European Wireless 2013, 16 – 18 April, 2013, Guildford, UK ISBN 978-3-8007-3498-6 © VDE VERLAG GMBH • Berlin • Offenbach, Germany

First Iteration Policies for Admission Control in Multiaccess Networks

Diego Pacheco-Paramo, Jorge Martinez-Bauset, Vicent Pla, Elena Bernal-Mor
Universitat Politècnica de València
Camino de Vera s/n, 46022, Valencia, Spain
Email: diepacpa@posgrado.upv.es,{jmartinez,vpla}@dcom.upv.es, elbermo@upvnet.upv.es

***Abstract*—This work explores approximate methods to solve Markov decision processes for large systems through Policy iteration.Two methods, one using an embedded discrete time Markov chain and the other using time scale separation, are defined and compared with the solution obtained using traditional Policy iteration. First step solutions are found and compared for a radio resource management problem with two radio access technologies and two service types. The approaches proposed considerably reduce the computational cost while closely approximate the optimal solution. The solutions are extended by increasing the number of steps of policy iteration and results show that it is possible to reach the performance of the optimal policy when several steps are required reducing the computational cost.**

I. Introduction

Markov decision processes (MDPs) are widely used for control problems where revenues depend on episodic actions. Different solving methods are used, such as dynamic programming or reinforcement learning. In problems where the state space and the transition probabilities are well known, dynamic programming is a reliable method which allows to find the optimal solution for a given cost function. Dynamic programming algorithms such as policy iteration and value iteration are commonly used, where the former is able to find the optimal solution in less iterations than the latter, although some variations of value iteration can be done in order to reduce the necessary number of iterations as shown in [1]. Different approximate methods have been proposed to reduce the computation complexity such as state aggregation [2] or linear approximations [3] applied to MDPs. However, these methods neither produce solutions that are necessarily close to the optimal, nor are efficient as it is stated in [4].

In policy iteration, the optimal policy is found by choosing an initial random policy, evaluating its cost/reward through *policy evaluation* and enhancing it through *policy improvement*. This two-phase iteration is repeated until no further improvement can be done, i.e. the optimal policy is found. Although it is well known that policy iteration converges in a few steps, when the dimensionality (size of the state space) of the Markov process that models the system dynamics is large, the computational cost to obtain the optimal policy is also large.

This work has been supported by the Spanish Government under project TIN2010-21378-C02-02. Diego Pacheco-Paramo was supported by the Spanish Ministry of Economy and Competitiveness under contract BES-2009-013162.

In [5] a first policy iteration solution is proposed, where the policy improvement phase is performed only once to obtain a suboptimal solution. This method drastically reduces the computational cost for large systems. In [6] a first iteration policy solution is proposed for a system where the cost to find a solution for an initial policy is low and it closely resembles the optimal.

In this work we also make use of the first iteration policy and compare its performance with the optimal solution for a radio resource management problem where two types of services and two radio access technologies are available [7]. The main objective of this work is the proposal and evaluation of two different methods to perform policy evaluation to obtain the first iteration policy. The first approach consists in iterating over the embedded discrete time Markov chain, which is found using uniformization [8]. The most important contribution of this work is the second approach. This solution is a variation of the typical time scale decomposition, since it relates the steady state probabilities of the fluid regime with the transitions of the quasi-stationary regime. To the best of our knowledge this dependance has not been studied before in the time scale decomposition context. Results for both approaches are compared with the optimal solution, and a considerable reduction of the computational cost is achieved, while obtaining a good precission.

The paper is organized as follows. First we define the system and the optimization problem. Then, we propose a solution method using the embedded discrete time Markov chain. In the next section we define the time scale separation approach and apply it to the system. Then, we compare both solutions and its use into the policy iteration context with the optimal solutions. Finally, we conclude the work with remarks about these results.

II. System description and optimization problem

Let us define a system that supports voice (streaming) and data (elastic) traffic. We assume that voice (data) call (session) arrivals follow a Poisson process with rate $\lambda_v(\lambda_d)$ and that the service time for voice calls is exponentially distributed with mean $1/\mu_v$. On the other hand, as data sessions generate elastic traffic, their sojourn time will depend on the available resources. The size of the flows generated by the data sessions are exponentially distributed with mean σ (in bits). If BR_d is the data bit rate experienced for a given user, then the

service time will be exponentially distributed with mean $1/\mu_d = \sigma/BR_d$. Clearly, BR_d might depend on the system state as discussed later.

