Post-Processing Improvements in Multi-Objective Optimization of General Single-Server Finite Queueing Networks

Gabriel L. de Souza, Anderson R. Duarte, Gladston J. P. Moreira, and Frederico R. B. Cruz

Abstract—**An alternative mathematical programming formulation is considered for a mixed-integer optimization problem in queueing networks. The sum of the blocking probabilities of a general service time, single server, and the finite, acyclic queueing network is minimized, and so are the total buffer sizes and the overall service rates. A multi-objective genetic algorithm (MOGA) and a particle swarm optimization (MOPSO) algorithm are combined to solve this difficult stochastic problem. The derived algorithm produces a set of efficient solutions for multiple objectives in the objective function. The implementation of the optimization algorithms is dependent on the generalized expansion method (GEM), a classical tool used to evaluate the performance of finite queueing networks. We carried out a set of computational experiments to attest to the efficacy and efficiency of the proposed approach. In addition, we present a comparative analysis of the solutions before and after post-processing. Insights obtained from the study of complex queue networks may assist the planning of these types of queueing networks.**

Index Terms—**Queueing networks, conflicting objectives, buffer allocation, particle swarm optimization.**

I. Introduction

The flow of services, users, and products, among others, always appears to be associated with some uncertainty. This uncertainty leads to the formation of queues to manage this flow. Of these processes, several can be modeled as queueing systems. In general, these processes are more complex and may be composed of interconnected queues or networks of queues. Queues configured in networks, in which each queue has an arrival rate λ and a service rate μ, are a natural generalization for various systems of practical interest.

The main interest in this article is to efficiently solve a mixed-integer optimization problem in networks of $M/G/1/K$ queues. In Kendall notation, $M/G/1/K$ represents a queue with independent and identically exponentially (Markovian) distributed times between the arrivals, a single server with generally and independently distributed service times, and a total capacity of K customers including the customer in service. We have shown an example of a network of queues in Figure 1.

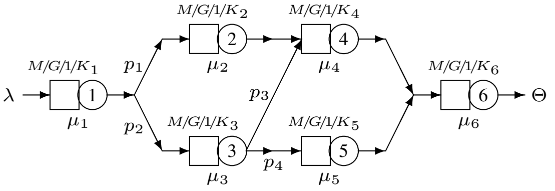


Fig. 1. An example of complex queueing network adapted from [1].

The discussion of queueing systems is usually associated with the classic queues in people's daily lives. Some processes can also be modeled as queueing systems; however, they are not noticeable in our daily routine. The investigation of the performance of queuing networks is closely linked with specific measures of queue performance. Several performance measures may be of practical interest in investigating performance in queueing networks. Many studies address the throughput θ as the measure of paramount interest. Achieving high throughput values leads to a high-performance queueing network. However, achieving high throughput values on a network requires a significant increase in resource consumption. As a result, a significant increase occurs in the costs of the queueing system. Therefore, a usual proposal is the maximization of throughput combined with the minimization of the overall service allocation $\sum \mu_i$ and the total capacity allocation of the system $\sum K_i$.

A new proposition to investigate the performance of a queueing network was discussed in [2] in which the usual goal of maximizing the throughput is replaced by the minimization of the sum of the blocking probabilities while preserving in the formulation the usual objectives associated with the resources (*i.e.*, the overall service allocation $\sum \mu_i$, and the total system capacity allocation $\sum K_i$). In such a proposition, the individualized effect of each queue on generating blockage of customers is considered. Thus, an internal system analysis is the way of reaching an overall improvement of the system. The objective is to provide low blocking probabilities across the queueing network, to increase the flow along with the system, and to improve the performance of the queueing network.

For a finite queue with a total capacity K, the probability that a customer will reach this queue and find precisely K customers is denoted by P_K, which is usually called *blocking probability*. The excessive occurrence of blockages reduces the overall performance of the queueing network and, therefore, high probabilities of blocking result in low efficiency of the queueing network [3].

A proposal for post-processing solutions based on solutions obtained from known efficient algorithms was presented in [4],

Gabriel L. de Souza and Gladston J. P. Moreira are with the Computing Department, Universidade Federal de Ouro Preto, Brazil e-mail: gabriel.souza@ufop.edu.br and gladston@ufop.edu.br.

Anderson R. Duarte is with the *Departamento de Estatística*, Universidade Federal de Ouro Preto, Brazil e-mail: anderson.duarte@ufop.edu.br.

Frederico R. B. Cruz is with the *Departamento de Estatística*, Universidade Federal de Minas Gerais, Brazil e-mail: fcruz@est.ufmg.br.

which has found refined improvements in previous solutions. Also this study used the formulation of maximization of throughput while minimizing the overall service allocation $\sum \mu_i$ and the total system capacity allocation $\sum K_i$. The target was to develop algorithms to redistribute capacity areas across the queueing network while preserving the full capacity previously allocated to the queueing network. Through this proposal, it was possible to increase the throughput.

Another post-processing proposal based on the sum of the blocking probabilities was described in [2]. Here we address again the formulation introduced by [2] but with some novelties including a more detailed presentation of the formulation and also a comprehensive investigation of the structure of the solutions obtained through an heuristic method to simultaneously minimize the sum of the blocking probabilities ($\sum_i P_{K_i}$) in an acyclic network of $M/G/1/K$ queues. In addition, this article introduces well-known metrics specially adapted to the evaluation of the improvements in the solutions obtained by the post-processing technique developed in [2].

There is a critical trade-off for this formulation (minimization of the blocking probabilities, overall service rates $\sum \mu_i$, and the total capacity of the system $\sum K_i$). Notice that the greater the buffer allocation and the service rates are in the system, the less the blocking probabilities for each queue are. However, the increase in the capacity and the service rates is highly costly. Thus, the objective here is to develop algorithms to minimize these expensive resources while simultaneously reducing the blocking probabilities.

The optimization approach seen in [2] produced Pareto-optimal solutions for more than one objective in the multi-objective function. However, the authors provided no critical evaluation of the behavior of these solutions before and after post-processing. They used no metrics to verify possible improvements through post-processing. The guarantee of improvements, through well-established metrics, can contribute to the decision-making agent opting for the use of the post-processing technique.

This study presents the contribution of a more detailed investigation of the proposition in [2]. In addition, we offer a more an extensive set of computational experiments and the inclusion of performance evaluation metrics for multi-objective algorithms to confirm the proposal's suitability.

