A Markov Decision Process-based Service Migration Procedure for Follow Me Cloud

Adlen Ksentini‡, Tarik Taleb*, Min Chen§
‡IRISA/University of Rennes 1
Email: adlen.ksentini@irisa.fr
*NEC Europe, Heidelberg, Germany
Email: talebtarik@ieee.org
§Department of Computer Science and Technology, Huazhong University of Science and Technology, China.
Email: minchen2012@hust.edu.cn

***Abstract*—The Follow-Me Cloud (FMC) concept enables service mobility across federated data centers (DCs). Following the mobility of a mobile user, the service located in a given DC is migrated each time an optimal DC is detected. The detailed criterion for optimality is defined by operator policy, but it may be typically derived from geographical proximity or load. Service migration may be an expensive operation given the incurred cost in terms of signaling messages and data transferred between DCs. Decision on service migration defines therefore a tradeoff between cost and user perceived quality. In this paper, we address this tradeoff by modeling the service migration procedure using a Markov Decision Process (MDP). The aim is to formulate a decision policy that determines whether to migrate a service or not when the concerned User Equipment (UE) is at a certain distance from the source DC. We numerically formulate the decision policies and compare the proposed approach against the baseline counterpart.**

I. INTRODUCTION

To cope with the ever-growing mobile traffic, mobile operators have been looking into the decentralization of their core network, along with devising traffic offload-based solutions [1][2]. On the other hand, the fast growing business of clouding computing is pushing for the deployment of regional Data Centers (DCs) [3][4]. Internet Service Providers (ISPs) are also pushing micro DCs closer to customers, enabling third parties to rent them out [11]. Connecting these geographically distributed micro DCs as well as macro DCs, together into a common resource pool, to deliver a variety of cloud services forms the so-called federated cloud. The distribution of cloud computing resources over different locations in the network is beneficial for different reasons such as increasing availability, reducing bandwidth cost, and reducing latency by locating resources nearby users.

Using decentralized mobile networks and federated cloud, mobile cloud services are best provisioned if users are receiving their services from optimal data centers via optimal data- and mobility-anchor gateways (e.g., Packet Data Network Gateway (P-GWs) and Serving Gateways (S-GWs) in the context of the Evolved Packet System (EPS). Only then, an optimal end-to-end connectivity can be ensured. It shall be noted that the detailed criteria for optimality may be defined by operator policy, but they may typically be derived from geographical proximity (to the user location), load, service type, hardware type of DCs, or a combination of these. The Follow Me Cloud (FMC[1]) concept, detailed in [5], describes how this can be achieved during the entire movement of a user. Indeed, FMC enables service mobility across federated DC. Following the mobility of a mobile user, the service located in a given DC is migrated each time an optimal DC is available. To achieve this goal, a smooth migration of the service is needed. That is, with no service disruption. However, it is worth noting that there are technical issues to consider when migrating services (typically Virtual Machines - VMs) between two DCs. These issues pertain to the time needed to transfer a VM between DCs, which can disturb the service continuity. This time depends on:

- the time required for converting a VM, particularly if DCs are not using the same hypervisor.
- the time required for transferring the service (VM) over the network.

The latter intuitively depends on the objects size, the connection speed and the Round Trip Time (RTT) between the DCs. RTT is of high importance as VMs are usually transferred using a FTP/TCP like application; whose performance largely depends on RTT. To fix this issue, solutions such as File Data Transfer (FDT) [6] can be used. It is also worth noting that the service migration decision relies on several attributes/criteria that depend on users expectation on the service (QoS/QoE, cost) and network/cloud provider policies, e.g., considering the service type [7], content size, urgency of the service/task, and/or user class.

Given the above, the migration of a service may become an expensive operation. Decision on service migration defines therefore a tradeoff between cost and user perceived quality. In this paper, we address this tradeoff by modeling the service migration procedure using a Markov Decision Process (MDP). The aim is to formulate a decision policy that determines whether to migrate a service or not when the concerned User Equipment (UE) is at a certain distance from the source DC.