The system state is represented by the 4-tuple $\boldsymbol{s}$=(v_1,v_2,d_1,d_2), where v_1 (d_1) represents the voice (data) calls (sessions) in TDMA, and v_2 (d_2) represents the voice (data) calls (sessions) in WCDMA. Therefore, the 4-tuples must fulfill the capacity conditions given by:

$$v_1 \cdot n_c + d_1 \leq n_c \cdot C. \tag{1}$$

$$v_2 \cdot V + d_2 \cdot D \leq \eta_{ul}, \tag{2}$$

where (1) and (2) refer to capacity conditions for TDMA and WCDMA respectively, n_c is the maximum number of data sessions that can share one channel in TDMA, C is the total number of channels in TDMA, V=$\left(\frac{W/BR_{w,v}}{(E_b/N_0)_v}+1\right)^{-1}$, D=$\left(\frac{W/BR_{w,d}}{(E_b/N_0)_d}+1\right)^{-1}$, W is the chip rate, $BR_{w,x}$ is the bit rate used for transmitting service x in WCDMA, $(E_b/N_0)_x$ is the bit energy to noise density required for service x, and η_{ul} is the uplink cell load factor. Three metrics are of our interest: voice (PB_v) and data (PB_d) blocking probabilities, and the total throughput. The voice (data) blocking probability refers to the probability of being in those states where a new voice (data) call (session) would be blocked. The total throughput takes into account the different contributions of voice and data sessions, and the fact that data sessions can share a channel in TDMA which is reflected in the min $(C - v_1, d_1)$ term of:

$$\begin{aligned} Th = \sum_{\boldsymbol{s}\in S} \Pi(\boldsymbol{s}) \big(& v_1 BR_{t,v} + v_2 BR_{w,v} \\ & + \min(C - v_1, d_1) BR_{t,d} + d_2 BR_{w,d} \big), \end{aligned} \tag{3}$$

where $\Pi(\boldsymbol{s})$ is the steady state probability of being in state $\boldsymbol{s}$ and $BR_{x,y}$ is the bit rate used for transmitting service y (voice or data) in technology x (TDMA or WCDMA).

In the context of Markov decision processes (MDPs) a policy assigns the action a_s to be performed on each possible state $\boldsymbol{s}$ whenever a call arrives to the system. Therefore, the system chooses an option from the action set A defined in Table I according to the type of the arriving call (voice or data) and the current state $\boldsymbol{s}$. Vertical handoff (VH) , which is the ability to change the access technology of an active call, is used in actions 3 and 4. However, some conditions must be fulfilled to perform these vertical handoffs. Action 3 can only be performed when a voice call arrives to the system and there is full capacity on WCDMA, where N is the number of data sessions that have to be switched from WCDMA to TDMA to allow the voice call to use a WCDMA channel. Therefore, action 3 promotes that voice calls are served by WCDMA. On the other hand, action 4 can only be performed when an arriving data call forces channel sharing on TDMA. Therefore, this type of VH is used to reduce data channel sharing.

TABLE I
SET OF ACTIONS A

a_s	*Description*
0	Block call
1	Send call to TDMA
2	Send call to WCDMA
3	VH for N data sessions from WCDMA to TDMA and the voice call is sent to WCDMA.
4	VH for 1 voice call from TDMA to WCDMA and the data call is sent to TDMA.

The optimization problem consists on finding the optimal policy Ψ^* for the blocking function:

$$F_{BP} = BP_v \cdot \alpha + BP_d \cdot (1 - \alpha), \tag{4}$$

which is the weighted sum of the voice (PB_v) and data (PB_d) blocking probabilities through the value α, which is set as 0.5 unless otherwise stated. The cost function associated to the objective function for each feasible state $\boldsymbol{s}$ is

$$c(\boldsymbol{s}) = 1 - (\alpha \cdot F_v(a_s) + (1 - \alpha) \cdot F_d(a_s)), \tag{5}$$

where $F_x(a_s)$ =1 if a_s is 1,2,3 or 4, and 0 otherwise, being x the service.

