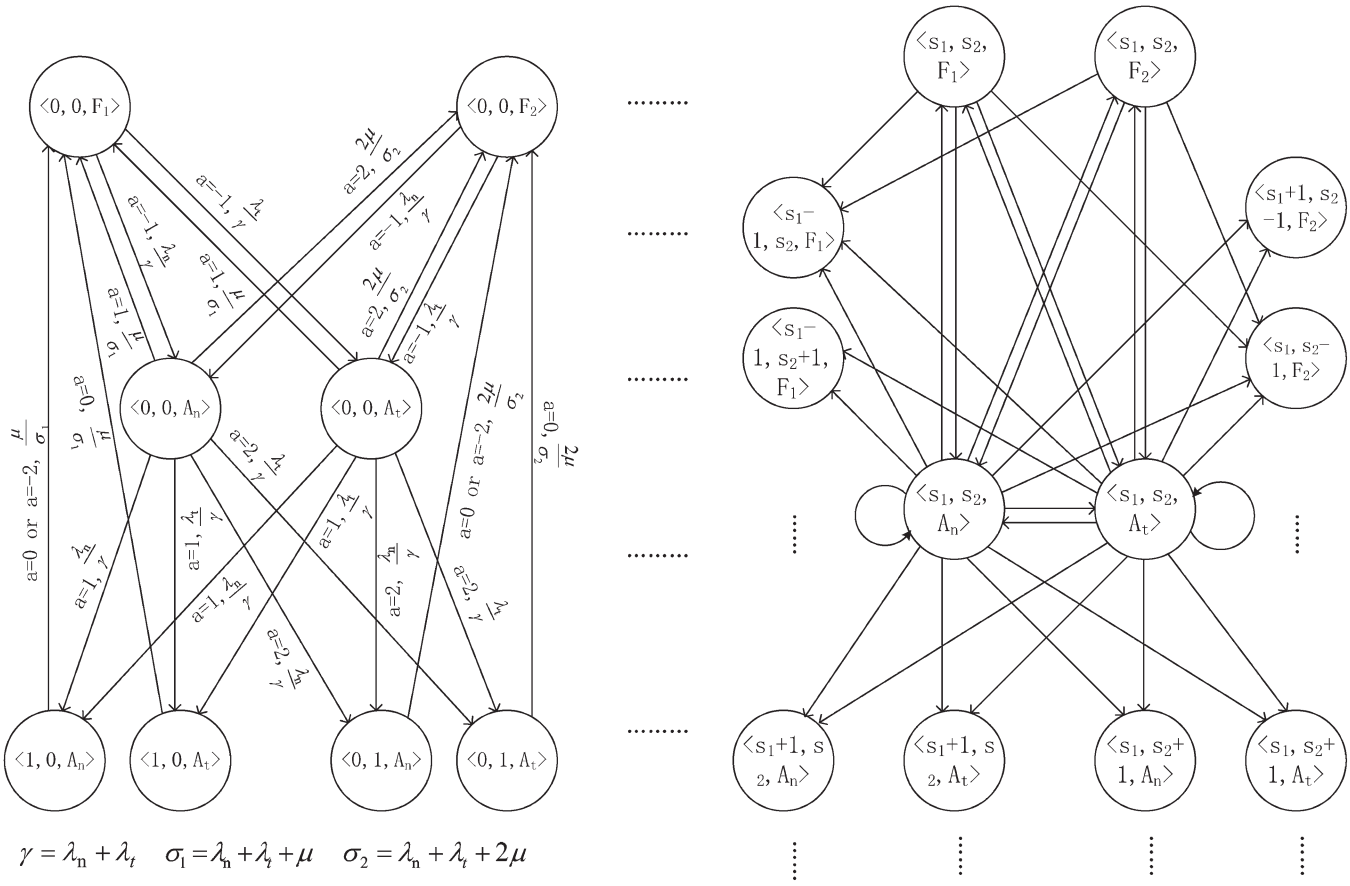

Fig. 3. State transition diagram.

completes and leaves the system (F_c), the long-term expected system rewards are maximized. The time duration between two continuous decision epochs also follows an exponential distribution. The state transition diagram with $C = 2$ is illustrated in Fig. 3, where the first item in the state transition diagram represents the action, and the second item presents the state transition probability. The three tuples in Fig. 3 represents the system state of the mobile cloud domain. Taking $\langle 1, 0, A_n \rangle$ as an example, the first two items in the three tuples indicate that the total number of services that occupy one and two VMs is 1 and 0, respectively, and the third item A_n in the three tuples denotes an arrival of a new service request.

Given the current state s and the selected decision a, we denote the time duration from this epoch to the next epoch by $\tau(s, a)$. Therefore, the mean rate of events for a given s and a, denoted as $\gamma(s, a)$, is the summation of the rates of all events in the system, which is given by

$$\gamma(s,a) = \tau(s,a)^{-1} = \begin{cases} \lambda_n + \lambda_t + \sum_{c=1}^{C} s_c c\mu & e \subseteq \{F_1, F_2, \ldots, F_C\} \\ & e \subseteq \{A_n, A_t\}, a = -1 \\ & e \subseteq \{A_n, A_t\}, a = 0 \\ \lambda_n + \lambda_t + \sum_{c=1}^{C} s_c c\mu + c\mu, & e \subseteq \{A_n, A_t\}, a = c \end{cases} \tag{8}$$

where λ_n and λ_t are the arrival rates of the new and transfer requests, respectively. When a departure occurs, or an arriving request is rejected or transferred, the total number of existing services in the cloud domain is $\sum_{c=1}^{C} s_c$, and thus, the rate of an existing service departing the system is $\sum_{c=1}^{C} s_c c\mu$. When a service request is admitted, we have $\sum_{c=1}^{C} s_c + 1$ services, which accounts for $\sum_{c=1}^{C} s_c c\mu + c\mu$.

We then evaluate the expected discounted reward (denoted as $r(s, a)$) during $\tau(s, a)$ based on the discounted reward model defined in [8] and [30], i.e.,

$$\begin{aligned} r(s,a) &= w(s,a) - o(s,a) E_s^a \left\{ \int_0^{\tau} e^{-\alpha t} dt \right\} \\ &= w(s,a) - o(s,a) E_s^a \left\{ \frac{[1 - e^{-\alpha\tau}]}{\alpha} \right\} \\ &= w(s,a) - \frac{o(s,a)}{\alpha + \gamma(s,a)} \end{aligned} \tag{9}$$

where α is a continuous-time discounting factor.