The remainder of this paper is structured as follows. Section II gives an overview of the FMC concept. Section III

---

[1] Whilst FMC is widely used to stand for Fixed Mobile Convergence, this abbreviation stands for Follow-Me Cloud throughout this paper.

978-1-4799-2003-7/14/$31.00 ©2014 IEEE

introduces the envisioned FMC service migration procedure based on MDP. The performance evaluation of the proposed approach is conducted in Section IV. Results are presented and discussed in the same. The paper concludes in Section V.

II. FOLLOW ME CLOUD CONCEPT

A. General concept

A typical network architecture that enables the FMC concept is depicted in Fig. 1. The figure illustrates two main components, namely FMC controller and DC/GW mapping entity, that can be two independent architecture components or functional entities collocated with existing nodes, or can run as a software on any DC of the underlying cloud. In this figure, both the cloud network and the mobile operator network are distributed. We also consider that DCs are mapped to a set of PDN-GWs (i.e., data anchor points in EPS) based on some metric, e.g., location or hop count. This mapping may be static or dynamic. In case of the latter, it could be that the topology information is being exchanged between FMC service provider and the Mobile Network Operator (MNO). Alternatively, an MNO entity/function could be in charge of updating the FMC service provider with such information either in a reactive or a proactive manner. Additionally, we assume that an FMC controller entity exists for managing distributed DC instances. Alternatively, distributed DCs coordinate among themselves in a Self-Organizing Network (SON) manner. In the envisioned FMC service, similar in spirit to Content Centric Networking (CCN), content served by the FMC service has some predefined hierarchy; e.g., “content ID = FMC-Service/ApplicationName.DataName.Characteristics”. For example, in case of Titanic movie, it could be that the “content ID = Video.Titanic.30min”; this means that this content is a video content, from Titanic movie and the frames to be played back are those from the 30^{th} minute since the beginning of the movie.

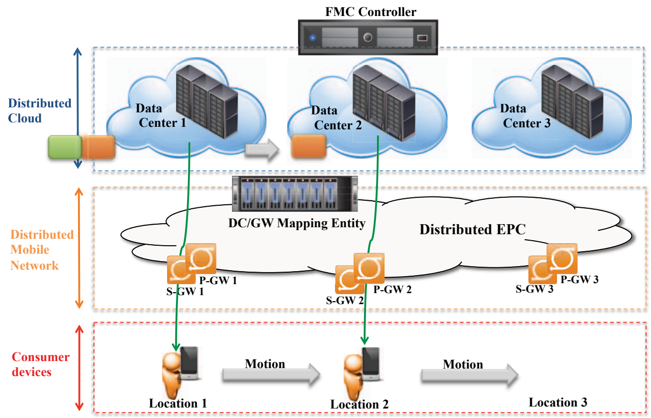


Fig. 1. Interworked federated cloud/distributed mobile network architecture.

To replace IP addressing by service/data identification, a specific application logic/plugin is installed at the UE and the DC servers. Indeed, requests from UEs for an application or a service available at the cloud are mapped to a unique session/service identifier. In other words, any IP session between a UE and a cloud server is identified as follows:

Session/Service ID= Function(UE_ID ; Content_ID)

This session/service ID is generated by the end-host (e.g., UE) that issues the service request and is communicated to the receiving end-host, which is the cloud server.

It shall be noted that the above proposed structure of the session/service identification ensures that all sessions used by the same UE or all sessions used by all UEs belonging to any mobile network will be uniquely identified and that there shall be no conflict in the session/service ID. Indeed, the usage of the UE ID (which is unique within and across different mobile networks) in the session/service ID serves to avoid any conflict in session/service ID among UEs, whereas the usage of content ID in the session/service ID helps in differentiating sessions received by the same UE.

For more details on the performance of the FMC concept, reader can refer to [8], where FMC was modeled by Markov chains.

