Approximate Product Form Solutions for Markov Chains

Johann Christoph Strelen
Rheinische Friedrich–Wilhelms–Universität Bonn

Abstract

An approximation technique for finite state Markov chains is proposed. The solutions are in product form and can be calculated iteratively. This iteration is equivalent to disaggregating macro probabilities, multiplying with transition probabilities, and aggregating the obtained state probabilities – we call it *DA iteration*. Large Markov chains with many states can be solved, so large that the state probabilities could neither be stored nor calculated in any computer.

Accurate approximations are obtained, and moreover, the accuracy can be influenced: The more aggregates are considered, the better the approximations are, if the aggregates are chosen suitably.

Practical experience is encouraging: The obtained accuracy compares favourably with other approximation methods and with exact solutions.

In this paper, we discuss steady state probabilities.

Introduction

Markov chains are widely used in stochastic modeling. We are interested in models of computer, communication, and manufacturing systems. There are discrete time and continuous time Markov chains. The latter can be transformed into discrete time Markov chains which behave equivalently. This is why our technique, which is designed for discrete time Markov chains, is useful for both types.

The goal of considering Markov chain models is to obtain performance measures like throughput of jobs, utilization and availability of system components, and response times. Usually, throughput, utilization, availability, and in some cases response times can be derived directly from state probabilities, but sometimes this is difficult for response times ([2],[5]). Steady state probabilities are solutions of typically large and sparse linear systems of equations.

The most important, well-known, and widely applied modeling paradigms in this field, namely queueing systems, queueing networks, and generalized stochastic Petri nets (GSPN), can be considered as abstract representations of Markov chains, if the

[0]This work was supported in part by the Deutsche Forschungsgemeinschaft

memoryless condition is satisfied – this can be assumed if all probability distributions are of Coxian type.

Unfortunately, the number of Markov chain states grows quickly as the complexity of the model increases. Therefore much effort has been made on the development of algorithms for large linear systems of equations, like power, Gauss-Seidel, and SOR algorithms; an overview is given in [11]. Nowadays, systems with some hundreds of thousands of states can be solved. A special class of algorithms is based on the principle of iterative aggregation and disaggregation, a well-established solution technique for Markov chains, see for example the survey of Schweitzer [12].

Another problem are different time scales which stem from rare events in the modeled systems. This may lead to very long iteration times when iterative algorithms are applied. Simultaneous iterative techniques ([6], [14], [15], [16]) cope with this.

Our *disaggregation aggregation (DA) iteration* is a new general approximate aggregation technique for finite Markov chains with discrete time. The calculated state probabilities are in product form; we call them *disaggregation aggregation product form (DAΠ)* solutions. Here, neither the complete state space, nor all state probabilities, nor the transition probability matrix are handled explicitly. Marginal probabilities are considered instead. Therefore Markov chains with really large state spaces can be treated, for example billions of states.

Time and space demands are higher than for many known, special purpose heuristics, but are by orders of magnitude lower than for exact solutions of Markov chains with many states. On the other hand, the results of DA iteration are usually more accurate than the heuristic solutions. Moreover, the accuracy can be influenced: The more aggregates are considered, the more accuracy can be obtained, if the aggregation is chosen suitably. If many aggregates are used, the aggregation is called *fine*.

It is interesting to note that the steady state probabilities of separable queueing networks are equal to DAΠ solutions calculated according to suitably chosen aggregates.

We analyzed quite a lot of Markov chain models applying DA iteration, and we compared the obtained results with the exact solutions and with results which were calculated using heuristic methods known from the literature. The accuracy of DAΠ solutions compares favourably.

In this paper, we consider steady state solutions. In section 1, DAΠ solutions and DA iteration are discussed. In section 2, we show that there exists always a product form approximation for any discrete probability distribution, which is as accurate as desired, with only a small number of factors. Practical aspects are treated in section 3. As an illustrative example, the techniques are applied to queueing networks with blocking.

1 Disaggregation-aggregation iteration

Irreducible aperiodic Markov chains $(Z(t),\ t \in I\!N)$, discrete and homogeneous in time, with finite state space $\mathcal{Z} = [1:n],\ n \in I\!N$, are considered.

A class $\{\mathcal{Z}_{k,0}, \ldots, \mathcal{Z}_{k,N_k}\},\ N_k \in I\!N,\ k \in I\!N$, of disjoint non-empty subsets of $\mathcal{Z}$ whose union is the full set $\mathcal{Z}$ itself is called *partition* $\mathcal{Z}_k$ of $\mathcal{Z}$. A class $\{\mathcal{Z}_1, \ldots, \mathcal{Z}_C\},\ C \in I\!N$, of partitions is called an *aggregation* AGG of $\mathcal{Z}$, and each of the subsets is an *aggregate*.

If for all *state tuples* $\mathbf{z} = (z_1, \ldots, z_C) \in [0 : N_1] \times \ldots \times [0 : N_C]$ the intersections $\mathcal{Z}_{1,z_1} \cap \ldots \cap \mathcal{Z}_{C,z_C}$ contain at most one element, the aggregation is said to *completely identify the states*. If the intersection is empty, the state tuple $\mathbf{z}$ is termed *infeasible*, otherwise *feasible*. In the sequel we consider only aggregations which completely identify the states. Thus, for each state $z \in \mathcal{Z}$ there is one and only one such state tuple $\mathbf{z}$. The set $\mathcal{Z}^T$ of all these feasible state tuples is another representation of the state space $\mathcal{Z}$. There is a one-to-one mapping $\mathcal{I} : \ \mathcal{Z}^T \ \text{ onto } \ \mathcal{Z}$.

$p^{(t)}(z) \triangleq \mathrm{P}\{Z(t) = z\}, \ \ z \in \mathcal{Z}$, denotes the state probabilities of the Markov chain at time t. For convenience, we use also $p^{(t)}(\mathbf{z}), \ \mathbf{z} \in \mathcal{Z}^T$, where $p^{(t)}(\mathbf{z}) = p^{(t)}(z)$ for $z = \mathcal{I}(\mathbf{z}), \ \mathbf{z} \in \mathcal{Z}^T$, holds.

The n-vector $\mathbf{p}^{(t)} = [p^{(t)}(1), \ldots, p^{(t)}(n)]$ contains the state probabilities, and $\mathbf{P} \triangleq [p(x,y)]_{x,y=1}^n$ denotes the matrix of the transition probabilities. For convenience, we use also $p(\mathbf{x},\mathbf{y}), \ \mathbf{x},\mathbf{y} \in \mathcal{Z}^T$, where $p(\mathbf{x},\mathbf{y}) = p(x,y)$ for $x = \mathcal{I}(\mathbf{x}), \ y = \mathcal{I}(\mathbf{y})$, holds.

Sometimes, if we are not interested in a particular time, or if steady state probabilities are considered, we omit the specification (t).

Each aggregate $\mathcal{Z}_{k,j}$ defines a *macro probability (marginal probability)* as follows:

$$r_k(j) \triangleq \mathrm{P}\{Z \in \mathcal{Z}_{k,j}\} = \sum_{z \in \mathcal{Z}_{k,j}} p(z), \ j = 0, \ldots, N_k, \ k = 1, \ldots, C. \tag{1}$$

We collect the marginal probabilities into tuples $\mathbf{r}_k \triangleq \big(r_k(0), \ldots, r_k(N_k)\big), k = 1, \ldots, C$, and, in turn, these into $\mathbf{R} \triangleq (\mathbf{r}_1, \ldots, \mathbf{r}_C)$.

By (1), the *aggregation function*

$$\mathcal{A} : [0,1]^n \to [0,1]^{N_1+1} \times \ldots \times [0,1]^{N_C+1}, \ \mathbf{p} \mapsto \mathbf{R},$$

is defined.

Sometimes *joint marginal probabilities*

$$r_{k_1,k_2}(j_1,j_2) \triangleq \mathrm{P}\{Z \in \mathcal{Z}_{k_1,j_1}, Z \in \mathcal{Z}_{k_2,j_2}\} = \sum_{z \in \mathcal{Z}_{k_1,j_1} \cap \mathcal{Z}_{k_2,j_2}} p(z),$$

$$j_i = 0, \ldots, N_{k_i}, \ \{k_1, k_2\} \subseteq [1 : C],$$

are used. Again we collect them into tuples $\mathbf{r}_{k_1,k_2}$.

Each partition $\mathcal{Z}_k, \ k \in [1 : C]$, defines a random variable $Z_k : \ \mathcal{Z}^T \to [0 : N_k], \ \mathbf{z} \mapsto z_k$. The marginal distribution $\mathbf{r}_k$ is the distribution of Z_k. Furthermore, sometimes it is more convenient to consider the random vector $\mathbf{Z} = (Z_1, \ldots, Z_C)$ for which $\mathrm{P}\{\mathbf{Z} = \mathbf{z}\} = p(\mathbf{z}), \ \mathbf{z} \in \mathcal{Z}^T$ holds, instead of the random variable Z.

A special aggregation is defined by the structure of the model as follows. Consider a system which consists of K subsystems, e.g. a queueing network with K nodes. If each aggregate is defined by the state z_k of one single subsystem, $\mathcal{Z}_{k,j} = \{\mathcal{I}(\mathbf{z}) \mid \mathbf{z} \in \mathcal{Z}^T, z_k = j\}, \ j = 0, \ldots, N_k, \ k = 1, \ldots, K$, where $[0 : N_k]$ is the set of all states of subsystem k, then we call the aggregation *canonical*.

