A Multi-Objective Approach for Buffer Allocation in General Queueing Networks

F. R. B. Cruz
Department of Statistics, Federal University of Minas Gerais, Brazil, fcruz@est.ufmg.br

L. Duczmal
Department of Statistics, Federal University of Minas Gerais, Brazil, duczmal@est.ufmg.br

T. van Woensel
Technische Universiteit Eindhoven, t.v.woensel@tm.tue.nl

J. MacGregor Smith
University of Massachusetts, jmsmith@ecs.umass.edu

The optimal buffer allocation problem in real-life systems is a difficult optimization problem. Besides being non-linear, different objectives conflict with each other: maximize the throughput and minimize the number of buffers allocated. In this paper, we present a solution methodology for the buffer allocation problem in single server general queueing networks based on a multi-objective approach. The complete set of all best solutions is derived by a genetic algorithm (GA), resulting in the Pareto set. The GA proves to be suitable for multi-objective problems because of the availability of a well-known formulation for dealing with these problems. Preliminary results attest for the efficacy and efficiency of the methodology proposed.

Key words: Buffer allocation; genetic algorithms; queues; networks.

1. Introduction

An important trade-off in practice is between the amount of buffer space versus the desired throughput. As buffer space is expensive, one would like to minimize the amount of buffers as much as possible. On the other hand, we would also like to maximize the throughput in the network. This throughput is affected by the buffers allocated, i.e. more buffers leads to a higher throughput. This latter shows the trade-off: minimize buffers and maximize throughput. From a modeling point of view, this problem can be formulated as a multi-objective optimization problem, with two objectives: buffers and throughput. This paper will present the first step into the direction of multi-objective optimization applied to the buffer allocation problem for general, single server queueing networks. In other words, we focus in this paper on networks of $M/G/1/K$ queues[1], which in Kendall's notation means Markovian arrivals, generally distributed service times, a single server, and a total capacity of K items, including the item in service.

This paper builds on a number of recent articles on the Buffer Allocation Problem (BAP). Both Cruz et al. (2007) and Smith and Cruz (2005) formulated the BAP as a single-objective optimization problem in which buffers are minimized subject to a minimum throughput constraint. More specifically, the BAP was defined by the following formulation with integer decision variables $x_i \equiv K$, for the ith $M/G/1/K$ queue:

$$Z = \min \sum_i x_i, \tag{1}$$

[1] Multi-servers queueing networks will be dealt in future research as this results in a three-objective problem: throughput, buffers, and servers.

s.t.:

$$\Theta(\mathbf{x}) \geq \Theta^{\min}, \tag{2}$$
$$x_i \in \mathbb{N}, \forall i, \tag{3}$$

which minimizes the total buffer allocation to the network, $\sum_i x_i$, subject to providing a minimum total throughput $\Theta^{\min}$. In this formulation, $\Theta^{\min}$ was some threshold throughput, not superior to the total external arrival rate, $\Lambda = \sum_i \Lambda_i$, and x_i is the buffer K allocation to the ith $M/G/1/K$ queue, including those in service.

It should be clear that in the above formulation the throughput is modeled as a constraint rather than as an objective. In Cruz et al. (2007) and Smith and Cruz (2005) this constraint is then relaxed via a Lagrangian relaxation technique, but some factors like the threshold throughput $\Theta^{\min}$ still needs to be determined beforehand. Setting this threshold throughput in a correct way is not a trivial task. Moreover, it might be perfectly well possible that a small decrease in throughput might result in a significant gain in terms of buffers used. This trade-off is not visible in the above formulation. In general, it shows that mono-objective optimization techniques may be arbitrary. More specifically, defining the weights for composing a single-objective function over different objectives, $\boldsymbol{\lambda}$, and other parameters such as the acceptable error, ϵ, minimum throughput, Θ (see Smith and Cruz 2005, Cruz et al. 2007) is difficult.

