Proceedings of the 2016 Winter Simulation Conference
T. M. K. Roeder, P. I. Frazier, R. Szechtman, E. Zhou, T. Huschka, and S. E. Chick, eds.

DISCRETE EVENT OPTIMIZATION: WORKSTATION AND BUFFER ALLOCATION PROBLEM IN MANUFACTURING FLOW LINES

Mengyi Zhang
Andrea Matta

Department of Industrial Engineering & Management
School of Mechanical Engineering
Shanghai Jiao Tong University
800 Dongchuan Road
Shanghai, 200240, CHINA

Giulia Pedrielli

School of Computing, Informatics,
& Decision Systems Engineering
Arizona State University
699 S Mill Avenue
Tempe, AZ 85281, USA

ABSTRACT

Resource and buffer allocation problems are well-known topics in manufacturing system research. A proper allocation of resource and space can significantly improve the system performance and reduce the investment cost. However, few works consider the joint problem because of its complexity. Recent research has shown that Discrete Event Optimization (DEO) framework, an integrated simulation-optimization approach based on mathematical programming, can be used to optimize buffer allocation of production lines, such as open and closed flow lines and pull controlled manufacturing systems. This paper proposes mathematical programming models for solving the joint workstation and buffer allocation problem in manufacturing flow lines constrained to a given target throughput. The problem is formulated in two different ways: an exact model using mixed integer linear programming formulation and approximate models using linear programming formulation. Numerical analysis shows that efficiency and accuracy can be both achieved by using approximate formulations in a math-heuristic procedure.

1 INTRODUCTION

Both resource allocation problems and buffer allocation problems are well-known topics of research in manufacturing system design. Among all related literatures, only a few works dealt with joint resource and buffer allocation problems, and all of them solved joint problems assuming the system layout is given, i.e. finding out the service rate and buffer capacity of each workstation other than designing the networks. The joint optimization of both server and buffer allocation of a single station system was solved by Shanthikumar and Yao (1987). Hillier and So (1995) proposed an enumeration method to find the optimal number of servers and buffer capacities in open networks. Spinellis, Papadopoulos, and Smith (2000) used a simulated annealing algorithm to solve the same joint allocation optimization problem for long production lines. Woensel et al. (2010) discussed the problem of acyclic configured M/G/c/K queuing networks under the assumption of Poisson arrivals and exponential service rates. To solve this problem, they used Lagrangian relaxation to approximate the joint buffer and server optimization problem, Powell's search method to optimize the relaxed problem, and a two-moment approximation to compute the mean throughput.

The whole problem analyzed in this paper can be decomposed as workstation allocation, workload allocation and buffer allocation. Workstation allocation and workload allocation are both resource allocation problems in manufacturing systems. Some examples of resource allocation are the server allocation, the assembly line balancing, the machine grouping problems in cellular manufacturing, the machine loading and tool allocation problems in flexible manufacturing systems. The server allocation problem has the goal to allocate parallel servers at each workstation (Boxma, Kan, and Vliet 1990). Assembly line balancing

problems partition tasks among workstations arranged along a flow oriented material handling equipment constrained by a cycle time, where the number of workstations is fixed. The survey of Becker and Scholl (2006) summarized assembly line balancing problems and the related methods for optimization and evaluation. Machine grouping problems in cellular manufacturing systems allocate different machines to each cell to maximize compatibility between machines and parts or to seek a trade-off between machine cost and intercell movement cost (Gunasingh and Lashkari 1989). Machine loading and tool allocation problems in flexible manufacturing systems help to decide which tools and operations of each part family are allocated to which machine (Sarin and Chen 1987). The resources allocated in these problems are machines. Demir, Tunali, and Eliiyi (2014) reviewed 110 articles on buffer allocation problems. Most of them proposed different algorithms like simulated annealing, tabu search and evolutionary algorithms for solving buffer allocation problems having an objective function of the throughput maximization. Only a few articles solved the optimization under the objective of minimization of total buffer space.

The joint problem to be solved in this paper, however, differs from all problems in previous works. Existing literatures deal with either resource allocation, or buffer allocation or joint server and buffer allocations of manufacturing systems where the number of workstations is fixed. This paper will help to design an open flow line by providing the total number of workstations, the workload at each workstation and the buffer capacity at each stage subject to a target throughput. Therefore, the analyzed problem is more general than others because it embraces three different problems related one each other.

Manufacturing systems are Discrete Event Systems (DES) whose simulation process can be analytically modeled into Mathematical Programming (MP) formulation (Chan and Schruben 2003). The times at which events occur in the simulation is the solution of this MP under the objective of minimization of the sum of all event times. Solving the MP model means finding the evolution trajectory of the system during the simulation. Optimization constraints like limited buffer capacity and a target throughput can also be formulated in the MP. This Discrete Event Optimization (DEO) framework, an integrated simulation-optimization method based on the MP, is proposed to optimize buffer allocation problem for a class of queuing systems, such as open flow lines (Matta 2008) and pull control manufacturing systems (Pedrielli, Matta, and Alfieri 2015b). The optimal solution from this method is the global optimal based on one simulation sample path. Other examples of enhancement of the DEO approach can be found in Tan (2015), Stolletz and Weiss (2013). Pedrielli (2013); Pedrielli, Matta, and Alfieri (2015a); Pedrielli, Matta, and Alfieri (2016) proposed a more general framework, i.e. not tailored to a specific simulation optimization problem. Specifically, Pedrielli, Matta, and Alfieri (2016) proposed a formal procedure that encompasses all the steps from the description of Event Relationship graphs (ERGs) for simulation-optimization (ERGLite formalism) to the generation of the mathematical programming formulation.

