A Matrix Pattern Compliant Strong Stochastic Bound

A. Busic and J.M. Fourneau
PRiSM, Université de Versailles Saint-Quentin en Yvelines,
45 Av. des Etats Unis, 78000 Versailles, France

Abstract

Stochastic bounds are a promising method to analyze QoS requirements. Indeed it is sufficient to prove that a bound of the real performance satisfies the guarantee. However, the time and space complexity issues are not well understood so far. We propose a new algorithm to derive a strong stochastic bound of a Markov chain, using a matrix pattern specifing the structural properties a bounding matrix should comply with. Thus we can obtain a simpler Markov chain bounding for which the numerical computation of the steady-state solution is easier.

1. Introduction

Despite considerable works (see for instance Stewart's book [16] and the recent LAA issue [9] devoted to this subject), the numerical analysis of Markov chains is still a very difficult problem when the state space is too large or the eigenvalues badly distributed. Fortunately enough, while modeling high speed networks, it is often sufficient to satisfy the requirements for the Quality of Service (QoS) we expect. Exact values of the performance indices are not necessary in this case and bounding some reward functions is often sufficient. Stochastic bounds are in general obtained with sample path arguments and coupling theorems applied to models transformation (see [14] for an example on Fair Queueing delays comparison based on sample-paths). Here, we only consider Markov chains and algorithmic operations on stochastic matrices. Indeed, in the last decade, many algorithmic techniques have been designed to model complex systems with large Markov chains (Stochastic Automata Network [15], Superposition of Stochastic Petri Nets [2], Stochastic Process Algebra [7]). So there are now several well-founded methods to model complex systems using Markov chains with large state space but these models can still not be solved.

The key idea of our methodology is to design a new chain such that the reward functions will be upper or lower bounds of the exact reward functions. This new chain is a simplified model of the former one to reduce the complexity of the numerical analysis. These bounds are based on stochastic orderings applied to Markov processes (see Stoyan [12] and other references therein).

The fundamental algorithm to obtain a strong stochastic ("st") bound was developped by Vincent [1]. But it has several drawbacks: the bounding matrix may be reducible and the time and space complexity for the storage and the numerical resolution are not considered. Unfortunately they can be very bad and even worse than the original problem. Thus we present here a new algorithm based on a matrix pattern to insure irreducibility, and control the storage and the resolution.

The remaining of the paper is as follows: in section 2, we introduce briefly "st" bounds and Vincent's algorithm. Section 3 is devoted to the matrix pattern approach and in section 4 we show two structures which can be represented by patterns and which simplify the computation.

2. Strong Stochastic Bounds

For the sake of simplicity, we restrict ourselves to Discrete Time Markov Chains (DTMC) with finite state space $E = \{1, \ldots, n\}$ but continuous-time models can be considered after uniformization. In the following, n will denote the size of matrix P and $P_{i,*}$ will refer to row i of P. First, we give a brief overview on "st" ordering for Markov chains and we obtain a set of inequalities to imply bounds which gives us the basic algorithm proposed by Vincent and Abu-Amsha [1].

2.1. A brief overview

Following [12], we define the strong stochastic ordering by the set of non-decreasing functions.

Definition 1 *Let X and Y be random variables taking values on a totally ordered space. Then X is said to be less than Y in the strong stochastic sense, that is, $X <_{st} Y$ if and only if $E[f(X)] \leq E[f(Y)]$ for all non decreasing functions f whenever the expectations exist.*

Property 1 *(Characterization for a finite state space.) If X and Y take values on $E = \{1, 2, \ldots, n\}$ with p and q as probability distribution vectors, then $X <_{st} Y$ if and only if $\sum_{j=k}^{n} p_j \le \sum_{j=k}^{n} q_j$ for $k = 1, 2, \ldots, n$.*

Important performance indices such as average population, loss rates or tail probabilities are non decreasing functions. Therefore, bounds on the distribution imply bounds on these performance indices as well. It is known for a long time that monotonicity [8] and comparability of the one step transition probability matrices of time-homogeneous MCs yield sufficient conditions for their stochastic comparison. This is the fundamental result we use in our algorithms. First let us define the st-comparability of the matrix and the st-monotonicity.