Our proposed solution technique needs *disaggregation*: Given positive macro probabilities $\mathbf{R}$, a distribution $\boldsymbol{\pi} = [\pi(1), \ldots, \pi(n)]$ over $\mathcal{Z}$ is defined as follows. First, it satisfies the macro probabilities, $\mathbf{R} = \mathcal{A}(\boldsymbol{\pi})$, and secondly, $\boldsymbol{\pi}$ maximizes the entropy. Thus, according to the *product form theorem*, see Kagan, Linnik, Rao [8, theorem 13.2.1, page 409], this maximum entropy distribution (MED) $\boldsymbol{\pi}$ is uniquely determined and has a product form,

$$\pi(\mathbf{z}) = \frac{1}{G}\prod_{k=1}^{C}\rho_{k,z_k}, \tag{2}$$

where the $\rho_{k,z_k} > 0$ are some real numbers,

$$r_k(j) = \frac{1}{G}\sum_{\substack{\mathbf{z}\in\mathcal{Z}^T\\ z_k=j}}\prod_{i=1}^{C}\rho_{i,z_i},\ j = 0,\ldots,N_k,\ k = 1,\ldots,C, \tag{3}$$

holds, and

$$G = \sum_{\mathbf{z}\in\mathcal{Z}^T}\prod_{k=1}^{C}\rho_{k,z_k} \tag{4}$$

is a normalizing constant. In fact, the MED is completely determined by all parameters $\rho_{k,j}$, $\boldsymbol{\rho}$ for short.

By (2) and (3), the *disaggregation function* is defined:

$$\mathcal{D} : [0,1]^{N_1+1} \times \ldots \times [0,1]^{N_C+1} \to [0,1]^n,\ \mathbf{R} \mapsto \boldsymbol{\pi}.$$

In general, the nonlinear equations (3) must be solved in order to calculate all MED parameters $\rho_{k,j}$ and G of the MED $\boldsymbol{\pi}$, given the marginal probabilities $\mathbf{R}$. This is easily accomplished in some special cases. For example, if the aggregation completely identifies the states, and if the state space is *rectangular*, i.e. if $\mathcal{Z}^T$ is a cartesian product, $\mathcal{Z}^T = [0 : N_1] \times \ldots \times [0 : N_C]$, the MED is given by

$$p(\mathbf{z}) = r_1(z_1) \cdot \ \ldots\ \cdot r_C(z_C),\ \mathbf{z} \in \mathcal{Z}^T, \tag{5}$$

which is easily verified (see [16]). Here $\rho_{k,j} = r_k(j)$, $j = 0,\ldots,N_k$, $k = 1,\ldots,C$, and $G = 1$ hold. This is generalized in section 3 for the case where $\mathcal{Z}^T$ consists of the union of cartesian products. Another simple case are the state probabilities in product form queueing networks.

Instead of calculating the steady state probabilities $\mathbf{q}$ of a Markov chain we determine an approximation $\mathbf{p} = \mathcal{D}(\mathbf{R})$ where $\mathbf{R} = \mathcal{T}(\mathbf{R})$ holds. Here the function $\mathcal{T}$ is defined as follows:

$$\mathcal{T} : [0,1]^{N_1+1} \times\ \ldots\ \times [0,1]^{N_C+1} \to [0,1]^{N_1+1} \times\ \ldots\ \times [0,1]^{N_C+1},\ \mathbf{R} \mapsto \mathcal{A}\Big(\mathcal{D}(\mathbf{R})\mathbf{P}\Big).$$

This approximation $\mathbf{p}$ and its marginal probabilities $\mathbf{R}$ are referred to as *disaggregation aggregation product form (DAΠ) solution*. In fact, only the marginal probabilities $\mathbf{R}$ and the MED parameters $\boldsymbol{\rho}$ and G must be calculated. This is practicable even if the state space is large, so large that neither all state probabilities nor the transition probabilities can be handled.

We calculate the DAΠ solutions using *disaggregation aggregation (DA) iteration*,

$$\mathbf{R}^{(t+1)} = \mathcal{T}(\mathbf{R}^{(t)}),\ t = 1,2,\ldots \tag{6}$$

until $\max_{k,j} |r_k^{(t+1)}(j) - r_k^{(t)}(j)|$ is sufficiently small, beginning with feasible marginal probabilities $\mathbf{R}^{(1)}$.

It is important to note that the function $\mathcal{T}$ must be evaluated without handling either all transition probabilities or all probabilities $\boldsymbol{\pi}^{(t)}$. Whether this is possible, depends on the transition probabilities and on the aggregation. According to our experience, such conditions are fulfilled very often in Markov chains originating from performance or reliability models. We have not yet found any model which could not be treated with DA iteration (we tried quite a lot). In section 3, these matters are treated.

More precisely, the conditions are fulfilled if the distribution functions $\varphi_k(j,l)$ of the random vectors $(Z_k^{(t)}, Z_k^{(t+1)})$,

$$\varphi_k(j,l) = \sum_{\substack{\mathbf{y}\in\mathcal{Z}^T \\ y_k=j}} \sum_{\substack{\mathbf{z}\in\mathcal{Z}^T \\ z_k=l}} p(\mathbf{y},\mathbf{z})p^{(t)}(\mathbf{y}),\ j,l = 0,\ldots,N_k,\ k=1,\ldots,C, \tag{7}$$

can be simplified substantially by collecting many summands into only a few expressions.

Observing

$$r_k^{(t+1)}(l) = \sum_{j=0}^{N_k} \varphi_k(j,l),\ l=0,\ldots,N_k, \tag{8}$$

$$\varphi_k(j,j) = r_k^{(t)}(j) - \sum_{\substack{l=0 \\ l\neq j}}^{N_k} \varphi_k(j,l),\ j=0,\ldots,N_k,\ k=1,\ldots,C, \tag{9}$$

follows. (9) is already simple in the sense mentioned above.

The conditional probabilities

$$\pi_k(j,l) \triangleq \mathrm{P}\{Z_k^{(t+1)} = l \mid Z_k^{(t)} = j\} = \frac{\varphi_k(j,l)}{r_k^{(t)}(j)},\ \ j,l=0,\ldots,N_k,\ k=1,\ldots,C, \tag{10}$$

are *aggregate transition probabilities* as the following *aggregate transition equations*

$$r_k^{(t+1)}(l) = \sum_{j=0}^{N_k} \pi_k(j,l) r_k^{(t)}(j),\ l=0,\ldots,N_k,\ k=1,\ldots,C, \tag{11}$$

hold. The aggregate transition probabilities are not homogeneous in time, in general. According to (9),

$$\pi_k(j,j) = 1 - \sum_{\substack{l=0 \\ l\neq j}}^{N_k} \pi_k(j,l),\ j=0,\ldots,N_k,\ k=1,\ldots,C, \tag{12}$$

holds. (8) or (11) are used to calculate the new macro probabilities $\mathbf{R}^{(t+1)}$ from the old ones, $\mathbf{R}^{(t)}$. In this way the function $\mathcal{T}$ is evaluated.

The nonlinear equations for the MED parameters $\boldsymbol{\rho}$, given the macro probabilities $\mathbf{R}$, can be solved iteratively as follows ([7]), if no closed form solution is available:

$$\rho_{k,j}^{[i+1]} = \frac{r_k(j)}{r_k^{[i]}(j)} \rho_{k,j}^{[i]},\ j=0,\ldots,N_k,\ k=1,\ldots,K,\ i=1,2,\ldots. \tag{13}$$

Here $\boldsymbol{\rho}^{[0]}$ are initial approximations of the unknown parameters $\boldsymbol{\rho}$, and $r_k^{[i]}(j)$ are the macro probabilities according to the MED parameters $\boldsymbol{\rho}^{[i]}$, see (3). But in fact, the iteration (13) can be omitted when steady state solutions are desired. It is sufficient to apply just one iteration step,

$$\rho_{k,j}^{(t+1)} = \frac{r_k^{(t+1)}(j)}{r_k^{(t)}(j)} \rho_{k,j}^{(t)} \tag{14}$$

within each DA iteration step (6).

We summarize, stating the

DA Iteration Algorithm

Input: Reasonable MED parameters $\boldsymbol{\rho}^{(1)}$ – e.g. all $\rho_{k,j}^{(1)} = 1$ – and a small bound $\epsilon > 0$

0) Calculate the normalization constant $G^{(1)}$ and the marginal probabilities $\mathbf{R}^{(1)}$. Let $t = 1$

1) Calculate the joint distribution functions $\varphi_k(j, l)$ for all j, l, k, or the aggregate transition probabilities $\pi_k(j, l)$ for all j, l, k

2) From these, calculate the new macro probabilities $\mathbf{R}^{(t+1)}$

3) Calculate the MED parameters $\boldsymbol{\rho}^{(t+1)}$ either directly using (5) or (28), or approximately using (14). In the latter case, calculate again the macro probabilities $\mathbf{R}^{(t+1)}$ from the new MED parameters $\boldsymbol{\rho}^{(t+1)}$. Calculate the normalizing constant $G^{(t+1)}$

4) If $\max_{j,k} |r_k^{(t+1)}(j) - r_k^{(t)}(j)| < \epsilon$ holds the iteration is completed. Otherwise let $t := t+1$ and goto step 1)

Results: The macro probabilities $\mathbf{R}^{(t+1)}$ and the MED parameters $\boldsymbol{\rho}^{(t+1)}$ and $G^{(t+1)}$. $\square$

Now we are going to prove the existence of DAII solutions. This is accomplished applying the following theorem, see e.g. [10, 6.3.2].

Brouwer Fixed-Point Theorem Let $\mathcal{T}' : \bar{\mathcal{P}}_n \subset I\!R^n \to I\!R^n$ be a continuous operator on the compact, convex set $\bar{\mathcal{P}}_n$, and suppose that $\mathcal{T}'(\bar{\mathcal{P}}_n) \subset \bar{\mathcal{P}}_n$. Then $\mathcal{T}'$ has a fixed point in $\bar{\mathcal{P}}_n$.

DA Product Form Existence Theorem The equation

$$\mathbf{R} = \mathcal{T}(\mathbf{R})$$

has a fixed point.