In this paper, we use the DEO framework to solve the joint workstation and buffer allocation problem. This work differs from previous researches in two aspects: (1) it presents a Mixed Integer Linear Programming (MILP) formulation as an exact representation of the joint workstation and buffer allocation optimization. (2) it develops a math-heuristic algorithm consisting of three steps based on Linear Programming (LP) approximate models of the system. This algorithm finds out the number of workstations in the first step and then allocates workload and buffer space. The MILP formulation of the joint problem, the LP approximate models for workstation allocation and the math-heuristic algorithm are original contributions.

This paper is organized as follows. The problem is described in the next section. The exact model and the approximate models using Mathematical Programming Representation (MPR) of DES for simulation-optimization are formulated in section 3. The three-step math-heuristic algorithm is introduced in section 4. Section 5 reports the application of the math-heuristic to some test cases. Finally, conclusions are drawn in the last section.

2 PROBLEM DESCRIPTION

Manufacturing systems considered in this paper are open flow lines composed of workstations and buffers (Figure 1). Processing times are randomly distributed, thus workstations can be assumed perfectly reliable.

Parts are processed from the first workstation to the last one sequentially. A workstation cannot process more than one part at a certain time, and a part cannot be processed by more than one workstation at the same time. Parts wait in the $(j-1)th$ buffer when jth workstation is working on a previous part. Because of random processing times and limited buffer capacities, workstations can be either working, starving or blocked. The last workstation is never blocked.

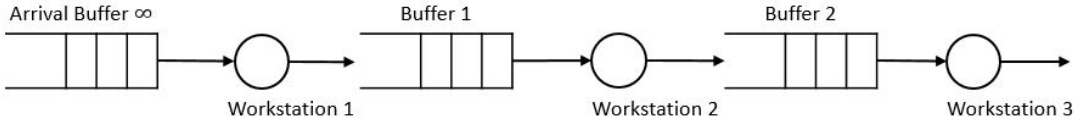

Figure 1: Example of open flow line with 3 workstations.

We want to minimize the investment cost of the system while guaranteeing a minimum production rate. The number of workstations influences the production rate. Since it is assumed that the total processing time to complete a part is given, the longer the line, the higher the throughput. However, the workload and buffer allocation problems change depending on the number of workstations. For instance, a 10-workstation line requires allocating 9 buffers, and splitting the process cycle in 10 partitions, but a 2-workstation line only requires allocating 1 buffer, and splitting the process cycle in 2 partitions. Therefore, workload and buffer allocations are two problems nested in the workstation allocation problem. It is clear that a joint optimization can be more effective.

Parameters for solving this joint problem are the expected total processing times of parts, distributions of processing times at each stage, the target throughput and the random numbers used to generate processing times in simulation. Expected total processing times of parts can be different, which makes the method presented in this paper be proper also to flexible manufacturing lines. Furthermore, processing times may not necessarily be exponential, which is usually an assumption in other researches on open flow lines.

The joint design problem will provide the number of workstations needed, the workload allocated among workstations and the buffer capacity at each stage given a target throughput where the total cost of the flow line is minimized. Workload of a workstation is defined as the proportion of expected processing time at the workstation, therefore it is a real number between 0 and 1. The workload allocated to workstations can also be limited by additional constraints related to manufacturing process, e.g. a bottleneck workstation, minimum workload at some workstations because of some special processing techniques, etc. The objective function, i.e. the total cost of a flow line, consists of workstation cost and buffer cost, and unit costs of both workstation and buffer space are given.

3 MODELING

3.1 Notation

In this section, according to the approach in Pedrielli, Matta, and Alfieri (2016), we present the ERGL model (Figure 2), which is then explained through the related integrated MPR with the notations below.

Parameters

U_M : the upper bound of workstation number.
U_B : the upper bound of buffer capacity at each stage.
N : the total number of parts in simulation experiment.
D : the number of parts in warm-up period.
C_M : the unit workstation cost.
C_B : the unit buffer capacity cost.
A_M : the adjusted unit workstation cost parameter in the approximate model.

A_B : the adjusted unit time buffer cost parameter in the approximate model.
A_i : the arrival time of ith part.
T_i : the expected total processing time of part i, which is the sum of processing time at all stages of the part.
α^* : the target throughput.
$z_{i,j}$: random numbers used to generate processing times $t_{i,j}$.

Event time decision variables
$t_{i,j} \in [0,+\infty)$: the processing time of part i at workstation j.
$F_{i,j} \in [0,+\infty)$: the finishing time of part i at workstation j.

