Server optimization of infinite queueing systems

András Mészáros and Miklós Telek

Abstract The problem of optimizing Markovian models with infinitely or finite but infeasible large state space is considered. In several practically interesting cases the state space of the model is finite and extremely large or infinite, and the transition and decision structures have some regular property which can be exploited for efficient analysis and optimization. Among the Markovian models with regular structure we discuss the analysis related to the birth death and the quasi birth death (QBD) structure.

Contents

Server optimization of infinite queueing systems 1
András Mészáros and Miklós Telek
1 Introduction 2
2 Basic definition and notations 4
3 Motivating examples 5
3.1 Optimization of a queueing system with 2 different servers 5
3.2 Optimization of a computational system with power saving mode 6
3.3 Structural properties of these motivating examples 7
4 Theoretical background 8
4.1 Subset measures in Markov chains 8
4.2 Markov chain transformation 10

András Mészáros
MTA-BME Information systems research group, Magyar Tudósok Körútja 2, 1117 Budapest, Hungary, e-mail: meszarosa@hit.bme.hu

Miklós Telek
Budapest University of Technology and Economics, Magyar Tudósok Körútja 2, 1117 Budapest, Hungary e-mail: telek@hit.bme.hu

4.3 Markov decision processes with a set of uncontrolled states 12
4.4 Infinite Markov chains with regular structure 14
5 Solution and numerical analysis of the motivating examples 15
5.1 Solution to the queue with two different servers 15
5.2 Solution to the power-saving model 16
6 Further examples 18
6.1 Optimization of a queuing system with 2 Markov modulated servers 18
6.2 Structural properties of the example with Markov modulated servers 19
7 Infinite MDPs with quasi birth death structure 19
7.1 Quasi birth death process 19
7.2 Solving MDPs with QBD structure 21
8 Solution and numerical analysis of MDPs with QBD structure 25
8.1 Solution of the example with Markov modulated servers . 25
8.2 Markov modulated server with 3 background states 27
9 Conclusion 27
References 28

1 Introduction

Queueing systems with discrete customers and infinite buffer form stochastic models with (countable) infinite state space. The problem of optimal control of such infinite queueing systems often occurs in practical applications. E.g., with the currently more and more widespread used of cloud computing resources the problem of optimal assignment of tasks or task fragments to service blocks is a very hot research topic.

One of the motivating examples of the current work is to find optimal server selection in a Markovian, work conserving (no server is idle when there is a waiting customer), multi server service unit when the servers might have temporal differences. In such a system with n servers the work conserving service policy defines the service process as long as there are at least n customers in the system, because the n oldest customers (assuming ordered service starts) have to be under service at the n servers. In contrast, when there are less then $n-1$ customers in the system and a new customer arrives the customer has to be directed to one of the idle servers. This choice of the idle server allows the optimization of the system behavior when the servers are at least temporarily different (for a graphical representation see Figure 1).

The dominant property of this motivating example is that an infinite state Markov model needs to be controlled such that decisions are possible only in a finite set of states. We use Markov Decision Processes (MDPs) for optimal control of such systems and investigate the special properties of the MDPs with infinite states and finite set of states with possible decisions.

Markov Decision Processes (MDPs) are prevalent for analysing decision problems in queueing systems. The MDP methodology can be used to find the exact optimum in many cases, however, with increasing the size of the examined system its computation time may become prohibitively large. Furthermore, if the system contains an infinite buffer, the standard MDP solution algorithms are not applicable anymore. However, there are cases when these systems can still be analyzed using the tools developed for finite MDPs. There are some general properties that often hold for MDP solutions. Perhaps the most fundamental of them is the threshold form of the optimal policy. A policy is of threshold form, if the optimal decision on a state can be determined by comparing a certain parameter of the state to a fixed value (called threshold). For instance accepting requests to a queue may be optimal until the queue length reaches a certain value. See e.g. [6] or [3] for more examples.

Apart from exact optimal solutions, one can get a quasi-optimal solution by using certain approximation techniques. One possible approach is the truncation of the state space. This may happen based on the physical model (e.g. the size of the buffer is constrained) as in [9] and [5] for example. Alternatively one can use only mathematical considerations as discussed by [1]. Another interesting approach is shown in [8], where a so-called deterministic simulative model is introduced. The essence of this model is that the original MDP is transformed in such a way that transitions of the new model all become deterministic.

Here we discuss another approach, the exact solution of MDP models with infinite or finite but large state spaces. We apply general results from Markov chain theory, e.g. the analysis of Markov chains measures associated with some subsets of states, which has been studied for a long time [2]. Based on the subset measures we introduce a Markov chain transformation with the replacement of one subset, which results in a smaller, thus more easily computable MDP model with the same optimal policy. For the application of this approach one needs to compute subset measures for subsets of infinitely many states if the original model is infinite, which is not possible in general, but there are cases when the regularity in the transition structure of the MDP can be exploited to compute the required subset measures.

The proposed methodology is used to compute the optimal control of some queueing systems. We study queueing systems with Poisson as well as with Markov modulated arrivals and a shared infinite queue with multiple (identical or different) Markovian servers and investigate the following question: *If there are multiple idle servers and there is a request to be served, which server do we choose to serve this request to obtain optimal system operation?*

In the following we present the specifics of the aforementioned transformation method and its application for some concrete examples. The rest of the chapter is organized as follows. Section 4 summarizes the basics of MDPs and the elements of the Markov chain transformation method including the computation of subset measures in general and for some special cases with regular Markov chain structures like the birth death structure and the quasi birth death structure. A set of examples and their analysis based on the proposed Markov chain transformation method are presented in Section 5. Throughout this chapter we are going to build on some basic queueing knowledge, like queue, server, buffer, Poisson process, Little law, work conserving service, ...

2 Basic definition and notations

In this section we restrict the scope of the paper, introduce the applied notations for MDPs and refer to some classical results that will be used later. In the following we will only consider continuous time time homogeneous MDPs without discount. Thus we will use the following definition for MDPs

Definition 1. Let us consider a process $X(t)$ on a continuous time Markov chain with state space S, a set of decisions $A = \{a_i\}$, a set of decision dependent generator matrices $Q = \{Q(a)|a \in A\}$ and a set of decision and state dependent cost rates $C = \{c^a(s)|a \in A, s \in S\}$. We say that the tuple (S,A,Q,C) is a continuous time Markov decision process.

In the following sometimes the C^a cost rate matrix will be used, which is a diagonal matrix constructed from the cost rates for decision a, such that

$$C^a{}_{i,j} = \begin{cases} c^a(i) & \text{if } i = j \\ 0 & \text{otherwise} \end{cases}$$

In this work we concentrate on optimizing for infinite horizon. Because there is no discount in the considered MDPs the goal function is the average cost rate of the process, i.e., the optimal strategy is

$$\pi^* = \arg\min_{\pi} \mathrm{E}_\pi \left[\lim_{k\to\infty} \frac{1}{T} \int_{t=0}^{T} c^{\pi(X(t))}(X(t))dt \right], \tag{1}$$

which is known to be the same as

$$\pi^* = \arg\min_{\pi} \sum_{s\in S} \alpha^\pi(s) c^{\pi(s)}(s), \tag{2}$$

where $c^{\pi(s)}(s)$ is the cost rate in state s if the strategy is π and $\alpha^\pi(s)$ is the steady state probability of being in state s for policy π.

We mention here that the previous description stands for pure strategies (i.e. we always make the same decision in a state with 1 probability). As shown in [4], there always exists a pure strategy that gives the optimum for the average reward rate problem.

We also note that, even though we only consider continuous time MDP examples, the same results hold for the discrete time counterparts. The method to related the continuous and the discrete time processes is referred to as uniformization. The discrete time counterpart of a continuous time MDP can be obtained by $P = \frac{1}{\Delta(Q)} Q + I$, where P is the transition matrix of the discrete time MDP and $\Delta(Q)$ is the largest absolute value in matrix Q, that is $\Delta(Q) = \max_{i,j}(|Q_{i,j}|)$.

3 Motivating examples

3.1 Optimization of a queueing system with 2 different servers

Let us consider an M/M/2 queueing system, i.e. a system with Poisson arrival process with parameter λ and 2 servers with exponential service times and parameters μ_1 and μ_2 respectively, see Figure 1. We assume a shared infinite queue and investigate the following question: If both servers are idle and there is a request to be served, which server do we choose to serve this request to obtain optimal system operation? An intuitive measure of optimality is the average expected sojourn time (system time) $\mathrm{E}(T)$, which is the sum of the average expected waiting time and service time.

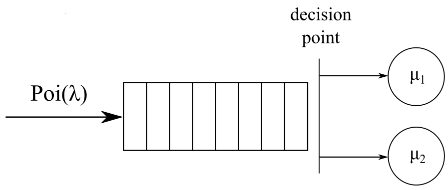

Fig. 1: M/M/2 queueing system with two different servers

We will utilize Little's law, which states that, $\mathrm{E}(n) = \bar{\lambda}\,\mathrm{E}(T)$, where $\mathrm{E}(n)$ is the expected value of average number of requests in the system and $\bar{\lambda}$ is the mean arrival intensity (in this case $\bar{\lambda} = \lambda$). Using this we will optimize $\mathrm{E}(n)$ as it is equivalent to the optimization of $\mathrm{E}(T)$ in the considered example because the decisions do not affect $\bar{\lambda}$.