Proof Instead of $\mathcal{T}$, we consider the operator

$$\mathcal{T}' : \bar{\mathcal{P}}_n \to \bar{\mathcal{P}}_n, \quad \mathbf{p} \mapsto \mathcal{D}\Big(\mathcal{A}(\mathbf{p})\Big) \cdot \mathbf{P}. \tag{15}$$

$\bar{\mathcal{P}}_n = \{\mathbf{p}' \in [0,1]^n \mid p'_1 + \ldots + p'_n = 1\} \subset I\!R^n$ is the set of all probability distributions over $[1:n]$. Obviously, $\bar{\mathcal{P}}_n$ is bounded and closed, therefore compact, and convex.

Now we prove that $\mathcal{T}'$ is defined and continuous. In the sequel, we only consider marginal probabilities $\mathbf{R}$, for which a probability distribution $\tilde{\mathbf{p}}$ exists such that $\mathbf{R} = \mathcal{A}(\tilde{\mathbf{p}})$

holds. Therefore we may assume that always a MED exists with the constraints that the marginal probabilities $\mathbf{R}$ are assumed.

Case 1 $\mathcal{Z}^T$ is rectangular. Firstly, let $\mathbf{p} \in \bar{\mathcal{P}}_n$, $\quad \mathbf{R} \triangleq \mathcal{A}(\mathbf{p})$. Clearly, $\mathbf{R}$ depends continuously on $\mathbf{p}$. Secondly, let

$$p'(\mathbf{z}) = r_1(z_1) \cdot \ldots \cdot r_C(z_C), \; \mathbf{z} \in \mathcal{Z}^T. \tag{16}$$

$\mathbf{p}'$ is a probability distribution, $\mathbf{p}' \in \bar{\mathcal{P}}_n$, which depends continuously on $\mathbf{R}$. In the next paragraph, we show, that (16) is equivalent to $\mathbf{p}' = \mathcal{D}(\mathbf{R})$. Thirdly, $\mathbf{p}'\mathbf{P}$ is a continuous mapping. Consequently, $\mathcal{T}'$ (15) is defined in this case, and it is continuous.

If all marginal probabilities $r_k(z_k)$ are positive, $\mathbf{p}'$ is a MED, according to the product form theorem. If there are marginal probabilities $r_k(z_k)$ which are zero, $\mathbf{p}'$ is also a MED, as can be seen as follows: Let

$$\tilde{\mathcal{Z}} = \mathcal{Z}^T \; \setminus \; \{\mathbf{z} \in \mathcal{Z}^T \mid \exists k, j : \mathbf{z} \in \mathcal{Z}_{k,j} \;\text{ where }\; r_k(j) = 0\}.$$

According to the product form theorem,

$$\tilde{p}(z) = p'(z), \; \mathbf{z} \in \tilde{\mathcal{Z}}, \tag{17}$$

is the MED, given the positive marginal probabilities as constraints. For each state $\mathbf{z} \in \mathcal{Z}_{k,j}$ where $r_k(j) = 0$, $p'(\mathbf{z}) = 0$ must hold if the constraints $\mathbf{R} = \mathcal{A}(\mathbf{p}')$ are given. This is fulfilled by $\mathbf{p}'$ in (16). Therefore $\mathbf{p}'$ maximizes the entropy and is in accordance with the constraints.

Case 2 $\mathcal{Z}^T$ is not rectangular, but a proper subset of $[0 : N_1] \times \ldots \times [0 : N_C]$. Therefore $\mathcal{Z}^R \triangleq [0 : N_1] \times \ldots \times [0 : N_C] \setminus \mathcal{Z}^T$ is not empty. Here, we use the *subset independence* feature of maximum entropy distributions, which is stated in the Shore and Johnson paper [13]. For our purposes, this can be formulated as follows. Let $\tilde{\mathbf{p}}$ denote a probability distribution over $\mathcal{Z}^R$ where all $\tilde{p}(\mathbf{z}) > 0$, $\mathbf{z} \in \mathcal{Z}^R$, $\mathbf{R}''' \triangleq \mathcal{A}(\tilde{\mathbf{p}})$, w a probability, $0 < w < 1$, and

$$\mathbf{R}'' = w\mathbf{R} + (1 - w)\mathbf{R}'''.$$

Clearly, $\mathbf{R}'' = \mathcal{A}\big(w\mathbf{p} + (1 - w)\tilde{\mathbf{p}}\big)$ holds. Therefore there is a MED $\mathbf{p}''$ of the form (16) obeying the constraints $\mathbf{R}'' = \mathcal{A}(\mathbf{p}'')$. According to the subset independence feature,

$$\mathbf{p}'' = w\mathbf{p}' + (1 - w)\mathbf{p}'''$$

holds, where $\mathbf{p}'$ is the unique MED over $\mathcal{Z}^T$ with the constraints $\mathbf{R} = \mathcal{A}(\mathbf{p}')$, and $\mathbf{p}'''$ the unique MED over $\mathcal{Z}^R$ with the constraints $\mathbf{R}''' = \mathcal{A}(\mathbf{p}''')$.

Keeping $\mathbf{R}'''$ and w fixed, $\mathbf{R}''$ depends continuously on $\mathbf{R}$. According to (16), applied to $\mathbf{R}''$ and $\mathbf{p}''$, $\mathbf{p}''$ depends continuously on $\mathbf{R}''$. Observing

$$\mathbf{p}' = \big(\mathbf{p}'' - (1 - w)\mathbf{p}'''\big)/w,$$

$\mathbf{p}'$ is defined, and depends continuously on $\mathbf{p}''$. Together, $\mathbf{p}'$ depends continuously on $\mathbf{R}$.

In each of the two cases, we saw that $\mathcal{T}'$ is defined. Moreover, $\mathbf{R} = \mathcal{A}(\mathbf{p})$, $\mathbf{p}' = \mathcal{D}(\mathbf{R})$, and $\mathbf{p}'\mathbf{P}$ are continuous in each case. Therefore, $\mathcal{T}'$ is continuous. Furtermore, for each $\mathbf{p} \in \bar{\mathcal{P}}_n$, $\mathcal{T}'(\mathbf{p})$ is a probability distribution, i.e. $\mathcal{T}'(\mathbf{p}) \in \bar{\mathcal{P}}_n$.

So far we have proved that $\mathcal{T}'$ and $\bar{\mathcal{P}}_n$ fulfil all assumptions of the Brouwer fixed-point theorem. As a consequence, $\mathcal{T}'$ has a fixed point in $\bar{\mathcal{P}}_n$, say $\mathbf{p}$.

Let $\mathbf{R} \stackrel{\triangle}{=} \mathcal{A}(\mathbf{p})$. In turn, $\mathcal{T}(\mathbf{R}) = \mathcal{A}\big(\mathcal{D}(\mathbf{R}) \cdot \mathbf{P}\big) = \mathcal{A}\Big(\mathcal{D}\big(\mathcal{A}(\mathbf{p})\big) \cdot \mathbf{P}\Big) = \mathcal{A}\big(\mathcal{T}'(\mathbf{p})\big) = \mathcal{A}(\mathbf{p}) = \mathbf{R}$ holds. Clearly, $\mathbf{R}$ is a fixed point of $\mathcal{T}$. This concludes the proof. □

Unfortunately, we have not yet found error estimates or a proof of convergence of the iteration. But we solved many models using DA iteration, among others polling, fork join, stochastic Petri net, and blocking models, and we found very accurate results; the iteration did never diverge. Moreover, the accuracy could be substantially ameliorated if more and suitably chosen aggregates are considered. These matters are discussed in section 3.

2 Motivation

In this section, we justify why it is worth applying DA iteration to Markov models.

First, we mention favourable experience with DAΠ approximations for steady state solutions of Markov models: We calculated DAΠ approximations for many models using DA iteration. Usually, we tried different aggregations for the same model, coarse grain with a small number of aggregates, which use a small amount of computing time, and fine grain with more aggregates, which need more computing time and storage space, but produce on the other hand even more accurate approximations. In section 3, these topics are pointed out.

In [17] and [16], DA iterative solutions for polling systems are treated, in [16] for queueing networks with a fork-join subnet, in [9] and [4] for some generalized stochastic Petri nets, and in [18] for queueing networks with repetitive service blocking.

The obtained accuracy compares favourably with results calculated using other methods from the literature, and with exact results.

We describe the blocking networks in some detail:

Example We consider queueing networks with repetitive service blocking. A network consists of $K \in I\!N$ nodes. In node k, $k = 1, \ldots, K$, there is a single exponential server, service rate μ_k. Buffer space is available for no more than ν_k jobs. If the model is open, jobs arrive according to a Poisson process with parameter λ_k. When the service of a job in node k is finished, the job tries to go to node l with probability $t_{k,l}$, $l \in [1 : K]$. If there is no buffer space available, the job remains in node k and instantaneously receives another service (*repetitive-service with random destination policy*). In an open network, a job which is completely served in node k leaves the network with probability $t_{k,0}$. In a closed network, there are $N \in I\!N$ jobs, and we assume without loss of generality $\nu_k \leq N$, $k = 1, \ldots, K$.

In section 3, we present the analysis of these blocking networks using DA iteration. □

Secondly, we look on general product form approximations $\pi(\mathbf{z}) = \rho_{1,z_1} \cdot \ldots \cdot \rho_{C,z_C}\rho_0$, $\mathbf{z} \in \mathcal{Z}^T$, for probability distributions $\mathbf{q}$. We will show that there is always such an approximation with a small number of factors (the MED parameters ρ_{k,z_k}), given any degree of accuracy, namely an ϵ' such that $||\boldsymbol{\pi} - \mathbf{q}||_1 \leq \epsilon'$ holds. The proof is constructive: The aggregation of the product form approximation $\boldsymbol{\pi}$ and the MED parameters are stated. This is accomplished directly, not using marginal probabilities as restrictions when maximizing the entropy.