Optimization decision variables
$m_j \in \{0,1\}$: if the jth workstation is allocated in the flow line, $m_j = 1$. Otherwise $m_j = 0$.
$s_j \in [0,1)$: workstation workload, the proportion of workload allocated at the jth workstation.
$x_{j,k} \in \{0,1\}$: if capacity of jth buffer is $k-1$, $x_{j,k} = 1$. Otherwise, $x_{j,k} = 0$.
$r_{j,k} \in [0,+\infty)$: time buffer capacity of the jth workstation (in the approximate model).

3.2 Exact MILP Model

The joint workstation and buffer allocation problem can be formulated in an MILP model that integrates both simulation and optimization. The model is formulated as follows:

$$min\{C_M \sum_{j=1}^{U_M} m_j + C_B \sum_{j=1}^{U_M-1} \sum_{k=1}^{U_B+1} (k-1)x_{j,k}\}$$

Subject to:

$$\sum_{j=1}^{U_M} s_j = 1 \tag{1}$$

$$s_j \leq m_j, \qquad \forall j = 1,2,\ldots,U_M \tag{2}$$

$$m_{j-1} \geq m_j, \qquad \forall j = 2,\ldots,U_M \tag{3}$$

$$\sum_{k=1}^{U_B+1} x_{j,k} = 1, \qquad \forall j = 1,2,\ldots,U_M - 1 \tag{4}$$

$$t_{i,j} = \phi(T_i s_j, z_{i,j}), \qquad \forall j = 1,2,\ldots,U_M, \forall i = 1,2,\ldots,N \tag{5}$$

$$F_{i,1} - t_{i,1} \geq A_i, \qquad \forall i = 1,2,\ldots,N \tag{6}$$

$$F_{i+1,j} - F_{i,j} - t_{i+1,j} \geq 0, \qquad \forall j = 1,2,\ldots,U_M, \forall i = 1,2,\ldots,N-1 \tag{7}$$

$$F_{i,j+1} - F_{i,j} - t_{i,j+1} \geq 0, \qquad \forall j = 1,2,\ldots,U_M - 1, \forall i = 1,2,\ldots,N \tag{8}$$

$$F_{i+k,j} - F_{i,j+1} - t_{i+k,j} + (1 - x_{j,k})M \geq 0, \qquad \forall j = 1,2,\ldots,U_M - 1, \forall k = 1,\ldots,U_B + 1, \forall i = 1,2,\ldots \tag{9}$$

$$\frac{N - D}{F_{N,U_M} - F_{D,U_M}} \geq \alpha^* \tag{10}$$

Constraint (1) states that the workload is completely allocated to the flow line. Constraints (2) describe that when the workload allocated to a workstation is non-zero, this workstation is allocated to the system; otherwise if the workload is zero, the related workstation is not allocated. Constraints (3) impose that all workstations allocated are in the first part of the line, the time for parts passing through unused workstations is 0 and this does not influence the manufacturing process in front. Constraints (6)-(9) describe the

production process which is presented with the ERGL model in Figure 2. Constraints (6) are derived from arcs from A_i to $F_{i,1}$, and it states that the ith part arrives at the line at time A_i. Constraints (7) are derived from horizontal arcs in the ERGL model, and state that one machine cannot process more than one part at a certain time. Constraints (8) are derived from vertical arcs in the ERGL model, and impose that a part cannot be processed by more than one machine at the same time ((6)-(8) (Chan and Schruben 2003)). Constraints (9) are derived from the arcs from $F_{i,j+1}$ to $F_{i+k,j}$, and describe that buffer capacity is finite: if capacity of jth buffer is equal to $k-1$ (which means $x_{j,k}=1$), part $i+k$ cannot enter the jth workstation before the ith part leaves the $(j+1)th$ workstation ((4) and (9) in Matta (2008)). Constraint (10) states that the designed production line should reach a minimum target throughput α^*.

