A combinatorial aggregation algorithm for computing the stationary distribution of a large Markov chain (Extended abstract)

Anna Gambin[1]* and Piotr Pokarowski[2]

[1] Institute of Informatics, Warsaw University
Banacha 2, 02-097 Warsaw, Poland
email: aniag@mimuw.edu.pl
[2] Institute of Applied Mathematics, Warsaw University
Banacha 2, 02-097 Warsaw, Poland
email: pokar@mimuw.edu.pl

Abstract. A new aggregation algorithm for computing the stationary distribution of a large Markov chain is proposed. This algorithm is attractive when the state space of Markov chain is large enough so that the direct and iterative methods are inefficient. It is based on grouping the states of a Markov chain in such a way that the probability of changing the state inside the group is of greater order of magnitude than interactions between groups. The correctness of the combinatorial aggregation is justified by the method of *forest expansions* developed recently in [14, 15]. In contrast to existing methods our approach is based on combinatorial and graph-theoretic framework and can be seen as an algorithmization of famous Matrix Tree Theorem. The general method is illustrated by an example of computing the stationary distribution. We establish also some preliminary results on the complexity of our algorithm. Numerical experiments on several benchmark examples show the potential applicability of the algorithm in real life problems.

1 Introduction

When considering complex systems, e.g. communication networks, the usual way to compute their important parameters (like throughput) is to study a probabilistic markovian model instead of the system itself. The usual task is to compute some of its characteristics (like stationary distribution), corresponding to the parameters of a modeled system. To this aim an appropriate system of linear equation must be solved.

Our attention is restricted to finite discrete-time Markov chains (although presented approach is valid also for continuous time). Such a Markov chain over a state space $\mathcal{S}$ is usually represented by a **transition probability matrix P** of order n, where n is the number of states in $\mathcal{S}$. The (i, j)-th element of $\mathbf{P}$, denoted $p_{i,j}$, is the one-step transition probability of going from state i to state j.

In what follows, boldface capital letters (e.g. $\mathbf{P}$) denote matrices, boldface lowercase letters (e.g. $\boldsymbol{\pi}$) denote column vectors, italic lowercase and uppercase letters (e.g. a) denote scalars and italic letters (e.g. $\mathcal{S}$) denote sets.

* This work was partially supported by the KBN grant 8 T11C 039 15