B. Service Migration

The possible need for a FMC service migration can be intuitively noticed when a UE changes its data anchor gateway (i.e., P-GW relocation), i.e., changes its IP address. Change of the IP address of the UE can be certainly noticed by the corresponding DC. A preliminary decision has to be first taken by UE and/or current DC on whether a service migration is worthwhile or not. This decision may be based on the service type (e.g., an ongoing video service with strict Quality of Service (QoS) requirements may be migrated) [16], content size (e.g., in case a user was watching a movie and the movie is about to finish at the time of the P-GW relocation, the UE may decide, at the FMC application layer, not to initiate the service migration), task type of the service (e.g., in case of Machine Type Communications, session of emergency warning services, delay-sensitive measurement reporting services have to be always migrated to the nearest DC), and/or user class. It is worth noting that the service migration decision (migrate or not) relies on several attributes/criteria (could be conflicting) that depend on users expectation on the service (QoS/QoE, cost) and network/cloud provider policies (at each P-GW relocation, load balancing, maximize using the DC resources).

Once it is deemed appropriate, by either UE or current DC, to migrate the service, the FMC plugin available at the DC may request the FMC controller to select the optimal DC with the right service and right content to serve the UE in its new location, and to initiate the service migration. As a service may consist of multiple cooperating sessions and pieces, the decision has to be made indicating whether the service has to be fully or partially migrated, and that is while considering the service migration cost; e.g., cost associated with the initiation of a new virtual machine at the target DC, cost (if any) associated with the release of resources at the source DC, and cost associated with the bandwidth consumption due to traffic to be exchanged between the DCs and also the FMC controller. An estimate of the cost/overhead to be possibly incurred shall be compared against benefits to the cloud in terms of traffic distribution and also to end users in terms of Quality of Experience (QoE). It shall be noted that there are different forms (e.g., state, data, images, etc), different technologies (e.g. VMware), and different approaches (e.g., Software as a Service SaaS, Platform as a Service PaaS, or Infrastructure as a Service IaaS) for service migration. The latter decides the former.

In the following section, we will demonstrate how the service migration procedure can be modeled through a Markov Decision Process. By using MDP, it is possible to derive the optimal policy that can be employed to decide to migrate or not a service.

III. A MDP MODEL FOR THE FMC SERVICE MIGRATION PROCEDURE

As stated earlier, for an efficient service migration, there is a tradeoff to achieve between reducing cost and maintaining good user QoE. To capture this tradeoff, we model the service migration decision as a MDP Process. The envisioned model decides on whether a service shall be migrated to an optimal DC or not, when the concerned UE is at distance d from the current DC. The MDP decision procedure is assumed to be implemented in the FMC controller. To formulate the service migration decision policy, we define a Continuous Time Markov Decision Process (CTMDP) that associates to each state an action, corresponding transition probabilities, and rewards.

Let s_t be the process describing the evolution of the system state and S denote the state space. In order to derive the state space, we assume that the 3GPP network is divided into hexagonal cells, whereby each cell belongs to a Tracking Area (TA) and each TA belongs to a Service Area (SA)[2], which is served by one S-GW connected to a P-GW. We assume that one DC is mapped to one SA. For the sake of simplicity and without losing generality, we consider a one dimension (1D) mobility model. In other words, a mobile user has only two possible destinations, namely a new SA with a probability p, or returning back to a visited SA with a probability (1-p). Higher values of p indicate that the user is moving far from the current DC. The residence time of a user in each SA follows an exponential distribution with mean $1/\mu$. Hence, the state space S is defined as $S = \{0, 1, \ldots, thr\}$. Here, thr represents the maximum distance (in terms of visited SAs) from where the service must be migrated to the optimal DC.