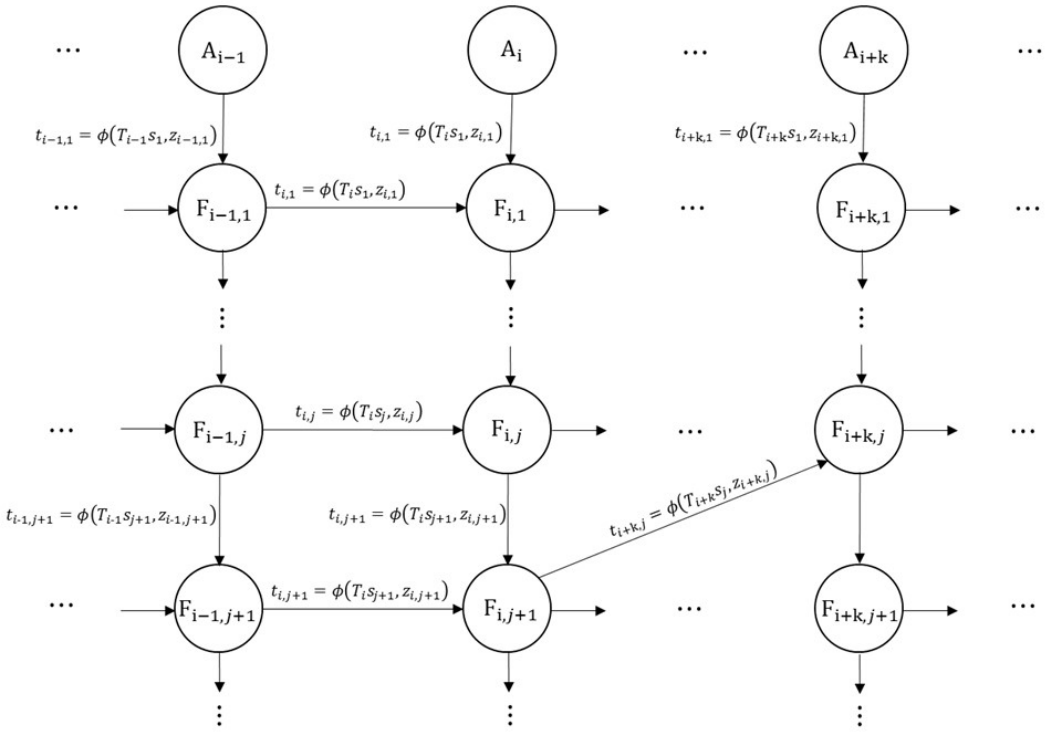

Figure 2: ERGlite Representation.

Constraints (5) deal with random generation of processing times, which are a function ϕ of the expected value $T_i s_j$ and the random numbers $z_{i,j}$. As the value of the decision variable s_j changes, also the generated processing times are modified accordingly. As we solve an MILP problem, the function ϕ should be a linear function of variables s_j to keep low the complexity of the model. Specifically, s_j and $z_{i,j}$ can be combined in an additive or a multiplicative way:

- *Additive combination* A function like $\phi = T_i(s_j + z_{i,j})$ can be used, where $z_{i,j}$ follows a zero-mean distribution. For example, if $z_{i,j}$ follows a uniform distribution on (-0.1, 0.1) and $t_{i,j} = T_i(s_j + z_{i,j})$, then $t_{i,j}$ also follows a uniform distribution on $(T_i(s_j - 0.1), T_i(s_j + 0.1))$.
- *Multiplicative combination* A function like $\phi = T_i s_j f(z_{i,j})$ can be used. In this case, distributions of $z_{i,j}$ and $t_{i,j}$ do not necessarily have the same shape. For example, if $z_{i,j}$ is uniformly distributed in interval (0,1), and $t_{i,j}$ is assumed to follow an exponential distribution with a mean $T_i s_j$, then $t_{i,j} = -T_i s_j ln(1 - z_{i,j})$.

Other constraints can also be considered to be more consistent with industrial reality if additional knowledge is available on the process. For example, constraints (11) impose that the second workstation is the bottleneck of the system. Another useful constraint is (12) which gives a lower bound to the workload of the *jth* workstation.

$$s_2 \geq s_j, \qquad \forall j \tag{11}$$

$$s_j \geq 0.2 \tag{12}$$

By solving this MILP model, the global optimal can be obtained using a single-replication experiment under the DEO framework. As the replication length increases, the optimal solution epi-converges to the optimum (Pedrielli et al. 2016). However, the number of variables and the number of constraints increase significantly as N, U_M or U_B increases. Specifically, the number of binary variables and the number of continuous variables in the model are $U_M U_B$ and $2NU_M + U_M$, respectively. The number of constraints containing binary variables is $NU_M U_B$, and the number of continuous constraints is $3NU_M$. Therefore, when designing long production lines or when considering long simulations, the computational complexity can be very high.

3.3 Approximate Model

One reason for high complexity of the MILP is the large number of integer variables. Thus, replacing these variables by continuous ones is important for solving long production line design or long simulations in reasonable computation time.

Constraints (2)-(4) and (9) and the objective function contain the binary variables m_j and $x_{j,k}$. By using constraints (13), Matta (2008) introduced an LP approximate formulation of buffer allocation binary variables $x_{j,k}$.

$$F_{i+k,j} - F_{i,j+1} \geq t_{i+k,j} - r_{j,k} \tag{13}$$

If the capacity of *jth* buffer is not less than k (which is equivalent to $r_{j,k} > 0$), part $i+k$ can enter machine j before part i leaves machine $j+1$. A larger $r_{j,k}$ means higher necessity to have the *kth* slot. Variables $r_{j,k}$ are also known as time buffer capacity and were extensively studied in Matta (2008); Pedrielli (2013); Pedrielli, Matta, and Alfieri (2015a); Pedrielli, Matta, and Alfieri (2015b).

The total buffer capacity formula in the objective function is replaced by

$$\sum_{j=1}^{U_M-1} \sum_{k=1}^{U_B} r_{j,k}.$$

The objective function is changed into formula (14), in which the minimization of workstation number is guaranteed by giving higher weight on additional workstations.

$$min\{A_M \sum_{j=1}^{U_M} j s_j + A_B \sum_{j=1}^{U_M-1} \sum_{k=1}^{U_B} r_{j,k}\} \tag{14}$$

where A_M and A_B are adjusted unit cost parameters.