Following the policy iteration method, the optimal policy Ψ^* is found by choosing an initial random policy Ψ_{ini}, evaluating its relative values through *policy evaluation*, improving the actions through *policy improvement*, and repeating until no further improvement is possible. In this work we are interested in first iteration policies, that is, policy improvement is performed only once. In the following sections two different methods to perform policy evaluation will be defined: the first uses iterations over the embedded discrete Markov chain and the second uses time scale separation.

III. 1ST STEP POLICY ITERATION THROUGH THE EMBEDDED DISCRETE TIME MARKOV CHAIN

In [7] the continuous time Markov chain (CTMC) is constructed according to the arrival and departure rates previously defined and MDPs are solved using policy iteration for different scenarios through the LSQR algorithm [9] , commonly used for large and sparse matrices. However, this method requires a large number of iterations and high internal precision to converge, which greatly increases its computational cost and thus reduces the scope of possible implementations. In this section we propose a method that partially alleviates the high computational cost by using the first step solution of policy iteration and the properties of the embedded discrete time Markov chain (eDTMC). The eDTMC can be obtained by dividing each outgoing transition rate by a factor γ which should be bigger than the largest aggregated outgoing transition rate. Then, an loop must be defined to keep the sum of transition probabilities equal to one. This method is known as uniformization [8]. The main advantage of the eDTMC is that the steady state probability distribution (SSPD) can be easily found iteratively. Once the transition matrix T

has been defined according to the initial policy Ψ_{ini}, the SSPD vector Π can be solved by iterating as follows:

$$\Pi_{ini}^{i+1} = \Pi_{ini}^{i} \cdot T(\Psi_{ini}), \tag{6}$$

where Π_{ini}^{0} is a vector of zeros with a 1 in its first position. Two stopping criteria have been defined: The quadratic error limit δ, $|\Pi_{ini}^{i+1} - \Pi_{ini}^{i}|^2 < \delta$, or the maximum number of iterations is reached, $i = it_{max}$. Having Π_{ini}, the mean cost of the initial policy $\bar{c}(\Psi_{ini})$ is found using:

$$\bar{c}(\Psi_{ini}) = BP_d \cdot (1-\alpha) + BP_v \cdot \alpha. \tag{7}$$

This is needed to iteratively obtain the relative values of the initial policy $\vec{r}(\Psi_{ini})$, through Howard's equation as:

$$\vec{r}^{i+1}(\Psi_{ini}) = \vec{c}(\Psi_{ini}) - \bar{c}(\Psi_{ini})\vec{e} + T(\Psi_{ini})\vec{r}^{i}(\Psi_{ini}), \tag{8}$$

where $\vec{c}(\Psi_{ini})$ is a column vector with the cost of being in each state in S, $\vec{e}$ is a column vector of ones and $\vec{r}^{0}(\Psi_{ini})$ is a zeros vector of proper size. Two stopping criteria have been defined: The quadratic error limit δ', $|\vec{r}^{i+1}(\Psi_{ini}) - \vec{r}^{i}(\Psi_{ini})|^2 < \delta'$, or the maximum number of iterations is reached, $i = it'_{max}$.

Once we have obtained the relative values of the initial policy $\vec{r}(\Psi_{ini})$, it is straightforward to perform *policy improvement* to obtain the actions a_n that define the first step policy Ψ_{fs} using:

$$\Psi_{fs}(a_n) = \arg\min \left\{ c_n(a_n) - \bar{c}(\Psi_{ini}) + \textstyle\sum_m t_{n,m}(a_n) r_m(\Psi_{ini}) \right\}, \tag{9}$$

where $c_n(a_n)$ is the cost associated to being in state n and taking action a, $t_{n,m}(a_n)$ is the probability of going from state n to state m for the Markov chain that uses policy Ψ_{ini} with action a, and $r_m(\Psi_{ini})$ is the relative value of the destination state, m.

