Aggregation strategies for large semi-Markov processes

Marcel C. Guenther Nicholas J. Dingle
Jeremy T. Bradley William J. Knottenbelt

Department of Computing, Imperial College London
180 Queen's Gate, London SW7 2BZ, United Kingdom
Email: {mcg05,njd200,jb,wjk}@doc.ic.ac.uk

Abstract High-level semi-Markov modelling paradigms (such as semi-Markov stochastic Petri nets and process algebras) are used to capture realistic performance models of computer and communication systems but have the drawback of generating huge underlying semi-Markov state spaces. Extraction of measures such as steady-state probabilities and passage-time distributions relies on sparse matrix representation for very large transition matrices.

Previous studies have shown that exact state-by-state aggregation of semi-Markov processes can be applied to reduce the number of states but that this comes at the cost of creating additional transitions between the remaining states. Our paper addresses this issue by presenting the concept of state space partitioning for aggregation. Partitioning the state space entails creating a number of non-intersecting subsets that span the entire state space. In contrast to previous algorithms that perform state-by-state aggregation on the unpartitioned graph, our new aggregation algorithm works on a partition-by-partition basis which allows more space-efficient aggregation.

The aggregation of partitions can be done in one of two ways. The first way is to use exact state-by-state aggregation to aggregate each individual state in a partition. We introduce different partitioning methods for this purpose. Furthermore we discuss partition sorting methods that determine the order in which we aggregate partitions. The order of partition aggregation has a significant impact on the connectivity of the aggregate state space, and thus the density of the transition matrix.

Once a partition is selected for aggregation, we have the further problem of determining the order in which the states should be aggregated within that partition. To address this, we present a new state sorting algorithm, which takes into account the exact number of new transitions that are created when aggregating a particular state. This technique is preferable to existing sorting methods which only approximate the number of newly created transitions.

A second partition aggregation approach is *atomic partition aggregation* which aggregates an entire partition in one go. In atomic aggregation, we introduce a technique derived from passage-time analysis to collapse a whole partition into a small number of semi-Markov transitions.

Most partitionings produced by existing graph partitioners are not suitable for use with our atomic partition aggregation techniques, and we therefore present a new deterministic partitioning method which we term *barrier partitioning*. We show that barrier partitioning is capable of splitting large semi-Markov models into two equally sized partitions such that first passage-time analysis can be performed using 50% less memory than existing algorithms, without compromising speed.