$q(j|s, a)$ is defined as the state transition probability from state s to state j when action a is chosen. We then derive the state transition probability, as shown in Fig. 3. For

the state $s = \langle s_1, s_2, \ldots, s_c, \ldots, s_C, A_n\rangle$, $q(j|s,a)$ can be obtained as

$$q(j|s,a) = \begin{cases} \frac{\lambda_n}{\gamma(s,a)}, & j = \langle s_1, \ldots, s_C, A_n\rangle \\ & a = 0, -2 \\ \frac{\lambda_t}{\gamma(s,a)}, & j = \langle s_1, \ldots, s_C, A_t\rangle \\ & a = 0, -2 \\ \frac{s_c c\mu}{\gamma(s,a)}, & j = \langle s_1, ., s_c - 1, ., s_C, F_c\rangle, s_c \geq 1 \\ & a = 0, -2 \\ \frac{(s_c+1)c\mu}{\gamma(s,a)}, & j = \langle s_1, ., s_c, ., s_C, F_c\rangle \\ & a = c \\ \frac{s_m m\mu}{\gamma(s,a)}, & j = \langle s_1, .s_m - 1, .s_c + 1, .s_C, F_m\rangle \\ & s_m \geq 1, m \neq c, a = c \\ \frac{\lambda_n}{\gamma(s,a)}, & j = \langle s_1, ., s_c + 1, ., s_C, A_n\rangle \\ & s_c \leq C - 1, a = c \\ \frac{\lambda_t}{\gamma(s,a)}, & j = \langle s_1, ., s_c + 1, ., s_C, A_t\rangle \\ & s_c \leq C - 1, a = c \end{cases} \tag{10}$$

where $c \in \{1, 2, \ldots, C\}$, $m \in \{1, 2, \ldots, C\}$, $m \neq c$.

For the states $s = \langle s_1, s_2, \ldots, s_c, \ldots, s_C, A_t\rangle$, $q(j|s,a)$ can be obtained as

$$q(j|s,a) = \begin{cases} \frac{\lambda_n}{\gamma(s,a)}, & j = \langle s_1, \ldots, s_C, A_n\rangle \\ & a = 0 \\ \frac{\lambda_t}{\gamma(s,a)}, & j = \langle s_1, \ldots, s_C, A_t\rangle \\ & a = 0 \\ \frac{s_c c\mu}{\gamma(s,a)}, & j = \langle s_1, ., s_c - 1, ., s_C, F_c\rangle \\ & s_c \geq 1, a = 0 \\ \frac{(s_c+1)c\mu}{\gamma(s,a)}, & j = \langle s_1, ., s_c, ., s_C, F_c\rangle \\ & a = c \\ \frac{s_m m\mu}{\gamma(s,a)}, & j = \langle s_1, .s_m - 1, .s_c + 1, .s_C, F_m\rangle \\ & s_m \geq 1, m \neq c, a = c \\ \frac{\lambda_n}{\gamma(s,a)}, & j = \langle s_1, ., s_c + 1, ., s_C, A_n\rangle \\ & s_c \leq C - 1, a = c \\ \frac{\lambda_t}{\gamma(s,a)}, & j = \langle s_1, ., s_c + 1, ., s_C, A_t\rangle \\ & s_c \leq C - 1, a = c \end{cases} \tag{11}$$

where $c \in \{1, 2, \ldots, C\}$, $m \in \{1, 2, \ldots, C\}$, $m \neq c$.

For the states $s = \langle s_1, s_2, \ldots, s_c, \ldots, s_C, F_c\rangle$, when a service leaves the system, there is no special action required, and $a = -2$; thus, the transition probability $q(j|s,a)$ is

$$q(j|s,a) = \begin{cases} \frac{\lambda_n}{\gamma(s,a)}, & j = \langle s_1, s_2, \ldots, s_C, A_n\rangle \\ \frac{\lambda_t}{\gamma(s,a)}, & j = \langle s_1, s_2, \ldots, s_C, A_t\rangle \\ \frac{s_c c\mu}{\gamma(s,a)}, & j = \langle s_1, s_2, ., s_c - 1, ., s_C, F_c\rangle, s_c \geq 1 \end{cases} \tag{12}$$

where $c \in \{1, 2, \ldots, C\}$.

Based on the derived transition probabilities, we can obtain the maximum long-term discounted reward using a discounted reward model defined in [8] and [30] as

$$\nu(s) = \max_{a \in Act_s} \left\{ r(s,a) + \lambda \sum_{j \in S} q(j|s,a)\nu(j) \right\} \tag{13}$$

where $\lambda = \gamma(s,a)/(\alpha + \gamma(s,a))$.

Letting $w = \lambda_n + \lambda_t + K * C * \mu < \infty$, $\tilde{q}(j|s,a)$, $\tilde{v}(s)$, and $\tilde{r}(s,a)$ are defined as the uniformed transition probability, long-term reward, and reward function, respectively. We derive the optimality equation of $\nu(s)$ after the uniformization as

$$\tilde{\nu}(s) = \max_{a \in \tilde{A}} \left\{ \widetilde{r}(s,a) + \tilde{\lambda} \sum_{j \in \mathcal{S}} \tilde{q}(j|s,a)\tilde{\nu}(j) \right\} \tag{14}$$

where $\widetilde{r}(s,a) \equiv r(s,a)(1 + \alpha\tau(s,a))/((\alpha + w)\tau(s,a))$, $\tilde{\lambda} = w/(w + \alpha)$, and

$$\tilde{q}(j|s,a) = \begin{cases} 1 - \frac{[1 - q(s|s,a)]}{\tau(s,a)w}, & j = s \\ \frac{q(j|s,a)}{\tau(s,a)w}, & j \neq s. \end{cases} \tag{15}$$

V. Performance Analysis

In this section, we analyze the performance of the proposed SMDP-based interdomain resource allocation scheme. An important performance metric is the dropping probability of the cloud system. When a request is rejected, there is no system income for the mobile cloud computing system, and the mobile user will experience a low QoE. Basically, a service request is dropped when both the home cloud domain and any of the adjacent cloud domains cannot accommodate it. Thus, the dropping probability of a new service request depends on the available resources in both the home cloud domain and the neighboring cloud domains. In the following, we will derive the dropping probabilities of new services and interdomain transfer services based on the proposed SMDP model.

