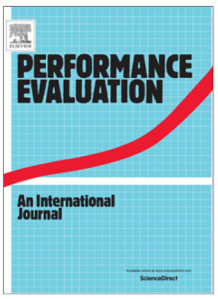

Power control in saturated fork-join queueing systems

Andrea Marin, Sabina Rossi

PII: S0166-5316(17)30087-1
DOI: http://dx.doi.org/10.1016/j.peva.2017.08.008
Reference: PEVA 1926

To appear in: *Performance Evaluation*

Please cite this article as: A. Marin, S. Rossi, Power control in saturated fork-join queueing systems, *Performance Evaluation* (2017), http://dx.doi.org/10.1016/j.peva.2017.08.008

Power control in saturated fork-join queueing systems

Andrea Marin[a,*], Sabina Rossi[a]

[a] *Università Ca' Foscari Venezia, Italy*

Abstract

The analysis of fork-join queueing systems has played an important role for the performance evaluation of distributed systems where parallel computations associated with the same job are carried out and a job is considered served only when all the parallel tasks it consists of are served and then joined. The fork-join nodes that we consider consist of $K \geq 2$ parallel servers each of which is equipped with two First Come First Served queues, namely the service-queue and the join-queue. The former stores the tasks waiting for being served while the latter stores the served tasks waiting to be joined. Under heavy load conditions, the variance of the service times associated with the tasks tends to cause long join-queue lengths. In this work, we propose an algorithm to dynamically control the servers' speeds (e.g., via frequency scaling), that aims at reducing the power consumption of the servers whose join-queue lengths are longer than the others'. Under Markovian assumptions, we provide a model for the performance evaluation of the system in saturation that allows us to derive the expression for the steady-state distribution, the system's throughput and balance index. Finally, we derive the analytical expression for the marginal state probabilities of each server and provide upper and lower bounds for the expected power consumption.

Keywords: Fork-join queues, Frequency Scaling, Energy Control, Time Reversibility

1. Introduction

Fork-join queueing stations have been extensively studied in the literature because of their wide applications in the context of distributed and parallel systems. Such queueing stations behave as follows: jobs arrive according to a certain arrival process and are forked into K tasks that are enqueued in the *service-queues* and then served by independent servers. Once a task is served, it is enqueued in the *join-queue* waiting for the service completions of all the other tasks of the job it belongs to. Once all the tasks of a job are served, the *join*

*Corresponding author

Email addresses: marin@unive.it (Andrea Marin), sabina.rossi@unive.it (Sabina Rossi)