Rather than working with, say, objectives $f_1(\mathbf{x})$ and $f_2(\mathbf{x})$ simultaneously, researchers usually try to solve the problem with some reasonable algorithm to generate an approximation to the Pareto set of solutions for the two objectives, as shown in Fig. 1. Starting from an initial hyperplane h whose inclination is defined by the given weights $\boldsymbol{\lambda}$ for each one of the objectives, that is, defining the objective funtion $z(\mathbf{x}) = \lambda_1 f_1(\mathbf{x}) + \lambda_2 f_2(\mathbf{x})$, some mono-objective optimization algorithm will find a sequence of parallel hyperplanes to h converging to the support hyperplane h^*, with only one intersection point y^* that belongs to the Pareto set Y^*.

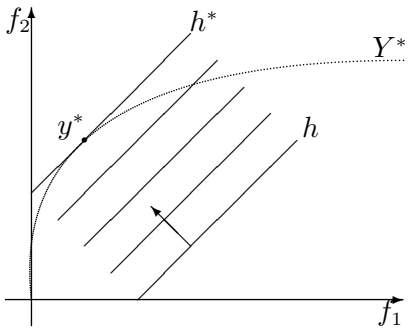

Figure 1 Pareto set and an approximate solution by means of a mono-objective algorithm.

In this article, we will do differently and determine the whole Pareto-optimal set, which is the set of optimal solutions for more than one objective in the objective functions (Chankong and Haimes 1983). This means that the decision maker will be able to better evaluate the effect of replacing one solution by another. The multi-objective approach allows for evaluating the loss in one objective (e.g. throughput) compared to a simultaneous enhancement in another objective (e.g. buffer allocation).

Here, we use a genetic algorithm (GA) approach in combination with the generalized expansion method, a well-known method to obtain the performance measures of the network. The GA is particularly suitable for multi-objective problems because they have been performing well in dealing with this type of problems (see e.g. Carrano et al. 2006, and the references therein).

The main contributions of this paper can be listed as follows:

1. The Buffer Allocation Problem is treated in a multi-objective approach, explicitly taking into account buffer space and throughput as decision variables. Both variables are optimized simultaneously resulting in the Pareto set of optimal solutions, which results in interesting insights with regards to the optimal configuration of the network.
2. We embed the BAP into a problem-specific GA. We show that the GA solves the multi-objective BAP in reasonable computation times.
3. The results show that the GA combined with the BAP gives the complete Pareto set. Moreover, the results obtained in Cruz et al. (2007) are shown to be on these Pareto sets.

Highlight of results
The main results obtained include the proposition of a multi-objective formulation that has the advantage of being less arbitrary than the mono-objective optimization techniques allowing the decision maker to choose among quite different trade-off solutions. Additionally, with regards to the GA, we show that a simple cross-over and mutation scheme handled the problem properly. Moreover, a straightforward peeling-off scheme solved the problem of selecting the correct Pareto set. Finally, the results obtained in the previous paper (Cruz et al. 2007) are shown to be special cases of the results presented here.

Structure of the paper
The organization of this paper is as follows. In Sec. 2, we present the buffer allocation and its multi-objective mathematical programming formulation. The GA specially developed for the buffer allocation problem is presented in detail in Sec. 2.1. The generalized expansion method as performance evaluation methodology is discussed in Sec. 2.2. In Sec. 3, preliminary computational results are presented to show the efficiency of the approach. Finally, we close the article with conclusions and topics for future research in the area (Sec. 4).

2. The Multi-Objective Buffer Allocation Problem

In this section, we define the buffer allocation problem for single-server general service queueing networks and present the multi-objective mathematical programming formulation used in this paper. The BAP is concerned with minimizing the buffer spaces while maximizing the number of users served per time unit, known as the throughput. The BAP is perhaps best formulated as a non-linear multiple-objective programming problem in which the decision variables, the buffer space, are the integers.

The Multi-Objective Buffer Allocation Problem (MOBAP) can be formulated by the following mathematical programming formulation:

$$\text{optimize } Z = \Big\{ f_1(\mathbf{x}), f_2(\mathbf{x}) \Big\} \tag{4}$$

s.t.:

$$x_i \in \{1, 2, \ldots\}, \forall i, \tag{5}$$

in which

$$f_1(\mathbf{x}) = \textstyle\sum_i x_i, \text{ the total buffer size,}$$
$$f_2(\mathbf{x}) = \Theta(\mathbf{x}), \text{ the throughput profit,}$$

and with integer decision variables $x_i \equiv K$, which are the total capacity (including those in service) for the ith $M/G/1/K$ queue.