It should be noted that this method requires the setting of four parameters that have an important impact in Ψ_{fs}, that is, δ and it_{max} for Π_{ini}, and δ' and it'_{max} for $\vec{r}(\Psi_{ini})$. For very small values of δ and δ' or high values of it_{max} and it'_{max} the computational cost will increase but the obtained values will be closer to the real value, and the improved policy will be better. Therefore, it is necessary to define the values of δ, δ', it_{max} and it'_{max} that can guarantee an acceptable precission. In our experiments, unless otherwise stated, the chosen values were: $\delta = \delta' = 10^{-3}$ and $it_{max} = it'_{max} = 500$.

IV. 1ST STEP POLICY ITERATION THROUGH TIME SCALE SEPARATION

Since the rate of events is higher for data users than for voice users, it is possible to assume that the former can see the latter as being still. By means of this approximation, a SSPD can be defined for each combination of data users $\boldsymbol{d}$=(d_1,d_2), where its state space is conditioned by (1) and (2) according to the combination of voice users (v_1,v_2), and the initial policy Ψ_{ini} chosen. Let us denote these conditional probabilities as $p((d_1,d_2)|(v_1,v_2))$. Unless otherwise stated, in this work Ψ_{ini} is a policy that sends voice calls to WCDMA until it is full, and then sends them to TDMA. On the other hand, data

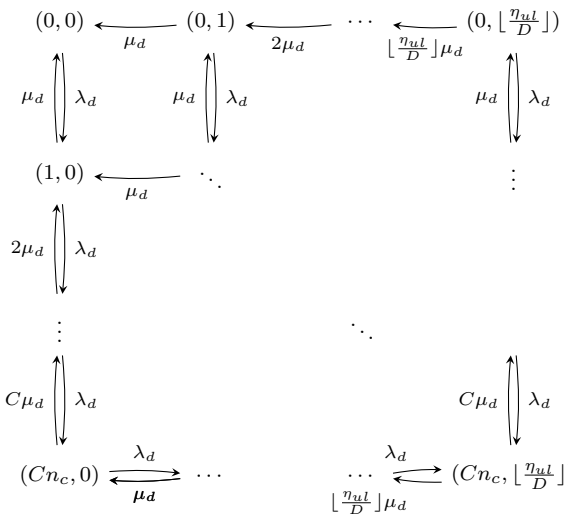

Fig. 1. Continuous-time Markov chain for a fast time scale subsystem.

sessions are sent to TDMA until it is full, and then are sent to WCDMA. Since v_1=$\{0,1,...,C\}$ and v_2=$\{0,1,...,\lfloor \frac{\eta_{ul}}{V} \rfloor\}$, a total of $(C+1) \cdot (\lfloor \frac{\eta_{ul}}{V} \rfloor + 1)$ independent two-dimensional systems must be solved. The CTMC of the fast time scale subsystem (FTSS) conditioned to $v_1 = 0$ and $v_2 = 0$ is shown in Fig 1.

On the other hand, when the time scale separation is sufficiently large, data sessions can achieve a permanent regime between two voice events, and therefore it can be assumed that its behavior is sufficiently represented by its mean. However, this approach does not consider the capacity limitation that data users are imposing over voice users, that is, a transition between two states of the slow time scale subsystem (STSS) can only exist if the conditions in (1) and (2) are fulfilled on both the initial and final states. For example when TDMA is fully occupied by voice calls, and a new voice call arrives to WCDMA, being the initial and final states $v_o = (C, 0)$ and $v_f = (C, 1)$ respectively, the transition can only occur when d_1=0 and $d_2 \leq (\eta_{ul}$ - $V)$ / D, that is, the voice transition is limited by the subset of data sessions (d_1, d_2) that mantain capacity consistency according to (1) and (2) for both $v_o = (C, 0)$ and $v_f = (C, 1)$. In order to model this phenomenon, we use the conditional probabilities $p((d_1,d_2)|(v_1,v_2))$ that were calculated in the previous step. Hence, the transition rate of going from an initial STSS state to a final STSS state is weighted by the sum of probabilities of being on those FTSS states where both the initial and final STSS states are feasible according to (1) and (2). For the initial policy Ψ_{ini}, the weighting probabilities Φ_{v_o,v_f} for going from initial state v_o to final state v_f are:

