Decompositional Analysis of Kronecker Structured Markov Chains

Yujuan Bao İlker N. Bozkurt Tuğrul Dayar
Xiaobai Sun Kishor S. Trivedi

August 20, 2007

Contact Info

- Yujuan Bao
  e-mail: yujuan.bao@google.com
- İlker N. Bozkurt, Tuğrul Dayar
  Department of Computer Engineering, Bilkent University,
  e-mail: {bozkurti,tugrul}@cs.bilkent.edu.tr
- Xiaobai Sun
  Department of Computer Science
  Duke University,
  e-mail: xiaobai@cs.duke.edu
- Kishor S. Trivedi
  Department of Electrical and Computer Engineering
  Duke University,
  e-mail: kst@ee.duke.edu

Introduction

- A decompositional iterative method for the steady-state analysis of Kronecker structured Markov chains.
- The Markovian system, which is formed by a composition of subsystems using the Kronecker sum operator for local transitions and the Kronecker product operator for synchronized transitions [1], is assumed to have irreducible subsystem matrices associated with local transitions.
- In contrast with [2], the interactions among subsystems, which are captured by synchronized transitions, are not assumed to be weak.

Related Work

- Ciardo and Trivedi's decomposition approach when the submodels are nearly independent.[2]
- Iterative method of Tomek and Trivedi [3]
- Based on decomposition, product form solution, approximative.
- Interaction response is assumed to be weak.
- Bucholz's iterative decomposition and aggregation methods [4, 5]
- Both approaches produce approximate results.
- Method in [4] is adaptive, states with high probability are considered in detail, and states with low probability are aggregated.

An Example Model

- Two submodels coupled by a snychronizing transition (recovery from total failure) with rate μ
- k-th subsystem has n_k redundant components, fails with rate λ_k, locally repaired independently with rate μ_k
- W synchronization matrix, two nonzero elements $W(n_1 n_2, 0) = \mu$, $W(n_1 n_2, n_1 n_2) = -\mu$
- $$Q_{(k)} = \begin{pmatrix} -\lambda_k & \lambda_k & & \\ \mu_k & -(\lambda_k + \mu_k) & \lambda_k & \\ & \ddots & \ddots & \ddots \\ & & \mu_k & -\mu_k \end{pmatrix}, k = 1, 2$$
- $Q = Q_{(1)} \oplus Q_{(2)} + W$

Steady State Equations

- Let π_{ij} be the probability of the model being in state (i,j) and let $\pi_{i(k)}$ submodel k being in state $i, k = 1, 2$.
- Steady-state balance equations in these local and global state variables can then be represented in the form of local equations for each submodel k, k = 1,2.
- $0 = -\lambda_k \pi_{0(k)} + \mu_k \pi_{1(k)} + \mu \pi_{n_1 n_2}$
- $0 = -(\lambda_k + \mu_k)\pi_{i(k)} + \lambda_k \pi_{i-1(k)} + \mu_k \pi_{i+1(k)}, i = 1 : n_k - 2,$
- $0 = \lambda_k \pi_{n_{k-1}(k)} - \mu_k \pi_{n_k(k)} - \mu \pi_{n_1 n_2}$

Exploiting the local equations

- $\mu = 0$, independent subsystems, Kronecker product of the local solutions give the solution
- Nearly independent submodels, Kronecker product solution gives an approximate solution
- Iterative scheme of Tomek and Trivedi [3] is used to improve approximation
- The coupling variable $\pi_{n_1 n_2}$ is replaced by $\pi_{n_1(1)} * \pi_{n_2(2)}$ in [2] (synchronization replacement)
- Refine the local solution of one submodel while holding fixed the iterative solution of other submodels.

Exploiting the local equations (cont.)

- The goodness of the approximation depends on the difference $\mu(\pi_{n_1 n_2} - \pi_{n_1(1)}\pi_{n_2(2)})$
- Idea: Calculate numerically significant deviation from the product form solution and use it with synchronizing transitions.
- $\pi = \pi_{(1)} \otimes \pi_{(2)} - y$
- $\pi_{(k)} Q_{(k)} = v_k(\pi), k = 1...2$
- Vectors v_k describe the global transitions
- For our small model
  $v_{(1)}(\pi) = (\mu\pi_{n_1 n_2}, 0, \ldots, 0, -\mu\pi_{n_1 n_2})$
  $v_{(2)}(\pi) = (\mu\pi_{n_1 n_2}, 0, \ldots, 0, -\mu\pi_{n_1 n_2})$
  with $\pi_{n_1 n_2} = \pi_{n_1(1)}\pi_{n_2(2)} - y_{n_1 n_2}$

Iterative Algorithm

- How to compute the global deviation? :
  $\pi Q = 0, \pi = \pi_{(1)} \otimes \pi_{(2)} - y$
  $\Rightarrow$yQ = $(\pi_{(1)} \otimes \pi_{(2)})Q$
- Let $Q = U - L$ be a split of the generator matrix Q, U is the upper triangular portion, L has the rest
- Local state response eqution
  $\pi_{(k)} Q_{(k)} = v_k(\pi), k = 1...K$
- Global deviation equations
  $yU = yL + \otimes_{k=1}^{K} \pi_{(k)} Q$
- Use above equations in an iterative method.