This remaining of this article is organized as follows. Section II briefly describes the literature of the area. Section III presents the methodological aspects of this study: the problem formulation in terms of a mathematical programming model, the multi-objective methodology, and the proposed post-processing approach based on particle swarm optimization. Section IV presents the simulation results for several basic settings. Section V completes the article with conclusions and possible themes for future research in this area.

II. Brief literature review

Many researchers are potential users of general single-server finite queueing network optimization. The importance of optimization problems in queueing networks stems from the possibility of providing improvements in several systems. Investigations in this direction reveal the possibility of understanding and improving several systems, including production line modeling [5], [6], industrial processes [7], [8], production systems [9], health systems [10], [11], traffic of vehicles and pedestrians [12]–[14], computer and communication systems [15]–[18], web-based applications with tiered configurations [19] with QoS requirements defined in terms of response time, throughput, availability, and security [20], among others.

This study's queueing network optimization problem is addressed through nature-inspired heuristic algorithms. Here, the multi-objective genetic algorithm (MOGA) (proposed by [21]), combined with a multi-objective particle swarm optimization algorithm (MOPSO) (proposed by [2]) are used. These optimization algorithms perform an approximate global search based on knowledge of several points in the search space [22], [23]. It is well known that these algorithms can be suitable for many multi-objective problems with complicated objective functions and constraints [24]–[29].

III. The Methodology

The interest here is the discussion of mathematical formulations for the performance optimization problem of a network of general finite single-server queues and the adaptation of algorithms to efficiently optimize such queueing networks. Several algorithms have been proposed to solve this problem, which is strongly dependent on the mathematical programming formulation used.

There are classic single-objective formulations in the literature: buffer allocation problem (BAP) [1], [3], server allocation problem (CAP) [30]–[32], and buffer and server allocation problem (BCAP) [33]. As a result of these formulations, multi-objective formulations propositions also emerged [2], [4], [21]. Particularly this study addresses the problem through the multi-objective formulation presented in [2].

A. Multi-objective Formulation

The optimization problem in queueing network may be defined on a digraph $\mathcal{D}(V, A)$ in which V is a finite set of m vertices (queues), and A is a finite set of arcs (connections between the queues).

The following multi-objective mathematical programming formulation is another possible way to formulate the optimization problems of $M/G/1/K$ networks (see [4], [21]):

$$\text{minimize}\, F(\mathbf{K}, \boldsymbol{\mu}) = \Big[f_1(\mathbf{K}), f_2(\boldsymbol{\mu}), f_3(\mathbf{K}, \boldsymbol{\mu})\Big], \tag{1a}$$

subject to:

$$\begin{array}{rcl} K_i & \in & \mathbb{N}, \quad \forall i \in \{1, 2, \ldots, m\}, \\ \mu_i & \geqslant & 0, \quad \forall i \in \{1, 2, \ldots, m\}, \end{array} \tag{1b}$$

which comprises the minimization of capacities and service rates, simultaneously with maximization of throughput. Then, $f_1(\mathbf{K}) = \sum_{i=1}^m K_i$ represents the total capacities, $f_2(\boldsymbol{\mu}) = \sum_{i=1}^m \mu_i$ represents the overall service rates, and $f_3(\mathbf{K}, \boldsymbol{\mu}) = -\Theta(\mathbf{K}, \boldsymbol{\mu})$ represents the throughput. The minus sign in the throughput is because this objective is to be maximized.

Notice that it is usual in the literature of the area to model the throughput as a constraint but this approach has an important drawback because the throughput constraint usually must be relaxed and it is not easy to define a suitable threshold.

Various possible formulations have been described in the literature for the optimization problem based on the throughput of the system, (Θ) [3], [4], [21], [33]–[40]. Here, a mathematical formulation for optimization is presented, focusing on the blocking probabilities of the queueing system [2], which prioritizes the minimization of the sum of the blocking probabilities in the system. Notice that the minimization of the total capacity allocation and the overall service allocation is also sought. Behind this choice is the idea of a more intuitive nature of blocking probability. The prioritization to minimize the blocking probability values ensures a greater degree of decoupling between the different queues of the network. That is to say, the queues will suffer less interlocking under low blocking probabilities.

B. Performance Evaluation

The Generalized Expansion Method (GEM) (see [41]) is quite efficient for estimating the blocking probability P_K in *single* $M/G/1/K$ *queues*. GEM is a computationally efficient and accurate method based on a two-moment approximation [42]:

$$P_K = \frac{\rho^{\left(\frac{2+\sqrt{\rho}s^2-\sqrt{\rho}+2(K-1)}{2+\sqrt{\rho}s^2-\sqrt{\rho}}\right)}(\rho-1)}{\rho^{\left(2\frac{2+\sqrt{\rho}s^2-\sqrt{\rho}+(K-1)}{2+\sqrt{\rho}s^2-\sqrt{\rho}}\right)}-1}, \tag{2}$$

in which $\rho < 1$ is a constraint that must be guaranteed. The system utilization ρ is defined as the ratio between the total arrival rate and the service rate, $\rho = \lambda/\mu$, and $s^2 = \mathrm{Var}(T_s)/\,\mathbb{E}^2(T_s)$ is the squared coefficient of variation of the service time (T_s). The approximation of P_K is shown to be accurate by previous studies for a wide range of values [1], [3], [43].

Additionally, in *single queues*, a fraction P_K of the arrivals cannot join the system. Then, P_K represents the probability that a customer arrives when there is no more waiting space. Therefore, only the fraction $(1 - P_K)$ of the arrivals can be served by the queue [44], resulting in a throughput of $\lambda(1 - P_K)$. Then, the throughput of this single queue is the fraction of customers, arriving at a rate of λ, who did not find the system blocked, and this throughput may be considered approximately Markovian (see [1]).

Accurate estimates for the performance measures of arbitrarily configured, finite queueing, acyclic networks have been successfully obtained by GEM, which is a repeated-trial method. GEM considers the delay effect generated by several possible blockages occurring in the flow of customers along with the queueing network. Employing an iterative procedure, GEM solves a set of simultaneous nonlinear equations leading to considerable improvement in the precision of the estimation of the performance measures of the queueing network. The method may be seen as a combination of node-by-node decomposition and repeated trials. Thus, each queue is analyzed separately, and corrections are made to account for interrelated effects between network queues.