For the treatment of general product form approximations $\boldsymbol{\pi}$ we do not require that the aggregation identifies the states completely. Thus the mapping $\mathcal{I}^{-1} : \mathcal{Z} \to \mathcal{Z}^T$ is not always one-to-one: Given a state $z \in \mathcal{Z}$, and letting $\mathbf{z} = (z_1, \ldots, z_C) = \mathcal{I}^{-1}(z)$ denote the state tuple belonging to z, $z \in \mathcal{Z}_{k,z_k}$ holds for all $k = 1, \ldots, C$, but there may exist another state $z' \in \mathcal{Z}$ with the same tuple, $\mathbf{z} = \mathcal{I}^{-1}(z')$.

Here we restrict ourselves to partitions consisting of two aggregates each, $\mathcal{Z}_k = \{\mathcal{Z}_{k,0}, \mathcal{Z}_{k,1}\}$, $k = 1, \ldots, C$.

The following theorem states the existence of a general product form approximation $\boldsymbol{\pi}$, which approximates a given probability distribution $\mathbf{q}$ over $\mathcal{Z}$ accurately. In the first part of the proof, the aggregation is provided, and the MED parameters $\rho_{k,j}$, $k = 1, \ldots, C$, $j = 0, 1$, are given.

Accurate Product Form Theorem Given a probability distribution $\mathbf{q} = \big(q(1), \ldots, q(n)\big)$ and a degree ϵ of accuracy, $0 < \epsilon \leq 0.01$, there exists an approximating product form distribution $\boldsymbol{\pi} = \big(\pi(1), \ldots, \pi(n)\big)$ with

$$C \leq \widehat{C} \triangleq \left\lceil \ln\left(\frac{\ln(n/\epsilon)}{-\ln(1-2\epsilon)} + 1\right) / \ln 2 \right\rceil \tag{18}$$

partitions, each of them with two aggregates. The accuracy is

$$||\mathbf{q} - \boldsymbol{\pi}||_1 < 4.2\epsilon, \tag{19}$$

or, more precisely,

$$\begin{aligned} |q(z) - \pi(z)| &< \frac{1.1}{n}\epsilon \quad \text{if} \quad q(z) \leq \mu, \\ \frac{|q(z) - \pi(z)|}{q(z)} &< 3.1\epsilon \quad \text{if} \quad \mu \leq q(z) \leq 1, \\ |q(z) - \pi(z)| &< 2.1\epsilon \quad \text{if} \quad 1 - \epsilon \leq q(z) \leq 1, \\ z &= 1, \ldots, n, \end{aligned} \tag{20}$$

where

$$\mu \triangleq (1 - 2\epsilon)^{2^{\widehat{C}} - 1}. \tag{21}$$

Proof The proof is constructive: We give the product form distribution and the aggregation belonging to it, and afterwards we will prove the quoted statements about it.

The aggregation consists of C partitions $\mathcal{Z}_k = \{\mathcal{Z}_{k,0}, \mathcal{Z}_{k,1}\}$, each with two aggregates. Let

$$\begin{aligned} C &= \widehat{C} \quad \text{(this is preliminary, perhaps } C \text{ will decrease later on),} \\ b &= 1 - 2\epsilon, \\ \rho_{k,0} &= 1, \;\; \rho_{k,1} = b^{2^{k-1}}, \;\; k = 1, \ldots, \widehat{C}, \\ l_z &= \lfloor \ln q(z) / \ln b + 0.5 \rfloor, \end{aligned} \tag{22}$$

$$
\begin{aligned}
\tilde{\pi}_z &= \begin{cases} \mu, & \text{if} \quad q(z) \le \mu, \\ b^{l_z}, & \text{if} \quad \mu < q(z) \le 1-\epsilon, \\ 1, & \text{if} \quad q(z) < 1-\epsilon, \end{cases} && (23) \\
\rho_0 &= 1/(\tilde{\pi}_1 + \ldots + \tilde{\pi}_n), \\
\pi(z) &= \rho_0 \tilde{\pi}_z, \; z \in \mathcal{Z}. && (24)
\end{aligned}
$$

The aggregation is obtained as follows. From $\mu < q(z) \le 1-2\epsilon$, $1 \le l_z \le 2^{\widehat{C}} - 1$ can easily be shown. Therefore we may represent the integers l_z as binary numbers,

$$
l_z = l_{z,1}2^0 + l_{z,2}2^1 + \ldots + l_{z,\widehat{C}}2^{\widehat{C}-1},
$$

which defines the figures $l_{z,k} \in \{0,1\}, \; k = 1, \ldots, \widehat{C}$, uniquely. Thus we have by (23)

$$
\begin{aligned}
\tilde{\pi}_z &= \mu = \rho_{1,1} \cdot \ \ldots \ \cdot \rho_{\widehat{C},1} && \text{if} \;\; q(z) \le \mu, \\
\tilde{\pi}_z &= b^{l_{z,1} \cdot 2^0} b^{l_{z,2} \cdot 2^1} \cdot \ \ldots \ \cdot b^{l_{z,\widehat{C}} \cdot 2^{\widehat{C}-1}} \\
&= \rho_{1,l_{z,1}} \cdot \ \ldots \ \cdot \rho_{\widehat{C},l_{z,\widehat{C}}} && \text{if} \;\; \mu < q(z) \le 1-\epsilon, \\
\tilde{\pi}_z &= 1 = \rho_{1,0} \cdot \ \ldots \ \cdot \rho_{\widehat{C},0} && \text{if} \;\; 1-\epsilon < q(z). && (25)
\end{aligned}
$$

Clearly,

$$
\tilde{\pi}_z = \rho_{1,l_{z,1}} \cdot \ \ldots \ \cdot \rho_{\widehat{C},l_{z,\widehat{C}}}
$$

where $l_{z,k} \triangleq 1$ if $q(z) < \mu$, and $l_{z,k} \triangleq 0$ if $1-\epsilon < q(z), \; k = 1, \ldots, \widehat{C}$, is in accordance with (25).

From this representation, a preliminary aggregation is obtained: A state $z \in \mathcal{Z}$ belongs to the aggregate $\mathcal{Z}_{k,j}$ iff in (25) $\rho_{k,j}$ is a factor of $\tilde{\pi}_z$, i.e.

$$
\mathcal{Z}_{k,j} = \{z \in \mathcal{Z} \mid l_{z,k} = j\}, \;\; j = 0,1, \;\; k = 1, \ldots, \widehat{C}.
$$

If an aggregate $\mathcal{Z}_{k,j}$ is empty, then ρ_0 is replaced by $\rho_0\rho_{k,1-j}$, the partition $\mathcal{Z}_k$ is discarded, the numbering of the remaining partitions is adjusted to $1, \ldots, C-1$, and C is reduced by 1.

If there are two partitions $\mathcal{Z}_k$ and $\mathcal{Z}_{k'}$, for which there are numbers $j, \, j' \in \{0,1\}$ such that $\mathcal{Z}_{k,j} = \mathcal{Z}_{k',j'}$ holds, then we replace $\rho_{k,j}$ by $\rho_{k',j'}\rho_{k,j}$ and $\rho_{k,1-j}$ by $\rho_{k',1-j'}\rho_{k,1-j}$, and discard the partition $\mathcal{Z}_{k'}$ like just explained.

This was the more interesting part of the proof, namely the definition of the approximating product form distribution and of the aggregation belonging to it.

The second part is to prove the accuracy (20), from which (19) follows immediately. The arguments are straightforward, lengthy, and consist in applying some Taylor series with remainders and simple estimates. This second part of the proof is given in the appendix. $\square$

Remark The aggregation has $C \le \widehat{C}$ partitions, each of them with two aggregates. This C is also the number of factors in the approximating product form. There are C degrees of freedom, namely the MED parameters $\rho_{k,1}, \;\; k = 1, \ldots, C$ (the parameters ρ_0 are unity, and the normalization constant is determined by the MED parameters). The number $\widehat{C}$ is $O(\log\log n)$ for ϵ fixed, and $O(\log 1/\epsilon)$ for n fixed. For example, if n varies between 1000 and 10^{27}, and ϵ between 0.01 and 10^{-5}, $\widehat{C}$ is not greater than 22.

As a consequence of the theorem, for any probability distribution $\mathbf{q}$ with $n-1$ degrees of freedom – n might be large – there exists an accurate product form approximation $\boldsymbol{\pi}$ with a small number C of degrees of freedom. This is the important feature for our motivation: It is worth searching for product form approximations.

3 Practical aspects

If a Markov model is to be analyzed with DA iteration, the aggregation, i.e. the partitions of the state space, must be defined. Moreover, in the DA iteration algorithm, given the MED parameters $\rho_{k,j}$, the normalization constant G, the marginal probabilities $r_k(j)$, and the aggregate transition probabilities $\pi_k(j,l)$ must be calculated in an efficient manner. The defining equations of these quantities, (4), (3), and (7), (10) are composed of too many summands.

For the definition of the aggregation, we consider the modeled system as being composed of components k, $k = 1,\ldots,K$. Each system component k may assume a state z_k of its own state space $\mathcal{S}_k$, e.g. $\mathcal{S}_k = [0 : N_k]$, $N_k \in I\!N$. Consequently, the state of the whole system can be described by K-tuples $\mathbf{z} = (z_1,\ldots,z_K) \in \mathcal{S}_1 \times \ldots \times \mathcal{S}_K$. Clearly, the state space is a subset of this cartesian product, $\mathcal{Z}^T \subset \mathcal{S}_1 \times \ldots \times \mathcal{S}_K$.

