On Local Optima Distribution in Buffer Allocation Problem for Production Line with Unreliable Machines ⋆

Alexandre Dolgui * **Anton Eremeev** ** **Viatcheslav Sigaev** ***

* *IMT Atlantique, Nantes, France*
** *Sobolev Institute of Mathematics SB RAS, Novosibirsk, Russia (e-mail: eremeev@ofim.oscsbras.ru).*
*** *Avtomatika-Servis LLC, Omsk, Russia (e-mail: sigvs@yandex.ru).*

Abstract: In this paper, we consider a buffer allocation problem in manufacturing flow lines with series-parallel network structure where nodes correspond to buffers of finite capacity, and arcs correspond to the machines. The machines are supposed to be unreliable, their time to failure and repair time are assumed to be exponentially distributed. Different machines may have different production rates and the production rates of all machines are assumed to be deterministic. The buffer allocation problem is to determine the capacities of all buffers with respect to a given optimality criterion, which is a function of the average production rate of the line, the buffer acquisition and installation cost and the inventory cost. In search for the optimum, the tentative solutions are evaluated by means of an approximate method based on the Markov models aggregation. We carry out computational experiments with the local search and genetic algorithms. It turns out that the "massif central" or "big valley" structure of the fitness landscape is present but only partially: The fitness of the local optima is negatively correlated with the distance to the best found solution, yet the set of local optima can not be encompassed by a ball of relatively small radius. Moreover, we show that in many problem instances, several clusters of local optima can be identified. The symmetries of the fitness function are discussed and suggested as the possible cause of the local optima clustering. Finally the performance of genetic algorithms is bfiefly discussed with respect to solutions clustering.

Keywords: Production line, Unreliable machines, Buffer allocation, Series-parallel network, Genetic algorithms, Local optima.

1. INTRODUCTION

Buffer capacity allocation problems arise in a wide range of flow line manufacturing systems, such as transfer lines, flexible manufacturing or robotic assembly systems. The parts are accumulated in the intermediate buffers when the machines downstream are less productive than the upstream machines. It is assumed that machines can break down and then go through repair. When a breakdown occurs, the corresponding machine is not used in production for a random repair time, which is independent of the total number of machines under repair. We assume that there is a sufficient number of raw parts at the input buffer and the finished parts depart from the system immediately. One of the key performance measures of a flow-line is the average production rate, i.e., the expected number of parts produced per unit of time in the steady state mode. We also consider the inventory cost and the buffers cost.

Evaluation of the manufacturing flow-line performance for given sizes of buffers is studied by Coillard and Proth (1984), Dallery and Gershwin (1992), Gershwin (1993), Heavey et al. (1993), Li and Meerkov (2009), and Tan and Gershwin (2009). A number of models to evaluate the performance of lines with unreliable machines and fixed sizes of buffers were proposed by these and other authors. Markov models and aggregation or decomposition techniques are often used to calculate the steady state throughput or other performance indicators for these lines. The optimization for buffer capacity allocation with respect to diverse optimality criteria for different types of lines was studied using such models by Smith and Daskalaki (1988), So (1997), Gershwin and Schor (2000), Kassoul et al. (2021), Shi and Gershwin (2009) and other authors.

1.1 The Buffer Allocation Problem Formulation

In this paper, we consider the buffer allocation problem for lines with a series-parallel network. An example of a line with a series-parallel network is shown in Fig. 1, where $M_1, \dots, M_7$ are machines and $B_0, \dots, B_5$ are buffers.

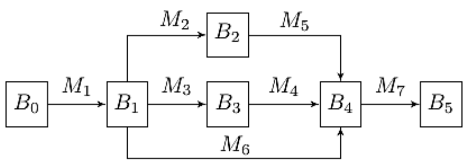

Fig. 1. Example of a line with a series-parallel network

⋆ The research is supported by Russian Science Foundation grant 21-41-09017.

We assume that a machine can be either operational or under repair. An operational machine may be *blocked* in case the downstream buffer is full. It may also be *starved* if there are no parts in the upstream buffer. Otherwise operational machines are working. In what follows, m denotes the number of machines in the system. A working machine i, $i = 1, \dots, m$, is assumed to have a constant cycle time C_i, so its average production rate is $u_i = 1/C_i$.