GEM is described in great detail in [45], in which is mentioned that the method creates, for each finite node j, an auxiliary vertex (h_j) that is modeled as an $M/G/\infty$ queue (see Fig. 2). For each entity placed in the system, vertex j may be blocked (with probability P_{K_j}) or may be unblocked (with probability $1 - P_{K_j}$). When blocking occurs, the entities are rerouted to vertex h_j and are delayed while node j is busy. Vertex h_j records the time that an entity has to wait, with a service rate μ'_h, given by GEM, before entering vertex j, and updates accordingly the effective arrival rate coming from vertex i to vertex j, $\lambda_{\text{eff}} = \lambda_i(1 - P_{K_i})$.

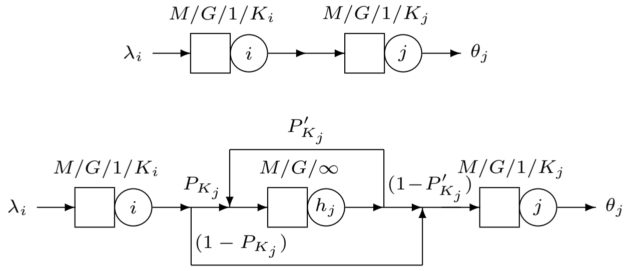


Fig. 2. Generalized expansion method for a tandem network.

The goal of GEM is to provide updates of the service rates of the nodes as follows:

$$\tilde{\mu}_i^{-1} = \mu_i^{-1} + P_{K_j}(\mu'_h)^{-1}, \tag{3}$$

in such a way that from Eq. (3) services rates μ_i can be updated for all nodes along ρ and consequently P_K, from Eq. (2).

GEM iteratively calculates updates of the system performance measures taking into account the delay effect generated by several possible blockages in the flow of customers in the queues. In this article, the throughput (Θ) is not considered its optimization objective. Instead, the sum of the blocking probabilities in each queue is considered. Notice that the computation of P_K, even though the approximation proposed in Eq. (2), is dependent on the knowledge of the queue arrival rate, λ. For the initial queue of a network (see Fig. 1), the arrival rate, λ, is known, but not for the subsequent queues. In some articles, the procedure applied to obtain these arrival rates considers approximations produced through the use of GEM.

The blocking probability P_{K_i} is calculated by GEM [41], and is dependent on λ_i, μ_i, and K_i. The μ_i and K_i values are decision variables of the optimization problem, but the arrival rate λ_i depends on the previous queue's throughput. For the sake of argument, consider a tandem queueing network, in which the computations performed assume that the arrival rate on the ith queue is dependent on the previous $(i-1)$th queue, given by:

$$\lambda_i = \lambda_{i-1}(1 - P_{K_{i-1}}), \tag{4}$$

in which $i \in \{2, \ldots, m\}$ and λ_1 is the external queueing network arrival, $\lambda_1 = \Lambda$.

C. Multi-objective Approach

The optimization problem presented by Eq. (1a) and (1b) will be solved by an adapted multi-objective evolutionary algorithm (MOEA). Generally speaking, an MOEA may be seen as an optimization algorithm that approximately performs global searches [22]. The application of the genetic operators of *mutation*, *crossover*, *selection*, and *elitism* ensure that the population converges to a mutually non-dominated approximation set of the Pareto front. In this article, the MOEA is the elitist non-dominated sorting genetic (NSGA-II) algorithm. Further details about this algorithm for the queueing network optimization problem are in [4].

A multi-objective particle swarm optimization (MOPSO) algorithm is applied after the NSGA-II optimization to improve its solutions. Given the mathematical programming formulation that has been proposed here, the convergence of NSGA-II can be enhanced by a post-processing algorithm like a MOPSO. The proposed MOPSO extends the single-objective PSO algorithm from [46].

Each possible solution for the resource allocation (capacities and service rates) is represented by a particle that aims to optimize the queueing network under study. Each particle in the proposed formulation can be represented by variables $(x_1, \ldots, x_\ell) = (K_1, K_2, \ldots, K_m, \mu_1, \mu_2, \ldots, \mu_m)$, with $\ell = 2m$.

Notice that the multi-objective optimization problem addressed here is a mixed-integer problem. Because of that, we must define a particle repair strategy. Indeed, changes to capacities are performed, and then integer values are used, as $K_i \geqslant 1$ must always be respected. Similarly, the restrictions associated with service rates are also appreciated because it is necessary to guarantee that $\rho < 1$. The queue arrival rate must be strictly less than the service rate μ. Then the feasibility of the investigated solutions is guaranteed. Each particle $1 \leqslant i \leqslant s$ has the following attributes, considering that s denotes the size of the swarm (population of particles):

- Position, $x_i = (x_{i,1}, x_{i,2}, \ldots, x_{i,\ell})$;
- Velocity, $v_i = (v_{i,1}, v_{i,2}, \ldots, v_{i,\ell})$;
- Personal best position, p_i;
- Global best position, g_i.

The proposed MOPSO approach for the optimization of the queueing network is presented as the pseudo-code shown in Algorithm 1:

Eq. (5) and (6) are responsible for the update of the speed and position of the particles, respectively:

$$v_i^{t+1} = w^t + r_1\big(p_i - x_i^t\big) + r_2\big(g_i - x_i^t\big), \tag{5}$$

$$x_i^{t+1} = x_i^t + v_i^{t+1}; \tag{6}$$

Eq. (7) is responsible for the update of the position for the integer variables:

$$x_i^{t+1} = \mathrm{int}\Big(x_i^t + v_i^{t+1}\Big); \tag{7}$$

The parameters and their values were defined as follows: r_1 and r_2 are positive random numbers with uniform distribution belonging to the interval $[0, 1.0]$, $w = 0.4$ is the inertia weight.

The literature presents several details about the implementation of MOPSO algorithms. The MOPSO just described it is an adaptation of the classical implementation described by [47], and simplified versions are found in [48], and more sophisticated and improved versions in [49] and in [50], which includes the mixed-integer mathematical programming formulations [51].