We can write

$$\mathrm{E}(n) = \sum_{i=0}^{\infty} \alpha_i n(i), \tag{3}$$

where α_i is the steady state probability of state i and $n(i)$ is the number of requests in state i. By comparing this with the formula for average reward rate in (2), we can see that the problem can be formalized as an average reward rate optimization using $c^a(i) = n(i)$.

In the example we consider work conserving schemes only. This means that the service of any request has to start as soon as there is an idle server. Consequently there is only one decision in the system: when a new request arrives to the empty queue we have to decide whether server 1 or server 2 should serve this request.

The generator matrix of the MDP corresponding to this system is

$$Q^a = \left(\begin{array}{cccc|ccc}
-\lambda & p_a\lambda & (1-p_a)\lambda & 0 & \cdots & & \\
\mu_1 & -\lambda-\mu_1 & 0 & \lambda & 0 & \cdots & \\
\mu_2 & 0 & -\lambda-\mu_2 & \lambda & 0 & \cdots & \\
0 & \mu_2 & \mu_1 & -\lambda-\mu_1-\mu_2 & \lambda & 0 & \cdots \\
\hline
\vdots & 0 & 0 & \mu_1+\mu_2 & -\lambda-\mu_1-\mu_2 & \lambda & \ddots \\
 & \vdots & \ddots & \ddots & \ddots & \ddots & \ddots
\end{array}\right), \tag{4}$$

The single decision of choosing between server 1 and 2 happens in the first state. In Q^a this decision is represented by p_a, which is the probability of choosing server 1 upon arrival of a new request in the empty state, that is, the two possible decisions are always choosing the first server ($a = 1$) and always choosing the second server ($a = 2$) with $p_1 = 1$ and $p_2 = 0$. We recall here that there always exists a pure optimal strategy, therefore one of these decisions is optimal.

The cost of each state is the actual number of requests in the system; consequently,

$$c^a(i) = \begin{cases} 0, & \text{for } i = 1 \\ 1, & \text{for } i = 2 \\ i-2, & \text{otherwise} \end{cases} \tag{5}$$

for $a = 1,2$. Note that the decisions do not affect the costs in this case, only the transitions.

3.2 *Optimization of a computational system with power saving mode*

In the second example we consider a system that executes simple computational tasks that can be decomposed to two steps, see Figure 2. The steps take an exponentially distributed time with μ_1 and μ_2 parameter respectively. Tasks arrive according to a Poisson process of parameter λ. Usage of resources induces a certain cost per time unit. Each waiting task requires the same amount of memory, generating cost with rate c_m, while the usage of the CPU generates cost with rate c_c. If the computer becomes idle it can either enter power saving mode, or remain in normal mode, which will be associated with c_i cost rate (power saving mode is assumed to have 0 cost rate). If a new task arrives while the computer is in power saving mode, the first part of the task takes an exponential time of μ_0 ($\mu_0 < \mu_1$) parameter. In other words, power saving mode costs less when the system is empty, but provides a slower service of the first request, which results in higher average CPU and memory usage costs. The operator has to decide if it is beneficial to use power saving mode.

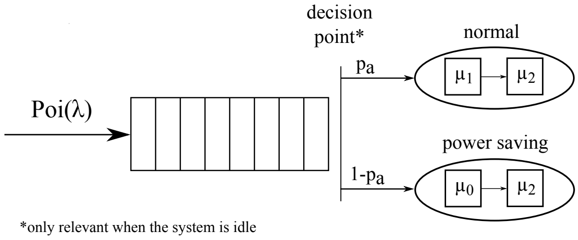

Fig. 2: computational system with power saving mode

If state 1 corresponds to the empty system and p_a represents the decision of power saving such that $p_a = 1$ if power saving mode is used and $p_a = 0$ if not, then the generator of this process can be written as

$$
Q^a = \left(\begin{array}{c|ccc|ccc|ccc|c}
-\lambda & p_a\lambda & (1-p_a)\lambda & 0 & 0 & 0 & 0 & & & & \\ \hline
0 & -\lambda-\mu_0 & 0 & \mu_0 & \lambda & 0 & 0 & & & & \\
0 & 0 & -\lambda-\mu_1 & \mu_1 & 0 & \lambda & 0 & & & & \\
\mu_2 & 0 & 0 & -\lambda-\mu_2 & 0 & 0 & \lambda & & & & \\ \hline
0 & 0 & 0 & 0 & -\lambda-\mu_0 & 0 & \mu_0 & \lambda & 0 & 0 & \\
0 & 0 & 0 & 0 & 0 & -\lambda-\mu_1 & \mu_1 & 0 & \lambda & 0 & \\
0 & \mu_2 & 0 & 0 & 0 & 0 & -\lambda-\mu_2 & 0 & 0 & \lambda & \\ \hline
 & & & & 0 & 0 & 0 & & & & \ddots \\
 & & & & 0 & 0 & 0 & & \ddots & & \\
 & & & & \mu_2 & 0 & 0 & & & & \\ \hline
 & & & & & & & \ddots & & &
\end{array}\right). \tag{6}
$$

3.3 Structural properties of these motivating examples

The main characteristics of the above described examples is associated with number of states of the queueing systems and the number of states where different decisions are possible. In both examples the overall state space is composed by infinitely many states, which inhibits the application of several standard MDP solution methods. On the other hand, the set of states in which decisions can be made (different actions can be chosen) is finite. These characteristic properties suggests the division of the set of states of such MDPs into two parts, the subset where decisions can be made and the complementer subset. Assuming that this structural properties are often present in MDP problems below we first introduce analysis results of Markov chains associated with state space division, and based on them we discuss a solution method of such MDPs.

4 Theoretical background

In this part we briefly summarize the notations and the basic mathematical structures used for the decomposition based analysis of the considered MDP models.

4.1 Subset measures in Markov chains

The analysis of Markov chain properties associated with disjoint subsets of states has been considered for a very long time [2]. We summarize the related results in this subsection based on [7]. For a more detailed explanation of the presented results the reader is referred to that textbook. We borrow the terminology from reliability theory, where the operational states are commonly denoted as up states and the failure states as down states, and apply an S_U, S_D state partitioning such that $S_U \cup S_D = S$ and $S_U \cap S_D = \emptyset$. With appropriate numbering of states (states with low indexes are in S_U) the associated partitioning of the generator matrix is

$$Q = \begin{pmatrix} Q_U & Q_{UD} \\ Q_{DU} & Q_D \end{pmatrix}. \tag{7}$$

There are various interesting performance measures associated with the sets S_U and S_D. Let $\gamma_U = \min(t|X(t) \in S_U)$ be the time to reach a state in S_U. Starting from state $i \in S_D$ the joint distribution of the time to reach S_U and the state first visited in S_U is

$$\Theta_{ij}(t) = Pr(X(\gamma_U) = j, \gamma_U < t|X(0) = i) \tag{8}$$

The associated density function is $\theta_{ij}(t) = \frac{d}{dt}\Theta_{ij}(t)$ and the matrix function of size $|S_D| \times |S_U|$ composed by these elements satisfies

$$\theta(t) = \{\theta_{ij}(t)\} = e^{Q_D t} Q_{DU}.$$

Several interesting performance measures can be derived from this joint distribution. For example, the distribution of the state first visited in S_U is obtained as

$$\{Pr(X(\gamma_U) = j|X(0) = i)\} = \lim_{t\to\infty} \Theta(t) = \int_{t=0}^{\infty} \theta(t)dt = (-Q_D)^{-1} Q_{DU}, \tag{9}$$

where $i \in S_D$ and $j \in S_U$.

The inverse of matrix Q_D and Q_U always exist if the Markov chain is irreducible and positive recurrent, which we will assume in the following. The elements of matrix $(-Q_D)^{-1}$ have important stochastic meaning related to the time spent in the states of Q_D during a visit to S_D, that is for $i, j \in S_D$

$$E(\text{time spent in state } j \text{ in } (0,\gamma_U)|X(0)=i)) = E(\int_t I_{\{X(t)=j,\gamma_U>t|X(0)=i\}}dt)$$
$$= \int_t Pr(X(t)=j,\gamma_U>t|X(0)=i)dt = [\int_t e^{Q_D t}dt]_{ij} = [(-Q_D)^{-1}]_{ij}$$

where $(0,\gamma_U)$ is the time interval of the visit to S_D, $I_{\{\bullet\}}$ is the indicator of event $\bullet$ and $[M]_{ij}$ refers to the i,j element of matrix M. The time to reach S_U starting from state $i \in S_D$ is phase type distributed with the following density function

$$\sum_{j\in S_U} \theta_{i,j}(t) = e_i\theta(t)\mathbb{1} = e_i e^{Q_D t} Q_{DU}\mathbb{1}, \tag{10}$$

where e_i is the ith unit row vector, i.e. a vector with all its elements being zero except for the ith element which is one, and $\mathbb{1}$ is the column vector with all elements equal to one. To simplify the notations instead of scalar equations we often use appropriate vector expressions. For example (10) can be written as

$$\theta(t)\mathbb{1}_U = e^{Q_D t} Q_{DU}\mathbb{1}_U.$$

The size of vector $\mathbb{1}$ is determined by the context (the size of the matrix it is multiplied with), but occasionally we emphasize the dimension by a subscript. For example $\mathbb{1}_U$ refers to the vector of size $|S_U|$. One can obtain the $S_U \to S_D$ counterparts of these measures by interchanging the role of S_U and S_D in the above expressions.