Let $\pi_{\langle s_1, s_2, \ldots, s_C, e\rangle}$ (or denoted by $\pi_{\langle \overline{s}, e\rangle}$) be the steady-state probability of state $\overline{s} = \langle s_1, s_2, \ldots, s_C, e\rangle$ in the home cloud service domain. According to different events, i.e., an arrival of a new service request, an arrival of an interdomain transfer service request, or a departure of a completed service with c VMs, $\pi_{\langle \overline{s}, e\rangle}$ can be further divided into three items, i.e., $\pi_{\langle \overline{s}, A_n\rangle}, \pi_{\langle \overline{s}, A_t\rangle}$, and $\pi_{\langle \overline{s}, F_c\rangle}$. Based on the transition probabilities derived in (10)–(12), we can obtain $\pi_{\langle \overline{s}, A_n\rangle}$ and $\pi_{\langle \overline{s}, A_t\rangle}$ as follows:

$$\begin{aligned} \pi_{\langle \overline{s}, A_n\rangle} = {} & \frac{\lambda_n}{\gamma(s,a)} \rho_{\langle \overline{s}, A_n\rangle} \pi_{\langle \overline{s}, A_n\rangle} + \frac{\lambda_n}{\gamma(s,a)} \rho_{\langle \overline{s}, A_t\rangle} \pi_{\langle \overline{s}, A_t\rangle} \\ & + \frac{\lambda_n}{\gamma(s,a)} \sum_{c=1}^{C} \rho_{\langle \overline{s}, A_n\rangle} \pi_{\langle \overline{s}, A_n\rangle} \\ & + \frac{\lambda_n}{\gamma(s,a)} \sum_{c=1}^{C} \rho_{\langle \overline{s_{-1}}, A_t\rangle} \pi_{\langle \overline{s_{-1}}, A_t\rangle} \\ & + \frac{\lambda_n}{\gamma(s,a)} \sum_{c=1}^{C} \pi_{\langle \overline{s}, F_c\rangle} \end{aligned} \tag{16}$$

$$\begin{aligned} \pi_{\langle \overline{s}, A_t\rangle} = {} & \frac{\lambda_t}{\gamma(s,a)} \rho_{\langle \overline{s}, A_n\rangle} \pi_{\langle \overline{s}, A_n\rangle} + \frac{\lambda_t}{\gamma(s,a)} \rho_{\langle \overline{s}, A_t\rangle} \pi_{\langle \overline{s}, A_t\rangle} \\ & + \frac{\lambda_t}{\gamma(s,a)} \sum_{c=1}^{C} \rho_{\langle \overline{s_{-1}}, A_n\rangle} \pi_{\langle \overline{s_{-1}}, A_n\rangle} \\ & + \frac{\lambda_t}{\gamma(s,a)} \sum_{c=1}^{C} \rho_{\langle \overline{s_{-1}}, A_t\rangle} \pi_{\langle \overline{s_{-1}}, A_t\rangle} \\ & + \frac{\lambda_t}{\gamma(s,a)} \sum_{c=1}^{C} \pi_{\langle \overline{s}, F_c\rangle} \end{aligned} \tag{17}$$

where $\overline{s} = \langle s_1, s_2, \ldots, s_C \rangle$, and $\overline{s_{-1}} = \langle s_1, s_2, \ldots, s_c - 1, \ldots, s_C \rangle$. $\rho_{\langle \overline{s}, A_n \rangle}$, $\rho_{\langle \overline{s}, A_t \rangle}$, $\rho_{\langle \overline{s_{-1}}, A_n \rangle}$, and $\rho_{\langle \overline{s_{-1}}, A_t \rangle}$ are parameters defined as

$$\rho_{\langle \overline{s}, A_n \rangle} = \begin{cases} 1, & a_{\langle \overline{s}, A_n \rangle} = 0, -2 \\ 0, & \text{otherwise} \end{cases}$$

$$\rho_{\langle \overline{s}, A_t \rangle} = \begin{cases} 1, & a_{\langle \overline{s}, A_t \rangle} = 0, -2 \\ 0, & \text{otherwise} \end{cases}$$

$$\rho_{\langle \overline{s_{-1}}, A_n \rangle} = \begin{cases} 1, & a_{\langle \overline{s_{-1}}, A_n \rangle} = c, \quad c = \{1, 2, \ldots, C\} \\ 0, & \text{otherwise} \end{cases}$$

$$\rho_{\langle \overline{s_{-1}}, A_t \rangle} = \begin{cases} 1, & a_{\langle \overline{s_{-1}}, A_t \rangle} = c, \quad c = \{1, 2, \ldots, C\} \\ 0, & \text{otherwise.} \end{cases}$$

Similarly, the steady-state probability $\pi_{\langle \overline{s}, F_c \rangle}$ is given by

$$\begin{aligned} \pi_{\langle \overline{s}, F_c \rangle} = {} & \frac{(s_c+1)c\mu}{\gamma(s,a)} \rho_{\langle \overline{s_{+1}}, A_n \rangle} \pi_{\langle \overline{s_{+1}}, A_n \rangle} \\ & + \frac{(s_c+1)c\mu}{\gamma(s,a)} \rho_{\langle \overline{s}, A_n \rangle} \pi_{\langle \overline{s}, A_n \rangle} \\ & + \frac{(s_c+1)c\mu}{\gamma(s,a)} \sum_{m=1, m \neq c}^{C} \rho_{\langle \overline{s_{\pm 1}}, A_n \rangle} \pi_{\langle \overline{s_{\pm 1}}, A_n \rangle} \\ & + \frac{(s_c+1)c\mu}{\gamma(s,a)} \rho_{\langle \overline{s_{+1}}, A_t \rangle} \pi_{\langle \overline{s_{+1}}, A_t \rangle} \\ & + \frac{(s_c+1)c\mu}{\gamma(s,a)} \rho_{\langle \overline{s}, A_t \rangle} \pi_{\langle \overline{s}, A_t \rangle} \\ & + \frac{(s_c+1)c\mu}{\gamma(s,a)} \sum_{m=1, m \neq c}^{C} \rho_{\langle \overline{s_{\pm 1}}, A_t \rangle} \pi_{\langle \overline{s_{\pm 1}}, A_t \rangle} \\ & + \frac{(s_c+1)c\mu}{\gamma(s,a)} \sum_{m=1}^{C} \pi_{\langle \overline{s_{+1}}, F_m \rangle} \end{aligned} \tag{18}$$

where $\overline{s_{+1}} = \langle s_1, s_2, \ldots s_c + 1, \ldots s_C \rangle$, and $\overline{s_{\pm 1}} = \langle s_1, s_2, \ldots s_c + 1, ..s_m - 1, \ldots s_C \rangle$. $\rho_{\langle \overline{s_{+1}}, A_n \rangle}$, $\rho_{\langle \overline{s}, A_n \rangle}$, $\rho_{\langle \overline{s_{\pm 1}}, A_n \rangle}$, $\rho_{\langle \overline{s_{+1}}, A_t \rangle}$, $\rho_{\langle \overline{s}, A_t \rangle}$, and $\rho_{\langle \overline{s_{\pm 1}}, A_t \rangle}$ are defined as

$$\rho_{\langle \overline{s_{+1}}, A_n \rangle} = \begin{cases} 1, & a_{\langle \overline{s_{+1}}, A_n \rangle} = 0, -2 \\ 0, & \text{otherwise} \end{cases}$$