$$\phi^{W}_{(v_1,v_2),(v_1,v_2+1)} = \sum_{d_1=0}^{n_c(C-v_1)} \sum_{d_2=0}^{\frac{\eta_{ul}-(v_2+1)V}{D}} p((d_1,d_2)|(v_1,v_2)), \tag{10}$$

$$\phi^T_{(v_1,v_2),(v_1+1,v_2)} = \sum_{d_1=0}^{n_c(C-(v_1+1))} \sum_{d_2=0}^{\frac{\eta_{ul}-(v_2)V}{D}} p((d_1,d_2)|(v_1,v_2)), \quad (11)$$

where ϕ^W is used for calls sent to WCDMA and ϕ^T is used for calls sent to TDMA. The resulting CTMC for the STSS using policy Ψ_{ini} is shown in Fig. 2, where elements $\boldsymbol{v}$=(v_1,v_2) represent that there are v_1 active voice sessions on TDMA and v_2 on WCDMA. It should be noted that voice calls are sent to WCDMA by default, and when capacity is full they are sent to TDMA. The STSS probability distribution p (v_1,v_2) is easily found by solving the two dimensional CTMC.

Using Ψ_{ini}, the SSPD for the complete system, Π_{ini}, is obtained by unconditioning $p((d_1,d_2)|(v_1,v_2))$ as follows:

$$\Pi(v_1,v_2,d_1,d_2) = p(v_1,v_2)\cdot p((d_1,d_2)|(v_1,v_2)). \quad (12)$$

Having Π_{ini} it is possible to obtain the mean cost of the initial policy $\bar{c}(\Psi_{ini})$ using (7) as it was done in the previous section. Also, we can perform *policy evaluation* by defining the discrete Markov chain of the whole system in order to solve the discrete version of Howard's equation in (8). To maintain consistency, the iterative solution uses the same values of the last section, that is, $\delta' = 10^{-3}$ and $it'_{max} = 500$. The first step of policy iteration is fulfilled by performing *policy improvement* as shown in (9) to obtain Ψ_{fi}. As it can be seen, the difference between both methods lies in how Π_{ini} is obtained, in order to perform *policy evaluation* and *policy improvement*. In this method, Π_{ini} is found solving multiple small sized continuous time Markov chains which are straightforward. While on the previous method, we had to define specific values for δ and it_{max} to reduce computational cost while at the same time maintaining a low error. In this solution, the influence that data user's occupancy have over the system's remaining capacity for voice users is modeled through the introduction of SSPD dependant transition rates, which expands the reach of time scale separation approaches.

V. Numerical Analysis

In previous sections we have defined two methods to find the steady state probability distribution of the initial policy, Π_{ini}, where the first method uses the embedded discrete time Markov chain and the second method uses a novel approach of time scale separation. In this section we compare the 1st step policy iteration solution of both approaches in terms of the computational cost and also in accuracy. The system is defined by the values in Table II.

In Fig. 3 we compare the blocking function for the initial policy chosen for both approximation methods, with the solution obtained by the time scale separation approach, the embedded discrete time Markov chain approach and the optimal solution as λ_v varies from 0.0996 to 0.4498 and $\lambda_d = 0.448$. As it can be seen, with the 1st step of policy improvement, both approximate solutions are close to the optimal for low values of λ_v. However, when $\lambda_v = 0.4498$, it is necessary to perform more iterations in order to find a solution that is closer to the optimal. This occurs due to two reasons: First, when λ_v is high, the optimal solution will use vertical handoff as defined in actions 3 and 4 of Table I more often, while the initial policy does not use vertical handoff. Therefore, it is necessary to change more actions to get close to the optimal solution and this can only be done by performing at least three more policy improvement steps. Secondly, since λ_v is very high and voice calls are not allowed to share TDMA channels as data sessions do, once voice calls have occupied all the resources on WCDMA, they will tend to use more resources of TDMA, reducing the ability of data sessions to share and therefore increasing the blocking function. Hence, in this case the first step solutions are not limited by the approach we choose, but by the initial policy, and that is why both solutions are very similar.

TABLE II
System scenario.