Definition 2 *Let P and Q be two stochastic matrices. $P <_{st} Q$ if and only if $P_{i,*} <_{st} Q_{i,*}$ for all i.*

Definition 3 *Let P be a stochastic matrix, P is st-monotone if and only if for all i, $j > i$, we have $P_{i,*} <_{st} P_{j,*}$*

Theorem 1 *Let $X(t)$ and $Y(t)$ be two DTMC and P and Q be their respective stochastic matrices. Then $X(t) <_{st} Y(t), t > 0$, if:*

- *$X(0) <_{st} Y(0)$,*
- *st-monotonicity of at least one of the matrices holds,*
- *st-comparability of the matrices holds, i.e. $P <_{st} Q$.*

Thus, assuming that P is not monotone and in order to obtain a strong upper bound monotone matrix Q for P, the previous theorem leads to the following set of inequalities on elements of Q :

$$\begin{cases} \sum_{k=j}^{n} P_{i,k} \le \sum_{k=j}^{n} Q_{i,k} & \forall\, i, j \\ \sum_{k=j}^{n} Q_{i,k} \le \sum_{k=j}^{n} Q_{i+1,k} & \forall\, i, j \end{cases} \quad (1)$$

2.2. Vincent's Algorithm

It is possible to derive a set of equalities, instead of inequalities. These equalities provide, once they have been ordered (in increasing order for i and in decreasing order for j in system (2)), a constructive way to design a stochastic matrix which yields a stochastic bound.

$$\begin{cases} \sum_{k=j}^{n} Q_{1,k} = \sum_{k=j}^{n} P_{1,k}, \ \forall j, \\ \sum_{k=j}^{n} Q_{i+1,k} = \max(\sum_{k=j}^{n} Q_{i,k}, \sum_{k=j}^{n} P_{i+1,k}), \ \forall\, i, j \end{cases} \quad (2)$$

The following algorithm [1] constructs an st-monotone upper bounding DTMC Q for a given DTMC P. For the sake of simplicity, we use a full matrix representation for P and Q. The sparse matrix version is straightforward. Note that due to the ordering of the indices, the summations $\sum_{j=l}^{n} q_{i-1,j}$ and $\sum_{j=l+1}^{n} q_{i,j}$ are already computed when we need them. However, we let them appear as summations to show the relations with inequalities (1).

Algorithm 1 Construction of a st-monotone upper bounding DTMC Q:
$q_{1,n} = p_{1,n}$;
for *i=2* **to** *n* **do** $q_{i,n} = \max(q_{i-1,n}, p_{i,n})$;
for *j=n-1* **to** *1* **do**
  $q_{1,j} = p_{1,j}$;
  for *i=2* **to** *n* **do**
    $q_{i,j} = \max(\sum_{k=j}^{n} q_{i-1,k}, \sum_{k=j}^{n} p_{i,k}) - \sum_{k=j+1}^{n} q_{i,k}$;
  end
end

It may happen that matrix $v(P)$ computed by Algorithm 1 is not irreducible, even if P is irreducible. Indeed due to the subtraction operation in inner loops, some elements of $v(P)$ may be zero even if the elements with the same indices in P are positive. To fix this problem, a new algorithm has been derived in [5] which tries not to delete transitions and to add subdiagonal entries. A necessary and sufficient condition on P has been obtained to ensure that the bound is irreducible.

However, this solution does not fix the storage problem. Indeed, it may be possible that the bound has many more positive elements than the matrix P and it may be even completely filled. It is easy to build a matrix P with $3n$ positive elements resulting in a completely filled bounding matrix. Furthermore, the Algorithm 1 builds a matrix which is, in general, as difficult as P to analyze.

3. Matrix Pattern Approach

We use the two sets of constraints of system (1) and add some structural properties to simplify the storage and the resolution of the bounding matrix.

3.1. Boolean pattern

Those additional constraints can be defined by a boolean pattern T imposing the exact graph structure of the bounding transition matrix.