$$\rho_{\langle \overline{s}, A_n \rangle} = \begin{cases} 1, & a_{\langle \overline{s}, A_n \rangle} = c, \quad c = \{1, 2, \ldots, C\} \\ 0, & \text{otherwise} \end{cases}$$

$$\rho_{\langle \overline{s_{\pm 1}}, A_n \rangle} = \begin{cases} 1, & a_{\langle \overline{s_{\pm 1}}, A_n \rangle} = m, \quad c = \{1, 2, \ldots, C\} \\ & m = \{1, 2, \ldots, C\}, \; m \neq c \\ 0, & \text{otherwise} \end{cases}$$

$$\rho_{\langle \overline{s_{+1}}, A_t \rangle} = \begin{cases} 1, & a_{\langle \overline{s_{+1}}, A_t \rangle} = 0, \text{-2} \\ 0, & \text{otherwise} \end{cases}$$

$$\rho_{\langle \overline{s}, A_t \rangle} = \begin{cases} 1, & a_{\langle \overline{s}, A_t \rangle} = c, \quad c = \{1, 2, \ldots, C\} \\ 0, & \text{otherwise} \end{cases}$$

$$\rho_{\langle \overline{s_{\pm 1}}, A_t \rangle} = \begin{cases} 1, & a_{\langle \overline{s_{\pm 1}}, A_t \rangle} = m, \quad c = \{1, 2, \ldots, C\} \\ & m = \{1, 2, \ldots, C\}, m \neq c \\ 0, & \text{otherwise.} \end{cases}$$

TABLE II
SIMULATION PARAMETERS

Parameter	Value	Parameter	Value
E_d	50	U_d	10
E_t	80	θ_d	60
δ_s	3	β	1
δ_d	30		

Since the sum of the steady-state probabilities for all states is equal to 1, we have

$$\sum_{S} \left(\pi_{\langle \overline{s}, A_n \rangle} + \pi_{\langle \overline{s}, A_t \rangle} + \pi_{\langle \overline{s}, F_c \rangle} \right) = 1. \tag{19}$$

By solving the equation sets of (16)–(19), we can obtain the probability of each state in the steady state. The dropping probability of a new service request, which is denoted by Pn_0, is the ratio of the sum probability of the rejected new services over the total probability of new service arrivals. Similarly, the dropping probability of an interdomain transfer service request, denoted by Pt_0, is the ratio of the sum probability of the rejected interdomain transfer services over the totally probability of interdomain transfer requests. Thus, we have

$$Pn_0 = \frac{\sum\limits_{a\langle \overline{s}, A_n \rangle = 0} \pi_{\langle \overline{s}, A_n \rangle}}{\sum\limits_{m=-2, m \neq -1}^{C} \left(\sum\limits_{a\langle \overline{s}, A_n \rangle = m} \pi_{\overline{s}, A_n \rangle} \right)} \tag{20}$$

$$Pt_0 = \frac{\sum\limits_{a\langle \overline{s}, A_t \rangle = 0} \pi_{\langle \overline{s}, A_t \rangle}}{\sum\limits_{m=-2, m \neq -1}^{C} \left(\sum\limits_{a\langle \overline{s}, A_t \rangle = m} \pi_{\langle \overline{s}, A_t \rangle} \right)}. \tag{21}$$

VI. PERFORMANCE EVALUATION

In this section, we evaluate the performance of the proposed interdomain resource allocation model using an event-driven simulator written in Matlab. In the simulation, the parameters are selected as an example for performance illustration. To further study the relationship between these parameters, we vary some parameters, such as the service arrival and departure rates for performance comparison. A mobile cloud service domain contains up to $K = 10$ resource units (i.e., VMs or cloud server clusters, and we use VMs in our simulation study), and the maximum number of VMs allocated to a service request is $C = 3$, i.e., a service can be assigned 1, 2, or 3 VMs based on the dynamic computing environments in the cloud domain. Each interdomain transfer request will be accepted by an adjacent domain with a certain probability p_t, which varies from 0.5 to 0.9. The arrival rates of new and interdomain transfer services are $\lambda_n = 7.2$ and $\lambda_t = 2.4$, respectively. The departure rate of a service using one VM is $\mu_1 = 6.6$, and thus, the departure rate of services using c VMs is $\mu_c (c \in \{1, 2, 3\})$, if not otherwise specified. The discount factor is set to $\alpha = 0.1$ to assure the convergence of the reward computation. We collect the simulation results of each experiment over 18 000 s and repeat each experiment for 1000 runs with different random seeds to calculate the average. The other parameters used in the simulation are listed in Table II.

TABLE III
DECISION TABLE OF NEW SERVICE
($\lambda_n = 7.2, \lambda_t = 2.4, \mu_1 = 6.6, K = 10, s_3 = 0, p_t = 0.5$)

$s_1 \backslash s_2$	0	1	2	3	4	5
0	3	3	3	3	1	0
1	3	3	3	2	1	-
2	3	3	3	1	0	-
3	3	3	2	1	-	-
4	3	2	1	-1	-	-
5	3	2	1	-	-	-
6	2	1	0	-	-	-
7	2	1	-	-	-	-
8	1	-1	-	-	-	-
9	1	-	-	-	-	-
10	0	-	-	-	-	-

TABLE IV
DECISION TABLE OF NEW SERVICE
($\lambda_n = 7.2, \lambda_t = 2.4, \mu_1 = 6.6, K = 10, s_3 = 0, p_t = 0.9$)

$s_1 \backslash s_2$	0	1	2	3	4	5
0	3	3	3	3	1	-1
1	3	3	3	2	1	-
2	3	3	3	1	-1	-
3	3	3	2	1	-	-
4	3	3	1	-1	-	-
5	3	2	1	-	-	-
6	3	1	-1	-	-	-
7	2	1	-	-	-	-
8	1	-1	-	-	-	-
9	1	-	-	-	-	-
10	-1	-	-	-	-	-

Tables III and IV tabulate the optimal resource allocation decisions or actions for new service requests under different interdomain transfer acceptance probabilities p_t. The numbers in the table represent actions made on the state $\langle s_1, s_2, s_3 \rangle$. For example, if there are no services in the system, when a new service request arrives, an action $a = 3$ is made that three VMs are allocated to the requesting service. If there are four users, and each user has been allocated two VMs, implying that there are only two VMs available, an action $a = 1$ is made that only one VM is allocated to the new service request. When the remaining computing resources are sufficient, action $a = 3$ is usually selected to achieve a higher utility gain over $a = 2$ or $a = 1$. On the other hand, when the available computing resources are limited, a more conservative decision is selected. When the available VMs in the home cloud domain cannot accommodate the requested service, the home domain will send a request to the adjacent cloud domains. A higher probability of p_t indicates that an adjacent domain has more available computing resources, and it is more likely to accept the transfer requests from the home cloud domain that has insufficient resources. It is also observed that more transfer decisions are made with a higher p_t.