Iterative Algorithm

- 1.Initial step: set $y^{[0]} = 0, \pi^{[0]} = 0$.
- 2.Local iteration(s): $\pi_{(k)}^{[i+1]} Q_k = v_k(\pi^{[i]}), k = 1...K$
- 3.Global iteration(s): $y^{[i+1]} U = y^{[i]} L + \otimes_{k=1}^{K} \pi_{(k)}^{[i+1]} Q$
- 4.Normalization and termination check: $\pi = \otimes_{k=1}^{K} \pi^{(k)} - y$ and normalize π such that $\pi e = 1$. Exit if the current iterate meets certain prescribed termination criteria; otherwise return to step 2.

Numerical Experiments
Problems

- Experiments are performed on 3 problems which are extensions of the given simple model.
- Failure rate $\lambda_k = 0.4$ and local repair rate $\mu_k = 0.3$
- Problem A and problem B have 3 subsystems, each subsystem has 10 states. There is one global transition.
- Problem A: $(9, 9, 9) \rightarrow (0, 0, 0)$ with rate μ_a
  Problem B: $(3, 0, 0) \rightarrow (1, 1, 1)$ with rate μ_b
- Problem C has 4 subsystems, each subsystem has 20 states. One global transition.

Numerical Experiments
Setting

- Platform: Intel Core2Duo processor with 4GB main memory running Linux.
- Proposed method is compared to Power method,Gauss-Seidel, BICGSTAB and GMRES.
- For GMRES, Krylov subpace of dimension 20 is used.
- Stopping criterion: a tolerance of 10^{-10} on the norm of the residual or maximum number of iterations of 5000

Numerical Experiments
Results

Table: ProblemA (rate == 0.1)

Method	Iterations	Residual	Cpu Time(secs)
Proposed	260	6.8e-11	6.0e-06
Power	760	7.4e-11	3.0e-06
GaussSeidel	320	9.7e-11	4.0e-06
GMRES(20)	95	1.6e-11	0.00e+0
BICGSTAB	72	4.2e-11	0.00e+0

Numerical Experiments
Results

Table: ProblemA (rate == 1)

Method	Iterations	Residual	Cpu Time(secs)
Proposed	200	6.5e-11	4.0e-06
Power	580	8.6e-11	2.0e-06
GaussSeidel	220	3.6e-11	3.0e-06
GMRES(20)	100	1.9e-11	0.00e+0
BICGSTAB	74	3.5e-11	0.00e+0

Numerical Experiments
Results

Table: ProblemB (rate == 0.1)

Method	Iterations	Residual	Cpu Time(secs)
Proposed	70	7.2e-11	1.0e-06
Power	840	9.8e-11	3.0e-06
GaussSeidel	340	7.5e-11	5.0e-06
GMRES(20)	99	1.6e-11	1.0e-06
BICGSTAB	80	8.5e-11	0.00e+0

Numerical Experiments
Results

Table: ProblemB (rate == 1)

Method	Iterations	Residual	Cpu Time(secs)
Proposed	110	9.5e-11	2.0e-06
Power	840	9.8e-11	3.0e-06
GaussSeidel	340	7.5e-11	4.0e-06
GMRES(20)	100	9.5e-11	1.0e-06
BICGSTAB	82	9.5e-11	0.00e+0

Numerical Experiments
Results

Table: ProblemC (rate == 0.1)

Method	Iterations	Residual	Cpu Time(secs)
Proposed	10	2.2e-12	3.4e-05
Power	2740	9.7e-11	1.8e-03
GaussSeidel	1080	6.6e-11	2.1e-03
GMRES(20)	800	6.6e-11	1.2e-03
BICGSTAB	164	1.4e-11	1.7e-04

Numerical Experiments
Results

Table: ProblemC (rate == 5)

Method	Iterations	Residual	Cpu Time(secs)
Proposed	10	1.8e-11	3.4e-05
Power	5000	1.1e-08	3.4e-03
GaussSeidel	1080	9.3e-11	2.1e-03
GMRES(20)	1019	3.1e-11	1.6e-03
BICGSTAB	142	9.2e-11	1.5e-04

Conclusion
Summary

- An iterative method for structured Markov Chains based on decomposition is presented.
- Currently works for systems whose state space is equal the product state space of the subsystems and whose local transition matrices are irreducible.
  (SAN like systems or HMMs with one macrostate)
- Comparable performance against some known iterative solution methods.
- GMRES and BICGSTAB perform better in most of the cases but they require more memory.

Conclusion
Future Work

- A thorough comparison of the solution method to state of the art iterative solvers on other problems.
- Handle the case of reducible subsystems.
- Extend the solution method to multiple macrostates if possible.
- Identify for which kinds of problems the method is superior to other methods.

- Tugrul Dayar. Analyzing Markov chains based on Kronecker products. *MAM 2006: Markov Anniversary Meeting, Eds. A. N. Langville, W. J. Stewart, Boson Books, Raleigh, North Carolina, USA, 2006, pp. 279–300.*
- G.Ciardo and K.S.Trivedi. A decomposition approach for stochastic reward net models. *Performance Evaluation,18:37-59, 1993*
- L.A.Tomek and K.S.Trivedi. Fixed point iteration in availability modelling. *Proceedings of the 5th International GI/ITG/GMA Conference on Fault-Tolerant Computing Systems, pages 229-240, 1991*
- Peter Buchholz. Adaptive Decomposition and Approximation For The Analysis of Stochastic Petri Nets *Performance Evaluation,56:23-52, 2004*


Peter Buchholz. Iterative Decomposition and Aggregation of Labeled GSPNs *International Conference on the Application and Theory of Petri Nets 1998, LNCS 1420: 226-245, 1998*