STATE SPACE DECOMPOSITION FOR LARGE MARKOV CHAINS

Maria Rieders

Northwestern University, Evanston, IL

Extended Abstract

This paper discusses various approaches for decomposing large Markov chains in a way that facilitates the use of aggregation type algorithms and increases the efficiency of such methods. For a Markov chain defined on state space \( \mathcal{N} = \{1, \ldots, N\} \) governed by transition probability matrix (t.p.m.) \( P \), we are interested in finding the stationary distribution \( \pi \), satisfying \( \pi^T = \pi^T P \), with \( \pi > 0, \pi^T e = 1 \), where \( e \) is the vector containing all ones. Aggregation disaggregation (A/D) algorithms are based on a decomposition of the state space \( \mathcal{N} \) into smaller groups of states \( \mathcal{N} = \bigcup_{m=1}^{M} L(m); \quad L(m) \cap L(n) = \emptyset, \text{ for } m \neq n \).

For an approximation \( \pi_1 \) of \( \pi \), one defines two mappings \( Y: \mathbb{R}^N \rightarrow \mathbb{R}^M \) (typically based on \( \pi_1 \)) and \( E: \mathbb{R}^M \rightarrow \mathbb{R}^N \), and an aggregate matrix \( A = Y P E \).

In the aggregation step one finds a probability vector \( \gamma \) that solves the aggregated system \( \gamma^T = \gamma^T A \). In the disaggregation step, one assigns a conditional probability vector \( y_{L(m)} \) to the states in lump \( L(m) \). This may be achieved by constructing a new Markov chain on \( L(m) \) representing flow within \( L(m) \) as well as flow to and from the other sets. A new approximation of \( \pi \) is then obtained by setting \( \pi_2|_{L(m)} = \gamma_m y_{L(m)} \). A/D algorithms proceed iteratively, until convergence is reached. Examples of A/D methods for Markov chains include Takahashi’s algorithm [7] and the replacement process algorithm of Sumita and Rieders [6].

Structural properties such as lumpability or near complete decomposability (NCD) and the choice of partition play a significant role in the performance of A/D algorithms. A/D methods generally perform best when the decomposition of the state space is aligned with the structure of the chain. In the following, we outline several heuristic schemes for automated state space decomposition.
Dynamic Decomposition

In the context of Markov decision processes, Bertsekas and Castaño [1] have proposed to adjust the state space decomposition dynamically as the algorithm proceeds. We outline here the work of Rieders and Hoyer [4] who have developed and implemented a similar scheme for Gauss-Seidel type A/D algorithms for Markov chains. Starting with an arbitrary initial decomposition, states are regrouped after several iterations based on estimates of the residual error terms. For two consecutive approximations $\pi_1$ and $\pi_2$ of the stationary distribution $\pi$, we define the error vectors $\epsilon_1 = \pi_1 - \pi$, and $\epsilon_2 = \pi_2 - \pi$. A result due to Haviv [2] shows that $\epsilon_2^T = \epsilon_1^T (I - \Pi)(I - P\Pi)^\#$, where $I$ denotes the identity matrix, $\Pi = E Y$, and superscript $\#$ stands for the Drazin inverse. The heuristic developed by Rieders and Hoyer uses an analogous result for Gauss-Seidel type algorithms. In this approach, one groups states together such that $\epsilon_1^T (I - \Pi) \epsilon_1$ becomes small. Approximating $\epsilon_1$ by $\pi_2 - \pi_1$, this results in assigning states $j$ to the same subset for which the relative errors $\epsilon_{1;j}/\pi_{2;j}$ are similar.

Extensive numerical experiments with this dynamic decomposition scheme resulted in the following observations: Implementation of this heuristic can dramatically improve the performance of A/D type algorithms, especially when little is known about the structure of the chain or when the initial decomposition does not reflect this structure. The heuristic should only be evoked after several iterations; the work by Rieders and Hoyer [4] has several concrete suggestions on when to regroup the state space.

Graph Theoretic Decomposition

For NCD Markov chains, one wants to find a partition of the state space $\mathcal{L} = \{L(1), \ldots, L(M)\}$ such that $\|P_{L(m) L(n)}\| = O(1)$, for $m = 1, \ldots, M$ and $\|P_{L(m) L(n)}\| = O(\epsilon)$, for all $m \neq n$, where $\epsilon$ is a sufficiently small positive number. Stewart and Wu [5] formulate the problem of finding the subsystems of an NCD Markov chain as the problem of finding the strongly connected subsets in a directed graph. This graph is created based on the transition probabilities, ignoring all probabilities that are less than a given threshold. In [3], Hsu and Rieders improve upon this technique by proposing to employ estimates of the ergodic flow between states $i$ and $j$; i.e. replacing $P_{ij}$ by $C_{ij} = \tilde{\pi}_i P_{ij}$ in the above procedures. Preprocessing of large Markov chains based on such heuristics is computationally inexpensive and is rewarded by fast convergence of A/D methods (see [5]).