Tables V and VI tabulate the optimal resource allocation decisions or actions for interdomain transfer requests under different request arrival rates. A higher request rate implies that more computing resources are demanded, and thus, a more conservative decision is made. It is also observed that no transfer decision is made for interdomain transfer requests due to the charge of transfer by the home domain and the extra communication costs involved in transfer services.

TABLE V
DECISION TABLE OF INTERDOMAIN TRANSFER DERVICE
($\lambda_n = 1.2, \lambda_t = 2.4, \mu_1 = 6.6, K = 10, s_3 = 0$)

$s_1 \backslash s_2$	0	1	2	3	4	5
0	3	3	3	3	2	0
1	3	3	3	2	1	-
2	3	3	3	2	0	-
3	3	3	2	1	-	-
4	3	3	2	0	-	-
5	3	2	1	-	-	-
6	3	2	0	-	-	-
7	2	1	-	-	-	-
8	2	0	-	-	-	-
9	1	-	-	-	-	-
10	0	-	-	-	-	-

TABLE VI
DECISION TABLE OF INTERDOMAIN TRANSFER SERVICE
($\lambda_n = 60, \lambda_t = 2.4, \mu_1 = 6.6, K = 10, s_3 = 0$)

$s_1 \backslash s_2$	0	1	2	3	4	5
0	3	3	2	2	1	0
1	3	3	2	1	1	-
2	3	2	2	1	0	-
3	3	2	1	1	-	-
4	3	2	1	0	-	-
5	2	1	1	-	-	-
6	2	1	0	-	-	-
7	1	1	-	-	-	-
8	1	0	-	-	-	-
9	1	-	-	-	-	-
10	0	-	-	-	-	-

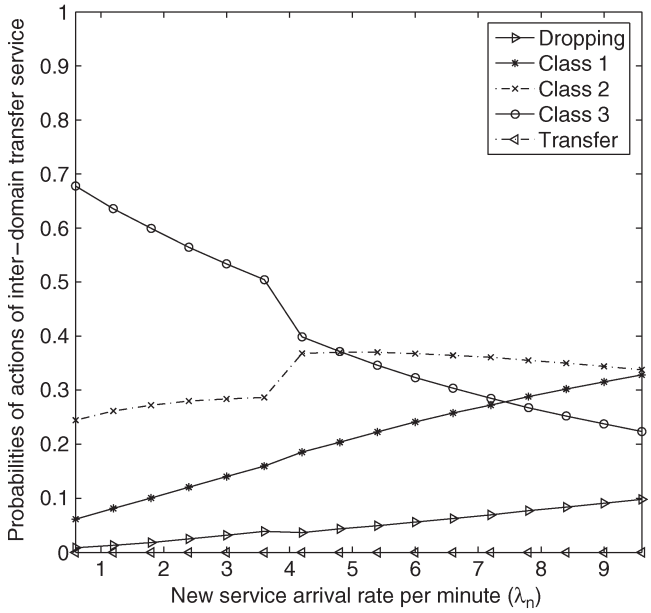

Fig. 4. Action probabilities of interdomain transfer service under various arrival rates of new services ($\lambda_t = 2.4, \mu_1 = 6.6, K = 10$).

The action probabilities for interdomain transfer services under different arrival rates are shown in Fig. 4. When the new request arrival rate is low, it is more likely that a request will be admitted and allocated with three VMs. When the arrival rates increase, the resource allocation decision becomes

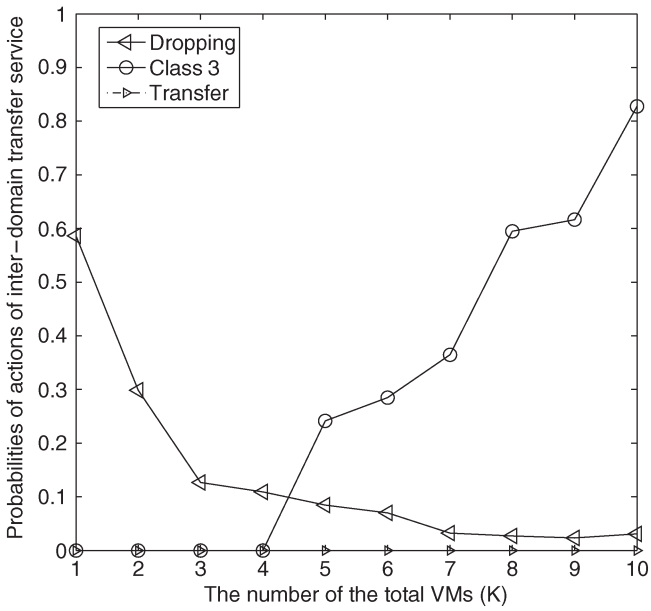

Fig. 5. Action probabilities of interdomain transfer service under various total computing resources ($\lambda_n = 7.2, \lambda_t = 2.4, \mu_1 = 6.6$).

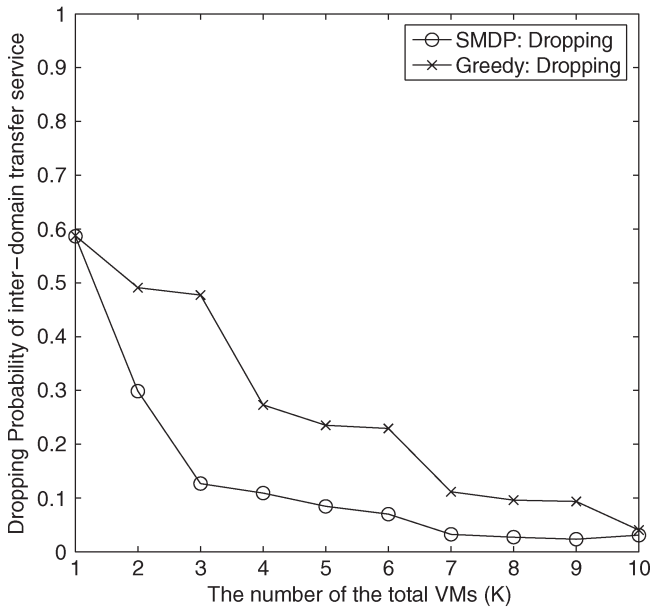

Fig. 7. Dropping probability of interdomain transfer service under various VMs ($\lambda_n = 7.2, \lambda_t = 2.4, \mu_1 = 6.6$).