Algorithm 1: Multi-objective Particle Swarm Algorithm

```
/* generate initial swarm */
$X \leftarrow \mathit{GenInitSwarm}(\texttt{swarmSize})$
$P \leftarrow X$
/* find non-dominated fronts $\mathcal{F} = (\mathcal{F}_1, \mathcal{F}_2, \ldots)$ */
$\mathcal{F} \leftarrow \mathit{NonDominSort}(X)$
$g \leftarrow \mathit{Random}(\mathcal{F})$
/* move swarm */
for $t = 0$; $t < \texttt{numIter}$, $t$++ do
    for $i = 0$; $i < \texttt{swarmSize}$; $i$++ do
        $v_i^{t+1} \leftarrow \mathit{Speed}(x_i^t, p_i, g)$
        $x_i^{t+1} \leftarrow \mathit{NewPosition}(x_i^t, v_i)$
        if $x_i^{t+1}$ $\mathit{dominates}$ $p_i$ then
            $p_i \leftarrow x_i^{t+1}$
        else
            if $p_i$ $\mathit{dominates}$ $x_i^{t+1}$ then
                $p_i \leftarrow p_i$
            else
                $p_i \leftarrow \mathit{Random}(x_i^{t+1}, p_i)$
            end if
        end if
    end for
    $\mathcal{F} \leftarrow \mathit{NonDominSort}(X)$
    $g \leftarrow \mathit{Random}(\mathcal{F})$
end for
write $\mathcal{F}$
```

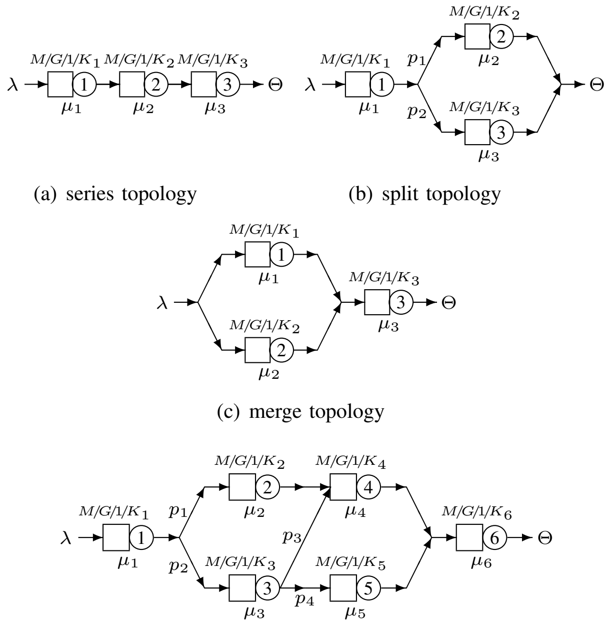
IV. Results

The post-processing code (multi-objective particle swarm optimization) was implemented in FORTRAN to use previous implementations of NSGA-II [21] and GEM [41]. All codes are available to authors upon request and should be used for educational and research purposes. We conducted the execution of the computational experiments on Intel(R) Core(TM) i3-2310M 2.10 GHz running Windows 10 Pro 64 bits, with 6.00GB of RAM.

According to previously reported studies [21], the best parameter group for NSGA-II is the combined use of the simulated binary crossover (SBX) and mutation, a mutation rate of 2%, 400 individuals seemed to be sufficient, the dispersion parameter should be approximately 8, and we set the maximum number of generations `numGen` to 4000 (this ensure a finite computation time). For MOPSO, the swarm size was set to 400 (equal to the number of individuals in the NSGA-II), and the maximum number of iterations of the algorithm was set to $4,000$. The parameters were defined as follows: r_1 and r_2 are positive random numbers with uniform distribution belonging to the interval $[0, 1.0]$, $w = 0.4$ is the inertia weight.

With the parameters set for the NSGA-II and MOPSO algorithms, computational experiments were conducted for queues networks in the topologies: series, split, and merge, and the generic mixed topology adapted from [1], as illustrated in Fig. 3.

For each topology under study, we analyzed three different values for the square coefficients of variation in service



(a) series topology

(b) split topology

(c) merge topology

(d) mixed topology

Fig. 3. Topologies tested, series, split, merge, and mixed topologies.

times, $s^2 \in \{0.5, 1.0, 1.5\}$, to characterize systems that are hypo-exponential, exponential (markovian), hyperexponential, respectively. A single entry is considered in the queueing network for all topologies, with $\lambda = 5.0$. In the investigations presented here, the routing vector in the split nodes is considered to be known, with equal probabilities. The routing probabilities in Figure 3(b) and Figure 3(d) are represented by (p_1, p_2) and (p_1, p_2, p_3, p_4), respectively. The specific case of optimizing routing probabilities to achieve a particular objective (for example, to maximize throughput) is the goal of other studies (see [40] for instance).

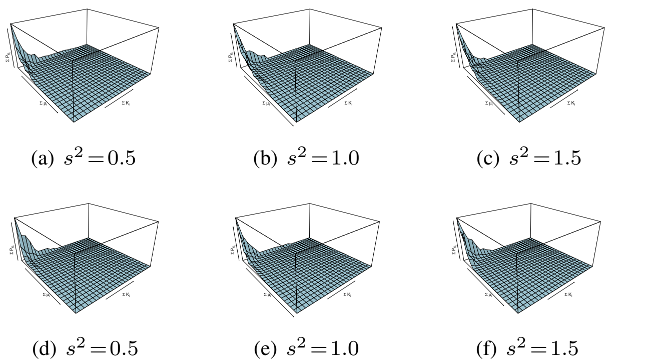
The main objective of the proposed post-processing is to readapt the allocation of resources to the queueing network. It is not an objective to ensure that every post-processed solution dominates the previous solution obtained by NSGA-II. In some cases, the solution does not dominate, nor is it dominated, i.e., it is a new solution distinct from the previous one concerning the allocation of resources.

Among all solutions, a named solution (a) provided by NSGA-II, after being post-processed by MOPSO, provides a solution denominated (a*). There is some possibility that the solution (a*) coincides with some solution (b) provided by NSGA-II but is distinct from solution (a). In practice, it would be a different solution. However, it is essential to observe the proportion of repeated solutions between the NSGA-II solutions and the post-processed solutions to verify this possibility. Table I represents the proportion of really new solutions generated through the post-processing technique.