Example In the queueing network with blocking, the system's components are the nodes. Their states are the numbers z_k of jobs in them, $k = 1,\ldots,K$. Therefore, a system's state is given by the $K-$tuple $\mathbf{z} = (z_1,\ldots,z_K)$. In an open queueing network,

$$\mathcal{Z}^T = \mathcal{S}_1 \times \ldots \times \mathcal{S}_K$$

where $\mathcal{S}_k = [0 : \nu_k]$ is the state space, and in a closed queueing network

$$\mathcal{Z}^T = \{\mathbf{z} \in \mathcal{S}_1 \times \ldots \times \mathcal{S}_K \mid z_1 + \ldots + z_K = N\}, \tag{26}$$

where

$$\mathcal{S}_k = [\tau_k : \nu_k], \quad \tau_k = \max\{0, N - \sum_{j\neq k} \nu_j\} :$$

A state τ_k in node k occurs if in no other node space is left. $\square$

In this section, we consider only aggregations which identify the states completely.

In section 1, canonical aggregation is defined as follows: There are $C = K$ partitions $\mathcal{Z}_k$, $k = 1,\ldots,K$, and the aggregates of each partition $\mathcal{Z}_k$ are defined by the states of the system component k:

$$\mathcal{Z}_{k,j} = \{z \in \mathcal{Z} \mid z_k = j\},\ j \in \mathcal{S}_k,\ k = 1,\ldots,K$$

(If z and $\mathbf{z}$ occur in the same formula, both denotations correspond: $\mathbf{z} = \mathcal{I}^{-1}(z)$).

Example In the example blocking networks, each aggregate $\mathcal{Z}_{k,j}$ consists of all system states where j jobs are in node k. $\square$

If more aggregates are applied, we call the aggregation *fine*. This can be accomplished as follows: A partition is defined by two or more system components, e.g. the partition $\mathcal{Z}_{(k,k')}$:

$$\mathcal{Z}_{(k,k'),(j,j')} = \{z \in \mathcal{Z} \mid z_k = j,\ z_{k'} = j'\},\ j \in \mathcal{S}_k,\ j' \in \mathcal{S}_{k'},\ k \neq k',\ k, k' \in [1:K]. \quad (27)$$

Example In our blocking networks, we may define aggregates according to the state of *two* nodes. Thus the dependencies between two such nodes are completely taken into account. In general, not all partitions $\mathcal{Z}_{(k_1,k_2)}$, $(k_1, k_2) \in [1:K]^2$, are considered, but only for a subset $\mathcal{K}_2 \subset [1:K]^2$ of node pairs.

Sometimes the dependencies between a certain node $k^* \in [1:K]$ and the other nodes are of particular importance, e.g. in a star topology. Then we try *center aggregation* with the partitions

$$\mathcal{Z}_{(k^*,k)},\ k = 1, \ldots, K,\ k \neq k^*,$$
$$\mathcal{K}_2 = \Big\{(k^*, k) \,\Big|\, k = 1, \ldots, K,\ k \neq k^*\Big\},$$

and the partition Z_{k^*}. Node k^* is called the *center*. In an open network, the number of aggregates is

$$(\nu_{k^*} + 1)\left(1 + \sum_{\substack{c=1\\c\neq k^*}}^{K} (\nu_c + 1)\right).$$

Sometimes the states of distinct pairs (k_1, k_2) of nodes depend strongly. Under such circumstances one can put these pairs into $\mathcal{K}_2$ and consider the according partitions $\mathcal{Z}_{(k_1,k_2)}$, $(k_1, k_2) \in \mathcal{K}_2$. If K is even, and if all nodes of the network are combined into such pairs,

$$\mathcal{K}_2 = \Big\{\big(\mathrm{per}(1), \mathrm{per}(2)\big),\ \big(\mathrm{per}(3), \mathrm{per}(4)\big), \ldots, \big(\mathrm{per}(K-1), \mathrm{per}(K)\big)\Big\}$$

holds, where $\big(\mathrm{per}(1), \ldots, \mathrm{per}(K)\big)$ is a permutation of $(1, \ldots, K)$. We call this *paired aggregation*. In an open network, the number of aggregates is

$$\sum_{(k_1,k_2)\in\mathcal{K}_2} (\nu_{k_1} + 1)(\nu_{k_2} + 1).$$

$\square$

In order to obtain a fine aggregation, also a *virtual system component* can be used. Its states $s \in \mathcal{S}$ are also a function of the system's states, $\sigma : \mathcal{Z} \to \mathcal{S}$, like the states of real components. Useful for the definition of fine grain aggregations are, for example, the number of jobs in an open queueing network, i.e. $s = z_1 + \ldots + z_K$, $\mathbf{z} \in \mathcal{Z}^T$, or the set of empty/non-empty system components, i.e.

$$\mathcal{S} = \{\mathbf{s} \in \{0,1\}^K \mid s_k = \mathrm{sign}(z_k),\ k = 1, \ldots, K,\ \mathbf{z} \in \mathcal{Z}^T\}.$$

In general,

$$\mathcal{S} = \{\sigma(z) \mid z \in \mathcal{Z}\}$$

holds.

Applying virtual system components, the aggregates may be defined by

$$\mathcal{Z}_{k,(j,s)} = \{z \in \mathcal{Z} \mid z_k = j,\ \sigma(z) = s, \},\ j \in \mathcal{S}_k,\ s \in \mathcal{S},\ k = 1, \ldots, K.$$

Another partition $\mathcal{Z}_C$, $C = K+1$, is defined solely the virtual component. Its aggregates are

$$\mathcal{Z}_{C,s} = \{z \in \mathcal{Z} \mid \sigma(z) = s\},\ s \in \mathcal{S}.$$

In this case, or similarly if the system's component C represented by the component state space $\mathcal{S}$ is a real one, the disaggregation is sometimes quite simple. We mean the case where the state space is the union of some disjoint cartesian products as follows:

$$\mathcal{Z} = \bigcup_{s \in \mathcal{S}} \mathcal{Z}_{C,s} \quad \text{with} \quad \mathcal{Z}_{C,s} = \mathcal{S}'_1(s) \times \ldots \times \mathcal{S}'_{C-1}(s) \times \{s\}$$

where the sets $\mathcal{S}'_k(s) \subset \mathcal{S}_k$ may depend on $s \in \mathcal{S}$.

Assume the marginal probabilities

$$r_k(j,s) = \mathrm{P}\{Z \in \mathcal{Z}_{k,(j,s)}\},\ j \in \mathcal{S}_k,\ s \in \mathcal{S},\ k = 1, \ldots, C-1,$$

and

$$r_C(s) = \mathrm{P}\{Z \in \mathcal{Z}_{C,s}\},\ s \in \mathcal{S},$$

are given. Consider a given value $s \in \mathcal{S}$. Because of the rectangular form of the subspace $\mathcal{Z}_{C,s}$,

$$\begin{aligned}
\mathrm{P}\{Z = z \mid Z \in \mathcal{Z}_{C,s}\} & \\
&= \mathrm{P}\{Z_1 = z_1 \mid Z \in \mathcal{Z}_{C,s}\} \cdot \ldots \cdot \mathrm{P}\{Z_{C-1} = z_{C-1} \mid Z \in \mathcal{Z}_{C,s}\} \\
&= \frac{r_1(z_1,s)}{r_C(s)} \cdot \ldots \cdot \frac{r_{C-1}(z_{C-1},s)}{r_C(s)},\quad z \in \mathcal{Z}_{C,s}
\end{aligned}$$

holds. From this, for all $z \in \mathcal{Z}_{C,s}$ and all $s \in \mathcal{S}$, in turn

$$\begin{aligned}
\mathrm{P}\{Z = z\} &= \mathrm{P}\{Z = z,\ Z \in \mathcal{Z}_{C,s}\} \\
&= \mathrm{P}\{Z = z \mid Z \in \mathcal{Z}_{C,s}\} \cdot \mathrm{P}\{Z \in \mathcal{Z}_{C,s}\} \\
&= \frac{r_1(z_1,s)}{r_C(s)} \cdot \ldots \cdot \frac{r_{C-1}(z_{C-1},s)}{r_C(s)} \cdot r_C(s),\quad s \in \mathcal{S},
\end{aligned}$$

hold. Thus

$$\begin{aligned}
\rho_{k,(j,s)} &= \frac{r_k(j,s)}{r_C(s)},\quad j \in \mathcal{S}'_k(s),\ k = 1, \ldots, C-1, \\
\rho_{C,s} &= r_C(s),\quad s \in \mathcal{S},
\end{aligned} \tag{28}$$

are the MED parameters.

Example In our blocking network, if center aggregation is applied, let $k^* = K = C$, i.e. $s \in \mathcal{S}$ are the states of the center. □

In general, the aggregation cannot be chosen as we did in the proof of the accurate product form theorem, as
– the exact solution is not known and
– the obtained aggregation could be of such a kind that the normalization constant G or the aggregate transition probabilities $\pi_k(j,l)$ cannot be calculated efficiently.

Instead, nearly dependent and independent components of the system are identified. Dependent components should be combined, in order to define together a partition as indicated in (27). Vice versa, independent components may define different partitions: If the state space is rectangular, and the components are completely independent, then the according DAΠ solution is exact and the transition probability matrix of the Markov chain is a Kronecker product, see e.g. [3]. Bärk [1] does some research on these topics.

If the state space is rectangular, $\mathcal{Z}^T = \mathcal{S}_1 \times \ldots \times \mathcal{S}_C$, e.g. in an open queueing network, the normalization constant

$$G = \sum_{\mathbf{z} \in \mathcal{Z}^T} \prod_{k=1}^{C} \rho_{k,z_k}$$

is efficiently calculated by

$$G = \left(\sum_{j \in \mathcal{S}_1} \rho_{1,j} \right) \cdot \ldots \cdot \left(\sum_{j \in \mathcal{S}_C} \rho_{C,j} \right). \tag{29}$$

This technique can easily be generalized if the state space is the union of some disjoint cartesian products.

