Performance Analysis of Dynamic Network Processes

Eli Upfal*
Brown University

Abstract

This tutorial covers various approaches for modeling and analyzing dynamic processes in networks. Modeling the dynamic performance as a stochastic process, we apply tools from discrete and continues time Markov processes theory, renewal theory and queueing theory to analyze the long term, steady state, performance of the processes. Non-stochastic approaches include adversarial queueing theory, and game theory techniques.

1. Introduction

Rigorous analysis of the dynamic performance of computer processes, in particular network processes, is one of the most challenging current goals in theory of computation. Past research in theoretical computer science has focused mainly on static computation problems. In static computation the input is known at the start of the computation and the goal is to minimize the number of steps till the process terminates with a correct output. Many important processes in today's computing are dynamic processes, whereby input is continuously injected to the system, and the algorithm (which is not supposed to terminate at all) is measured by its long term (steady state) performance. Examples of dynamic computer processes include: contention resolution protocols; routing and communication algorithms; and load balancing protocols.

The primary performance measure of a dynamic processes is its *stability* conditions. Roughly speaking, a system is stable if in the long run the number of new arriving requests is no larger than the number of requests processed by the system. The goal here is to characterize the (most general) input conditions (deterministic or stochastic) under which the system is stable. When a system is stable we are also interested in its speed or efficiency, measured by the (maximum or expected) time to satisfy a request.

Most of the tutorial focuses on stochastic analysis of dynamic processes. In this approach we model the process as an infinite stochastic process, where the probabilistic space is defined by the distribution of input requests and/or by the random steps of the protocol. Worst case analysis rarely gives an interesting insight on the actual performance of a dynamic process. A worst case adversary can generate extremely hard sequences of requests, and the performance of the algorithm on these "pathological" cases does not accurately represent the efficiency of the algorithm. To offset the affect of rare cases it is useful to analyze the performance of dynamic systems under some stochastic assumptions on the stream of inputs. Such assumptions are more realistic in the dynamic setting, in particular when requests are originated by a number of independent processors, than in the case of static analysis. The stochastic process that generates the stream of requests might be stationary, periodic, or even bursty. The goal is to obtain results that are valid under the weakest set of assumptions.

Stochastic analysis of dynamic processes builds on the rich theory of stochastic processes, in particular the theory Markov chains, more general stationary processes, and queueing theory. However, in many cases one needs to develop new tools to address the specific problems pose by computer related processes, which are discrete and involved complicated dependency conditions.

The drawback of the stochastic approach is that it requires some statistical assumptions about the stochastic process that generates the request stream. We breifly discuss an alternative approach, *adversarial (or deterministic) queueing theory* in which the stochastic assumptions are replaced by deterministic restrictions on the requests stream.

Another interesting approach employs game theoretical techniques to characterizing the system performance. In this approach individual proceeses in the network are modeled as players in a noncolaborative game, and the network steady state performance is characterized by an equialibium of the corresponding game.

*Dept. of Computer Science, Brown University, Providence, RI 02912. E-mail: `eli@cs.brown.edu`. Work supported in part by NSF grants CCR-0121154, and DMI-0121495.

Proceedings of the 44th Annual IEEE Symposium on Foundations of Computer Science (FOCS'03)
0272-5428/03 $17.00