The main advantage of this approximate LP model is the higher efficiency compared with the exact model, while the disadvantage is the loss of accuracy, especially for thebuffer allocation problem. Indeed, minimizing (14) leads that upstream workstations have higher workloads, and therefore, higher buffer capacities may be needed for such an unbalanced line. To solve this problem, a three-step math-heuristic algorithm is introduced in section 4.

4 THREE-STEP MATH-HEURISTIC ALGORITHM

The algorithm iteratively uses two models to approximately find out the optimal system configuration. The procedure is illustrated in Figure 3. The approximate model can be decomposed into two models. One model solves the workstation number with infinite buffers. The second model solves the workload and buffer allocation problem with the fixed workstation number N_M. This decomposition works under the assumption that workstation cost is much higher than buffer cost, which means adding an extra workstation is never considered as a good solution if target throughput can be fulfilled by increasing buffer capacity.

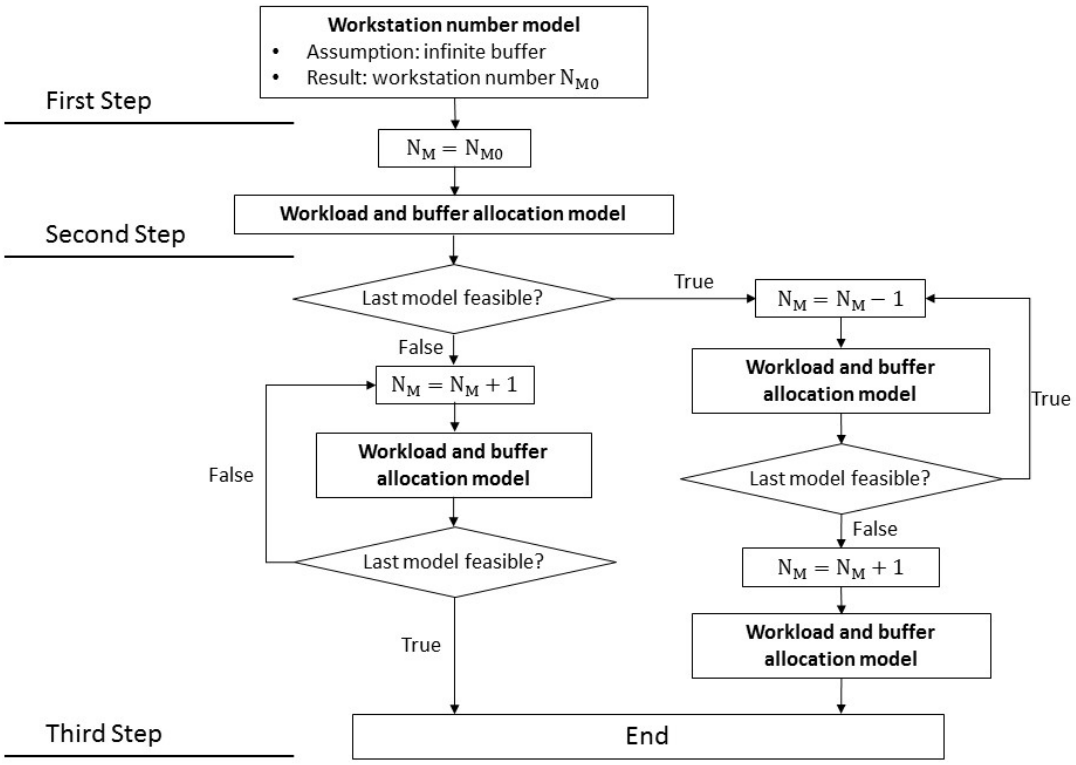

Figure 3: Algorithm outline.

The workstation number model consists of constraints (1), (5)-(8) and(10) and the following objective function:

$$min\{\sum_{j=1}^{U_M} js_j\}.$$

The workload and buffer allocation model consists of constraints (1),(5)-(8),(10) and (13) and the following objective function:

$$min\{\sum_{j=1}^{M-1}\sum_{k=1}^{U_B} r_{j,k}\}.$$

The detailed algorithm is described as follows.

Algorithm 1

Step 1 The workstation number
Solve the workstation number model, and the solution is an approximate workstation number N_{M0}.
Step 2 First iteration of workload and buffer allocation
Workload and buffer allocation model is solved with $N_M = N_{M0}$.
if this model is feasible **then**
  $b = true$
else
  $b = false$
end if
Step 3 Tuning
if $b = true$ **then**
  while $b = true$ **do**
    Solve the workload and buffer allocation model with $N_M = N_M - 1$.
    if *The model is infeasible* **then**
      $b = false$
    end if
  end while
  Solve the workload and buffer allocation model with $N_M = N_M + 1$.
else
  while $b = false$ **do**
    Solve the workload and buffer allocation model with $N_M = N_M + 1$.
    if *The model is feasible* **then**
      $b = ture$
    end if
  end while
end if