Based on the joint distribution (8), for later use, we also present the conditional mean time spent in S_D supposing that the first state visited in S_U is j. For $i \in S_D$ and $j \in S_U$

$$E(\gamma_U|X(0)=i,X(\gamma_U)=j) = \frac{E(\gamma_U I_{\{X(\gamma_U)=j\}}|X(0)=i)}{Pr(X(\gamma_U)=j|X(0)=i)} = \tag{11}$$
$$= \frac{[\int_{t=0}^{\infty} t\theta(t)dt\mathbb{1}]_{ij}}{[\int_{t=0}^{\infty}\theta(t)dt\mathbb{1}]_{ij}} = \frac{[(-Q_D)^{-2}Q_{DU}]_{ij}}{[(-Q_D)^{-1}Q_{DU}]_{ij}}.$$

Let α be the stationary probability vector of the Markov chain with generator Q. Then α is the solution of the linear system $\alpha Q = 0$ with normalizing equation $\sum_{i\in S}\alpha_i = \alpha\mathbb{1} = 1$. Let α_U and α_D be the parts of vector α associated with subsets S_U and S_D respectively. Using (7) the partitioned form of the linear system is

$$\alpha_U Q_U + \alpha_D Q_{DU} = 0 \text{ and } \alpha_U Q_{UD} + \alpha_D Q_D = 0,$$

from which we obtain a linear system for α_U

$$\alpha_U(Q_U - Q_{UD}Q_D{}^{-1}Q_{DU}) = 0. \tag{12}$$

The Markov chain with state space S_U and generator $Q_U + Q_{UD}(-Q_D)^{-1}Q_{DU}$ is referred to as censored Markov chain. It is obtained from the original Markov chain by switching off the clock when the Markov chain visits S_D and switching on the clock when the Markov chain visits S_U.

The censored Markov chain defines the stationary probability of the states in S_U through (12) apart from a normalizing constant, because $\sum_{i\in S_u} \alpha_i = \alpha_U \mathbb{1}_U$ is not known based on (12). Intuitively, (12) defines the direction of vector α_U, but does not define its norm. To compute the norm $||\alpha_U|| = \alpha_U \mathbb{1}_U$ we calculate the time spent in S_U and S_D in consecutive visits. Let $T_U(n)$ ($T_U(n)$) be the time of the nth visit to S_U (S_D) and let us denote its limit by $T_U = \lim_{n\to\infty} T_U(n)$ ($T_D = \lim_{n\to\infty} T_D(n)$). The portion of time spent in S_U defines the norm of α_U by the following relation

$$\alpha_U \mathbb{1}_U = \frac{E(T_U)}{E(T_U)+E(T_D)} = \frac{1}{1+\frac{E(T_D)}{E(T_U)}} .$$

$E(T_U)$ can be obtained as the inverse of the stationary rate form S_U to S_D, that is

$$E(T_U) = \frac{1}{\alpha_U Q_{UD} \mathbb{1}} ,$$

and $E(T_D)$ can be computed from the distribution in (10), where the ν_D initial distribution in S_D is characterized by the stationary distribution in S_U and a state transition from S_U to S_D, that is

$$E(T_D) = \nu_D \int_{t=0}^{\infty} t\theta(t)dt \mathbb{1} = \frac{\alpha_U Q_{UD}}{\alpha_U Q_{UD}\mathbb{1}} \int_{t=0}^{\infty} t\theta(t)dt \mathbb{1} = \tag{13}$$

$$= \frac{\alpha_U Q_{UD}}{\alpha_U Q_{UD}\mathbb{1}}(-Q_D)^{-2}Q_{DU}\mathbb{1} = \frac{\alpha_U Q_{UD}}{\alpha_U Q_{UD}\mathbb{1}}(-Q_D)^{-1}\mathbb{1} = \tag{14}$$

$$= E(T_U)\alpha_U Q_{UD}(-Q_D)^{-1}\mathbb{1},$$

where we used $(-Q_D)^{-1}Q_{DU}\mathbb{1} = \mathbb{1}$, which comes from the fact that the row sum of matrix Q is zero, that is $Q_{DU}\mathbb{1} + Q_D\mathbb{1} = 0$. Dividing the last expression by $E(T_U)$ gives

$$\alpha_U \mathbb{1}_U = \frac{E(T_U)}{E(T_U)+E(T_D)} = \frac{1}{1-\alpha_U Q_{UD} Q_D{}^{-1}\mathbb{1}} . \tag{15}$$

4.2 Markov chain transformation

There are practically interesting cases when the analysis of some performance measures is essentially related to only one subset of the states, say subset S_U. (As it is discussed below, in the context of MDPs we are going to consider cases when decisions can be made only in a subset of the states and the considered optimization problem is such that no decision is made in the rest of the sates.) In these cases it is possible to modify the Markov chain in the other subset, S_D, such that the important performance measures associated with S_U remain unchanged. For example, if we are interested only in α_U, the stationary distribution in S_U, it is possible to introduce a modified Markov chain with generator

$$\hat{Q} = \begin{pmatrix} Q_U & \hat{Q}_{UD} \\ \hat{Q}_{DU} & \hat{Q}_D \end{pmatrix}, \tag{16}$$

such that the stationary distribution $\hat{\alpha}$ is identical with the original stationary distribution α for the subset S_U that is $\hat{\alpha}_U = \alpha_U$.

The following example demonstrates this case.

Example 1. Let us consider the infinite birth-death Markov chain with birth rate λ, death rate μ and $S_U = \{0, 1, \ldots, n-1\}$, $S_D = \{n, n+1, \ldots\}$. We introduce $\hat{S}_D = \{n\}$ with associated matrix blocks

$$\hat{Q}_{UD} = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ \lambda \end{pmatrix}, \ \hat{Q}_D = \begin{pmatrix} -\mu+\lambda \end{pmatrix}, \ \hat{Q}_{DU} = \begin{pmatrix} 0 \ldots 0 \ \mu-\lambda \end{pmatrix}.$$

The stationary distribution in S_U is identical for this modified Markov chain and the original one.

The Markov chain transformation in this example is rather intuitive because it retains the following essential properties

- The only possible transition from S_U to S_D ($\hat{S}_D$) is the transition from state $n-1$ to state n.
- The mean time spent in S_D, which is $\frac{1}{\mu-\lambda}$, is identical with the mean time spent $\hat{S}_D$.
- The only possible transition from S_D ($\hat{S}_D$) to S_U is the transition from state n to state $n-1$.

However, these simple properties do not have to hold in general. The following theorem provides a general rule for a Markov chain transformation which maintains the stationary distribution in a subset of states.

Theorem 1. *The stationary distribution of the Markov chain with generator Q and with generator $\hat{Q}$ are identical for S_U if the following conditions hold*

$$Q_{UD}(-Q_D)^{-1}Q_{DU} = \hat{Q}_{UD}(-\hat{Q}_D)^{-1}\hat{Q}_{DU} \tag{17}$$

and

$$Q_{UD}(-Q_D)^{-1}\mathbb{1} = \hat{Q}_{UD}(-\hat{Q}_D)^{-1}\mathbb{1} \ . \tag{18}$$

Proof. The linear system that characterizes the direction of α_U according to (12) is identical with the one characterizing the direction of $\hat{\alpha}_U$ based on $\hat{Q}$ due to (17). In order to ensure the identity of the α_U and $\hat{\alpha}_U$, we still need the sums of the stationary probabilities in S_U to be identical in the two systems, that is $\alpha_U \mathbb{1}_U = \hat{\alpha}_U \mathbb{1}_U$, which comes from (18) using (15).

In addition to the stationary distribution in a wide range of applications (including MDPs) it is important to maintain reward measures as well.

Theorem 2. *The stationary reward rate of a Markov reward model with generator Q and reward rate matrix C and with generator $\hat{Q}$ and reward rate matrix $\hat{C}$ are identical if* (17), (18), $C_U = \hat{C}_U$ *and the following condition holds*

$$Q_{UD}(-Q_D)^{-1}C_D\mathbb{1} = \hat{Q}_{UD}(-\hat{Q}_D)^{-1}\hat{C}_D\mathbb{1}. \tag{19}$$

Proof. The stationary reward rate in the modified Markov reward model is

$$\begin{aligned}\hat{\alpha}\hat{C}\mathbb{1} &= \hat{\alpha}_U\hat{C}_U\mathbb{1}_U + \hat{\alpha}_D\hat{C}_D\hat{\mathbb{1}}_D = \hat{\alpha}_U(\hat{C}_U\mathbb{1}_U + \hat{Q}_{UD}(-\hat{Q}_D)^{-1}\hat{C}_D\hat{\mathbb{1}}_D)\\ &= \alpha_U(C_U\mathbb{1}_U + Q_{UD}(-Q_D)^{-1}C_D\mathbb{1}_D) = \alpha C\mathbb{1}\end{aligned}$$

where we used $\hat{\alpha}_D = \hat{\alpha}_U\hat{Q}_{UD}(-\hat{Q}_D)^{-1}$ in the second equation and $\hat{\alpha}_U = \alpha_U$ (which comes from Theorem 1) in the third equation.