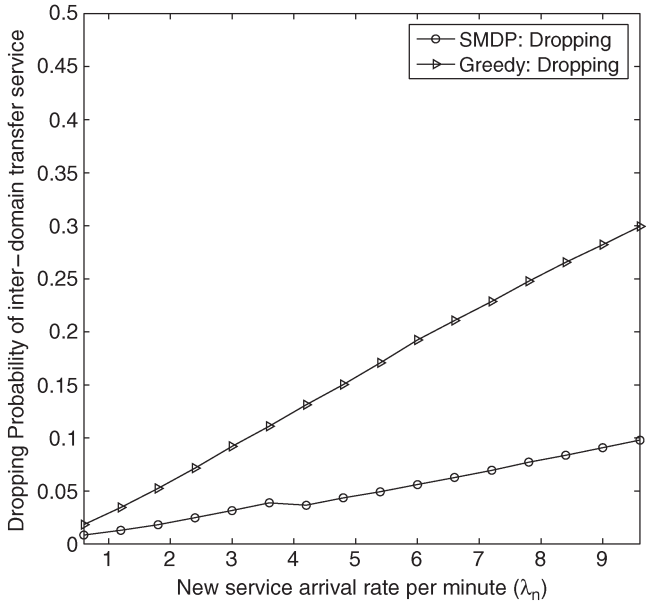

Fig. 6. Dropping probability of interdomain transfer service of adjacent domain under various arrival rates of new service ($\lambda_t = 2.4, \mu_1 = 6.6, K = 10$).

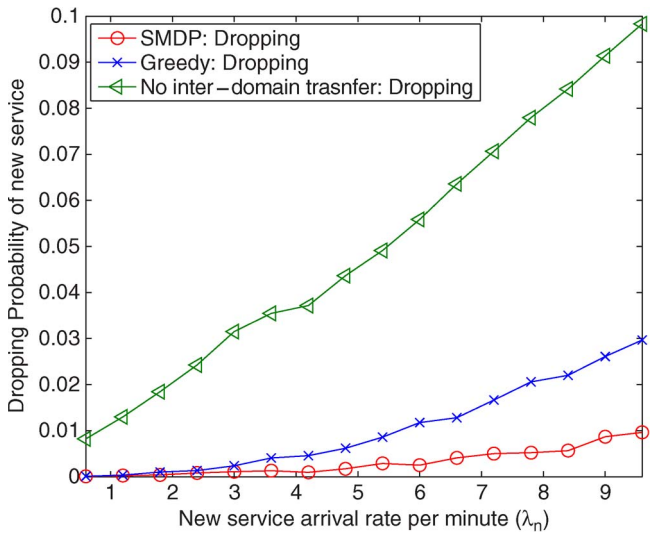

Fig. 8. Dropping probability of new service under various arrival rates ($\lambda_t = 2.4, \mu_1 = 6.6, K = 10$).

conservative, and the probability of allocating two and one VMs increases accordingly. The action probability under different numbers of VMs is compared in Fig. 5. With more available computing resources, i.e., VMs, in a cloud domain, the dropping probability keeps at a very low level. We also compare the dropping probability of the proposed SMDP-based model with the greedy algorithm in Figs. 6–8. In the greedy algorithm, the cloud system always allocates the maximum number of VMs to the requesting service when the system can afford to achieve the highest system reward at the decision epoch. It can be seen that the dropping probability increases with the arrival rate of new services in general and decreases when the total number of VMs in a cloud service domain increases. Our proposed SMDP-based model achieves much lower dropping probability than the greedy algorithm because our algorithm considers not only the current system gain but the expected long-term system rewards as well.

We further compare the expected system rewards of the SMDP-based model and the greedy scheme. The expected system reward of interdomain transfer services under different service arrival rates is compared in Fig. 9. When the arrived service requests exceed the system capacity, more requests will be rejected, and as a result, the expected system rewards decrease. By increasing the number of VMs in a cloud domain, more requests can be admitted in the cloud, and thus, a high system reward can be achieved, as shown in Fig. 10. The reward of new services under different service rates is shown in Fig. 11. When the new service rate is very low, almost all

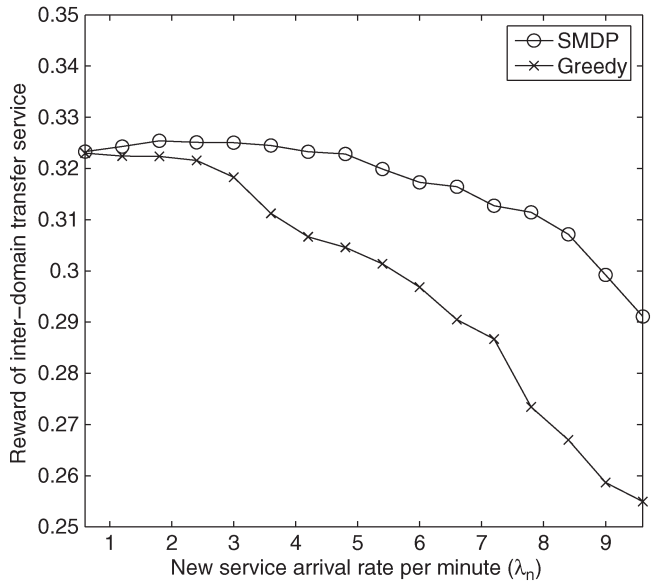

Fig. 9. Reward of interdomain transfer service under various arrival rates of new service ($\lambda_t = 2.4, \mu_1 = 6.6, K = 10$).

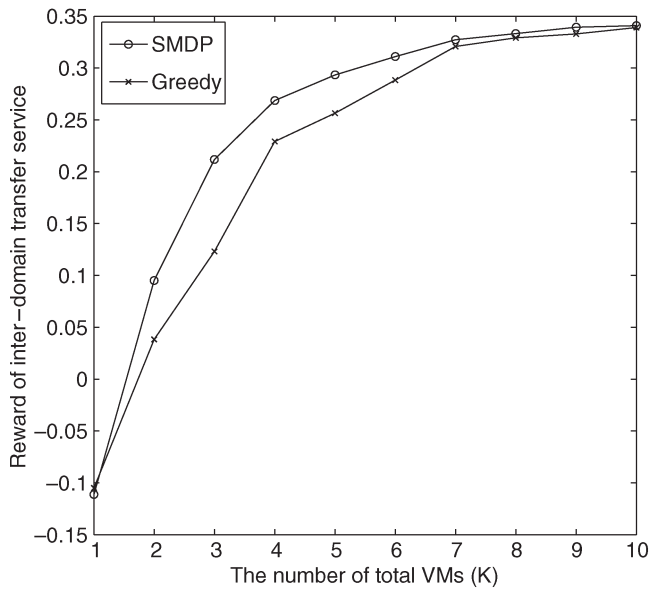

Fig. 10. Reward of interdomain transfer service under different number of VMs ($\lambda_n = 7.2, \lambda_t = 2.4, \mu_1 = 6.6$).

services can be admitted, and the system rewards increase with the rate. However, when the computing resource of a cloud domain is used up, some requests are rejected that degrades the system rewards. Thus, the system reward is a concave function of the service rates. From all figures, it can be seen that our proposed scheme significantly outperforms the greedy scheme. The reason is that, for the greedy scheme, a larger number of VMs are allocated when a service request arrives, and thus, it takes the risk of rejecting the next service request when the available VMs are not sufficient. The proposed SMDP-based interdomain resource allocation model is relatively conservative for decision making by considering both the instant lump sum income and the system expenses.