WCDMA	TDMA
W=3.84 Mcps	C = 8
$(E_b/N_0)_v$=6.5 dB	n_c= 3
$(E_b/N_0)_d$=5 dB	$BR_{t,v}$=12.2 kbps
$BR_{w,v}$=12.2 kbps	$BR_{t,d}$=44.8 kbps
$BR_{w,d}$=44.8 kbps	
η_{ul}= 1	
Clients	
μ_v= 0.0083	
σ= 1 Mb	

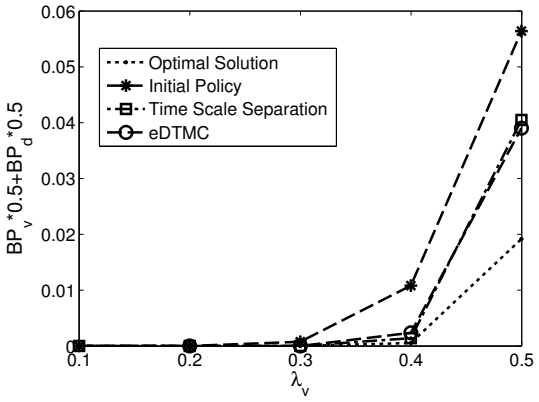

Fig. 3. Blocking function for various λ_v.

In Fig. 4 we follow the same approach and compare the blocking function value for the initial policy with the solution obtained by the time scale separation approach, the embedded discrete time Markov chain approach and the optimal solution as λ_d varies from 0.3584 to 1.792 and $\lambda_v = 0.0833$. In this case, the 1st step of policy iteration is enough to obtain solutions that closely resemble the optimal, and this occurs for the whole range of λ_d. In this case the impact of vertical handoff in blocking probability is diminished by the ability of data sessions to share channels in TDMA.

So far we have seen that the solutions of both approaches are very similar for all the range of λ_v and λ_d. In Table III, it is shown that the computational cost of both solutions is

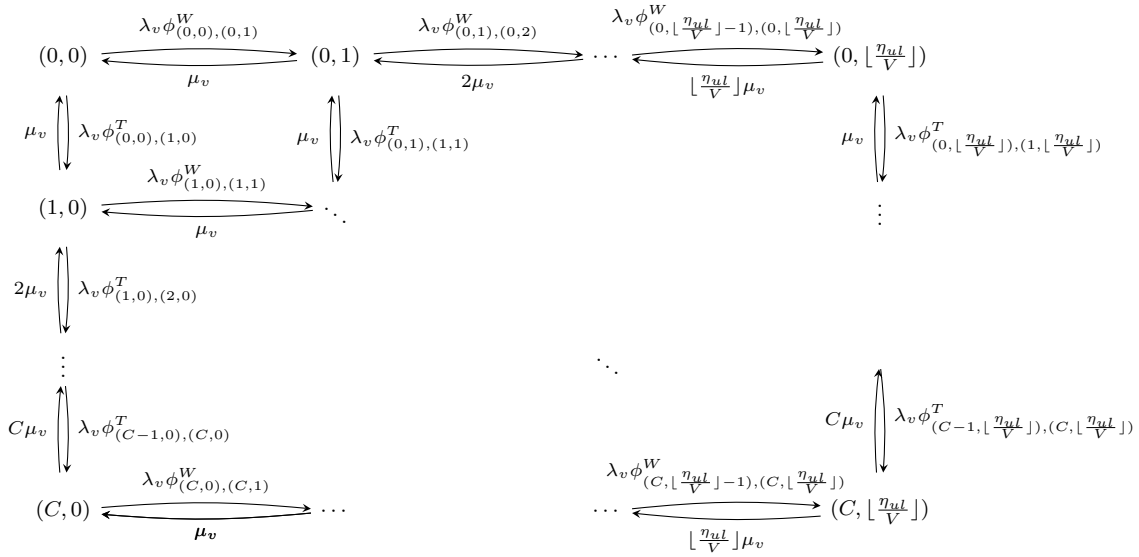

Fig. 2. Continuous-time Markov chain for the slow time scale subsystem.