According to Theorem 1 and 2 one can replace a Markov chain with generator Q with a Markov chain with generator $\hat{Q}$ if the required performance measures are associated only with the stationary probabilities in S_U, and (17) and (18) hold. This replacement remains valid for reward measures as well if (19) holds additionally.

We note that (17) is about the identity of two matrices of size $|S_U| \times |S_U|$ and the rank of those matrixes is

$$r = \text{rank}(Q_{UD}(-Q_D)^{-1}Q_{DU}) = \min(\text{rank}(Q_{UD}), \text{rank}(Q_{UD})). \tag{20}$$

Consequently the size of the transformed Markov chain should be at least $|S_U| + r$. For example, in Example 1 we have $r = 1$, because $\text{rank}(\hat{Q}_{DU}) = \text{rank}(\hat{Q}_{UD}) = 1$ and the transformed Markov chain has $n+1$ states.

4.3 Markov decision processes with a set of uncontrolled states

The above discussed state space division based analysis approaches can be efficiently used for the analysis of MDPs where decisions are possible only in a subset of states. More precisely, when there are states in the Markov chain where the Q^a_{ij} transition rates and the $c^a(i)$ associated cost are independent of the decision, that is $Q^a_{ij} = Q_{ij}$ and $c^a(i) = c(i)$, $\forall a \in A$. Unfortunately the efficient application of the space division depends on the properties of the considered problem. We consider some special cases below.

4.3.1 Decisions only in subset$_1$ without an effect on the transitions to subset$_2$

If the MDP is such that decisions are made only in subset$_1$ and it has no effect on the transitions to subset$_2$, then the the generator matrix has the form

$$Q^a = \begin{pmatrix} Q_1^a & Q_{12} \\ Q_{21} & Q_2 \end{pmatrix}.$$

In this case we can apply the association subset$_1$=S_U and subset$_2$=S_D and use the results of Theorem 1 and 2 in order to obtain a simple MDP problem with generator matrix

$$Q^a = \begin{pmatrix} Q_1^a & \hat{Q}_{12} \\ \hat{Q}_{21} & \hat{Q}_2 \end{pmatrix}.$$

4.3.2 Decisions only in subset$_1$ with an effect on the transitions to subset$_2$

If the MDP is such that decisions are made only in subset$_1$ and it has effect on the transitions to subset$_2$ then the the generator matrix has the form

$$Q^a = \begin{pmatrix} Q_1^a & Q_{12}^a \\ Q_{21} & Q_2 \end{pmatrix}.$$

In this case we can apply the association subset$_1$=S_U but we need to use the following decision dependent version of Theorem 1

Theorem 3. *The stationary reward rate of the MDP with generator and reward matrix*

$$Q^a = \begin{pmatrix} Q_U^a & Q_{UD}^a \\ Q_{DU} & Q_D \end{pmatrix}, \quad C^a = \begin{pmatrix} C_U^a & 0 \\ 0 & C_D \end{pmatrix},$$

and the MDP with generator and reward matrix

$$\hat{Q}^a = \begin{pmatrix} Q_U^a & \hat{Q}^a{}_{UD} \\ \hat{Q}_{DU} & \hat{Q}_U \end{pmatrix}, \quad \hat{C}^a = \begin{pmatrix} C_U^a & 0 \\ 0 & \hat{C}_D \end{pmatrix},$$

are identical for any policy if the following conditions hold

$$Q_{UD}^a Q_D{}^{-1} Q_{DU} = \hat{Q}^a{}_{UD} \hat{Q}_D{}^{-1} \hat{Q}_{DU}, \tag{21}$$

$$Q_{UD}^a Q_D{}^{-1} \mathbb{1} = \hat{Q}^a{}_{UD} \hat{Q}_D{}^{-1} \mathbb{1}, \tag{22}$$

and

$$Q_{UD}^a Q_D{}^{-1} C_D \mathbb{1} = \hat{Q}^a{}_{UD} \hat{Q}_D{}^{-1} \hat{C}_D \mathbb{1}. \tag{23}$$

Proof. The proof of Theorem 3 directly follows from the proofs of Theorem 1 and 2.

4.3.3 Decisions only in subset$_1$ with limited boundary to the other set

If the MDP is such that decisions are made only in subset$_1$ but the transitions from subset$_1$ towards the rest of the states can reach only a part of the complementer subset without decision, denoted as subset$_2$, and the remaining part of the subset without decision, denoted as subset$_3$, cannot be reached from subset$_1$, then the generator matrix has the form

$$Q^a = \begin{pmatrix} Q_1^a & Q_{12}^a & 0 \\ Q_{21} & Q_2 & Q_{23} \\ Q_{31} & Q_{32} & Q_3 \end{pmatrix}.$$

In this case we can apply the association subset$_1 \cup$ subset$_1$=S_U and subset$_3$=S_D and with these set definitions the results of Theorem 1 and 2 are directly applicable again.

4.4 *Infinite Markov chains with regular structure*

Thanks to Theorem 1 - 3 Markov chain transformations where the original and the transformed problem have different sizes can be applied in the analysis of MDPs with a set of uncontrolled states. These transformations can be efficiently used when the original problem has a finite or even infinite state space. In this work we focus on the application of Markov chain transformation methods with infinite state space. In case of general infinite state MDPs with completely irregular structure the application of Theorem 1 - 3 is rather difficult, but in the majority of the practically interesting cases infinite state MDPs have some regular structure. We consider two of the simplest structures below.

4.4.1 Birth death process

An MDP has a birth-death structure when (with appropriate numbering of states) state transitions are possible only to neighboring states. A birth-death structure can contain level dependent and level independent rates. Example 1 discusses the case of level independent rates. Here we focus on the level dependent case. Let the arrival and departure rates at state $k < n$ be $\lambda_k(a)$ and μ_k and at state $k \geq n$ be λ_k and μ_k respectively. Furthermore let $S_U = \{0, 1, \ldots, n-1\}$ and $S_D = \{n, n+1, \ldots\}$. Similar to Example 1 we can transform the MDP such that $\hat{S}_D = \{n\}$ with associated matrix blocks

$$\hat{Q}^a{}_{UD} = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ \lambda_{n-1}(a) \end{pmatrix}, \quad \hat{Q}_D = \begin{pmatrix} -\hat{\mu} \end{pmatrix}, \quad \hat{Q}_{DU} = \begin{pmatrix} 0 \ldots 0 \ \hat{\mu} \end{pmatrix}.$$

The rate from $\hat{S}_D$ to S_U, $\hat{\mu}$, can be computed form the recursive relation on the mean time spent in the set $S_k = \{k, k+1, \dots\}$, denoted by T_k, that is

$$T_k = \frac{1}{\lambda_k + \mu_k} + \frac{\lambda_k}{\lambda_k + \mu_k} T_{k+1}$$

where $\hat{\mu} = \frac{1}{T_n}$. If λ_k and μ_k are independent of k then this relation results in $\hat{\mu} = \mu - \lambda$ as in Example 1. If λ_k and μ_k are state dependent then the recursive relation needs to be solved based on the specific form of state dependence. Finally the unknown reward rate $\hat{c}$ can be computed based on (19).

5 Solution and numerical analysis of the motivating examples

In this section we provide some specific examples for the usage of the transformation techniques presented in the previous section.

5.1 Solution to the queue with two different servers

As marked in (4) we select the first four states as S_U and the rest as S_D.