We call a state space

$$\mathcal{Z}^T = \{ \mathbf{z} \in \mathcal{S}_1 \times \ldots \times \mathcal{S}_C \mid z_1 + \ldots + z_C = N \}, \tag{30}$$

where $\mathcal{S}_k = [u_k : o_k]$, a *closed form* state space, because closed queueing networks have a state space of this kind. In this case, the normalization constant is efficiently calculated using convolution. Let $\boldsymbol{\rho}_k$, $k = 1, \ldots, C$, denote the vectors $(\rho_{u_k}, \ldots, \rho_{o_k})$ of DME parameters. The normalization constant G is given by the N^{th} component of the convolution vector of all vectors $\boldsymbol{\rho}_k$, $k = 1, \ldots, C$:

$$G = \left(\boldsymbol{\rho}_1 \otimes \ldots \otimes \boldsymbol{\rho}_C \right)_N. \tag{31}$$

There are more general state spaces, for which both simplification techniques are applied.

The efficient calculation of a marginal probability $r_k(j)$ (3) can be done accordingly: In formula (4) for the normalization constant G, $\rho_{k,l}$ is replaced for all $l \neq j$ by zero, and the sum is divided by G.

More general, probabilities $\mathrm{P}\{\mathbf{Z} \in \mathcal{X}\} = \sum_{\mathbf{z} \in \mathcal{X}} \rho_{1,z_1} \cdot \ldots \cdot \rho_{C,z_C} / G$ where $\mathbf{Z} = (Z_1, \ldots, Z_C)$, $\mathcal{X} = \mathcal{X}_1 \times \ldots \times \mathcal{X}_C$, $\mathcal{X}_k \subset \mathcal{S}_k$, $k = 1, \ldots, C$, can efficiently be calculated as follows:

If the state space is rectangular,

$$\mathrm{P}\{\mathbf{Z} \in \mathcal{X}\} = \frac{1}{G} \prod_{k=1}^{C} \mathrm{P}\{Z_k \in \mathcal{X}_k\} \tag{32}$$

holds. If the state space is closed form (30),

$$\mathrm{P}\{\mathbf{Z} \in \mathcal{X} \cap \mathcal{Z}^T\} = (\mathbf{x}_1 \otimes \ldots \otimes \mathbf{x}_C)_N / G \tag{33}$$

holds where $\mathbf{x}_k = (x_{k,0}, \ldots, x_{k,N_k})$,

$$x_{k,j} \triangleq \begin{cases} \rho_{k,j}, & \text{if } j \in \mathcal{X}_k, \\ 0 & \text{otherwise}, \end{cases} \quad j \in \mathcal{S}_k,\ k = 1, \ldots, C.$$

The proof of the convolution formula (33) is straightforward.

Now we turn to the problem of efficiently calculating the aggregate transition probabilities. In general, this task needs $\mathrm{O}(n^2)$ operations. This is why we restrict the considered class of state transition matrices.

We assume that the Markov chain transitions from state $\mathbf{y}$ into state $\mathbf{z}$, $\mathbf{y} \neq \mathbf{z}$, $\mathbf{y}, \mathbf{z} \in \mathcal{Z}^T$, can be classified as follows: Each state transition belongs to one and only one of the *transition classes*, which are numbered $1, \ldots, \bar{\nu}$.

A transition $\mathbf{y}$ into $\mathbf{z}$ of transition class ν may occur only if $\mathbf{y} \in \mathcal{U}_\nu$, where $\mathcal{U}_\nu$ denotes the *source states set* of this class. The *destination state function* $u_\nu : \mathcal{Z}^T \to \mathcal{Z}^T$ defines the new states, i.e. a transition of class ν leads from state $\mathbf{y} \in \mathcal{U}_\nu$ into state $u_\nu(\mathbf{y})$. u_ν consists of component functions $u'_{\nu,k}$,

$$u_\nu = \Big(u'_{\nu,1}(\mathbf{y}), \ldots, u'_{\nu,C}(\mathbf{y}) \Big). \tag{34}$$

Transitions of class ν occur with probability α_ν.

In order to reduce the complexity of the calculation of the aggregate transition probabilities, we restrict the sets of source states to

$$\mathcal{U}_\nu = \mathcal{U}_{\nu,1} \times \ \ldots \ \times \mathcal{U}_{\nu,C} \bigcap \mathcal{Z}^T \tag{35}$$

where

$$\mathcal{U}_{\nu,k} \subset \mathcal{S}_k,\ k = 1, \ldots, C,\ \nu = 1, \ldots, \bar{\nu},$$

and we assume, that the destination state functions $u'_{\nu,k}$ depend only on y_k. In this case, there are functions $u_{\nu,k} : \mathcal{U}_{\nu,k} \to \mathcal{S}_k$, for which $u_{\nu,k}(y_k) = u'_{\nu,k}(\mathbf{y})$, $\mathbf{y} \in \mathcal{U}_\nu$, holds. Therefore we may consider

$$u_\nu(\mathbf{y}) = \Big(u_{\nu,1}(y_1), \ldots, u_{\nu,C}(y_C) \Big) \tag{36}$$

instead of (34).

The introduction of transition classes not only helps to obtain the lower complexity, but also serves to structure the design of the Markov models.

Example In an open queueing network, the system's state may change due to an arrival at node k from outside. The source state set is

$$\mathcal{U}_\nu = \{ \mathbf{z} \in \mathcal{Z}^T \mid z_k < \nu_k \}.$$

That means, there is space left in node k for an arriving job. The destination state function is given by

$$\begin{aligned} u_{\nu,k}(z_k) &= z_k + 1, \\ u_{\nu,i}(z_i) &= z_i, \qquad \text{if } i \neq k. \end{aligned}$$

The transition probability is

$$\alpha_\nu = \lambda_k \Delta.$$

Δ is the randomization constant: Initially, the Markov chain is continuous in time. Let $\mathbf{Q}$ denote its generator matrix. We consider the discrete time Markov chain with the transition probability matrix $\mathbf{P} = \mathbf{Q}\Delta + \mathbf{I}$ instead, where $\Delta = 0.99/\max_x |q_{x,x}|$. It has the same steady state probabilities as the continuous time Markov chain. By the factor 0.99, numerical problems are avoided, which may occur due to rounding errors.

Other state transitions take place when a job of node k leaves the network. The according transition class ν is defined as follows.
The source state set is

$$\mathcal{U}_\nu = \{\mathbf{z} \in \mathcal{Z}^T \mid z_k > 0\},$$

the destination state function

$$\begin{aligned} u_{\nu,k}(z_k) &= z_k - 1, \\ u_{\nu,i}(z_i) &= z_i, \qquad \text{if } i \neq k, \end{aligned}$$

and the transition probability

$$\alpha_\nu = \mu_k(1 - t_{k,1} - \ldots - t_{k,K})\Delta.$$

The third kind of state transitions occur when a job leaves node k for node i, $i \neq k$. In node k, there must be at least one job, and in node i must be buffer space available. Therefore, the source state set is

$$\mathcal{U}_\nu = \{\mathbf{z} \in \mathcal{Z}^T \mid z_k > 0,\ z_i < \nu_i\}.$$

The states of the nodes k and i change,

$$\begin{aligned} u_{\nu,k}(z_k) &= z_k - 1, \\ u_{\nu,i}(z_i) &= z_i + 1, \\ u_{\nu,m}(z_m) &= z_m, \qquad \text{if } m \neq i,\ k, \end{aligned}$$

and the transition probability is

$$\alpha_\nu = \mu_k t_{k,i} \Delta.$$

□

Applying the transition classes, the state transition probabilities of the Markov chain can be expressed as follows:

$$\begin{aligned} p(\mathbf{y},\mathbf{z}) &= \sum_{\nu=1}^{\bar{\nu}} I(\mathbf{y} \in \mathcal{U}_\nu) I\Big(u_\nu(\mathbf{y}) = \mathbf{z}\Big)\alpha_\nu, \ \ \mathbf{y} \neq \mathbf{z}, \ \ \mathbf{y},\mathbf{z} \in \mathcal{Z}^T, \\ p(\mathbf{y},\mathbf{y}) &= 1 - \sum_{\nu=1}^{\bar{\nu}} I(\mathbf{y} \in \mathcal{U}_\nu)\alpha_\nu, \ \ \mathbf{y} \in \mathcal{Z}^T, \end{aligned} \tag{37}$$

where $I : \{\text{true, false}\} \to \{0, 1\}$ denotes the indicator function.

From the definition (7), (10) of the aggregation transition probabilities, one obtains for $j \neq l$, using the expressions (37) for the transition probabilities, the sets $\mathcal{U}_\nu$ of source states (35), and the destination state functions u_ν (36), in turn

$$\begin{aligned}
&\pi_k(j,l)r_k(j) \\
&= \sum_{\substack{\mathbf{y}\in\mathcal{Z}^T\\ y_k=j}} \sum_{\substack{\mathbf{z}\in\mathcal{Z}^T\\ z_k=l}} p(\mathbf{y},\mathbf{z})p(\mathbf{y}) \\
&= \sum_{\substack{\mathbf{y}\in\mathcal{Z}^T\\ y_k=j}} \sum_{\substack{\mathbf{z}\in\mathcal{Z}^T\\ z_k=l}} \sum_{\nu=1}^{\bar\nu} I(\mathbf{y}\in\mathcal{U}_\nu) I\Big(u_\nu(\mathbf{y})=\mathbf{z}\Big)\alpha_\nu p(\mathbf{y}) \\
&= \sum_{\nu=1}^{\bar\nu}\alpha_\nu \sum_{\substack{\mathbf{y}\in\mathcal{U}_\nu\\ y_k=j}} I\Big(u_{\nu,k}(y_k)=l\Big)p(\mathbf{y}) \\
&= \sum_{\nu=1}^{\bar\nu}\begin{cases}\alpha_\nu \mathrm{P}\{\mathbf{Z}\in\mathcal{U}_{\nu,1}\times\ldots\times\mathcal{U}_{\nu,k-1}\times\{j\}\times\mathcal{U}_{\nu,k+1}\times\ldots\times\mathcal{U}_{\nu,C}\}, \\ \qquad \text{if } j\in\mathcal{U}_{\nu,k},\ u_{\nu,k}(j)=l, \\ 0, \qquad \text{otherwise.}\end{cases}
\end{aligned} \tag{38}$$

The probability in formula (38) can be evaluated efficiently using the formulae (32) or (33), respectively.