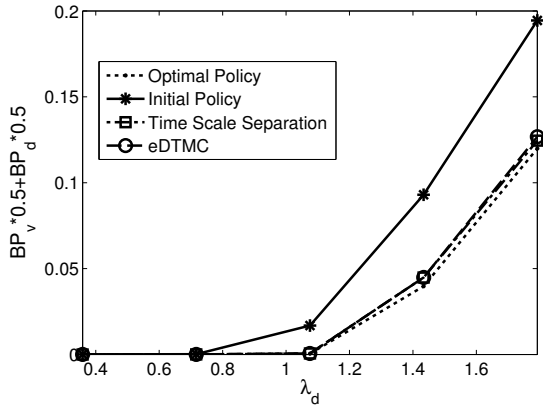

Fig. 4. Blocking function for various λ_d.

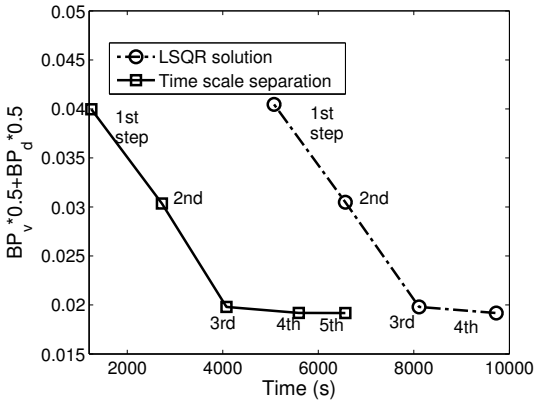

Fig. 5. Computational cost of LSQR in CTMC vs 1st step time scale separation and eDTMC

also very similar for each point of Figures 3 and 4 using a 2 Quad CPU @2.4GHz, 3.24 GB RAM desktop. Let us recall that the optimal solution for $\lambda_v = 0.4998$ and $\lambda_d = 0.448$ was obtained in $9724.99s$ after 4 steps, using the LSQR algorithm. Therefore, the proposed methods achieve a reduction to less than 10% of the original computational cost.

The main advantage that the time scale separation approach has over the eDTMC approach is that it reduces the parameter set to be defined from four to two, which is important as was discussed in Section III, specially if we want to perform more steps, since a high error in the first step will propagate into the next steps. In fact, selection of these parameters is so critical that if we mantain the values of $\delta = 10^{-3}$ and $it_{max} = 500$, and keep performing *policy iteration*, we will obtain a final blocking function value of 0.0392185 after three steps. This of course is far from the optimal solution, and very close to the value obtained in the 1st step policy, thus no real improvement has been done. This occurs because δ and it_{max} are not good enough to accurately perform *policy evaluation* and therefore the information used in *policy improvement* is flawed, and the error grows on each step performed.

In Fig. 5 we compare the blocking function value on each step of *policy improvement* for two solution methods: In the first solution, we use the 1st step time scale separation to obtain the initial mean cost and then perform several steps of *policy iteration* based on the eDTMC, but in this case the chosen values are $\delta' = 10^{-6}$ and $it'_{max} = 50000$. The second solution is based on the LSQR algorithm and uses a CTMC to solve the system. It should be noted that the initial policy in both cases is the one that will send voice calls to WCDMA and data sessions to TDMA. The values of $\lambda_v = 0.4998$ and $\lambda_d = 0.448$ where already used in Fig. 3, and the second method corresponds to the optimal solution showed there.

Since we increased the number of it_{max}, the cost of the 1st step policy is $1246.766s$, over 70% more than the policies

TABLE III
SOLUTION TIMES

	Time Scale Separation	eDTMC solution
λ_v=0.09996, λ_d=0.448	716.36 s	717.28 s
λ_v=0.1999, λ_d=0.448	719.85 s	687.67 s
λ_v=0.2999, λ_d=0.448	726.65 s	696.87 s
λ_v=0.3998, λ_d=0.448	723.09 s	678.19 s
λ_v=0.4998, λ_d=0.448	715.48 s	720.88 s
λ_v=0.0833, λ_d=0.3584	721.01 s	721.36 s
λ_v=0.0833, λ_d=0.7168	712.58 s	723.96 s
λ_v=0.0833, λ_d=1.075	747.34 s	724.46 s
λ_v=0.0833, λ_d=1.434	717.31 s	721.91 s
λ_v=0.0833, λ_d=1.792	727.39 s	733.80 s

seen in figures 3 and 4. However, in this case after five steps we obtain a policy that closely resembles the optimal, with a blocking function value of 0.01917517 while the optimal has a value of 0.01917065, which means a relative error of $2.35 \cdot 10^{-3}\%$. It is also worth noting that the policy obtained using the eDTMC reaches in the third step a blocking function value of 0.01980314, which is very close to the optimal but it was found in $4076.31s$, about $1000s$ less than the time than it took to obtain the 1st step policy obtained using LSQR ($5072.51s$), which has a blocking function value of 0.04046063. Therefore, it can be seen that a significant reduction in computational cost and a close to optimal solution can be obtained using the proposed methods.