Notice that the upper part of this system is a birth death process, thus we can use the results from Example 1 to get

$$\hat{Q}^a = \left(\begin{array}{cccc|c} -\lambda & p_a\lambda & (1-p_a)\lambda & 0 & \\ \mu_1 & -\lambda-\mu_1 & 0 & \lambda & 0 \\ \mu_2 & 0 & -\lambda-\mu_2 & \lambda & 0 \\ 0 & \mu_2 & \mu_1 & -\lambda-\mu_1-\mu_2 & \lambda \\ \hline 0 & 0 & 0 & \mu_1+\mu_2-\lambda & -\mu_1-\mu_2+\lambda \end{array}\right). \tag{24}$$

We chose $S_U = \{1,2,3,4\}$ and $S_D = \{5\}$, as it is indicated in the transition rate matrix. We can apply the previously presented cost transformation in (42) by noticing that this system is a special QBD where $G = [1]$, i.e., it is a 1×1 matrix with its only element being 1, from which Z and consequently $C_{i\to j}$ can be calculated. By substituting into (38) and using notation $\mu = \mu_1 + \mu_2$ we obtain

$$[\hat{C^a}]_{5,5} = \frac{\sum_{i=0}^{\infty} \frac{1}{\mu} (\frac{\lambda}{\mu})^i (i+3)}{\sum_{i=0}^{\infty} \frac{1}{\mu} (\frac{\lambda}{\mu})^i}.$$

We can use $\sum_{i=0}^{\infty} x^i = \frac{1}{1-x}$ and $\sum_{i=0}^{\infty} i x^i = \frac{x}{(1-x)^2}$ to simplify the expression and get the modified cost function

$$[\hat{C}]_{i,i} = \begin{cases} 0, & \text{for } i = 1 \\ 1, & \text{for } i = 2,3 \\ 2, & \text{for } i = 4 \\ 3 + \frac{\lambda}{\mu - \lambda}, & \text{for } i = 5 \end{cases} \tag{25}$$

The MDP described by $\hat{Q}^a$ and $\hat{C}$ can be solved using standard solution algorithms. Let us consider a specific example with $\lambda = 10$, $\mu_1 = 1$, $\mu_2 = 100$. Using these values we get $\mathrm{E}(n) = 0.15$ for $a = 1$ ($p_a = 0$) and $\mathrm{E}(n) = 0.19$ for $a = 2$ ($p_a = 1$). Unsurprisingly the optimal decision is choosing the faster server whenever it is possible. In this example the optimal strategy is trivial. It can be shown analytically that choosing the faster server is always optimal. For more complex systems; however, giving an analytical solution may be impossible.

5.2 *Solution to the power-saving model*

Starting from state 5 the generator is a QBD with block independent transition rates. Thus we will transform the MDP while keeping the first five states unchanged, that is, we choose $S_U = \{1, \ldots, 5\}$.

While this problem is more complicated than the previous one, we can exploit an important structural characteristic to transform the system to finite states without the usage of the matrix analytic methodology. We will create the same additional states as with the previously proposed transformation method in Section 7.2, but use elementary arguments to obtain the $\omega_{i,j}$ and $[G]_{i,j}$ parameters in the transition rates. Let use notation $\tau_{k,j} = E(\gamma_U | X(0^+) = k, X(\gamma_U) = j),\ k \in S_D,\ j \in S_U$. From (36) it is clear that $\omega_{i,j} = \sum_{k \in S_D} \frac{Pr(X(0^+)=k|X(0^-)=i}{Pr(X(0^+ \in S_D)} \tau_{k,j}$. Thus, if we can calculate $\tau_{k,j}$, $\omega_{i,j}$ can be calculated as well.

Note that Q^a has a QBD structure with group independent blocks starting from group 2. Let us denote the ith state of group n by (n,i). Because of the block independent QBD structure of the generator we can write

$$\tau_{(n,i) \to (n-1,j)} = \tau_{(n+1,i) \to (n,j)}, \forall n > 1,\ i,j = 1,2,3, \tag{26}$$

that is, the time to reach state j of group $n-1$ from state i of group n does not depend on the actual value of n. Furthermore note that the states of group n can only be reached from higher groups through state $(n,1)$ for $n \geq 1$. Consequently $\tau_{(n,i) \to (1,2)}$ can be expressed as

$$\tau_{(n,i) \to (1,2)} = \tau_{(2,1) \to (1,2)} + \tau_{(3,1) \to (2,1)} + \cdots + \tau_{(n,1) \to (n-1,1)} + \tau_{(n,i) \to (n,1)} \tag{27}$$

We can write recursive relations similar to the one for birth death processes. For example

$$\tau_{(2,1)\to(1,2)} = \frac{\lambda}{\lambda+\mu_0}\left(\frac{1}{\lambda+\mu_0} + \tau_{(3,1)\to(1,2)}\right) + \frac{\mu_0}{\lambda+\mu_0}\left(\tau_{(2,3)\to(1,2)} + \frac{1}{\lambda+\mu_0}\right). \tag{28}$$

Here the first term is the expected time it takes to reach state 2 from state 5 if the first event is the arrival of a new request weighted by the probability $\frac{\lambda}{\lambda+\mu_0}$ of such an event. The second term corresponds to the other possibility, i.e., the current request is served before a new request arrives. The probability of this event is $\frac{\mu_0}{\lambda+\mu_0}$. In this case the expected time to reach state 2 is $\frac{1}{\lambda+\mu_0} + \tau_{(2,3)\to(1,2)}$. We can derive expressions for $\tau_{(2,2)\to(1,2)}$ and $\tau_{(2,3)\to(1,2)}$ using the same approach. Thus we get

$$\tau_{(2,2)\to(1,2)} = \frac{\lambda}{\lambda+\mu_1}\left(\frac{1}{\lambda+\mu_1} + \tau_{(2,2)\to(1,2)}\right) + \frac{\mu_1}{\lambda+\mu_1}\left(\frac{1}{\lambda+\mu_1} + \tau_{(2,3)\to(1,2)}\right) \tag{29}$$

$$\tau_{(2,3)\to(1,2)} = \frac{\lambda}{\lambda+\mu_2}\left(\frac{1}{\lambda+\mu_2} + \tau_{(3,3)\to(1,2)}\right) + \frac{\mu_2}{\lambda+\mu_2}\frac{1}{\lambda+\mu_2}. \tag{30}$$

Furthermore, from (27) we have $\tau_{(3,1)\to(1,2)} = \tau_{(3,1)\to(2,1)} + \tau_{(2,1)\to(1,2)}$, $\tau_{(3,2)\to(1,2)} = \tau_{(3,2)\to(2,1)} + \tau_{(2,1)\to(1,2)}$, $\tau_{(3,3)\to(1,2)} = \tau_{(3,3)\to(2,1)} + \tau_{(2,1)\to(1,2)}$, additionally we have $\tau_{(3,2)\to(2,1)} = \tau_{(2,2)\to(1,2)}$, $\tau_{(3,3)\to(2,1)} = \tau_{(2,3)\to(1,2)}$. Using these the attained linear equation system can be easily solved for $\tau_{(2,1)\to(1,2)}, \tau_{(2,2)\to(1,2)}, \tau_{(2,3)\to(1,2)}$, however it results in rather complicated expressions, therefore we do not present the actual solutions. From S_D we always reach S_U in state 2, thus we only need to introduce states $\hat{s}_{2\to 2}$, $\hat{s}_{3\to 2}$, and $\hat{s}_{4\to 2}$ to create a transformed version of the MDP, and for these we only need the previously given τ parameters. Furthermore, from the definition of G it is clear that

$$[G]_{i,j} = \begin{cases} 1, \text{ if } j = 2 \\ 0, \text{ otherwise} \end{cases} \tag{31}$$

Thus we have the necessary τ (and consequently ω) and G values to calculate the elements of $\hat{Q}_{UD}$, $\hat{Q}_D$ and $\hat{Q}_{DU}$ using formulas (35) and (37).

The original cost function of this system is

$$[C^a]_{i,i} = \begin{cases} pc_i, & \text{for } i = 1 \\ c_n + \lfloor \frac{i+2}{3} \rfloor c_m, & \text{for } i \geq 2. \end{cases} \tag{32}$$

To calculate the modified costs we can use the same (45) formula as in the case of the M/M/2 system, utilizing (31) to calculate Z.

Let us take an example where the request arrival rate and service rates are $\lambda = 5,\ \mu_0 = 2,\ \mu_1 = 10,\ \mu_2 = 20$, the cost rate of not entering power saving mode is $c_i = 20$ and the cost rate of memory consumption and CPU usage are $c_m = 2,\ c_n = 10$ respectively. In this case we get

$$\hat{Q}_{UD} = \begin{pmatrix} 5 & 0 & 0 \\ 5 & 0 & 0 \\ 5 & 0 & 0 \end{pmatrix}, \quad \hat{Q}_{DU} = \begin{pmatrix} 1.67 & 0 & 0 \\ 0.45 & 0 & 0 \\ 5 & 0 & 0 \end{pmatrix}, \quad \hat{C}_D = \begin{pmatrix} 28.6 & 0 & 0 \\ 32.5 & 0 & 0 \\ 28 & 0 & 0 \end{pmatrix}$$

and the average cost rate is approximately 38.5 if we use power saving mode and 17.5 if we do not, which means that power saving mode should not be used.

The example did not require the usage of numerical methods for calculating G. The main reason for this is that, when the request was served, the system could go to only one state. Consequently the structure of the G matrix was very special and its values could be derived using elementary arguments. For the same reason the calculation of $\hat{Q}_{UD}$ and $\hat{Q}_{DU}$ could also be done using elementary tools. In the following examples the structure of the generator becomes even more complex, thus the usage of the previously presented will be necessary.

6 Further examples

In the remaining examples we will examine queueing systems with a Markov background process. The point process with Markov background process is referred to as Markov Arrival Process (MAP). The series of inter event times of a MAP form a dependent series of the random variables (in general). We use this series as the consecutive service times of a server, which is some times referred to as Markov Service Process, or MAP service times. The states of the Markov background process are often referred to as phases.