5 NUMERICAL ANALYSIS

In this section the application of the proposed math-heuristic algorithm is reported on three cases. In all the cases, the distribution of processing times of the second workstation is a symmetric triangular distribution with $width = 0.2T_i$, i.e. $t_{i,2}$ are distributed on $(T_i(s_2 - 0.1), T_i(s_2 + 0.1))$, $z_{i,2}$ follows a triangular distribution with minimum value -0.1, maximum value 0.1 and the peak of the probability density function at 0. Therefore, function ϕ in constraint (5) becomes

$$t_{i,2} = T_i(z_{i,2} + s_2)$$

Processing times at other workstations are exponentially distributed, i.e. $z_{i,j}$ follows a uniform distribution in $(0,1)$ and the following expression is used:

$$t_{i,j} = -T_i s_j ln(1 - z_{i,j}),\ j \neq 2.$$

All parts arrive at time 0 ($A_i = 0,\ \forall i$). Expected total processing times are 1 time unit for 50% of the parts or 0.5 time unit for 50% of the parts. Boundaries of the problem are $U_M = 10$ and $U_B = 20$. The total part number N is equal to 20000, and the warm-up period consists of 500 parts (identified with Welch's approach). Unit workstation cost C_M is 100 and unit buffer slot cost C_B is 1. The same joint allocation problem is also solved using OptQuest in Arena by running 1000 iterations, where each iteration executes 10 simulations for comparison. Results are verified by simulating the optimal system in Arena with 100000

parts and checking the satisfaction of the throughput constraint (the throughput values are presented in column - α verified in Table 3 with a half width 95% confidence level less than 0.01).

The first case is the design of a flow line with a bottleneck at the second workstation ($s_2 \geq s_j$). The target throughput is varied from 1.5 to 6 parts per time unit. Figure 4 shows the total costs of the systems derived from themath-heuristic and OptQuest. The cost provided by the heuristic is lower than that provided by OptQuest by 8.2% on the average and the gap can be up to 31.1%. In Table 1, we compare two system configurations for $\alpha^* = 5$ by using different methods.

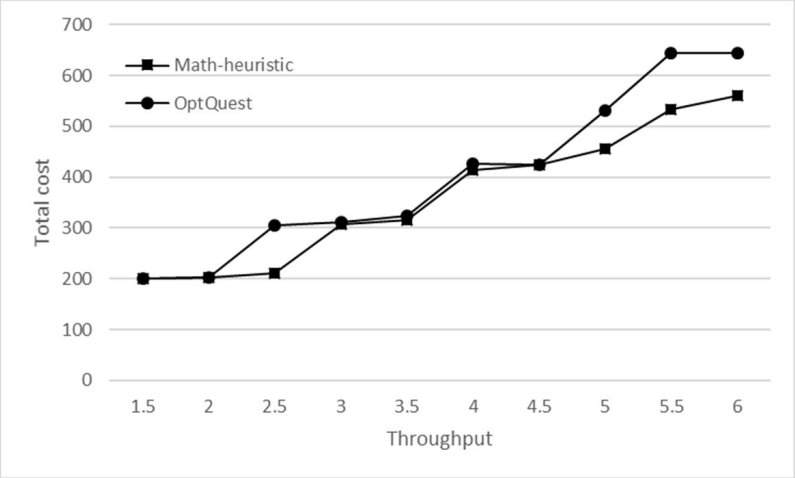

Figure 4: Case 1: comparison of math-heuristic and OptQuest.

Table 1: Case 1: optimal configurations ($\alpha^* = 5$).

Method	Number of Workstations	Total buffer capacity	Workload	Stage buffer capacity	Total cost
Math-Heuristic	4	56	0.25, 0.26, 0.24, 0.25	19, 19, 18	456
OptQuest	5	31	0.23, 0.24, 0.17, 0.12, 0.24	16, 6, 1, 8	531

The second case is the design of more unbalanced lines with a constraint of bottleneck at the second workstation ($s_2 \geq 1.2s_j$). The target throughput is varied from 1.5 to 6 parts per time unit. Other parameters and distribution assumptions are the same as in case 1. Figure 5 shows the total costs of systems derived from the math-heuristic and OptQuest. The cost provided by the heuristic is lower than that provided by OptQuest by 12.7% on the average and the gap can be up to 38.2%. In Table 2, we compare two system configurations for $\alpha^* = 5$ by using different methods.

Table 2: Case 2: optimal configurations ($\alpha = 5$).

Method	Number of Workstations	Total buffer capacity	Workload	Stage buffer capacity	Total cost
Math-Heuristic	5	22	0.20, 0.24, 0.19, 0.17, 0.20	7, 6, 5, 4	522
OptQuest	7	71	0.16, 0.20, 0.05, 0.14, 0.14, 0.17, 0.14	20, 5, 20, 5, 1, 20	771

Table 3 shows that solutions provided by the math-heuristic in both cases can guarantee the throughput target.