Definition 4 *A boolean pattern T for a matrix P is a boolean matrix (i.e. $t_{i,j} \in \{0, 1\}, \forall i, j$) of the same size as P. A bounding matrix Q complies with a boolean pattern T for P if and only if it satisfies:*

$$\forall i, j,\ q_{i,j} > 0 \text{ if and only if } t_{i,j} = 1. \quad (3)$$

The following algorithm constructs an st-monotone upper bounding DTMC Q compliant with a boolean pattern T for a given DTMC P. If such a bounding does not exist, the algorithm sends a message of non-compatibility of T and P.

Algorithm 2 Construction of the st-monotone upper bounding DTMC compliant with a boolean pattern T:

$\epsilon \in (0,1)$;
for $i = 1$ **to** n **do**
  last = -1;
  for $j = n$ **to** 1 **do**
    if $(i > 1$ *and* $j < n)$ **then**
    $temp = \max(0, \max(\sum_{k=j}^{n} q_{i-1,k}, \sum_{k=j}^{n} p_{i,k}) - \sum_{j=k+1}^{n} q_{i,k})$;
    else if $(i > 1)$ **then** $temp = \max(p_{i,n}, q_{i-1,n})$;
    else if $(j < n)$ **then**
    $temp = \max(0, \sum_{k=j}^{n} p_{1,k} - \sum_{k=j+1}^{n} q_{1,k})$;
    else $temp = p_{1,n}$;
    switch $t_{i,j}$ **do**
      case 0
        $q_{i,j} = 0$;
        if $temp > 0$ **then**
          if $last > 0$ **then** $q_{i,last} = q_{i,last} + temp$;
          else STOP : non-compatible!;
      case 1
        last = j;
        if $temp > 0$ **then** $q_{i,j} = temp$;
        else if $\sum_{k=j+1}^{n} q_{1,k} < 1$ **then**
        $q_{i,j} = \epsilon \times (1 - \sum_{k=j+1}^{n} q_{i,k})$;
        else STOP : non-compatible!;
    end
  end
end

3.2. Generalized pattern

We might not always want to impose such a strict structure of a bound. Moreover, it should be possible to use some information on the transitions in the initial matrix P. See for instance the contraints for the irreducibility of a bound [5]. We introduce other letters in the pattern alphabet in order to increase the structure defining possibilities.

We call an alphabet a set of symbols with rules associated to each symbol. For instance, for the boolean pattern the alphbet is $\mathcal{A} = \{0, 1\}$ with the rules: $R(0) : t_{i,j} = 0 \Rightarrow q_{i,j} = 0; R(1) : t_{i,j} = 1 \Rightarrow q_{i,j} > 0$.

Definition 5 *A pattern T for a matrix P is a matrix of the same size as P which elements are the symbols from an alphabet set $\mathcal{A}$. A bounding matrix Q complies with a pattern T for P if and only if it satisfies the rules associated to each pattern element.*

In the following we will use the alphabet set $\mathcal{A} = \{0, 1, \star, w, s\}$ with the associated rules:

letter	rule
0	$t_{i,j} = 0 \Rightarrow q_{i,j} = 0$
1	$t_{i,j} = 1 \Rightarrow q_{i,j} > 0$
$\star$	no additional constraints
s	$t_{i,j} = s$ & $p_{i,j} > 0 \Rightarrow q_{i,j} > 0$
w	$t_{i,j} = w$ & $p_{i,j} > 0 \Rightarrow q_{i,j} > 0$ if possible

For example, the pattern T corresponding to the IMSUB algorithm yielding an irreducible upper bound [5] is given by

$$T_{i,j} = \begin{cases} 1, & \text{if } j = i - 1, \\ s, & \text{if } j > i, \\ w, & \text{elsewhere.} \end{cases} \quad (4)$$

To obtain an algorithm deriving an st-monotone upper bounding DTMC compliant with a given pattern T for a DTMC P, it is sufficient to add the case blocks corresponding to a new letters $\star, s, w$ to the algorithm 2.