Definition 2. Markov Arrival Process (MAP) is a point process modulated by a background Markov chain. The transition rates which modify the state of the background Markov chain but are not associated with an event of the point process are collected into matrix S_0 and the transition rates which might or might not modify the state of the background Markov chain and are associated with an event of the point process are collected into matrix S_1. The diagonal elements of S_0 are defined such that $Q = S_0 + S_1$ is the generator of the background Markov chain (with zero row sums).

MAPs form a quite general framework for modeling point processes with different correlation structure and marginal distributions while making a simple description and analysis of the overall stochastic model possible.

6.1 Optimization of a queuing system with 2 Markov modulated servers

First let us consider a two server system very much like in the first example, with the only difference being that the servers are identical and they perform service according to a Markov Arrival Process. To avoid confusion we will call the state of a service MAP “phase”, and retain the term “state” for the states of the MDP. (We recall again that the events of the MAP are the service events in this case.) We presume that the internal state of a server (the phase of the MAP) may only change if that server is not idle. Otherwise our assumptions are the same as before: we assume a Poisson arrival process with parameter λ and a shared infinite queue and investigate the following question: If both servers are idle and there is a request to be served, which server do we choose to serve this request to obtain optimal system operation? We choose the average expected sojourn time as the measure of optimality but work with the expected value of the average number of requests in the system which are proportional according to Little's law for the same reason as in the M/M/2 example. Consequently the cost of each state is the number of requests for that state ($C(i) = n(i)$) just like in the M/M/2 example. Also in this case we restrict our inspection to work conserving schemes.

6.2 Structural properties of the example with Markov modulated servers

The state transition structure of the MDP describing the behavior of the queuing system with 2 Markov modulated servers is different from the birth-death structure of the previous examples, because apart of the number of customers in the system the system state has to contain information about the "phase" of the Markov modulated servers. With a proper lexicographical numbering of states the set of states with identical number of customers are continuously indexed (an are commonly referred to as "level"). Due to the fact that a transition can change the number of customers in the system at most by one nonzero transition rates are possible only between neighboring levels. Introducing matrix blocks that contain the state transitions between levels we obtain a similar birth death structure as in (24) on the level of matrix blocks. This transition matrix structure is referred to Quasi birth death structure and is studied in the next section.

7 Infinite MDPs with quasi birth death structure

7.1 Quasi birth death process

Another regular structure of infinite MDPs with practical interest is the quasi birth death (QBD) structure [7] (all results of this subsection are available in [7]). The QBD structure is a generalization of the birth death structure, where the states are divided into groups of finite sizes and transitions are possible only inside a group and between neighboring groups. If the states are numbered according to increasing group identifiers then the transition matrix has the form

$$\begin{array}{|c|c|c|c|c|}
\hline
\mathbf{L_0} & \mathbf{F_0} & & & \\
\hline
\mathbf{B_1} & \mathbf{L_1} & \mathbf{F_1} & & \\
\hline
 & \mathbf{B_2} & \mathbf{L_2} & \mathbf{F_2} & \\
\hline
 & & \mathbf{B_3} & \mathbf{L_3} & \mathbf{F_3} \\
\hline
 & & & \ddots & \ddots \\
\hline
\end{array},$$

where $\mathbf{L_k}$ contains the transitions inside group k, $\mathbf{F_k}$ contains the transitions from group k to group $k+1$, $\mathbf{B_k}$ contains the transitions from group k to group $k-1$, and the idle blocks indicate blocks with zero elements. The size of the groups might be different, but $\mathbf{L_k}$ is an invertible square matrix if the Markov chain is irreducible and positive recurrent.

We introduce a partitioning based on the groups of the QBD. Let sets $S_1, S_2, \ldots$ be defined such that S_n contains the states of group n. Then matrix $G_n(t)$ describes the joint distribution of time to reach S_{n-1} and the state visited first in S_{n-1} starting from a state in S_n. A similar joint distribution is described by matrix $\Theta(t)$ in (8), but here matrix $G_n(t)$ corresponds to the group based partitioning of the QBD.

$$[G_n(t)]_{i,j} = Pr(X(\gamma_{n-1}) = j, \gamma_{n-1} < t | X(0) = i), \; i \in S_n, \; j \in S_{n-1}, \qquad (33)$$

where, like before, $\gamma_n = min(t|X(t) \in S_n)$.

The transform domain expressions for $G_n(t)$ is

$$sG_n(s) = B_n + L_n G_n(s) + F_n G_{n+1}(s) G_n(s)$$

from which the distribution of the state visited first in group $n-1$ is the solution of the recursive equation

$$0 = B_n + L_n G_n + F_n G_{n+1} G_n$$

and the measure related with the mean time to reach group $n-1$, $G'_n = \lim_{s\to 0} \frac{d}{ds} G_n(s)$, can be obtained from

$$G_n = L_n G'_n + F_n G'_{n+1} G_n + F_n G_{n+1} G'_n.$$

There are rather few practically interesting cases when the solution of this recursive equation is available for group dependent transition rates. In practical applications the case of group independent transition rates is much more common.

If the transition rates are block independent, that is, $B_k = B$, $L_k = L$, $F_k = F$ ($\forall k \geq n$), then the matrix expressions simplify to

$$0 = B + LG + FG^2 \qquad (34)$$

and

$$G = LG' + FG'G + FGG'.$$

The first one is a quadratic matrix equation whose minimal non-negative solution can be computed by efficient numerical procedures. When G is known, the second equation is a Sylvester equation for G'.

One of the fundamental statements of group independent QBD theory is that the steady state probability of states has a matrix geometric distribution, i.e.

$$\alpha_{n+1} = \alpha_n R,$$

where α_n is a vector containing the steady state probabilities of states in S_n. Matrix R can be calculated from G as

$$R = F(-L - FG)^{-1}.$$

In the next section we use G, R and other associated matrices to transform MDPs whose uncontrolled set has a (group independent) QBD structure.

7.2 Solving MDPs with QBD structure

In this subsection we present a specific method for the transformation of MDPs with a set of uncontrolled states using the partitioning of 4.3.1.

When the uncontrolled QBD blocks are of size n, the rank of matrix $Q_{UD}Q_D{}^{-1}Q_{DU}$ in (20) is at most n. In this section we present a Markov chain transformation method which maintains the steady state reward rate of the MDP according to Theorem 1 and 2 The new Markov chain is such that during a given visit to $\hat{S}_D$ only a single state is visited before the transition back to S_U. The key idea of the transformation is to assign a state in the transformed MDP to each possible transition from S_U to S_U through a visit in S_D. Matrix $Q_{UD}(-Q_D)^{-1}Q_{DU}$ is composed of a single (potentially) non-zero block of size $n \times n$ associated with the $S_U \to S_D \to S_U$ transition from the last block of S_U to the same block, since transitions are possible only to the neighboring blocks. This non-zero matrix block is composed of n^2 elements. We introduce a modified MDP such that $\hat{S}_D$ is composed of n^2 elements. The associated $\hat{Q}_{UD}$, $\hat{Q}_D$, $\hat{Q}_{DU}$, are defined as follows. Each of $\hat{Q}_{UD}$, $\hat{Q}_D$ and $\hat{Q}_{DU}$ contain (at most) one non-zero elements per row. It means that transition $i \in S_U \to S_D \to j \in S_U$ is described with a $i \in S_U \to \hat{S}_D \to j \in S_U$ transition where the only state visited in $\hat{S}_D$ is associated with the described $i \in S_U \to S_D \to j \in S_U$ transition and is denoted by $s_{i \to j}$. See Figure 3.

There are (at most) n^2 such state transitions and associated states. If transition $i \in S_U \to S_D \to j \in S_U$ is impossible for a given pair of states in the last block of S_U then impossible state transitions and associated $s_{i \to j}$ states can be eliminated from $\hat{S}_D$, which results in less than n^2 states in $\hat{S}_D$.

The transition rate from i to $s_{i \to j}$ is the i, j element of the matrix block in $Q_{UD}(-Q_D)^{-1}Q_{DU}$ associated with the last block of S_U, that is

$$\beta_{ij} = [Q_{UD}(-Q_D)^{-1}Q_{DU}]_{ij} \ . \tag{35}$$

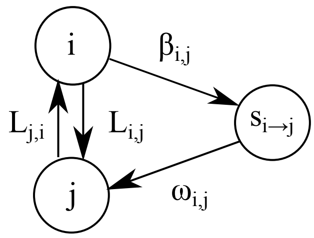

Fig. 3: Transitions in the transformed Markov chain: $i, j \in S_U$, $s_{i\to j} \in \hat{S}_D$.