The aggregate transition probabilities $\pi_k(j,j)$ are efficiently calculated using (12).

Example Here, we consider only closed queueing networks with blocking. The only transitions, which occur, are of the third kind: a job changes from a node to another. Observing the actual source state sets, destination state functions, and transition probabilities, one obtains the aggregat transition probabilities, using (38),

$$\pi_k(j,l)r_k(j) = \begin{cases}\sum_{k\neq i}\mu_k t_{k,i}\Delta \mathrm{P}\{Z_k=j, Z_i<\nu_i\} & \text{if } j>\tau_k,\ l=j-1, \\ \sum_{k\neq i}\mu_i t_{i,k}\Delta \mathrm{P}\{Z_k=j, Z_i>\tau_i\} & \text{if } j<\nu_k,\ l=j+1.\end{cases}$$

Clearly, $\mathrm{P}\{Z_k=j, Z_i<\nu_i\}$ equals $\mathrm{P}\{Z_k=j\}-\mathrm{P}\{Z_k=j, Z_i=\nu_i\}$ (observe that the last probability is 0 if $(j+\nu_i)\notin[\tau_k+\tau_i : N]$).

If $|j-l|>1$, the aggregate transition probability is 0. □

At last we mention some numerical results for open queueing networks with repetitive service blocking.

Example A tandem network: $K=4$ nodes, capacity $\nu=3$ in all nodes,
arrivals at node 1, $\lambda_1=1$, service rates $\mu_1=\mu_2=\mu_4=1$, $\mu_3=1,2,\ldots,20$,
transfer probabilities $t_{1,2}=t_{2,3}=t_{3,4}=t_{4,0}=1$.

The maximum absolute error of the macro probabilities is 0.13 with canonical aggregation, and 0.02 with center aggregation. It occurs when $\mu_3=20$. □

Example A split topology: $K=4$ nodes, $\nu_1=5$, $\nu_2=3$, $\nu_3=2$, $\nu_4=3$,
arrivals at node 1, $\lambda_1=0.1,0.2,\ldots,1.9$,
transfer probabilities $t_{1,2}=0.2$, $t_{1,3}=0.3$, $t_{1,4}=0.5$, $t_{2,0}=t_{3,0}=t_{4,0}=1$,
service rates $\mu_1=1$, $\mu_2=0.6$, $\mu_3=0.4$, $\mu_4=0.3$.

The maximum absolute error of the macro probabilities is 0.022 when canonical aggregation is used. It occurs for $\lambda_1=0.5$. When center aggregation is used, it is 0.002 and occurs for $\lambda_1=0.6$. □

Example A merge topology: $K = 4$ nodes, $\nu_1 = 5$, $\nu_2 = 3$, $\nu_3 = 2$, $\nu_4 = 3$,
arrivals at nodes 2,3, and 4: $\lambda_2 = 1$, $\lambda_3 = 0.5$, $\lambda_4 = 0.5$,
transfer probabilities $t_{1,0} = t_{2,1} = t_{3,1} = t_{4,1} = 1$,
service rates $\mu_2 = \mu_3 = 0.5$, $\mu_4 = 1$, $\mu_1 = 1.1, 1.2, \ldots, 2.5$.

The maximum absolute error of the macro probabilities is 0.016 when canonical aggregation is used. It occurs for $\mu_1 = 2.0$. When center aggregation is used, it is 0.0015 and occurs for $\mu_1 = 2.2$. □

Example A symmetrical network:
$K = 4$ nodes, capacities $\nu_1 = \nu_3 = 3$, $\nu_2 = \nu_4 = 2$,
arrivals at nodes 1 and 3, $\lambda_1 = \lambda_3 = 0.01, 0.02, \ldots, 0.2$,
transfer probabilities $t_{1,2} = t_{3,4} = 0.8$, $t_{1,3} = t_{3,1} = 0.2$, $t_{3,0} = t_{4,0} = 1$,
service rates $\mu_1 = \mu_3 = 0.5$, $\mu_2 = \mu_4 = 0.1$.

The maximum absolute error is 0.11 when canonical aggregation is used. It occurs for $\lambda_1 = \lambda_3 = 0.08$. When paired aggregation is used, $\mathcal{K}_2 = \{(1,2), (3,4)\}$, it is 0.011 and also occurs for $\lambda_1 = \lambda_3 = 0.08$. □

Example The tandem topology was analyzed, but here with 9 nodes, $\mu_3 = 1$, and the capacity $\nu = 9$ in all nodes, which has 10^9 system states. On an Atari 1040ST, the computing time for DA iteration with canonical aggregation was 238 seconds. This demonstrates that Markov models with a really large state space can be treated with the DA iteration technique. □

Conclusion

Product form approximations for the steady state probabilities of Markov chains and a technique for calculating them are proposed, which are suitable for huge state spaces. Unfortunately, we have not yet found error estimates or a proof of convergence of the iteration. But we solved many models using DA iteration, among others polling, fork join, stochastic Petri net, and blocking models, and we found very accurate results; the iteration did never diverge. Moreover, by choosing aggregations with more aggregates – fine grain aggregation – the accuracy can be substantially ameliorated.

The just mentioned open questions, different time scales, the development of a tool, and time-dependent probabilities are topics of future research. The latter can be obtained via DA iteration: If the Markov chain is discrete in time, the DA solutions $\mathbf{p}^{(t)}$, $t = 1, 2, \ldots$, are approximate time-dependent state probabilities. If the Markov chain is continuous in time, randomization can be applied onto the DA solutions in order to obtain approximate time-dependent solutions.

Appendix

Proof of the accurate product form theorem, part 2.

We begin estimating the number $\mu = b^{2^{\widehat{C}}-1}$: From

$$\widehat{C} \geq \ln\left(\frac{\ln(n/\epsilon)}{-\ln b} + 1\right) / \ln 2,$$

according to (18), we obtain

$$b^{2^{\widehat{C}}-1} \leq \epsilon/n,$$

and using the definition of μ,

$$\mu \leq \epsilon/n. \tag{39}$$

Case $q(z) \leq \mu$: From $\widetilde{\pi}_z = \mu$, $|\widetilde{\pi}_z - q(z)| \leq \mu$ follows, and using (39), one gets

$$|\widetilde{\pi}_z - q(z)| \leq \epsilon/n. \tag{40}$$

Case $\mu < q(z) \leq 1 - \epsilon$: Let

$$r_z \triangleq \frac{q(z) - \widetilde{\pi}_z}{q(z)} = 1 - \widetilde{\pi}_z/q(z) = 1 - b^{l_z - c_z}, \tag{41}$$

where $c_z \triangleq \ln q(z)/\ln b$, i.e. $q(z) = b^{c_z}$. $l_z = \lfloor c_z + 0.5 \rfloor$ follows using (22), and in turn

$$c_z - 0.5 < l_z \leq c_z + 0.5, \quad -0.5 < l_z - c_z \leq 0.5, \quad b^{-0.5} > b^{l_z - c_z} \geq b^{0.5},$$

and using (41)

$$1 - b^{-0.5} < r_z \leq 1 - b^{0.5}. \tag{42}$$

Considering the Taylor series

$$b^{-0.5} = (1-2\epsilon)^{-0.5} = 1 + \epsilon + \frac{3}{2}(1-\theta)^{-5/2}\epsilon^2 \quad \text{where} \quad 0 \leq \theta \leq 2\epsilon,$$

and observing $0 \leq \epsilon \leq 0.1$,

$$b^{-0.5} \leq 1 + 1.0157\ldots\epsilon$$

holds, and accordingly

$$b^{0.5} = (1-2\epsilon)^{0.5} = 1 - \epsilon - \frac{1}{2}(1-\theta)^{-3/2}\epsilon^2 \quad \text{where} \quad 0 \leq \theta \leq 2\epsilon,$$

$$b^{0.5} \geq 1 - 1.0051\ldots\epsilon.$$

Applying this to (42),

$$-1.0157\ldots\epsilon < r_z \leq 1.0051\ldots\epsilon,$$

$$|r_z| < \hat{r} \triangleq 1.016\epsilon, \tag{43}$$

and using (41),

$$|q(z) - \widetilde{\pi}_z| < \hat{r}q(z) \tag{44}$$

follow.

Case $1 - \epsilon < q(z)$: Here $\widetilde{\pi}_z$ is unity, and therefore

$$|\widetilde{\pi}_z - q(z)| < \epsilon \tag{45}$$

holds.