case $\star$
  last = j;
  $q_{i,j} = temp$;
case s, w
  last = j;
  if $temp > 0$ **then** $q_{i,j} = temp$;
  else
    $q_{i,j} = 0$;
    if $p_{i,j} > 0$ **then**
      if $\sum_{k=j+1}^{n} q_{1,k} < 1$ **then**
      $q_{i,j} = \epsilon \times (1 - \sum_{k=j+1}^{n} q_{i,k})$;
      else if $t_{i,j} = s$ **then** STOP : non-compatible!;

We say that the transition matrix P of an DTMC and a pattern T are compatible if and only if there exist at least one st-monotone bounding DTMC which transition matrix complies with the pattern T. It is possible to show that our algorithm (with the alphabet described above) results by such a bounding DTMC for each pattern T compatible with a given DTMC.

We have checked our algorithm (sparse version implementation) using Single Input Pattern (see 4.1) against IMSUB Algorithm described in [5] on the exemple of the loss rate in a RR queue with Pushout Algorithm presented in [5]. The computation times are quite similar when the number of blocks is small.

Buffer	Size	Imsub Patt. (4)	SI Pattern 10 blocks	100 blocks	IMSUB [5]
500	251502	12.8s	16.4s	77.8s	18.3s
1000	1003002	52.5s	66.5s	315.7s	72.9s

4. Examples of Pattern

In the following we illustrate this principle with two matrix patterns associated to simple resolution methods: the single-input macro chains and the stochastic complement.

4.1. Single Input Macro State Markov Chain

Feinberg and Chiu [3] have studied chains divided into macro-states where the transition entering a macro-state must go through exactly one node. This node is denoted as the input node of the macro-state. They have developed an algorithm to efficiently compute the steady-state distribution by decomposition. It consists of the resolution of the macro-state in isolation and the analysis of the chain reduced to input nodes. Unlike ordinary lumpability, the assumptions of the theorem are based on the graph of the transitions and do not take into account the real transition rates.

We assume that for every macro state, the input state is the last state of the macro state. Then the pattern for this structure has the following block form:

$$T = \begin{bmatrix} A_1 & B_{12} & \cdots & B_{1K} \\ B_{21} & A_2 & \cdots & B_{2K} \\ \vdots & \vdots & \ddots & \vdots \\ B_{K1} & B_{K2} & \cdots & A_K \end{bmatrix}, \; B_{ij} = \begin{bmatrix} 0 & \cdots & 0 & \star \\ \vdots & \ddots & \vdots & \vdots \\ 0 & \cdots & 0 & \star \end{bmatrix},$$

$\forall i, j, i \neq j$, where K denotes the number of the macro-states. There is no additional constraints on the structure of the blocks A_i corresponding to the transitions inside the macro-state i, i.e. $\forall i,\ A_i(k,l) = \star, \forall k, l$. The blocks corresponding to the transitions between the macro-states have the non zero elements only in the last column.

This structure have been used by several authors even if their proofs of comparison are usually based on sample-path theorem [6, 13, 10].

4.2. Partition and stochastic complement

The stochastic complement was initially proposed by Meyer in [11]. Here we propose a completely different idea based on an easy resolution of the stochastic complement due to Quessette [4]. Let us consider a block decomposition of Q: $\begin{pmatrix} A & B \\ C & D \end{pmatrix}$, where A and D are square matrices. We know that $I - D$ is not singular if Q is not reducible [11]. We decompose π into two components π_0 and π_1 to obtain the stochastic complement formulation for the steady-state equation: $\begin{cases} \pi_0\, R = 0 \\ \pi_0\, r = 1 \\ \pi_1 = \pi_0\, H \end{cases}$ where $H = B(I - D)^{-1}$, $R = I - A - HC$ and $r = e_0 + He_1$.

Following Quessette [4], we chose to partition the states such that matrix D is upper triangular. It should be clear that this partition is not mandatory for the theory of stochastic complement. However it simplifies the computation of H.