The transition rate from $s_{i\to j}$ to j is computed based on the conditional mean time spent in S_D supposed that the process moves to S_D from state i and the first state visited in S_U is j. When the initial state in S_D is known this quantity is provided in (11). In our case we need to consider the distribution of the initial state in S_D as well. For $i, j \in S_U$

$$E(\gamma_U | X(0^-) = i, X(0^+) \in S_D, X(\gamma_U) = j) = \tag{36}$$
$$= \frac{E(\gamma_U I_{\{X(\gamma_U)=j\}} | X(0^-) = i, X(0^+) \in S_D)}{Pr(X(\gamma_U) = j | X(0^-) = i, X(0^+) \in S_D)} = \frac{[Q_{UD}(-Q_D)^{-2}Q_{DU}]_{ij}}{[Q_{UD}(-Q_D)^{-1}Q_{DU}]_{ij}} .$$

The transition rate from $s_{i\to j}$ to j is the inverse of the conditional mean time in (36), that is

$$\omega_{ij} = \frac{[Q_{UD}(-Q_D)^{-1}Q_{DU}]_{ij}}{[Q_{UD}(-Q_D)^{-2}Q_{DU}]_{ij}} . \tag{37}$$

With this definition matrix $\hat{Q}_D$ is a diagonal matrix (with negative diagonal elements) and matrix $(-\hat{Q}_D)^{-1}\hat{Q}_{DU}$ is a kind of mapping matrix with only one nonzero element per row whose value is 1. The identity of $Q_{UD}(-Q_D)^{-1}Q_{DU}$ and $\hat{Q}_{UD}(-\hat{Q}_D)^{-1}\hat{Q}_{DU}$, which is required for Theorem 1 to hold, comes from the fact that the only nonzero element of $\hat{Q}_{UD}$ in the row associated with state i is equal with the appropriate element of $Q_{UD}(-Q_D)^{-1}Q_{DU}$ and the multiplication with matrix $(-\hat{Q}_D)^{-1}\hat{Q}_{DU}$ maps this element to the appropriate position.

The identity of $Q_{UD}(-Q_D)^{-1}\mathbb{1}$ and $\hat{Q}_{UD}(-\hat{Q}_D)^{-1}\mathbb{1}$, can be obtained as follows. Matrix $(-\hat{Q}_D)^{-1}$ is a diagonal matrix whose element associated with $s_{i\to j}$ is the expression on the right hand size of (36). The only non-zero matrix element of $\hat{Q}_{UD}$ associated with that state is $[Q_{UD}(-Q_D)^{-1}Q_{DU}]_{ij}$. The product of the two is $[Q_{UD}(-Q_D)^{-2}Q_{DU}]_{ij}$. When we sum up these quantities for all states in $\hat{S}_D$ we obtain $Q_{UD}(-Q_D)^{-2}Q_{DU}\mathbb{1} = Q_{UD}(-Q_D)^{-1}\mathbb{1}$.

The reward rate of state $s_{i\to j}$ is defined as

$$C_{i\to j} = \frac{[Q_{UD}(-Q_D)^{-1}C_D(-Q_D)^{-1}Q_{DU}]_{ij}}{[Q_{UD}(-Q_D)^{-2}Q_{DU}]_{ij}} . \tag{38}$$

We still need to show that the reward rates in $\hat{S}_D$ are defined such that they fulfill the conditions of Theorem 2. Since matrix $(-\hat{Q}_D)^{-1}$ is diagonal with diagonal elements given in (36) the product $(-\hat{Q}_D)^{-1}\hat{C}_D$ is also diagonal with diagonal elements $\frac{[Q_{UD}(-Q_D)^{-1}C_D(-Q_D)^{-1}Q_{DU}]_{ij}}{[Q_{UD}(-Q_D)^{-1}Q_{DU}]_{ij}}$. Multiplying this diagonal element with the i to $s_{i\to j}$ transition of matrix $\hat{Q}_{UD}$ we have $[Q_{UD}(-Q_D)^{-1}C_D(-Q_D)^{-1}Q_{DU}]_{ij}$. Summing up this quantity for the destination state j we have

$$Q_{UD}(-Q_D)^{-1}C_D(-Q_D)^{-1}Q_{DU}\mathbb{1} = Q_{UD}(-Q_D)^{-1}C_D\mathbb{1} \ .$$

We note that there are more than one reward rate definition which fulfills Theorem 2. The one in (38) is such that the above described $(Q,C) \to (\hat{Q},\hat{C})$ Markov reward model transformation does not modify the Markov reward model (apart of potential renumbering of states) whose original structure (defined by matrix Q) complies with the structure of matrix $\hat{Q}$ depicted in Figure 3.

7.2.1 QBD measures associated infinite sets

Already Theorem 1 and 2 indicate that the Markov chain transformation approach is applicable only if we can compute the measures on the left hand size of (17)-(19). If S_D is composed by a finite number of states it is a trivial computational task with $\mathscr{O}(|S_D|^3)$ complexity. If S_D is composed by an infinite number of states it is a more difficult problem which has a nice solution only in a limited number of cases. One those cases is Markov chain with group independent QBD structure. In that case the Q_{UD} Q_D and Q_{DU} matrices have the following structure.

$$Q_{UD} = \begin{array}{|c|c|c|c|}\hline & & & \cdots \\ \hline & & & \cdots \\ \hline \mathbf{F} & & & \cdots \\ \hline\end{array}, \quad Q_D = \begin{array}{|c|c|c|c|}\hline \mathbf{L} & \mathbf{F} & & \\ \hline \mathbf{B} & \mathbf{L} & \mathbf{F} & \\ \hline & \mathbf{B} & \mathbf{L} & \ddots \\ \hline & & \ddots & \ddots \\ \hline\end{array}, \quad Q_{DU} = \begin{array}{|c|c|c|}\hline & & \mathbf{B} \\ \hline & & \\ \hline & & \\ \hline \vdots & \vdots & \vdots \\ \hline\end{array}.$$

Due to the block structure of matrix Q_D its inverse is a full matrix. When, to compute $Q_{UD}(-Q_D)^{-1}Q_{DU}$, we multiply this full matrix with Q_{UD} from the left and with Q_{DU} from the right only the upper left block of $(-Q_D)^{-1}$ plays role in the result and that block is computable based on the process restricted to the first group S_D [7] as $(-L-FG)^{-1}$. The essential main value of this expression is that a block of an infinite matrix inverse can be computed by a finite matrix inverse. Consequently the only non-zero block of matrix $Q_{UD}(-Q_D)^{-1}Q_{DU}$, its lower left block, equals to FZB, where matrix Z is defined as $Z = (-L-FG)^{-1}$. That is

$$[Q_{UD}(-Q_D)^{-1}Q_{DU}]_{ij} = [FZB]_{ij} \ , \tag{39}$$

where the left hand side of the equation refers to the i,j element of the non-zero block. We note that $R = FZ$ and $G = ZB$, which can be used to simplify the nota-

```
X = I;
S = 0;
Repeat
  X = RXG;
  S = S + X;
Until ||X|| < ε;
```

Table 1: Procedure 1

```
X = I; Y = I;
S = 0; k = 1;
Repeat
  X = RX; Y = YG;
  S = S + X C_Dk Y; k++;
Until ||X C_Dk Y|| < ε;
```

Table 2: Procedure 2

tions. Unfortunately, the computation of $Q_{UD}(-Q_D)^{-2}Q_{DU}$ requires the evaluation of further blocks of the infinite matrix inverse $(-Q_D)^{-1}$, because all blocks of the upper row and the left column of $(-Q_D)^{-1}$ contribute to the upper left block of $(-Q_D)^{-2}$. The kth block of the upper row of $(-Q_D)^{-1}$ is $Z(FZ)^{k-1} = ZR^{k-1}$, and the kth block of the left column of $(-Q_D)^{-1}$ is $(ZB)^{k-1}Z = G^{k-1}Z$. Using these relations, we have

$$[Q_{UD}(-Q_D)^{-2}Q_{DU}]_{ij} = [\sum_{k=1}^{\infty} R^k G^k]_{ij} \ , \tag{40}$$

where the infinite sum, $S = \sum_{k=1}^{\infty} R^k G^k$, can be computed by the following simple iterative procedure in Table 1. For positive recurrent Markov chains the infinite summation converges to a finite limit, because the spectral radius of R is less than 1 and the spectral radius of G is 1.

The block structure of the reward rate matrix is

$$C_D = \begin{array}{|c|c|c|} \hline C_{D1} & & \\ \hline & C_{D2} & \\ \hline & & \ddots \\ \hline \end{array} \ ,$$

where matrices C_{Dk} denote the reward rate matrix associated with the kth group of S_D. Utilizing the knowledge on the upper row of $(-Q_D)^{-1}$ the overall reward measure associated with a visit to S_D can be computed as

$$Q_{UD}(-Q_D)^{-1}C_D\mathbb{1} = \sum_{k=1}^{\infty} R^k C_{Dk}\mathbb{1} \ . \tag{41}$$

The evaluation of this infinite sum depends on the properties of the reward rate matrix. The infinite summation converges to a finite limit if the Markov chain is positive recurrent and the C_{Dk} series increases sub-exponentially. In practice, the most common case is when C_{Dk} is proportional to k which results in a finite limit for positive recurrent Markov chains.