For calculating the approximation errors, we need bounds for the normalization constant $\rho_0 = 1/(\widetilde{\pi}_1 + \ldots + \widetilde{\pi}_n)$:

Case where there is no state $z \in \mathcal{Z}$ with $q(z) > 1 - \epsilon$:

$$\begin{aligned}
\rho_0^{-1} &= \sum_{\substack{z\in\mathcal{Z}\\ q(z)\le\mu}} \widetilde{\pi}_z + \sum_{\substack{z\in\mathcal{Z}\\ q(z)>\mu}} q(z)(1-r_z) \quad \text{(using (41))} \\
&= \sum_{z\in\mathcal{Z}} q(z) + \sum_{\substack{z\in\mathcal{Z}\\ q(z)\le\mu}} \Big(\widetilde{\pi}_z - q(z)\Big) - \sum_{\substack{z\in\mathcal{Z}\\ q(z)>\mu}} q(z) r_z \qquad (46)\\
&\geq 1 - \sum_{\substack{z\in\mathcal{Z}\\ q(z)\le\mu}} |\widetilde{\pi}_z - q(z)| - \widehat{r} \sum_{\substack{z\in\mathcal{Z}\\ q(z)>\mu}} q(z) \\
&\geq 1 - \epsilon - \widehat{r} \quad \text{(using (40) and (43)).} \qquad (47)
\end{aligned}$$

Observing the Taylor series

$$\frac{1}{1-\epsilon-\widehat{r}} = 1 + \epsilon + \widehat{r} + \left(\frac{1}{1+\theta}\right)^3 (\epsilon+\widehat{r})^2 \quad \text{where} \quad -\epsilon - \widehat{r} \le \theta \le 0,$$

$$\begin{aligned}
\rho_0 &\leq 1 + \left(1 + \frac{\epsilon+\widehat{r}}{(1-\epsilon-\widehat{r})^3}\right)(\epsilon+\widehat{r}) \\
&\leq 1 + 2.059\ldots\epsilon \qquad (48)
\end{aligned}$$

follows.

Vice versa, from (46) one gets

$$\begin{aligned}
\rho_0^{-1} &\leq 1 + \sum_{\substack{z\in\mathcal{Z}\\ q(z)\le\mu}} |\widetilde{\pi}_z - q(z)| + \widehat{r} \sum_{\substack{z\in\mathcal{Z}\\ \mu<q(z)}} q(z) \\
&\leq 1 + \epsilon + \widehat{r}. \qquad (49)
\end{aligned}$$

Observing the Taylor series

$$\frac{1}{1+\epsilon+\widehat{r}} = 1 - \epsilon - \widehat{r} + \left(\frac{1}{1+\theta}\right)^3 (\epsilon+\widehat{r})^2 \quad \text{where} \quad 0 \le \theta \le \epsilon + \widehat{r},$$

using (49), one obtains

$$\rho_0 \geq 1 - \epsilon - \widehat{r} = 1 - 2.016\epsilon.$$

Case where there is a state $z' \in \mathcal{Z}$ with $q_{z'} > 1 - \epsilon$: Here, $\widetilde{\pi}_{z'} = 1$.

$$\begin{aligned}
\rho_0^{-1} &= \sum_{\substack{z\in\mathcal{Z}\\ q(z)\le\mu}} \widetilde{\pi}_z + \sum_{\substack{z\in\mathcal{Z}\\ \mu<q(z)\le 1-\epsilon}} q(z)(1-r_z) + 1 \qquad (50)\\
&\leq 1 + \epsilon + (1+\widehat{r}) \sum_{\substack{z\in\mathcal{Z}\\ \mu<q(z)\le 1-\epsilon}} q(z) \\
&\leq 1 + 2\epsilon + \epsilon\widehat{r} \qquad (51)
\end{aligned}$$

using (39), (23), and $\sum_{\substack{z\in\mathcal{Z}\\ \mu<q(z)\le 1-\epsilon}} q(z) \le 1 - q_{z'} \le \epsilon$.

Observing the Taylor series

$$\frac{1}{1+2\epsilon+\epsilon\widehat{r}} = 1 - 2\epsilon - \epsilon\widehat{r} + \left(\frac{1}{1+\theta}\right)^3 (2\epsilon - \epsilon\widehat{r})^2 \quad \text{where} \quad 0 \le \theta \le 2\epsilon + \epsilon\widehat{r},$$

one obtains from (51)

$$\rho_0 > 1 - 2.010\ldots\epsilon. \tag{52}$$

Vice versa, from (50) one gets

$$\begin{aligned}\rho_0^{-1} &\geq 1 + \sum_{\substack{z\in\mathcal{Z}\\ q(z)\leq\mu}} \widetilde{\pi}_z + (1-\widehat{r}) \sum_{\substack{z\in\mathcal{Z}\\ \mu<q(z)\leq 1-\epsilon}} q(z) \\ &\geq 1. \end{aligned} \tag{53}$$

In each of the two cases,

$$1 - 2.016\epsilon \leq \rho_0 \leq 1 + 2.059\ldots\epsilon \tag{54}$$

holds, and if there is a state $z' \in \mathcal{Z}$ with $q_{z'} > 1-\epsilon$

$$1 - 2.010\ldots\epsilon < \rho_0 \leq 1 \tag{55}$$

holds.

Now we are prepared to give the approximation errors:

Case $q(z) \leq \mu$:

$$\begin{aligned}\pi(z) &= \widetilde{\pi}_z\rho_0 = \mu\rho_0 \leq \frac{\epsilon}{n}(1 + 2.059\ldots\epsilon) \\ &\leq 1.020\ldots\epsilon/n,\end{aligned}$$

using (25), (39), (54). Clearly,

$$|\widetilde{\pi}_z - q(z)| < 1.1\epsilon/n$$

holds.

Case $\mu < q(z) \leq 1-\epsilon$: In turn, using (24), (41)

$$\begin{aligned}\frac{q(z)-\pi(z)}{q(z)} &= \frac{q(z)-\widetilde{\pi}_z\rho_0}{q(z)} = \frac{q(z)-\widetilde{\pi}_z-\widetilde{\pi}_z(\rho_0-1)}{q(z)} \\ &= r_z - \frac{\widetilde{\pi}_z}{q(z)}(\rho_0-1) = r_z - (1-r_z)(\rho_0-1)\end{aligned}$$

holds, and with (43), (54) one gets

$$\begin{aligned}\frac{|q(z)-\pi(z)|}{q(z)} &\leq \widehat{r} + (1+\widehat{r})2.059\ldots\epsilon \\ &< 3.1\epsilon.\end{aligned}$$

Case $q(z) > 1-\epsilon$: From $\pi(z) = \widetilde{\pi}_z\rho_0 = \rho_0$, and using (55),

$$|q(z) - \pi(z)| \leq 2.1\epsilon$$

follows. This completes the proof of (20).

From (20), (19) follows directly. □

Acknowledgements

The author would like to thank B. Bärk who has been engaged successfully in the DA project for years, W. Everling and M. Heckler for valuable discussions, and the students who worked at the DA idea.

References

[1] B. Bärk, Aggregationsmethoden für Markovketten, 1996.

[2] O. J. Boxma, Workloads and waiting times in single-server systems with multiple customer classes, *Queueing Systems* **5** (1989) 185–214.

[3] G. Ciardo and K. S. Trivedi, A decomposition approach for stochastic reward net models, *Performance Evaluation* **18** (1) (1993) 37–59.

[4] J. F. Gierse, Zustandsraum-Aggregation für fast-unabhängige stochastische Petrinetze, Master's thesis, Universität Bonn, 1995.

[5] T. Heinrichs, B. Bärk, and J. C. Strelen, Wartezeiten für Pollingsysteme mittels numerischer Modelle, in: B. Walke and O. Spaniol, eds., *Messung, Modellierung und Bewertung von Rechen- und Kommunikationssystemen* (Springer, Berlin, 1993) 173–185.

[6] A. Jennings and W. J. Stewart, Simultaneous iteration for partial eigensolution of real matrices, *J. Inst. Maths Applics* **15** (1975) 351–361.

[7] V. Jonas, Iterative Analyse von Blockiernetzen mit alternierender Aggregation und Disaggregation, Master's thesis, Universität Bonn, 1992.

[8] A. M. Kagan, J. V. Linnik, and C. R. Rao, *Characterization Problems in Mathematical Statistics* (Wiley, New York, 1973).

[9] R. Nitzgen, Aggregations-Disaggregationsverfahren für verallgemeinerte stochastische Petrinetze, Master's thesis, Universität Bonn, 1993.

[10] J. M. Ortega and W. C. Rheinboldt, *Iterative Solution of Nonlinear Equations in Several Variables* (Academic Press, New York, London, 1970).

[11] B. Philippe, Y. Saad, and W. J. Stewart, Numerical methods in Markov chain modeling, *Operations Reseach* **40** (6) (1992).

[12] P. J. Schweitzer, A survey of aggregation-disaggregation in large Markov chains, in: W. J. Stewart, ed., *Numerical Solution of Markov Chains* (Marcel Dekker, New York, Basel, Hong Kong, 1991) 63–88.

[13] J. E. Shore and R. W. Johnson, Axiomatic derivation of the principle of maximum entropy and the principle of minimum cross-entropy, *IEEE Trans. Inform. Th.* **26** (1) (1980) 26–37.

[14] W. J. Stewart, A comparison of numerical techniques in Markov modeling, *Comm. ACM* **21** (2) (1978) 144–152.

[15] J. C. Strelen, A fast simultaneous iteration technique for the analysis of Markov chains, Interner Bericht II/89/1, Institut für Informatik, Universität Bonn, 1989.

[16] J. C. Strelen, Iterative Analyse von Markov-Modellen mit alternierender Aggregation und Disaggregation, in: A. Lehmann and F. Lehmann, eds., *Messung, Modellierung und Bewertung von Rechensystemen IFB 286* (Springer, Berlin, 1991) 320–336.

[17] J. C. Strelen and B. Bärk, An approach to the numerical analysis of multiple-queue cyclic service systems, in: G. Stiege and J. S. Lie, eds., *Messung, Modellierung und Bewertung von Rechensystemen und Netzen IFB 218* (Springer, Berlin, 1989) 75–88.

[18] J. C. Strelen, B. Bärk, J. Becker, and V. Jonas, Analysis of queueing networks with blocking using a new aggregation technique, *Submitted for publication*.