In the third case, we choose two tests, each from the last two cases. The two tests are repeated using the math-heuristic algorithm with 10 different random sample paths. $\alpha^* = 6$ is chosen from case 1, and

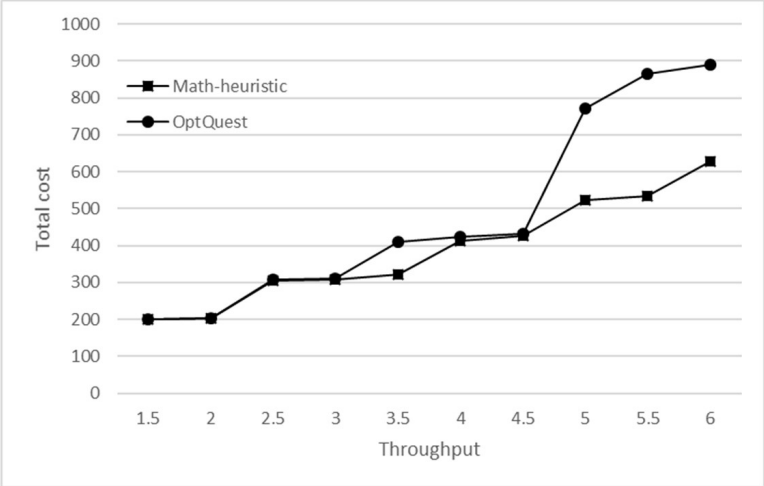

Figure 5: Case 2: comparison of math-heuristic and OptQuest.

Table 3: Throughputs verified using long simulation $N = 100000$.

	Case 1		Case 2	
α^*	α verified of math-heuristic	α verified of OptQuest	α verified of math-heuristic	α verified of OptQuest
1.5	1.9	1.9	1.9	1.6
2	2.1	2.1	2.1	2.1
2.5	2.5	2.8	2.8	2.8
3	3.2	3.2	3.2	3.1
3.5	3.6	3.7	3.5	3.6
4	4.3	4.2	4.2	4.3
4.5	4.6	4.6	4.6	4.6
5	5.0	5.2	5.3	5.7
5.5	5.7	5.7	5.6	6.7
6	6.1	6.5	6.2	6.8

$\alpha^* = 4$ is chosen from case 2. Table 4 and Table 5 show the results of these experiments. Similar with the first two cases, all results provided by the math-heuristic are better than the OptQuest results. It is possible to notice that the solution found by the heuristic is quite stable. Indeed, the number of allocated workstations does not change in the ten replications, whereas the total allocated buffer ranges from 59 to 63 ($\alpha^* = 6$) and from 13 to 15 ($\alpha^* = 4$).

The computation time is around 10 minutes on average for solving one problem using the math-heuristic algorithm, while the time for OptQuest is around 20 minutes. Experiments show that the proposed algorithm is both efficient and accurate.

CONCLUSION

This work proposes different MPRs and a math-heuristic algorithm based on the MPRs for solving the joint workstation and buffer allocation problems, both of which are global search methods. Numerical results show that the proposed math-heuristic is both efficient and accurate. However, the exact model in large scale cannot be solved in reasonable computational time. Future work will be dedicated to solve efficiently the integrated simulation-optimization model by using decomposition approaches from MILP theory.

Table 4: Case 3: results of 10 different sample paths using the same parameters and constraints as in case 1 with $\alpha^* = 6$.

Number of Workstations	Total buffer capacity	Workload	Stage buffer capacity	α verified
5	59	0.21, 0.21, 0.2, 0.19, 0.19	19, 13, 14, 13,	6
5	59	0.21, 0.21, 0.2, 0.19, 0.19	18, 14, 14, 13	6.1
5	59	0.21, 0.21, 0.2, 0.19, 0.19	18, 14, 14, 13	6.1
5	60	0.21, 0.21, 0.19, 0.19, 0.2	19, 14, 14, 13	6
5	62	0.21, 0.21, 0.19, 0.19, 0.2	19, 15, 15, 13	6
5	60	0.21, 0.21, 0.19, 0.19, 0.2	18, 14, 14, 14	6
5	62	0.21, 0.21, 0.19, 0.19, 0.2	19, 14, 15, 14	6.1
5	63	0.21, 0.21, 0.2, 0.19, 0.19	19, 15, 15, 14,	6.1
5	62	0.21, 0.21, 0.19, 0.19, 0.2	19, 14, 16, 13	6
5	63	0.21, 0.21, 0.2, 0.19, 0.19	19, 15, 15, 14	6.1

Table 5: Case 3: results of 10 different sample paths using the same parameters and constraints as in case 2 with $\alpha^* = 4$.

Number of Workstations	Total buffer capacity	Workload	Stage buffer capacity	α verified
4	14	0.24, 0.29, 0.22, 0.25	5, 5, 4	4.2
4	14	0.24, 0.29, 0.22, 0.25	5, 5, 4	4.2
4	14	0.24, 0.29, 0.22, 0.25	5, 5, 4	4.2
4	14	0.24, 0.29, 0.22, 0.25	5, 5, 4	4.2
4	14	0.24, 0.29, 0.22, 0.25	5, 5, 4	4.2
4	15	0.24, 0.29, 0.22, 0.25	6, 5, 4	4.3
4	13	0.24, 0.29, 0.22, 0.25	5, 4, 4	4.2
4	13	0.24, 0.29, 0.22, 0.25	5, 4, 4	4.2
4	14	0.24, 0.29, 0.22, 0.25	5, 5, 4	4.2
4	14	0.27, 0.27, 0.21, 0.25	6, 4, 4	4.3

REFERENCES

Becker, C., and A. Scholl. 2006. "A Survey on Problems and Methods in Generalized Assembly Line Balancing". *European Journal of Operational Research* 168 (3): 694–715.