One possible pattern for this decomposition, imposing an irreducible upper bound is:

$$T^A_{i,j} = \begin{cases} 1, & \text{if } j = i - 1, \\ \star, & \text{elsewhere;} \end{cases} \quad T^B_{i,j} = \begin{cases} 1, & \text{if } i = j = 1, \\ \star, & \text{elsewhere;} \end{cases}$$

$$T^C_{i,j} = \begin{cases} 1, & \text{if } i = n_1 \\ & \text{and } j = n_0, \\ \star, & \text{elsewhere;} \end{cases} \quad T^D_{i,j} = \begin{cases} 1, & \text{if } j = i + 1, \\ 0, & \text{if } j < i, \\ \star, & \text{elsewhere.} \end{cases}$$

5. Conclusions

Strong stochastic bounds are not limited to sample-path proofs. It is now possible to compute bounds of the steady-state distribution directly from the chain. This approach may be specially useful for high speed networks modeling where the performance requirements are thresholds. However, the knowledge of the model characteristics is essential to choose an adequate pattern and to obtain the bounds of satisfying quality for the measurements of QoS. Generalizations to other orderings or to computation of transient measures are still important problems to analyse.

References

[1] O. Abu-Amsha and J.-M. Vincent. An algorithm to bound functionals of markov chains with large state space. In *4th INFORMS Conference on Telecommunications*, 1998.

[2] S. Donatelli. Superposed generalized stochastic petri nets: Definition and efficient solution. In *Proceedings of the 15th International Conference on Application and Theory of Petri Nets*, pages 258–277. Springer-Verlag, 1994.

[3] B. N. Feinberg and S. S. Shiu. A method to calculate steady-state distributions of large markov chains by aggregating states. *Oper. Res.*, 35(2):282–290, 1987.

[4] J. Fourneau and F. Quessete. Graphs and stochastic automata networks. In *Proceedings of the 2nd International Workshop on the Numerical Solution of Markov Chains*, 1995.

[5] J.-M. Fourneau, M. Lecoz, and F. Quessette. Algorithms for an irreducible and lumpable strong stochastic bound. *Linear Algebra and its Applications*, 386:167–185, 2004.

[6] L. Golubchik and J. C. S. Lui. Bounding of performance measures for a threshold-based queueing system with hysteresis. In *Proceedings of the 1997 ACM SIGMETRICS international conference on Measurement and modeling of computer systems*, pages 147–157. ACM Press, 1997.

[7] J. Hillston and L. Kloul. An efficient kronecker representation for pepa models. In *PAPM'2001*, 2001.

[8] J. Keilson and A. Kester. Monotone matrices and monotone markov processes. *Stochastic Processes and Their Applications*, 5:231–241, 1977.

[9] *Linear Algebra and its Applications*, volume 386. Elsevier, 2004.

[10] J. C. S. Lui, R. R. Muntz, and D. Towsley. Computing performance bounds of fork-join parallel programs under a multiprocessing environment. *IEEE Trans. Parallel Distrib. Syst.*, 9(3):295–311, 1998.

[11] C. D. Meyer. Stochastic complementation, uncoupling markov chains, and the theory of nearly reducible systems. *SIAM Rev.*, 31(2):240–272, 1989.

[12] A. Muller and D. Stoyan. *Comparison Methods for Stochastic Models and Risks*. John Wiley & Sons, Ltd, 2002.

[13] R. R. Muntz, J. C. S. Lui, and D. Towsley. Bounding the mean response time of the minimum expected delay routing policy: An algorithmic approach. *IEEE Trans. Comput.*, 44(12):1371–1382, 1995.

[14] N. Pekergin. Stochastic delay bounds on fair queueing algorithms. In *Proc. of INFOCOM'99*, pages 1212–1220, 1999.

[15] B. Plateau. On the stochastic structure of parallelism and synchronization models for distributed algorithms. In *Proc. of the SIGMETRICS Conference on Measurement and Modeling of Computer Systems*, pages 147–154, 1985.

[16] W. J. Stewart. *Introduction to the Numerical Solution of Markov Chains*. Princeton University Press, 1994.