Table I shows that the post-processing strategy used provides a significant proportion of new solutions. However, at this moment, it is only possible to ensure that the solutions provided are new. It is still impossible to ensure that such solutions are effectively better than the previous ones. It is possible to guarantee that the previous ones do not dominate such solutions. On the other hand, it is impossible to guarantee that such solutions dominate the previous ones. By dominance criteria, many of these solutions are at an equal level. For that, decision-making would be up to the queueing network manager. Figs. 4, 5, 6, and 7 present the results obtained for all squared coefficients of variation tested, examining in detail the solutions for all topologies. These figures show the 3-d surface ($\sum K_i \times \sum \mu_i \times \sum P_{K_i}$) provided by NSGA-II and the post-processed surface by MOPSO.

TABLE I
PROPORTION OF NEW SOLUTIONS BY MOPSO POST-PROCESSING.

s^2	Topology			
	serie	split	merge	mixed
0.5	0.8500	0.8450	0.8550	0.8125
1.0	0.8350	0.8475	0.8675	0.7725
1.5	0.8750	0.8550	0.8700	0.8100



(a) $s^2 = 0.5$

(b) $s^2 = 1.0$

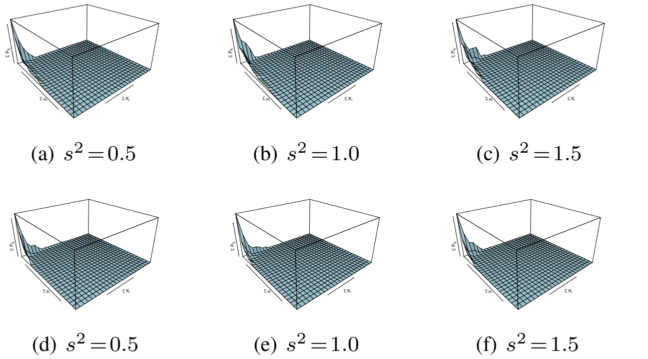
(c) $s^2 = 1.5$

(d) $s^2 = 0.5$

(e) $s^2 = 1.0$

(f) $s^2 = 1.5$

Fig. 4. Results for the series topology: (a), (b), (c), surfaces draw, solutions initially provided by the NSGA-II; (d), (e), (f), surfaces draw, solutions that have been improved by the MOPSO.



(a) $s^2 = 0.5$

(b) $s^2 = 1.0$

(c) $s^2 = 1.5$

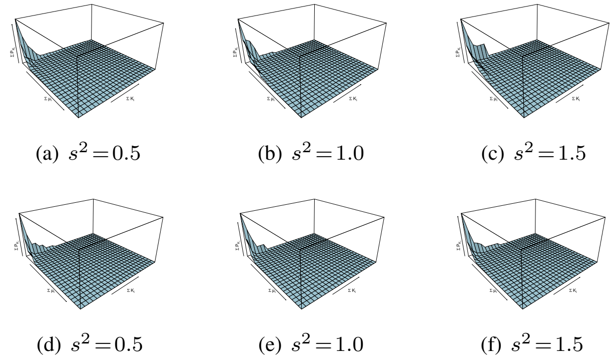
(d) $s^2 = 0.5$

(e) $s^2 = 1.0$

(f) $s^2 = 1.5$

Fig. 5. Results for the split topology: (a), (b), (c), surfaces draw, solutions initially provided by the NSGA-II; (d), (e), (f), surfaces draw, solutions that have been improved by the MOPSO.

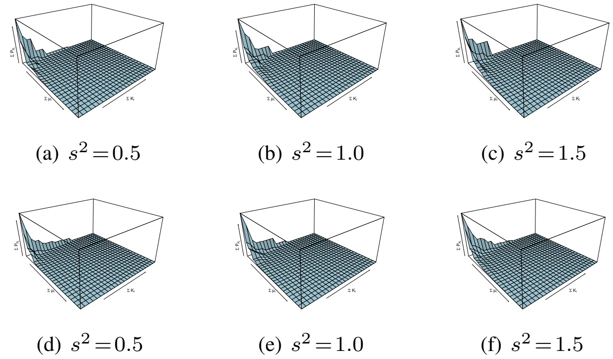
Table I and Figs. 4, 5, 6, and 7, inform those post-processed solutions are promising. However, decision-making, with this information alone, is still dubious. A reduction in the number of buffers is a reduction in total cost. However, if a reduction in buffers occurs simultaneously with some increase in service rates, will this new solution be adequate or less adequate (from an economic point of view)? In several real-life problems, the answer to this question is directly linked to costs.



(a) $s^2 = 0.5$ (b) $s^2 = 1.0$ (c) $s^2 = 1.5$

(d) $s^2 = 0.5$ (e) $s^2 = 1.0$ (f) $s^2 = 1.5$

Fig. 6. Results for the merge topology: (a), (b), (c), surfaces draw, solutions initially provided by the NSGA-II; (d), (e), (f), surfaces draw, solutions that have been improved by the MOPSO.



(a) $s^2 = 0.5$ (b) $s^2 = 1.0$ (c) $s^2 = 1.5$

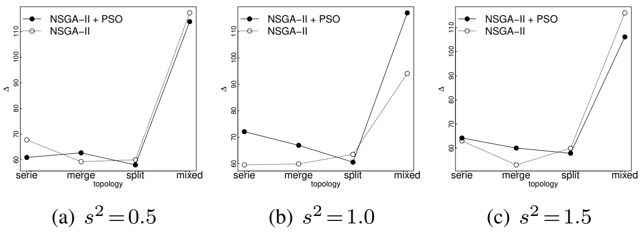
(d) $s^2 = 0.5$ (e) $s^2 = 1.0$ (f) $s^2 = 1.5$

Fig. 7. Results for the mixed topology: (a), (b), (c), surfaces draw, solutions initially provided by the NSGA-II; (d), (e), (f), surfaces draw, solutions that have been improved by the MOPSO.

A. Performance Evaluation

The performance of multi-objective optimization algorithms is often assessed using diversity indicators. The interest is to measure the quality of the spread of solutions obtained by the algorithm used [52]. The best solution proposal tends to be the greatest spread but with the most uniform possible spacing between solutions.