Boxma, O. J., A. R. Kan, and M. V. Vliet. 1990. "Machine Allocation Problems in Manufacturing Networks". *European Journal of Operational Research* 45 (1): 47–54.

Chan, W. K., and L. W. Schruben. 2003. "Properties of Discrete Event Systems from Their Mathematical Programming Representations". In *Proceedings of the 2003 Winter Simulation Conference*, edited by S. Chick, P. J. Sanchez, D. Ferrin, and D. J. Morrice, 496–502. New Orleans, Louisiana: Institute of Electrical and Electronics Engineers, Inc.

Demir, L., S. Tunali, and D. T. Eliiyi. 2014. "The State of the Art on Buffer Allocation Problem: a Comprehensive Survey". *Journal of Intelligent Manufacturing* 25 (3): 371–392.

Gunasingh, K. R., and R. S. Lashkari. 1989. "Machine Grouping Problem in Cellular Manufacturing Systems–an Integer Programming Approach". *The International Journal of Production Research* 27 (9): 1465–1473.

Hillier, F. S., and K. C. So. 1995. "On the Optimal Design of Tandem Queueing Systems with Finite Buffers". *Queueing Systems* 21 (3-4): 245–266.

Matta, A. 2008. "Simulation Optimization with Mathematical Programming Representation of Discrete Event Systems". In *Proceedings of the 2008 Winter Simulation Conference*, edited by T. Jefferson, J. Fowler, S. Mason, R. Hill, L. Moench, and O. Rose, 1393–1400. Miami, Florida: Winter Simulation Conference.

Pedrielli, G. 2013. *Discrete Event Systems Simulation-Optimization: Time Buffer Framework*. Ph. D. thesis, Mechanical Engineering Department, Politecnico di Milano, Italy.

Pedrielli, G., A. Matta, and A. Alfieri. 2015a. "Discrete Event Optimization: Single-Run Integrated Simulation-Optimization Using Mathematical Programming". In *Proceedings of the 2015 Winter Simulation Conference*, edited by C. M. Macal, M. D. Rossetti, L. Yilmaz, I. Moon, W. K. Chan, and T. Roeder, 3557–3568. Huntington Beach, California: Institute of Electrical and Electronics Engineers, Inc.

Pedrielli, G., A. Matta, and A. Alfieri. 2015b. "Integrated Simulation-Optimization of Pull Control Systems". *International Journal of Production Research* 53 (14): 4317–4336.

Pedrielli, G., A. Matta, and A. Alfieri. 2016. "DEO: Integrated Simulation-Optimizatin of Queueing Systems". *Working Paper*.

Sarin, S. C., and C. S. Chen. 1987. "The Machine Loading and Tool Allocation Problem in a Flexible Manufacturing System". *International Journal of Production Research* 25 (7): 1081–1094.

Shanthikumar, J. G., and D. D. Yao. 1987. "Optimal Server Allocation in a System of Multi-Server Stations". *Management Science* 33 (9): 1173–1180.

Spinellis, D., C. Papadopoulos, and J. G. Smith. 2000. "Large Production Line Optimization Using Simulated Annealing". *International Journal of Production Research* 38 (3): 509–541.

Stolletz, R., and S. Weiss. 2013, June. "Buffer Allocation Using Exact Linear Programming Formulations and Sampling Approaches". In *Preprints of the 2013 IFAC Conference on Manufacturing Modelling, Management and Control*. Saint Petersburg, Russia.

Tan, B. 2015. "Mathematical Programming Representations of the Dynamics of Continuous-Flow Production Systems". *IIE Transactions* 47 (2): 173–189.

Woensel, T. V., R. Andriansyah, F. Cruz, J. G. Smith, and L. Kerbache. 2010. "Buffer and Server Allocation in General Multi-Server Queueing Networks". *International Transactions in Operational Research* 17 (2): 257–286.

AUTHOR BIOGRAPHIES

MENGYI ZHANG is M.S. student of Department of Industrial Engineering and Management at Shanghai Jiao Tong University. Her research focuses on simulation-optimization based on mathematical programming. Her email address is myra@sjtu.edu.cn.

ANDREA MATTA is Distinguished Professor at the Department of Industrial Engineering and Management at Shanghai Jiao Tong University, where he currently teaches stochastic models and simulation. His research area includes analysis and design of manufacturing and health care systems. His email address is matta@sjtu.edu.cn.

GIULIA PEDRIELLI is currently Assistant Professor for the School of Computing, Informatics, and Decision Systems Engineering at Arizona State University, and previously Research Fellow for the Department of Industrial & Systems Engineering at National University of Singapore. Her research focuses on stochastic simulation-optimization in both single and multipleobjectives framework. She is developing her research in meta-model based simulation optimization and learning for simulation and simulation optimization. Her email address is giulia.pedrielli.85@gmail.com.