We denote by A= (a_1, a_2) the vector describing the actions available to the FMC controller at each epochs (i.e., when a UE performs handoff and enters into another SA). Action a_2 is used if the service is migrated to an optimal DC, while action a_1 is used if the UE is still served by the same DC. Note that, depending on the current state, the set of available actions is different. Fig. 2 shows the CTMDP procedure associated to the FMC service migration procedure. The FMC controller observes the current state s of the network and associates a set of possible actions A_s to it, taken upon arrival to it from the previous state. For a given action a, an instantaneous reward $r(s, s', a)$ is associated to a transition from state s to another state s'. The corresponding formal representation of the CTMDP process is as follows:

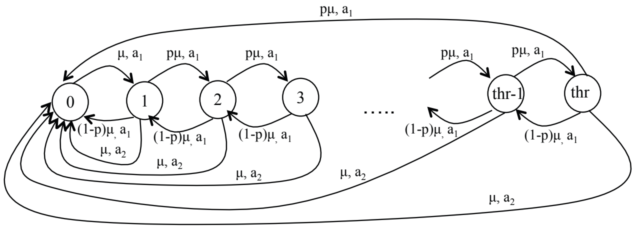


Fig. 2. CTMDP of the service migration procedure.

$$(S, A, (A_s, s \in S), q(s'|s, a), r(s, s', a))$$

For particular states, the set of possible actions A reduces to a subset A_s. A policy P associates an action $a(s|P)$ to a state s. Let Q be the transition matrix, with $q(s|s')$ denoting the transition rate between states s and s' in S, which, in the FMC scenario, represents a UE moving from one SA to another SA. By construction, we define a policy as a function of the actual state. The decision of migrating a service or not is taken by observing only the actual state. Since this process is Markovian (i.e., the sojourn time in a SA is following an exponential distribution), the controlled process is then Markovian. To resolve the MDP process, we use an equivalent Decision Time Markov Decision Process (DTMDP) for the mentioned CTMDP process with a finite state space S. It is worth noting that the state space is finite. We argue this by the fact that after a certain distance (thr) from the current DC, the service is automatically migrated to the optimal DC. For each $s \in S$, we denote by A_s the finite set of allowed actions in that state. This DTMDP process can be found by uniformization and discretization of the initial process as follows:

- When all the transition rates in matrix Q are bounded, the sojourn times in all states are exponential with bounded parameters $tr(s|s, a)$. Therefore, a $sup_{(s \in S, a \in A_s)} tr(s|s, a)$ exists and there is a constant value c such as:
  $$sup_{(s \in S, a \in A_s)} [1 - p(s, a)] tr(s|s, a) \leq c < \infty$$
  where $p(s|s, a)$ denotes the probabilities of staying in the same state after the next event. We can now define an equivalent uniformized process with state-independent exponential sojourn times with parameter c, and transition probabilities:
  $$p(s'|s, a) = \begin{cases} 1 - \frac{([1 - p(s|s)] tr(s|s, a))}{c} & s = s' \\ \frac{p(s'|s) tr(s'|s, a)}{c} & s \neq s' \end{cases}$$

By posing $c = \mu$, the transition probabilities of the DTMDP procedure are defined as follows:

---

[2] Generally speaking, a TA can be serviced by multiple SAs. However, for the sake of simplicity, we assume that each TA is serviced by one single SA.

$$p(j|s,a) = \begin{cases} 1 & j=0, s\neq 0, s\neq thr, a=a_1 \\ & or\ j=1, s=0, a=a_2 \\ p & j=s+1, s\neq 0, s\neq thr, a=a_1 \\ & or\ j=0, s=thr, a=a_1 \\ 1-p & j=s-1, s\neq 0, a=a_1 \\ 0 & Otherwise \end{cases}$$

It is important to note that when the system is in state $s = thr$, the only available action is a_1. If the UE moves to another SA where the distance exceeds the threshold thr, the service migration is automatically triggered.