A comparative analysis between the solutions provided by NSGA-II and the solutions subsequently post-processed by MOPSO needs adequate metrics for comparing multi-objective solution sets. The Δ metric was introduced in [53], $\Delta = \frac{1}{|S|-1}\sum_{i=1}^{|S|-1}|d_i - \bar{d}|$, in which d_i is the Euclidean distance between consecutive elements belonging to the Pareto front S. A Pareto-set is better to spread when the value of the Δ metric is lower. In other words, a smaller value of Δ implies more diversified solutions. Fig. 8 presents the results of the Δ metric for the solutions provided by NSGA-II and the solutions subsequently post-processed by MOPSO.



(a) $s^2 = 0.5$ (b) $s^2 = 1.0$ (c) $s^2 = 1.5$

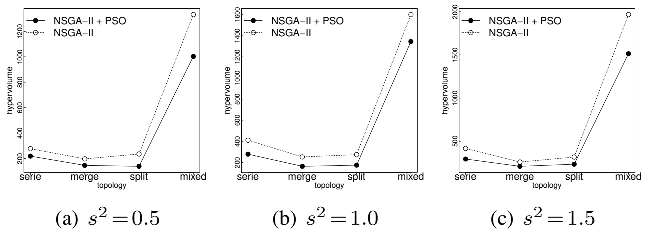
Fig. 8. Evolution of Δ metric for hypo-exponential, Markovian, and hyper-exponential queueing networks.

Four different topologies were evaluated, with three values for the square of the coefficient of variation. Thus, there are 12 different configurations under study. For 6 of these configurations, the Pareto front obtained after post-processing showed greater diversity. This fact shows a balance with the previous solutions provided by the NSGA-II algorithm. The behavior of post-processed solutions was superior in variety for systems with hypo-exponential service. In this situation, the post-processed solutions for the merge topology did not show greater diversity. The solutions in Markovian systems, using the NSGA-II algorithm, did not show greater diversity only for the split topology. There is a balance between solutions for hyperexponential service, with superiority for post-processed solutions in the split and mixed topologies. Post-processed solutions showed greater diversity whenever there was no Markovian service for the most complex (mixed) topology among those evaluated. Objectively, we notice an oscillation in the spreading levels in the Pareto-front, but with similar results between NSGA-II and NSGA-II post-processed by MOPSO. The differences in spreading are just stochastic fluctuations and are not indicative of effective improvement or worsening due to post-processing.

Another comparison strategy may be to consider the hypervolume metric (HV) [54]. The hypervolume measures the volume of the space dominated by the Pareto front. The hypervolume evaluates both the coverage and the diversity of the solutions. The hypervolume between the Pareto front and the origin in the objective space will be considered for this investigation. Thus, the superior Pareto front has the smallest verified hypervolume. The hypervolume metric used is defined by:

$$HV(S) = \text{volume}\left(\bigcup_{i=1}^{|S|} v_i\right)$$

where S is the Pareto front generated by the algorithm, v_i is the hypercube formed by the solution $s_i \in S$ and the origin of the objective space.



(a) $s^2 = 0.5$ (b) $s^2 = 1.0$ (c) $s^2 = 1.5$

Fig. 9. Evolution of HV metric for hypo-exponential, Markovian, and hyperexponential queueing networks.

In Fig. 9, the Pareto front obtained by the post-processing strategy was better in all evaluated configurations. Unlike verified for the Δ metric, this fact shows that the post-processing through the MOPSO algorithm provides an effective gain in the solutions previously offered by the NSGA-II algorithm.

V. Conclusions and Final Remarks

This article discusses an alternative form of mathematical formulation for the multi-objective optimization problem. The sum of blocking probabilities in the queueing network is adopted as a minimization objective. Thus, there is a reduction of blockages in the queueing network flow, and general performance is improved. We minimized the sum of the blocking probabilities of a general-service time, single-server, finite, acyclic queueing network along with the total capacity and the overall service rate. The main objective of this article was to evaluate algorithms for post-processing solutions. We obtained previous solutions from efficient known multi-objective evolutionary algorithms of a finite general-service acyclic queueing network.

The performance evaluation shows that the post-processing strategy through multi-objective particle swarm optimization is promising because it can improve the hypervolume of the Pareto front previously obtained. The computational results presented confirm that the evolutionary algorithm produces very effective sub-optimal solutions to the problem. However, the previous Pareto front could be significantly improved by a post-processing strategy with a reasonable amount of extra computational effort.

The improvement on the Pareto front occurred in some situations in the evaluation by indicator Δ. Besides, improvements are observed in all cases assessed for the hypervolume metric, independently of the topology. The proposed post-processing strategy appeared to be appropriate for the stochastic optimization problem under consideration, and it might also be suitable for other similar optimization problems.

Topics for future research in this area include considering different types of queues, such as multi-server Markovian queues, $M/M/c$, finite multi-server Markovian queues, $M/M/c/K$, and so on.

References

[1] J. MacGregor Smith and F. R. B. Cruz, "The buffer allocation problem for general finite buffer queueing networks," *IIE Transactions*, vol. 37, no. 4, pp. 343–365, 2005.

[2] G. L. Souza, A. R. Duarte, G. J. P. Moreira, and F. R. B. Cruz, "A novel formulation for multi-objective optimization of general finite single-server queueing networks," in *IEEE Congress on Evolutionary Computation (CEC)*, pp. 1–8, 2020.

[3] F. R. B. Cruz, A. R. Duarte, and T. van Woensel, "Buffer allocation in general single-server queueing networks," *Computers & Operations Research*, vol. 35, no. 11, pp. 3581–3598, 2008.

[4] F. R. B. Cruz, A. R. Duarte, and G. L. Souza, "Multi-objective performance improvements of general finite single-server queueing networks," *Journal of Heuristics*, vol. 24, no. 5, pp. 757–781, 2018.

[5] S. Yelkenci Kose and O. Kilincci, "A multi-objective hybrid evolutionary approach for buffer allocation in open serial production lines," *Journal of Intelligent Manufacturing*, vol. 31, no. 1, pp. 33–51, 2020.

[6] I. Zennaro, S. Finco, R. Aldrighetti, and D. Battini, "Buffer size evaluation in a bottle plant production system: a comparison between different solving methods," *International Journal of Services and Operations Management*, vol. 42, no. 4, pp. 500–524, 2022.

[7] J. MacGregor Smith, "Simultaneous buffer and service rate allocation in open finite queueing networks," *IISE Transactions*, vol. 50, no. 3, pp. 203–216, 2018.