The above formulated MOBAP is solved using a powerful class of optimization heuristic methods, the GA's. In Sec. 2.1, we describe the GA used in more detail. After this, we briefly discuss in Sec. 2.2 the Generalized Expansion Method which is used to find the performance measures for the considered finite queueing network.

2.1. A genetic algorithm

The GA's are suitable for the MOBAP because of their well-established efficiency for dealing with multi-objective problems in general (Fonseca and Fleming 1995, Coello 2000). The instance of multi-objective GA used in this article is shown in Fig. 2, adapted from Carrano et al. (2006). The GA's are optimization algorithms to perform an approximate global search relaying on the information obtained from the evaluation of several points in the search space and obtaining a population of these points that converges to the optimum through the application of the genetic operators *mutation*, *crossover*, *selection*, and *elitism* (Takahashi et al. 2003). Each one of these operators may be implemented in several different ways, each one of them characterizing an instance of the GA.

algorithm
  read graph, $G(V, A)$
  read arrival and service rates, $\lambda_v, \mu_v, \forall\, v \in V$
  $P_0 \leftarrow \emptyset$
  $P_1 \leftarrow$ **GenerateInitialPopulation**(`popSize`)
  $O_1 \leftarrow$ **ExtractParetoSet**(P_1)
  for $i = 1$ **until** `numGen` **do**
    $R_t \leftarrow P_t \cup P_{t-1}$
    $P_{t+1} \leftarrow$ **FitnessCrossoverMutation**(R_t)
    $O_{t+1} \leftarrow$ **ExtractParetoSet**(P_{t+1})
  end for
  write $O_{\texttt{numGen}+1}$
end algorithm

Figure 2 Multi-objective genetic algorithm - NSGA-II.

In the special case of the multi-objective optimization problems, the operators *selection* and *elitism* must be specially structured to correctly identifying the best individuals. Operators *mutation* and *crossover* are independent on the multi-objective nature of the problem (Takahashi et al. 2004). The multi-objective GA evolves the whole population toward the Pareto set, instead of a single point and in a single run the whole Pareto set, or a large portion of it, will be found. This explains the superiority of the GA's over the deterministic algorithms, which are able to find only one Pareto point for each run (Fonseca and Fleming 1995, Coello 2000). Critical to the convergence of the GA are the population size, `popSize`, and the number of generations, `numGen`, among other parameters. Further details on the GA will be postponed to the full paper.

2.2. Performance Evaluation

In order to solve the MOBAP optimization problem, we need an estimate for the throughput, $\Theta(\mathbf{x})$. An algorithm available is the Generalized Expansion Method (GEM), successfully used in the past to estimate performance measures for arbitrarily configured finite queueing networks. Well described in many articles, in particular in the article by Kerbache and Smith (2000), the GEM is basically a combination of node-by-node decomposition and repeated trials, in which each queue is analyzed separately and then corrections are made in order to take into account the interrelation between the queues in the network. The GEM uses type I blocking, that is, the upstream node gets blocked if the service on a customer is completed but it cannot move downstream due to the queue at the downstream node being full. This is sometimes referred to as blocking after service, which is prevalent in most production and manufacturing, transportation, and similar systems. Further details will not be given in this article.

Following Kerbache and Smith (2000), the GEM creates for each finite queue, represented by vertex j, an auxiliary vertex h_j, modeled as an $M/G/\infty$ queue (see Figure 3). When an entity arrives to the system, vertex j may be blocked with probability p_{K_j}, or unblocked, with probability

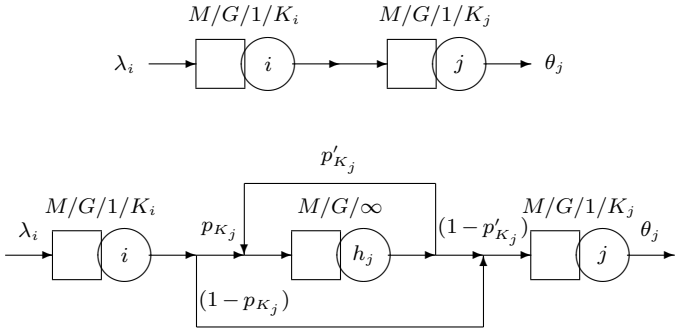

Figure 3 Generalized expansion method.