It is supposed that machines may break down only when they are working. The time to fail and time to repair for each machine are assumed to be random values with exponential distributions. Let T_b^i denote the average time till failure, and let $\lambda_i = 1/T_b^i$ be the failure rate for a machine i, $i = 1, ..., m$, if this machine is working. Similarly, let T_r^i and $\mu_i = 1/T_r^i$ denote respectively the time to repair and the repair rate for machine i, conditioned that this machine is under repair. Given our assumptions, the system has the steady state mode (see e.g. Sevast'yanov (1962)). The performance of the system in this mode is the most important for applications.

Let h_j be the capacity of buffer B_j, $j = 1, \dots, n$. Denote the vector of decision variables as $H = (h_1, h_2, \dots, h_n) \in Z_+^n$, where Z_+ is the set of non-negative integers.

The optimization criterion used in this paper is:

$$\max \phi(H) = T_{am} R(V(H)) - Q(H) - J(H), \qquad (1)$$

where

- T_{am} amortization time of the line (line life);
- $V(H)$ average production rate (steady state throughput);
- $R(V)$ revenue related to the production rate V;
- $J(H)$ cost of buffer configuration H;
- d_j maximal admissible capacity of buffer B_j;
- $Q(H) = c_1 q_1(H) + \ldots + c_n q_n(H)$ average steady state inventory cost, where $q_j(H)$ is the average steady state number of parts in buffer b_j, for $j = 1, \dots, n$.

The function $\phi(H)$ has to be maximized, subject to the constraints $h_1 \leq d_1$, $h_2 \leq d_2, \dots$, $h_n \leq d_n$, bounding the admissible buffer size. Functions $R(V)$ and $J(H)$ are assumed to be monotone and non-decreasing. $J(H)$ may be a linear function, or e.g. a step-function to model some standard buffer capacities, or may be a penalty function, imposing a penaly on solutions where the total capacity of all buffers exceeds some upper bound.

Exact computation of the production rate and inventory levels in a line with more than two serial machines is problematic due to the exponential growth of the number of states in the corresponding Markov model. Therefore, most of the techniques developed for the analysis of such systems are based on analytical approximations or simulations. Most of the analytical approximations are based on the two-machine Markov models, and either aggregation (De Koster, 1987) or decomposition (Dallery et al., 1989; Gershwin, 1987; Li, 2005). Simulation models require more computational resources but may be applied to a wider class of systems (Dolgui and Svirin, 1995; Sörensen and Janssens, 2004).

In this paper, we use the two-machine Markov model, independently developed by Levin and Pasjko (1969), Dubois and Forestier (1982) and Coillard and Proth (1984). For any tentative buffer allocation, the production rate is evaluated using the aggregation algorithm (Dolgui, 1993) which is similar to the techniques from (Terracol and David, 1987; Dolgui and Svirin, 1995).

The aggregation algorithm consists in recursive replacement of two adjacent machines by a single machine. The parameters λ^*, μ^*, c^* of each emerging machine are calculated from differential equations corresponding to the two-machine Markov model. After n iterations of such aggregation procedure the system reduces to a single machine with parameters λ^*, μ^*, c^* and the estimate of the overall production rate $V(H)$ is given by $c^*\mu/(\lambda^* + \mu)$. The steady state inventory levels are found in each application of the two-machine Markov model.

The buffer allocation problem is known to be NP hard as shown by Dolgui et al. (2013, 2018) and therefore it features some properties of the well-known combinarotial optimization problems, one of such properties is that often it is easy to find a locally optimal solution (computable in polynomial time w.r.t. the problem input size), although it is hard to find the global optimum (requires exponential time in the worst case).

1.2 "Big Valley" or "Massif Central"

In many combinatorial optimization problems, local optima of the objective function (or fitness) tend to be grouped in a "big valley“ (in the case of minimization problems) or "massif central“ (in the case of maximization problems). This fitness landscape structure has been observed e.g. in NK-landscapes (Stuart and Simon, 1987), in the traveling salesman problem (TSP) (Boese et al., 1993; Hains et al., 2011), in the graph bisection (Boese et al., 1993), and flowshop scheduling (Reeves, 1999). More precisely, the "big valley“ or "massif central“ is described by the following two statements (Boese et al., 1993):

(1) Values of the objective function in the local optima tend to deteriorate with increasing distance to the global optimum (i.e. there is a correlation of objective function in the local optima with the distance to the global optimum).
(2) Local optima are located relatively close both to each other and to the global optimum (they are located in a ball, which is smaller than the whole search space by several orders of magnitude).