If the reward rate is group dependent in S_D, then numerical iterations are required to compute the infinite summation. If the reward rate is group independent in S_D, that is $C_{Dk} = \bar{C}_D$ for $\forall k \geq 1$, then

$$Q_{UD}(-Q_D)^{-1}C_D\mathbb{1} = \sum_{k=1}^{\infty}(FZ)^k\bar{C}_D\mathbb{1} = FZ(I-FZ)^{-1}\bar{C}_D\mathbb{1}\ ,$$

which expression, on the right hand side, contains operations with computable finite matrices only. Assuming group dependent reward rates and utilizing again the upper row and the left column of $(-Q_D)^{-1}$, we have

$$[Q_{UD}(-Q_D)^{-1}C_D(-Q_D)^{-1}Q_{DU}]_{ij} = [\sum_{k=1}^{\infty}R^kC_{Dk}G^k]_{ij}\ , \qquad (42)$$

where the infinite sum can be computed by the numerical procedure in Table 2. For positive recurrent Markov chains this infinite summation has the same convergence behavior as the one in (41).

Finally, we summarize the QBD specific measures of the Markov chain transformation for later use

$$\beta_{ij} = [FZB]_{ij}\ , \qquad (43)$$

$$\omega_{ij} = \frac{[FZB]_{ij}}{[\sum_{k=1}^{\infty}R^kG^k]_{ij}}\ , \qquad (44)$$

$$C_{i\to j} = \frac{[\sum_{k=1}^{\infty}R^kC_{Dk}G^k]_{ij}}{[\sum_{k=1}^{\infty}R^kG^k]_{ij}}\ . \qquad (45)$$

8 Solution and numerical analysis of MDPs with QBD structure

8.1 Solution of the example with Markov modulated servers

Let us denote by S_0, S_1 the corresponding matrices of the MAP and use the standard $\otimes$ and $\oplus$ notation for the Kronecker product and Kronecker sum operators respectively. Furthermore let us denote by I_x the identity matrix of size x. Then the generator matrix of the describing MDP is

$$Q = \left(\begin{array}{cc|cccc} L_0 & F_0 & 0 & \cdots & & \\ B_1 & L_1 & F_1 & 0 & \cdots & \\ \hline 0 & B_2 & L & F & 0 & \\ \vdots & 0 & B & L & F & \ddots \\ & \vdots & \ddots & \ddots & \ddots & \ddots \end{array}\right). \qquad (46)$$

The blocks of Q are

$$L_0 = -\lambda I_4, \quad F_0 = \lambda \left(P\, I_4 - P\right), \quad B_1 = \begin{pmatrix} I_2 \otimes S_1 \\ S_1 \otimes I_2 \end{pmatrix},$$
$$L_1 = \begin{pmatrix} I_2 \otimes S_0 - \lambda I_4 & 0 \\ 0 & S_0 \otimes I_2 - \lambda I_4 \end{pmatrix}, \quad F_1 = \lambda \begin{pmatrix} I_4 \\ I_4 \end{pmatrix},$$
$$B_2 = \left(S_1 \otimes I_2 \; I_2 \otimes S_1\right), \quad L = S_0 \oplus S_0 - \lambda I_4, \quad F = \lambda I_4,$$
$$B = S_1 \otimes I_2 + I_2 \otimes S_1,$$

where

$$P = \begin{pmatrix} z & 0 & 0 & 0 \\ 0 & p & 0 & 0 \\ 0 & 0 & 1-p & 0 \\ 0 & 0 & 0 & z \end{pmatrix}, \tag{47}$$

where z is an arbitrary value in $[0,1]$ The cost function is

$$C^a{}_{i,i} = \begin{cases} 0, & \text{for } 1 \geq i \leq 4 \\ 1, & \text{for } 5 \geq i \leq 12 \\ \lfloor \frac{i}{4} \rfloor c_m, & \text{otherwise .} \end{cases} \tag{48}$$

In this queueing system there is one simple question to be answered: If both servers are idle, one of them is in phase 1 and the other one is in phase 2, which server has to process the next arriving customer to have a minimal average system time? In the generator this decision is represented by p in matrix P. If $p = 1$ we choose the server in phase 1, if $p = 0$ we choose the server 2.

Let us take a specific example, where $\lambda = 10$ and

$$S_0 = \begin{pmatrix} -0.1 & 0.05 \\ 0 & -100 \end{pmatrix}, \quad \begin{pmatrix} 0.05 & 0 \\ 5 & 95 \end{pmatrix}. \tag{49}$$

and let $U = \{1,2,3,4\}$ as indicated by the partitioning in (46).

Based on intuition and the results of the M/M/2 system the optimal strategy is to choose the server which can serve the customer faster. This means that we compare the mean service time starting from phase 1 and phase 2, i.e., $t_1 = \underline{e_1}^T (-S_0)^{-1} \mathbb{1}$ and $t_2 = \underline{e_2}^T (-S_0)^{-1} \mathbb{1}$, and if the first expression is smaller, we choose the server in phase 1 ($p = 1$), otherwise the one in phase 2 ($p = 0$), in this case $t_1 = x$, $t_2 = x$, thus $p = 1$ should be optimal. If we solve the MDP, however, we find that $\mathrm{E}(n) \approx 0.11$ if $p = 0$ and $\mathrm{E} \approx 0.098$ if $p = 1$; i.e., it is better to choose the server which serves the customer slower. This counter-intuitive result can be interpreted the following way. If we use the faster server for the first customer, the probability of finishing the service before a new arrival is high, as the mean service time of the faster state is smaller than the mean inter-arrival time of a new customer. Upon service there is a chance that the server moves to the slower state, leaving the system with two servers in the phase with higher service time. In this state there is a higher chance that more than 2 consecutive customers arrive before the first customer can be served, which leads to a higher average system time. In other words, assigning the customer to the faster server leads to a more deteriorated state after service completion, while assigning the customer with the server in the slower phase, there is a chance that the server will move to the faster state upon service, thus the state of the system improves. One can think of this effect as the repair of the server at the cost of a slower service. Extensive numerical investigations suggest that choosing the server with higher service time is optimal for any M/MAP(2)/2 system regardless of the other characteristics of the service MAP and the intensity of arrivals.

8.2 Markov modulated server with 3 background states

In the previous example a simple - although counterintuitive - rule could be made for the optimal decision. For even slightly more complicated systems this becomes increasingly difficult. Let us take the same system as before just change the service MAP from a MAP(2) to a MAP(3):

$$S_0 = \begin{pmatrix} -1 & 0 & 0 \\ 0 & -2.3 & 0 \\ 0 & 0 & -100 \end{pmatrix}, \quad S_1 = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 2.3 \\ 100 & 0 & 0 \end{pmatrix}.$$

Solution of this system is done the same way as before. Let us set $\lambda = 1.2$. In this case the optimal strategy is to always prioritize the server in phase 1 and choose the server in phase 2 over the one in phase 3. This is in accordance with the results of the M/MAP(2)/2 case, i.e, we choose the slowest available server. For $\lambda = 1.5$, however, it is better to choose the server in phase 3, if the other server is in phase 2. This example demonstrates, that, even for very simple cases, the optimal strategy cannot be determined based on intuition or simple examination of the system. In these cases the numerical solution of the problem is required.

Acknowledgement

This work is partially supported by the OTKA K101150 projects.

9 Conclusion

We have considered the problem of numerical analysis of MDPs with very large and infinite state spaces, where decisions can be made only in a finite subset of states. To handle such MDP models we introduced a general framework for model transformations of MDP such that the modified model has the same optimal policy as the original one. The applicability of this framework depends on the computability of some performance measures associated with a subset of states of the MDP model. We presented the computation of those subset measures in case of two special Markov chain structures the birth death and the quasi birth death structure. We applied the proposed methodology for a set of application examples where the optimal control of queueing systems with infinite buffer is of interest.

References

1. E. Altman. *Constrained Markov decision processes*, volume 7. CRC Press, 1999.
2. J. A. Buzacott. Markov approach to finding failure times of repairable systems. *IEEE transactions in reliability*, R-19(4):128–134, 1970.
3. D. Efrosinin. *Controlled Queueing Systems with Heterogeneous Servers*. PhD thesis, University of Trier, 2004.
4. J. Filar and K. Vrieze. *Competitive Markov Decision Processes*. Springer, 1997.
5. K. Jagannathan, S. Mannor, I. Menache, and E. Modiano. A state action frequency approach to throughput maximization over uncertain wireless channels. *Internet Mathematics*, 9(2-3):136–160, 2013.
6. Y. Kocaga and A. Ward. Admission control for a multi-server queue with abandonment. *Queueing Systems*, 65(3):275–323, 2010.
7. G. Latouche and V. Ramaswami. *Introduction to matrix analytic methods in stochastic modeling*, volume 5. Siam, 1999.
8. A. Y. Ng and M. Jordan. Pegasus: A policy search method for large mdps and pomdps. In *Proceedings of the Sixteenth conference on Uncertainty in artificial intelligence*, pages 406–415. Morgan Kaufmann Publishers Inc., 2000.
9. J. Slegers, I. Mitrani, and N. Thomas. Optimal dynamic server allocation in systems with on/off sources. In *Formal Methods and Stochastic Models for Performance Evaluation*, pages 186–199. Springer, 2007.