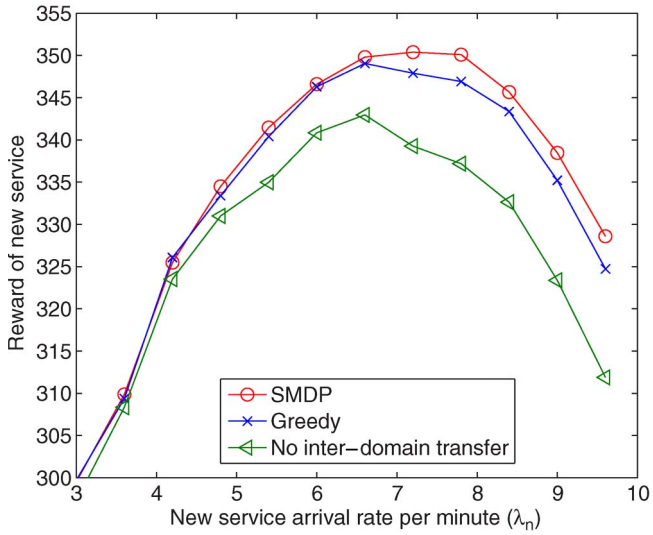

Fig. 11. Reward of new service under various arrival rates of new service ($\lambda_t = 2.4, \mu_1 = 6.6, K = 10$).

VII. Conclusion and Future Work

In this paper, we have developed an SMDP-based computing model for interdomain services in a cloud computing system considering both the system gain, the expenses of computing resources, and the communication costs. The optimal decision is made such that the overall system rewards are maximized.

In our future work, we will analyze the optimal system resources toward the maximal system rewards under a given dropping probability constraint for a large-scale cloud system.

References

[1] M. Armbrust, A. Fox, R. Griffith, A. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, and M. Zaharia, "Above the clouds: A Berkeley View of cloud computing," Dept. Elect. Eng. Comput. Sci., Univ. California, Berkeley, CA, Tech. Rep. UCB/EECS-2009-28, 2009.

[2] D. Huang, X. Zhang, M. Kang, and J. Luo, "Mobicloud: A secure mobile cloud framework for pervasive mobile computing and communication," in *Proc. 5th IEEE Int. Symp. Service-Oriented Syst. Eng.*, 2010, pp. 27–34.

[3] E. Cuervo, A. Balasubramanian, D.-K. Cho, A. Wolman, S. Saroiu, R. Chandra, and P. Bahl, "MAUI: Making smartphones last longer with code offload," in *Proc. ACM MobiSys*, 2010, pp. 49–62.

[4] M. Satyanarayanan, P. Bahl, R. Caceres, and N. Davies, "The case for VM-based cloudlets in mobile computing," *IEEE Pervasive Comput.*, vol. 8, no. 4, pp. 14–23, Oct.–Dec. 2009.

[5] B. Chun and P. Maniatis, "Augmented smartphone applications through clone cloud execution," in *Proc. USENIX HotOS XII*, 2009, p. 8.

[6] R. Jain, "Quality of experience," *IEEE Multimedia*, vol. 11, no. 1, pp. 95–96, Jan.–Mar. 2004.

[7] Secure Networking and Computing (SNACT) Research Group, Mobicloud. [Online]. Available: http://mobicloud.asu.edu/

[8] M. Puterman, *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. New York: Wiley, 2005.

[9] S. M. Ross, *Introduction to Probability Models,* 9th ed. New York: Elsevier, 2007.

[10] C. E. L. Thomas, H. Cormen, R. L. Rivest, and C. Stein, *Introduction to Algorithms,* 3rd ed. Cambridge, MA: MIT Press, 2009.

[11] X. H. Li, H. Zhang, and Y. F. Zhang, "Deploying mobile computation in cloud service," in *Proc. 1st Int. Conf. CloudCom*, 2009, pp. 301–311.

[12] X. Zhang, J. Schiffman, S. Gibbs, A. Kunjithapatham, and S. Jeong, "Securing elastic applications on mobile devices for cloud computing," in *Proc. ACM Workshop Cloud Comput. Security*, 2009, pp. 127–134.

[13] J. Oberheide, K. Veeraraghavan, E. Cooke, J. Flinn, and F. Jahanian, "Virtualized in-cloud security services for mobile devices," in *Proc. 1st Workshop Virtualization Mobile Comput.*, 2008, pp. 31–35.

[14] S. Goyal and J. Carter, "A lightweight secure cyber foraging infrastructure for resource-constrained devices," in *Proc. 6th IEEE Workshop Mobile Comput. Syst. Appl.* , 2004, pp. 186–195.