$(1-p_{K_j})$. Under blocking, the entities are rerouted to vertex h_j for a delay while node j is busy. Vertex h_j helps to accumulate the time an entity has to wait before entering vertex j and to compute the effective arrival rate to vertex j. The interested reader is referred to Cruz and Smith (2007).

3. Computational results and discussion

All algorithms were implemented in C and are available upon request. The experiments were run for tandem, split, and merge queues. The complete results were generated using the following experimental design: the arrival rates considered were $\lambda = \{1.0, 2.0, 4.0\}$, the service rates, $\mu_i = 10.0$, $\forall\, i$, resulting in a system utilization $\rho = \{0.1, 0.2, 0.4\}$, combined with several values for the squared coefficient of variation, $c_s^2 = \{0.5, 1.0, 2.0\}$, and number of nodes, $|V| = \{3, 7, 15\}$. In Figure 4, we give a small selection of representative results obtained. The complete results will be available in the full paper. Each of the three Pareto curves for the specific settings contained the original optimal solution found in the mono-objective approach as presented in Cruz et al. (2007). In Figure 4, this solution is circled. Extra insights are provided now because based on the Pareto curves the decision maker could choose to reduce a bit in throughput but to gain a significant number of expensive buffer spaces, or vice versa. Of course, the exact decisions made are highly dependent upon the underlying cost structure.

4. Conclusions and Future Research

In this paper we developed a Multi-Objective Buffer Allocation Problem (MOBAP) for single server general queueing networks. Combining the Generalized Expansion Method as the performance evaluation tool with a GA gives insightful Pareto curves. These curve explicitly show the trade-off between buffer spaces and throughput. Experimental results confirm optimal solutions found in earlier papers and show the merits of the approach. Future research involves the multi-objective optimization applied to zero-buffer systems and to the joint buffer and server allocation problem.

Acknowledgments

The research of prof. Frederico Cruz has been partially funded by the CNPq (*Conselho Nacional de Desenvolvimento Científico e Tecnológico*) of the Ministry for Science and Technology of Brazil, grants 201046/1994-6, 301809/1996-8, 307702/2004-9, 472066/2004-8, and 472877/2006-2, by the FAPEMIG (*Fundação de Amparo à Pesquisa do Estado de Minas Gerais*), grants CEX-289/98, CEX-855/98, and TEC-875/07 and PRPq-UFMG, grant 4081-UFMG/RTR/FUNDO/PRPq/99.