[8] J. O. Hernández-Vázquez, S. Hernández-González, J. A. Jiménez-García, M. D. Hernández-Ripalda, and J. I. Hernández-Vázquez, "Enfoque híbrido metaheurístico ag-rs para el problema de asignación del buffer que minimiza el inventario en proceso en líneas de producción abiertas en serie," *Revista Iberoamericana de Automática e Informática Industrial*, vol. 16, no. 4, pp. 447–458, 2019.

[9] J. MacGregor Smith, F. R. B. Cruz, and T. van Woensel, "Topological network design of general, finite, multi-server queueing networks," *European Journal of Operational Research*, vol. 201, no. 2, pp. 427–441, 2010.

[10] E. Almehdawe, B. Jewkes, and Q.-M. He, "Optimization in a two-stage multi-server service system with customer priorities," *Journal of the Operational Research Society*, vol. 70, no. 2, pp. 326–337, 2019.

[11] A. Ingolfsson, E. Almehdawe, A. Pedram, and M. Tran, "Comparison of fluid approximations for service systems with state-dependent service rates and return probabilities," *European Journal of Operational Research*, vol. 283, no. 2, pp. 562–575, 2020.

[12] F. R. B. Cruz, T. Van Woensel, J. MacGregor Smith, and K. Lieckens, "On the system optimum of traffic assignment in $M/G/c/c$ state-dependent queueing networks," *European Journal of Operational Research*, vol. 201, no. 1, pp. 183–193, 2010.

[13] R. Khalid, M. K. M. Nawawi, L. A. Kawsar, N. A. Ghani, A. A. Kamil, and A. Mustafa, "Optimal routing of pedestrian flow in a complex topological network with multiple entrances and exits," *International Journal of Systems Science*, vol. 51, no. 8, pp. 1325–1352, 2020.

[14] J. Liu, L. Hu, X. Xu, and J. Wu, "A queuing network simulation optimization method for coordination control of passenger flow in urban rail transit stations," *Neural Computing and Applications*, vol. 33, no. 17, pp. 10935–10959, 2021.

[15] N. U. Ahmed and X. H. Ouyang, "Suboptimal RED feedback control for buffered TCP flow dynamics in computer network," *Mathematical Problems in Engineering*, vol. 2007, no. Article ID 54683, p. 17 pages, 2007.

[16] J. Chen, C. Hu, and Z. Ji, "An improved ARED algorithm for congestion control of network transmission," *Mathematical Problems in Engineering*, vol. 2010, no. Article ID 329035, p. 17 pages, 2010.

[17] V. Inzillo, F. De Rango, and A. A. Quintana, "A self clocked fair queuing MAC approach limiting deafness and round robin issues in directional MANET," in *2019 Wireless Days (WD)*, pp. 1–6, IEEE, 2019.

[18] E. Pourjavad and E. Almehdawe, "Optimization of the technician routing and scheduling problem for a telecommunication industry," *Annals of Operations Research*, vol. 315, no. 1, pp. 371–395, 2022.

[19] K. Chaudhuri, A. Kothari, R. Pendavingh, R. Swaminathan, R. Tarjan, and Y. Zhou, "Server allocation algorithms for tiered systems," *Algorithmica*, vol. 48, no. 2, pp. 129–146, 2007.

[20] D. A. Menascé, "QoS issues in web services," *IEEE Internet Computing*, vol. 6, no. 6, pp. 72–75, 2002.

[21] F. R. B. Cruz, G. Kendall, L. While, A. R. Duarte, and N. C. L. Brito, "Throughput maximization of queueing networks with simultaneous minimization of service rates and buffers," *Mathematical Problems in Engineering*, vol. 2012, no. Article ID 692593, p. 19 pages, 2012.

[22] K. Deb, *Multi-objective optimisation using evolutionary algorithms*. New York, NY: John Wiley & Sons, Inc., 2001.

[23] C. A. Coello Coello, G. B. Lamont, D. A. Van Veldhuizen, *et al.*, *Evolutionary algorithms for solving multi-objective problems*, vol. 5. Springer, 2007.

[24] W. Leong and G. G. Yen, "PSO-based multiobjective optimization with dynamic population size and adaptive local archives," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 38, no. 5, pp. 1270–1293, 2008.

[25] V. Hajipour and S. H. R. Pasandideh, "Proposing an adaptive particle swarm optimization for a novel bi-objective queuing facility location model," *Economic Computation and Economic Cybernetics Studies and Research*, vol. 46, no. 3, pp. 223–240, 2012.

[26] M. Sharafi and T. Y. ELMekkawy, "Multi-objective optimal design of hybrid renewable energy systems using PSO-simulation based approach," *Renewable Energy*, vol. 68, pp. 67–79, 2014.

[27] W. Deng, H. Zhao, X. Yang, J. Xiong, M. Sun, and B. Li, "Study on an improved adaptive PSO algorithm for solving multi-objective gate assignment," *Applied Soft Computing*, vol. 59, pp. 288–302, 2017.

[28] P. Azimi and A. Asadollahi, "Developing a new bi-objective functions model for a hierarchical location-allocation problem using the queuing theory and mathematical programming," *Journal of Optimization in Industrial Engineering*, vol. 12, no. 2, pp. 149–154, 2019.

[29] D. R. X. Oliveira, G. J. P. Moreira, A. R. Duarte, A. L. F. Cançado, and E. Luz, "Spatial cluster analysis using particle swarm optimization and dispersion function," *Communications in Statistics - Simulation and Computation*, pp. 1–18, 2019.

[30] J. MacGregor Smith, F. R. B. Cruz, and T. van Woensel, "Optimal server allocation in general, finite, multi-server queueing networks," *Applied Stochastic Models in Business & Industry*, vol. 26, no. 6, pp. 705–736, 2010.

[31] J. MacGregor Smith, "Optimal workload allocation in closed queueing networks with state dependent queues," *Annals of Operations Research*, vol. 231, no. 1, pp. 157–183, 2015.

[32] A. R. Duarte, “The server allocation problem for Markovian queueing networks,” *International Journal of Services and Operations Management*, vol. (*to appear*), pp. 1–16, 2022.

[33] H. S. R. Martins, F. R. B. Cruz, A. R. Duarte, and F. L. P. Oliveira, “Modeling and optimization of buffers and servers in finite queueing networks,” *OPSEARCH*, vol. 56, no. 1, pp. 123–150, 2019.