In the remainder of this section, we use the DTMDP version. For $t \in N$, let s_t, a_t and r_t denote state, action and reward at time t of the DTMDP procedure, respectively. Let $P^a_{(s,s')} = p\left[s_{(t+1)} = s' | s_t = s, s_{(t+1)} = s', a_t = a\right]$ denotes the transition probabilities and $R^a_{(s,s')} = E\left[r_{(t+1)} | s_t = s, s_{(t+1)} = s', a_t = a\right]$ denotes the expected reward associated to the transitions. A policy π is mapping between a state and an action, and can be denoted as $a_t = \pi(s_t)$, where $t \in N$. Accordingly, a policy $\pi = (\theta_1, \theta_2, \theta_3, \ldots, \theta_N)$ is a sequence of decision rules to be used at all decision epochs. In this work, we restrict ourselves to only deterministic policies, as they are simple to implement [9]. When a UE handoffs from a particular SA to another SA, the FMC controller has to decide to migrate the service using action a_2 or not migrating it using action a_1. For each transition a reward is obtained. This reward is a function of the cost of migrating a service (zero in case of no migration) and the quality obtained from the new state. The cost of migrating a service is defined as follows:

$$g(a) = \begin{cases} C_m & a = a_1 \\ 0 & a = a_2 \end{cases}$$

where C_m denotes the cost of migrating all the service or a part of it. Therefore the reward function is derived as follows:

$$r(s,s,a) = q(s) - g(a)$$

where $q(s)$ quantifies the quality perceived by a UE connected to the source DC from a distance s in terms of number of visited SAs. Note that the quality is inversely proportional to the distance from the source DC. For instance, $q(0)$ is the maximum quality that a UE can obtain as the UE connects then to the optimal DC. Generally, the quality function can expressed in the form $q(s) = q(0) - k$, where k denotes a predetermined factor. Given a discount factor $\gamma \in [0,1]$ and an initial state s, we denote the total discount reward policy $\pi = (\theta_1, \theta_2, \theta_3, \ldots, \theta_N)$ as follows:

$$v^\pi_\gamma = \lim_{N\to\infty} E^\pi_\gamma\{\textstyle\sum_{t=1}^{N} \gamma^{t-1} r_t\} = E^\pi_\gamma\{\textstyle\sum_{t=1}^{\infty} \gamma^{t-1} r_t\}$$

Given the uniformization of CMTC, $r(s,s',a)$ explicitly depends on the transitions between states. According to [9], the new reward function $r(s,s',a)$ is obtained as follows:

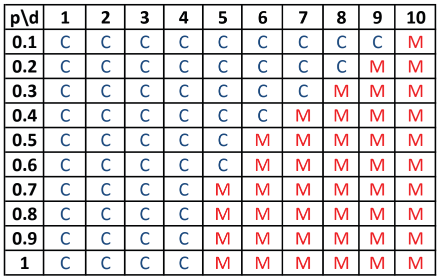
$$r'(s,s',a) = r(s,s',a)\frac{\alpha+\beta(s,s',a)}{\alpha+c}$$

where $\beta(s,s',a)$ is the transition rate between states s and s' when using action a, and α is a predetermined parameter. With the new formulation of the reward function and the uniformization of CMTC, we can use the discounted models as in discrete models to resolve the system [9]. Let $v(s)$ denotes the maximum discounted total reward, given the initial state s. That is, $v(s) = max_{\pi\in\Pi} v^\pi(s)$. From [9], the optimality equations are given by

$$v(s) = max_{\pi\in\Pi}\{r'(s,s'a) + \textstyle\sum_{s'\in S} \gamma P\left[s'|s,a\right] v(s')\}$$

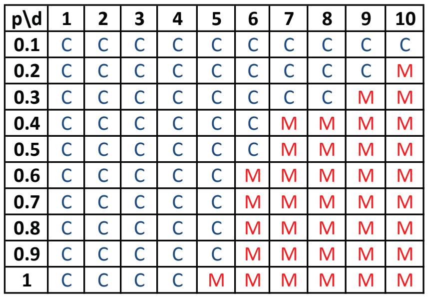
The solutions of the optimality equations correspond to the maximum expected discounted total reward $v(s)$ and the optimal policy $\pi^*(s)$. It is worth mentioning that the optimal policy $\pi^*(s)$ indicates the decision as to which network and which data center the UE is to be attached and to be connected, respectively, given the state s. There are several algorithms that can be used to resolve the optimization problem given by the above optimality equations. Value iteration and policy iteration are two noticeable examples.