[15] Q. Wang, C. Wang, J. Li, K. Ren, and W. Lou, "Enabling public verifiability and data dynamics for storage security in cloud computing," in *Proc. ESORICS*, Saint Malo, France, Sep 2009, pp. 355–370.
[16] C. Wang, K. Ren, W. Lou, and J. Li, "Towards publicly auditably secure data storage services," *IEEE Netw. Mag.*, vol. 24, no. 4, pp. 19–24, Jul./Aug. 2010.
[17] L. X. Cai, L. Cai, X. Shen, and J. W. Mark, "Resource management and QoS provisioning for IPTV over mmWave-based WPANs with directional antenna," *ACM Mobile Netw. Appl.*, vol. 14, no. 2, pp. 210–219, Apr. 2009.
[18] H. T. Cheng and W. Zhuang, "Novel packet-level resource allocation with effective QoS provisioning for wireless mesh networks," *IEEE Trans. Wireless Commun.*, vol. 8, no. 2, pp. 694–700, Feb 2009.
[19] L. X. Cai, X. Shen, and J. W. Mark, "Efficient MAC protocol for ultrawideband networks," *IEEE Commun. Mag.*, vol. 47, no. 6, pp. 179–185, Jun. 2009.
[20] G. Naddafzadeh-Shirazi, P.-Y. Kong, and C.-K. Tham, "Distributed reinforcement learning frameworks for cooperative retransmission in wireless networks," *IEEE Trans. Veh. Technol.*, vol. 59, no. 8, pp. 4157–4162, Oct. 2010.
[21] H.-P. Shiang and M. van der Schaar, "Online learning in autonomic multihop wireless networks for transmitting mission-critical applications," *IEEE J. Sel. Areas Commun.*, vol. 28, no. 5, pp. 728–741, Jun. 2010.
[22] A. Alshamrani, L. Xie, and X. Shen, "Adaptive admission control and channel allocation policy in cooperative ad hoc opportunistic spectrum networks," *IEEE Trans. Veh. Technol.*, vol. 59, no. 4, pp. 1618–1629, May 2010.
[23] C. Leong, W. Zhuang, Y. Cheng, and L. Wang, "Optimal resource allocation and adaptive call admission control for voice/data integrated cellular networks," *IEEE Trans. Veh. Technol.*, vol. 55, no. 2, pp. 654–669, Mar. 2006.
[24] X. Shen, W. Zhuang, H. Jiang, and J. Cai, "Medium access control in ultrawideband wireless networks," *IEEE Trans. Veh. Technol.*, vol. 54, no. 5, pp. 1663–1677, Sep. 2005. (Invited Paper).
[25] H. Liang, D. Huang, and D. Peng, "On economic mobile cloud computing model," in *Proc. Int. Workshop Mobile Comput. Clouds (MobiCloud in conjunction with MobiCASE)*, 2010, pp. 1–12.
[26] HP, Integrity Non-Stop Computing, 2011. [Online]. Available: http://h20223.www2.hp.com/nonstopcomputing/cache/76385-0-0-0-121.html
[27] Citrix, Xen Server High Availability, 2011. [Online]. Available: http://support.citrix.com/servlet/KbServlet/download/21018-102-479340/HA-de ep-2.pdf
[28] VMWare Inc., Vmware High Availability, 2011. [Online]. Available: http://www.vmware.com/products/high-availability/overview.html
[29] G. Wei, A. V. Vasilakos, Y. Zheng, and N. Xiong, "A game-theoretic method of fair resource allocation for cloud computing services," *J. Supercomput.*, vol. 54, no. 2, pp. 252–269, 2009.
[30] S. O. H. Mine and M. L. Puterman, *Markovian Decision Process*. Amsterdam, The Netherlands: Elsevier, 1970.

Hongbin Liang (S'09) received the B.Sc. degree in communication engineering from Beijing University of Post and Telecommunication, Beijing, China, in 1995 and the M.Sc. degree in electrical engineering from Southwest Jiaotong University, Chengdu, China, in 2001. He is currently working toward the Ph.D. degree with Southwest Jiaotong University.

From 2001 to 2009, he was a Software Engineer with the Motorola R&D Center of China, where he focused on system requirement analysis and Third-Generation Partnership Project protocol analysis. From 2009 to 2011, he was a visiting Ph.D. student with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON, Canada. His current research interests focus on resource allocation, quality-of-service, security and efficiency in cloud computing, and wireless sensor networks.

Lin X. Cai (S'09–M'11) received the M.A.Sc. and Ph.D. degrees in electrical and computer engineering from the University of Waterloo, ON, Canada, in 2005 and 2009, respectively

She is currently a Postdoctoral Research Fellow with Princeton University, Princeton, NJ. Her research interests include green communication and networking, resource management for broadband multimedia networks, and cross-layer optimization and quality-of-service provisioning.

Dijiang Huang (M'00–SM'11) received the B.S. degree from Beijing University of Posts and Telecommunications, Beijing, China, in 1995 and the M.S. and Ph.D. degrees from the University of Missouri-Kansas City, in 2001 and 2004, respectively.

He is currently an Associate Professor with the School of Computing Informatics and Decision System Engineering, Arizona State University, Tempe. He is an Associate Editor of the *Journal of Network and System Management*. His current research interests are computer networking, security, and privacy.

Dr. Huang is an Editor of the IEEE COMMUNICATIONS SURVEYS AND TUTORIALS. He has served as an organizer for many international conferences and workshops. His research is supported by the National Science Foundation, the Office of Naval Research (ONR), the Air Force Research Laboratory, HP, and the Consortium of Embedded System. He has received the ONR Young Investigator Award and the HP Innovation Research Program Award.

Xuemin (Sherman) Shen (M'97–SM'02–F09) received the B.Sc. degree in electrical engineering from Dalian Maritime University, Dalian, China, in 1982 and the M.Sc. and Ph.D. degrees in electrical engineering from Rutgers University, Camden, NJ, in 1987 and 1990, respectively.

He is a Professor and a University Research Chair with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON, Canada. He was the Associate Chair for Graduate Studies from 2004 to 2008. He is a coauthor/editor of six books and has published more than 600 papers and book chapters in wireless communications and networks, control, and filtering. His research focuses on resource management in interconnected wireless/wired networks, wireless network security, wireless body area networks, vehicular ad hoc, and sensor networks.

Dr. Shen is an Engineering Institute of Canada Fellow and a Distinguished Lecturer of the IEEE Vehicular Technology Society and of the Communications Society. He served as the Technical Program Committee Chair for the 2010 IEEE Vehicular Technology Conference (VTC'10) Fall, the Symposia Chair for the 2010 IEEE International Communications Conference (ICC'10), the Tutorial Chair for IEEE VTC'11 Spring and IEEE ICC'08, the Technical Program Committee Chair for IEEE Globecom'07, the General Co-Chair for Chinacom'07 and QShine'06, the Chair for IEEE Communications Society Technical Committee on Wireless Communications, and Peer-to-Peer Communications and Networking. He also serves/served as the Editor-in-Chief for IEEE NETWORK, *Peer-to-Peer Networking and Application*, and *IET Communications*; a Founding Area Editor for IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS; an Associate Editor for IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, *Computer Networks*, *ACM/Wireless Networks*, etc.; and the Guest Editor for the IEEE JOURNAL ON SELECTED AREAS ON COMMUNICATIONS, IEEE WIRELESS COMMUNICATIONS, IEEE COMMUNICATIONS MAGAZINE, *ACM Mobile Networks and Applications*, etc. He received the Excellent Graduate Supervision Award in 2006 and the Outstanding Performance Award in 2004, 2007, and 2010 from the University of Waterloo, the Premier's Research Excellence Award in 2003 from the Province of Ontario, Canada, and the Distinguished Performance Award in 2002 and 2007 from the Faculty of Engineering, University of Waterloo. He is a Registered Professional Engineer in Ontario.

Daiyuan Peng (M'05) received the M.S. degree in applied mathematics and the Ph.D. degree in mobile communications from Southwest Jiaotong University, Chengdu, China, in 1987 and 2005, respectively.

He is currently a Professor with Southwest Jiaotong University. His research interests include sequence analysis and design, algebraic coding theory, cryptography, and information security.