[34] L. Kerbache and J. MacGregor Smith, “Multi-objective routing within large scale facilities using open finite queueing networks,” *European Journal of Operational Research*, vol. 121, no. 1, pp. 105–123, 2000.

[35] F. R. B. Cruz, “Optimizing the throughput, service rate, and buffer allocation in finite queueing networks,” *Electronic Notes in Discrete Mathematics*, vol. 35, pp. 163 – 168, 2009. LAGOS’09 - V Latin-American Algorithms, Graphs and Optimization Symposium.

[36] T. van Woensel, R. Andriansyah, F. R. B. Cruz, J. MacGregor Smith, and L. Kerbache, “Allocation in general multi-server queueing networks,” *International Transactions in Operational Research*, vol. 17, no. 2, pp. 257–286, 2010.

[37] T. van Woensel, R. Andriansyah, F. R. B. Cruz, J. MacGregor Smith, and L. Kerbache, “Buffer and server allocation in general multi-server queueing networks,” *International Transactions in Operational Research*, vol. 17, no. 2, pp. 257–286, 2010.

[38] R. Andriansyah, T. van Woensel, F. R. B. Cruz, and L. Duczmal, “Performance optimization of open zero-buffer multi-server queueing networks,” *Computers & Operations Research*, vol. 37, no. 8, pp. 1472–1487, 2010.

[39] F. R. B. Cruz, T. van Woensel, and J. MacGregor Smith, “Buffer and throughput trade-offs in $M/G/1/K$ queueing networks: A bi-criteria approach,” *International Journal of Production Economics*, vol. 125, no. 2, pp. 224–234, 2010.

[40] T. van Woensel and F. R. B. Cruz, “Optimal routing in general finite multi-server queueing networks,” *PLoS ONE*, vol. 9, p. e102075, July 2014.

[41] J. MacGregor Smith, “Optimal design and performance modelling of $M/G/1/k$ queueing systems,” *Mathematical and Computer Modelling*, vol. 39, no. 9-10, pp. 1049–1081, 2004.

[42] T. Kimura, “A transform-free approximation for the finite capacity $M/G/s$ queue,” *Operations Research*, vol. 44, no. 6, pp. 984–988, 1996.

[43] J. MacGregor Smith, “$M/G/c/k$ blocking probability models and system performance,” *Performance Evaluation*, vol. 52, no. 4, pp. 237–267, 2003.

[44] D. Gross, J. F. Shortle, J. M. Thompson, and C. M. Harris, *Fundamentals of queueing theory*. New York, NY: Wiley - Interscience, 4th edition ed., 2009.

[45] L. Kerbache and J. MacGregor Smith, “The generalized expansion method for open finite queueing networks,” *European Journal of Operational Research*, vol. 32, pp. 448–461, 1987.

[46] J. Kennedy and R. Eberhart, “Particle swarm optimization,” in *Proceedings, IEEE International Conference on Neural Networks*, vol. 4, pp. 1942–1948, 1995.

[47] C. A. Coello Coello and M. S. Lechuga, “MOPSO: A proposal for multiple objective particle swarm optimization,” in *Proceedings of the 2002 Congress on Evolutionary Computation. CEC’02 (Cat. No.02TH8600)*, vol. 2, pp. 1051–1056, 2002.

[48] V. Trivedi, P. Varshney, and M. Ramteke, “A simplified multi-objective particle swarm optimization algorithm,” *Swarm Intelligence*, vol. 14, no. 2, pp. 83–116, 2020.

[49] Z. Fan, T. Wang, Z. Cheng, G. Li, and F. Gu, “An improved multiobjective particle swarm optimization algorithm using minimum distance of point to line,” *Shock and Vibration*, vol. 2017, pp. 1–16, 2017.

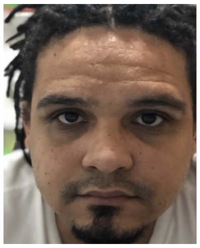
[50] C. Jia and H. Zhu, “An improved multiobjective particle swarm optimization based on culture algorithms,” *Algorithms*, vol. 10, no. 2, p. 46, 2017.

[51] X. Zhao, Y. Jin, H. Ji, J. Geng, X. Liang, and R. Jin, “An improved mixed-integer multi-objective particle swarm optimization and its application in antenna array design,” in *5th IEEE International Symposium on Microwave, Antenna, Propagation and EMC Technologies for Wireless Communications*, pp. 412–415, 2013.

[52] S. Jiang, Y.-S. Ong, J. Zhang, and L. Feng, “Consistencies and contradictions of performance metrics in multiobjective optimization,” *IEEE transactions on cybernetics*, vol. 44, no. 12, pp. 2391–2404, 2014.

[53] K. Deb, S. Agarwal, A. Pratap, and T. Meyarivan, “A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II,” in *International Conference on Parallel Problem Solving from Nature*, pp. 849–858, Springer, 2000.

[54] E. Zitzler and L. Thiele, “Multiobjective optimization using evolutionary algorithms—a comparative case study,” in *International conference on parallel problem solving from nature*, pp. 292–301, Springer, 1998.



Gabriel L. de Souza holds a Bachelor’s degree in statistics as well as a Master’s degree in computer science from the Universidade Federal de Ouro Preto (UFOP), in 2014 and 2020, respectively. He is currently a research scholar seeking a Ph.D. degree in computer science at UFOP.

Anderson R. Duarte holds a Bachelor’s degree in mathematics as well as a Master’s degree and a Doctorate in statistics from the Universidade Federal de Minas Gerais, in 2000, 2005 and 2009, respectively. He is currently an Associate Professor at the Department of Statistics at the Universidade Federal de Ouro Preto and conducts research in multi-objective optimization, simulation and operations research.

Gladston J. P. Moreira holds a Master’s degree in mathematics from the Federal University of Minas Gerais in 2003, and a Doctorate in electrical engineering in 2011. He is currently an Associate Professor at the Department of Computing at the Universidade Federal de Ouro Preto. His research interests include multi-objective optimization, pattern recognition and spatial statistics.

Frederico R. B. Cruz holds a Bachelor’s degree in electrical engineering (1988) as well as a Master’s degree (1991) and a Doctorate (1997) in computer science from the Universidade Federal de Minas Gerais, where he is full professor in the Department of Statistics and conducts research in operations research.