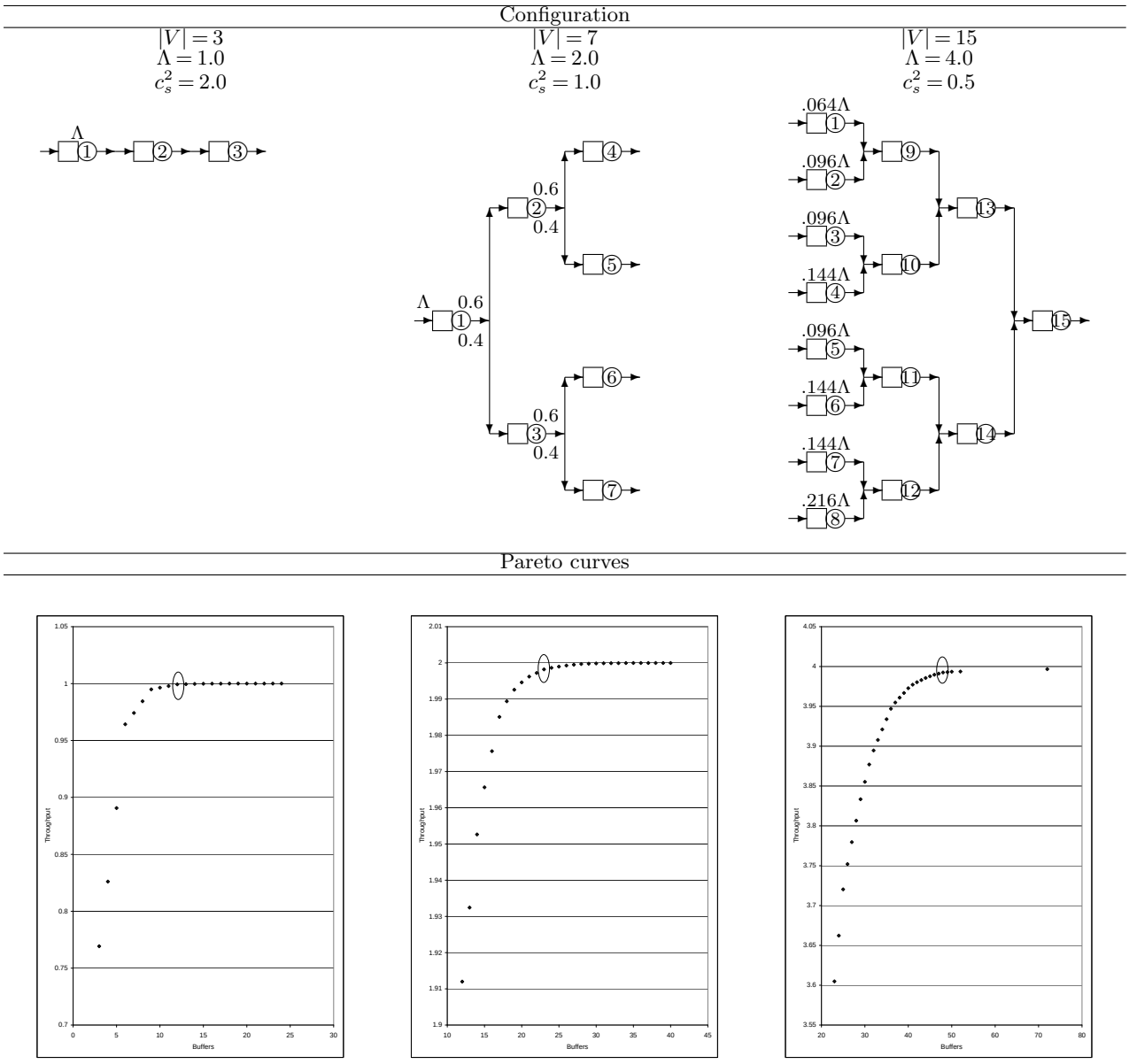

Figure 4 **Selection of tested topologies.**

References

Carrano, E. G., L. A. E. Soares, R. H. C. Takahashi, R. R. Saldanha, O. M. Neto. 2006. Electric distribution network multiobjective design using a problem-specific genetic algorithm. *IEEE Transactions on Power Delivery* **21**(2) 995–1005.

Chankong, V., Y. Y. Haimes. 1983. *Multiobjective Decision Making: Theory and Methodology*. Elsevier, Amsterdam, The Netherlands.

Coello, C. A. C. 2000. An updated survey of GA-based multiobjective optimization techniques. *Proceedings of the ACM Computing Surveys*, vol. 32. 109–143.

Cruz, F. R. B., A. R. Duarte, T. van Woensel. 2007. Buffer allocation in general single-server queueing network. *Computers & Operations Research* 1–16 (in press).

Cruz, F. R. B., J. MacGregor Smith. 2007. Approximate analysis of $M/G/c/c$ state-dependent queueing networks. *Computers & Operations Research* **34**(8) 2332–2344.

Fonseca, C. M., P. Fleming. 1995. An overview of evolutionary algorithms in multiobjective optimization. *Evolutionary Computing* **3**(1) 1–16.

Kerbache, L., J. MacGregor Smith. 2000. Multi-objective routing within large scale facilities using open finite queueing networks. *European Journal of Operational Research* **121**(1) 105–123.

Smith, J. MacGregor, F. R. B. Cruz. 2005. The buffer allocation problem for general finite buffer queueing networks. *IIE Transactions on Design & Manufacturing* **37**(4) 343–365.

Takahashi, R. H. C., R. M. Palhares, D. A. Dutra, L. P. S. Gonalves. 2004. Estimation of Pareto sets in the mixed h_2/h_∞ control problem. *International Journal of Systems Science* **35** 55–67.

Takahashi, R. H. C., J. A. Vasconcelos, J. A. Ramirez, L. Krahenbuhl. 2003. A multiobjective methodology for evaluating genetic operators. *IEEE Transactions on Magnetics* **39**(3) 1321–1324.