Simulation-based approach for solving resource allocation problem of queueing systems with performance constraint

Mengyi Zhang, Andrea Matta, Arianna Alfieri

ARTICLE HISTORY
Compiled February 21, 2020

1. Introduction

Queueing systems have seen wide applications as the abstract model of manufacturing and service systems. The design and control of queueing systems represents an important category of optimization problems in the operations research field. A particular case is the resource allocation problem with performance constraints, which refers to deciding the number of resources with a minimal cost guaranteeing a target performance.

In this work, an element is defined as resource if and only if the system performance is non-decreasing on it. For instance, the capacity of a buffer can be a resource when the performance of interest is the steady state throughout. Another example is the number of parallel servers in a station when the performance of interest is the average system time in the steady state. A counter example is the population in closed systems. Resource allocation problem widely exists in literature and practice. Buffer allocation problem of allocation problem in manufacturing systems and server allocation problem in call centers.

When the target performance is set as a constraints, the resource allocation in queueing systems becomes a complicated problem. First, there is no closed-form evaluation approach for complicated systems, for instance, with blocking or complicated scheduling rules in a single queue, so we are facing an optimization problem with limited information (the performance only or with gradient) from a non-convex evaluation function in most of the time. Therefore, there is no particular structure from the evaluation methods that could be used for solution searching. Second, the problem is a constrained optimization problem. There is no efficient algorithms for constrained discrete optimization problems providing the optimum or the optimality gap of the incumbent.

This work develops a simulation-based sample-path exact algorithm for solving the resource allocation problem of generic queueing systems. The algorithm is composed of submodels generating feasibility cuts and upperbound (the incumbent) and global model generating the submodel and lower bound. In a submodel, two types of feasibility cuts are generated, namely approximate cuts and combinatorial cuts. An execution of event-based simulator can provide the triggering relationship among events, and the events and the relationships form vertices and arcs of a graph. This work proposes an algorithm to generate the approximate gradient once an infeasible solution is simulated, and an approximate cut is then derived. The combinatorial cuts, however, is an

inequality specifying the fact that a solution cannot be feasible if its resource capacity is smaller than an infeasible solution in all the dimensions, which are exact cuts. The aggressive cuts and combinatorial cuts are dynamically added to the submodel, which is linear relaxation of the resource allocation problem, and after solving the LP and applying rounding to the solution for several times, a set of solutions, both feasible and infeasible, can be obtained. A submodel will provide a set of combinatorial cuts and the best feasible solution to the global model. The global model is integer programming, and once it is solved, the optimum will be the lower bound and the optimal solution is used to defining the domain of the submodel.

The contribution of this work is that it provides an exact algorithm for the resource allocation problem. That is to say, the optimum will be found if the time limit is infinity, or the upper bound and lower bound of the optimum is provided within finite time horizon. Moreover, the algorithm is generic for queue systems. These generality comes from the fact that it uses the event-based simulator instead of analytical approaches for calculating the approximate gradient, thus the Markovian properties are not assumed, and simulation has strong capability in modeling blocking or complex scheduling rules.