IV. NUMERICAL RESULTS



p\d	1	2	3	4	5	6	7	8	9	10
0.1	C	C	C	C	C	C	C	C	C	M
0.2	C	C	C	C	C	C	C	C	M	M
0.3	C	C	C	C	C	C	C	M	M	M
0.4	C	C	C	C	C	C	M	M	M	M
0.5	C	C	C	C	C	M	M	M	M	M
0.6	C	C	C	C	C	M	M	M	M	M
0.7	C	C	C	C	M	M	M	M	M	M
0.8	C	C	C	C	M	M	M	M	M	M
0.9	C	C	C	C	M	M	M	M	M	M
1	C	C	C	C	M	M	M	M	M	M

Fig. 3. The optimal policy construction – $\tau = 0.1$.



p\d	1	2	3	4	5	6	7	8	9	10
0.1	C	C	C	C	C	C	C	C	C	C
0.2	C	C	C	C	C	C	C	C	C	M
0.3	C	C	C	C	C	C	C	C	M	M
0.4	C	C	C	C	C	C	M	M	M	M
0.5	C	C	C	C	C	C	M	M	M	M
0.6	C	C	C	C	C	M	M	M	M	M
0.7	C	C	C	C	C	M	M	M	M	M
0.8	C	C	C	C	C	M	M	M	M	M
0.9	C	C	C	C	C	M	M	M	M	M
1	C	C	C	C	M	M	M	M	M	M

Fig. 4. The optimal policy construction – $\tau = 0.5$.

To derive the service migration policy, we used a Matlab implementation of the value iteration algorithm presented in [10]. We set thr to 10, which means that a service migration is automatically triggered if the UE is at a distance higher than 10 (in term of number of visited SAs) from the source DC. Without any specific purpose in mind, the factor k of the quality function is set to one. We introduce a new metric, denoted by τ, as the ratio between the cost and the maximum quality $q(0)$. Two scenarios are envisioned:

- $\tau = 0.1$, which represents a low cost compared to the quality obtained if a service migration is launched. For instance, this configuration corresponds to the case of only a part of a service is migrated.

- $\tau = 0.5$, which indicates that the cost is not negligible compared to the quality obtained if a service migration is launched.

Figs. 3 and 4 illustrate the optimal policy constructions for the two above-mentioned scenarios. The intersection between i and j represents the action (a=C continuation, a=M migration of the service) to be taken by the FMC controller, where i is the distance from the serving DC and j is the probability p. We remark that the optimal policy construction has a threshold-based form, since from a certain value of the distance, the policy recommends service migration. This distance is inversely proportional to the probability p. For instance, when p=0.8, the proposed policy recommends service migration, if d=6 and d=5, for the first and second scenarios, respectively. This is trivial as high probability means that the UE is moving far from the current DC, while low values indicate that the UE will most likely remain in the vicinity of the current DC or will come back to the service area of the current DC. Furthermore, we remark that the ratio τ has an impact on the optimal policy construction, since there are less service migrations when τ is high. This is intuitive as the incurred cost is not negligible in comparison to the achieved gain when migrating a service of the UE.

In case of random walk (p=0.5), we remark that for both scenarios, the optimal policy recommends service migration when the distance exceeds 5, which represents a good tradeoff between cost and quality. Considering this case of random walk mobility, we compare the proposed scheme against two approaches, one considers no service migration (until reaching the distance thr) and another enforcing service migration for each state. The comparison is in terms of the expected total reward if the action is taken from a state s (distance s) as shown in Figs. 5 and 6. These figures plot the obtained results for the two considered scenarios, $\tau = 0.1$ and $\tau = 0.5$, respectively.

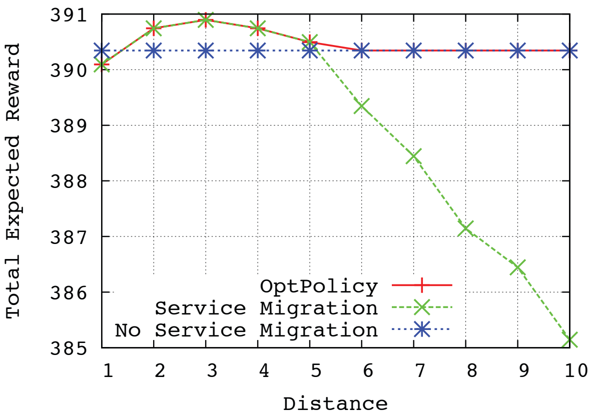


Fig. 5. The expected total reward – $\tau = 0.1$.