VI. CONCLUSIONS

In this work we have evaluated two approximate methods to solve a Markov decision process using the first step of policy iteration. We studied a resource management problem for a heterogeneous network with two radio access technologies (WCDMA and TDMA) and two types of services (voice and data). The first method uses the embedded discrete time Markov chain to obtain the mean cost of the proposed initial policy of the system and this result is used to perform policy improvement. This method requires the definition of four parameters in order to obtain a required accuracy. The second method, which is the most important contribution of this paper, is based on the time scale separation assumption which states that due to the difference of time scales of data and voice events, it is possible to approximate the total system as two separate subsystems solved sequentially. In the first subsystem, data users see voice users as being still. In the second subsystem, data users influence is perceived by voice users affecting the arrival transition rates. The introduction of these conditional transition rates expands the reach of traditional time scale separation beyond the quasi-stationary and flow regimes solutions as boundaries. The mean cost obtained through this method is used to perform a single step of policy improvement as in the previous case, but setting only two parameters for accuracy.

The two solutions are compared against the optimal solution which was found using several steps of policy iteration based on the continuous time Markov chain of the system and solved through the LSQR algorithm. Results showed that the solutions obtained using the proposed methods are very close to the optimal solution for most scenarios, except when the arrival rate of voice calls is high. Also, it was shown that the two approaches introduced have a similar computational cost, and therefore its performance is almost identical, although the first one can use any initial policy and the second can only use initial policies that respect the time scale separation properties.

Finally, it was shown that thanks to the computational cost reduction of the methods proposed, it is possible to perform more steps of policy iteration to obtain solutions that closely resemble the optimal in less time than the needed to perform a single step of policy iteration with LSQR. As future work it would be desirable to extend the time scale separation approach to the policy improvement step to exploit the advantages in computational cost reduction that it brings for the solution of MDPs for large systems.

REFERENCES

[1] C. W. Zobel and W. T. Scherer,"An empirical study of policy convergence in Markov decision process value iteration", Computers and Operations Research, Vol. 32 Issue 1, January 2005.
[2] D.P. Bertsekas and D.A. Castanon, "Adaptive aggregation for infinite horizon dynamic programming", IEEE Transactions on automatic control, Vol 34, Issue 6, June 1989, p 589-598.
[3] P. Schweitzer, A. Seidmann, "Generalized polynomial approximations in Markovian decision processes", Journal of mathematical analysis and applications, Vol 110, Issue 2, p 568-582.
[4] O. Shlakhter, "Acceleration of Iterative Methods for Markov Decision Processes", Thesis, University of Toronto, 2010.
[5] T. J. Ott, K. R. Krishnan, "Separable routing: A scheme for state-dependent routing of circuit switched telephone traffic", Annals of Operations Research, Volume 35, Issue 1, 1992, pp 43-68.
[6] J. van Leeuwaarden, S. Aalto, J. Virtamo:,"Load balancing in cellular networks using first policy iteration", Technical Report, Networking Laboratory, Helsinki University of Technology, 2001.
[7] D. Pacheco-Paramo, V. Pla, V. Casares-Giner and Jorge Martínez-Bauset,"Optimal radio access technology selection on heterogeneous networks", Physical Communication, Volume 5, Issue 3, September 2012, p 253-271.
[8] W.K. Grassman:, "Transient solutions in markovian queueing systems", Computers and Operations Research, Volume 4, Issue 1, 1977, p 47-53.
[9] C. Paige, M. Saunders, "LSQR: An Algorithm for Sparse Linear Equations and Sparse Least Squares", ACM Trans. Math. Software, 1982.