From these figures, we clearly observe that the proposed scheme achieves the highest total expected reward for both scenarios. This is attributable to the fact that the proposed scheme ensures a good trade-off between quality gained from a service migration and incurred cost. It always takes the action that maximizes the total expected reward. We also remark that the expected reward is higher in the first scenario, as the cost is lower and hence the reward function is maximized.

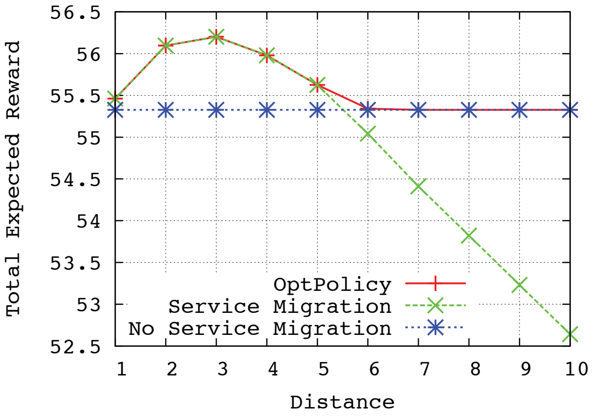


Fig. 6. The expected total reward – $\tau = 0.5$.

V. CONCLUSION

In this paper, we defined a MDP-based model for service migration to achieve the concept of follow me cloud. The proposed model allows the formulation of policies that achieve a good tradeoff between user QoE and cost incurred by a service migration. By implementing this model, the FMC controller can optimally decide if a service needs to be migrated or not. Numerical results show that the proposed service migration decision mechanism always achieves the maximum expected gain (reward) compared to two other policies, namely the policy that triggers service migration each time a UE enters into a new SA and the policy that launches service migration only when the UE is distant by k SAs from the current DC.

REFERENCES

[1] K. Samdanis, T. Taleb, and S. Schmid, “Traffic Offload Enhancements for eUTRAN”, in IEEE Communications Surveys & Tutorials journal, Vol. 11, No. 3, Aug. 2012, pp. 884-896.

[2] T. Taleb, K. Samdanis, and F. Filali, “Towards Supporting Highly Mobile Nodes in Decentralized Mobile Operator Networks”, in Proc. IEEE ICC 2012, Ottawa, Canada, Jun. 2012.

[3] R. Miller, AOL Gets Small with Outdoor Micro Data Centers, Data Center Knowledge, Jul. 2012.

[4] R. Miller, Solar-Powered Micro Data Center at Rutgers, Data Center Knowledge, May 2012.

[5] T. Taleb and A. Ksentini, “Follow Me Cloud: Interworking Distributed Clouds & Distributed Mobile Networks”, to appear in IEEE Network Magazine, Sep. 2013.

[6] File Data Transfer, http://monalisa.cern.ch/FDT/.

[7] T. Braun, A.Mauthe, V. Siris, “Service-Centric Networking Extensions”, in Proc. of ACM Symposium on Applied Computing 2013, Coimbra, Portugal.

[8] T. Taleb and A. Ksentini, ‘An Analytical Model for Follow Me Cloud”, in Proc. of Globecom 2013, Atlanta, USA, Dec. 2013.

[9] M. Puterman, “Markov Decision Process: Discrete stochastic dynamic programming”, Wiley-Interscience 1994.

[10] INRAs matlab MDP toolbox, http://www.inra.fr/mia/MDPtoolbox.

[11] B. Frank, et. al., Pushing “CDN-ISP Collaboration to the Limit”, ACM SIGCOMM Computer Communication Review, 43(3), 2013.