The presence of such structure partly explains good performance of genetic algorithms (GAs). If different local optima are found in the GA population and the new solution is built by means of a crossover operator, then the intuition suggests that such algorithm should have good chances to find the global optimum. This is supported by the theoretical analysis in the case of the Jump benchmark of Dang et al. (2016), and by the experimental studies, e.g. of Hains et al. (2011). In this respect, identification of the "big valley“ or "massif central“ structure, or the absence of such structure, is of great practical interest.

1.3 Contribution of the Paper

On the basis of computational experiments we show that the distribution of local optima for many instances of the buffer allocation problem on lines with a series-parallel structure has multiple clusters. The problem features causing such structures are discussed. The “massif central” structure is identified but only in part: While the negative correlation between the objective function in local optima and their distance to the global solution is present, yet the concentration of all local optima in a tiny fraction of the search space is not confirmed. The observed structures of fitness landscape appear to be similar to those in (Hains et al., 2011), and preliminary experiments suggest that in both cases the GA combined with a local search is able to locate the cluster of high quality local optima.

2. INVESTIGATION OF LOCAL OPTIMA PROPERTIES

The distance to global optimum in our case, we will calculate in the metric l_1. To verify the “massif central” structure, we developed a method for finding the number of integer points in a ball of a given radius at the intersection with a parallelepiped whose faces are parallel to coordinate planes. The method was obtained by reducing the problem to a combinatorial formulation, already considered earlier for the case of non-negative integer points, using generating functions, see Sachkov (1982).

2.1 Problem Instances Used In Computational Experiment

For computational experiments we use three series of problems:

Series	Number of Lines	Number of Machines
AS	8	4 – 14
BN	10	5
VP	4	5

Table 1. List of series

The AS series consists of instances created from lines 1,2,6,7,8 from Ancelin and Semery (1987) with real data from Renault production. A distinctive feature of the 7 and 8 lines is the presence parallel sections (see figures 2, 3). Line parameters 1,2,6,7,8 are given in tables 2-6.

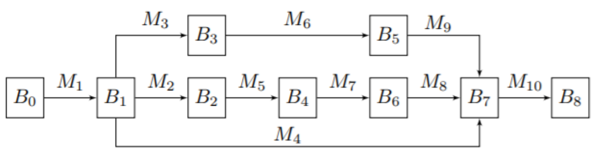

Fig. 2. Line structure AS7

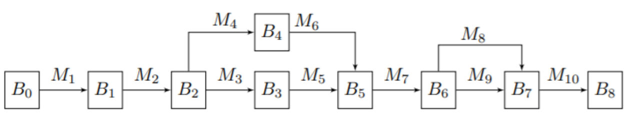

Fig. 3. Line structure AS8

Instances bn5.1-bn5.10 consist of 10 lines from Dolgui et al. (2000) with $m = 5$. There are two bottlenecks in each line. By bottleneck we mean here a line section of two relatively slow machines and a buffer between them.

Problems vp6.9-vp6.10, vp7.9-vp7.10 are defined on serial five-machine lines of Vouros and Papadopoulos (1998).

buffers		machines			
i	d_i	j	T_j^{O}	T_j^{B}	U_i
1	20	1	244.2	150	10
2	17	2	255.3	300	10
3	38	3	176	75	10
4	48	4	184	600	10
		5	192	450	10

Table 2. Problem parameters *AS1*

Buffers		Machines			
i	d_i	j	T_j^{O}	T_j^{B}	U_i
1	0	1	10000	440	22
2	50	2	20000	440	23
3	20	3	5000	430	22
4	50	4	40000	520	23
5	0	5	30000	430	24
6	80	6	2442	440	22
7	20	7	1840	520	23
8	100	8	1680	430	21
9	100	9	2208	920	24

Table 3. Problem parameters *AS2*

Buffers		Machines			
i	d_i	j	T_j^{O}	T_j^{B}	U_i
1	60	1	29880	22000	385
2	60	2	29880	22000	426
3	50	3	876000	22300	330
4	70	4	29880	22000	372
5	60	5	33250	27500	316
6	80	6	144000	8500	340
7	45	7	102300	74000	340
8	25	8	113300	7200	340
9	35	9	540000	60000	380
10	80	10	538800	349000	350
11	40	11	5064000	73700	400
12	45	12	468000	306000	400
13	65	13	1032000	54000	319
		14	45600	31120	319

Table 4. Problem parameters *AS6*

Buffers		Machines			
i	d_i	j	T_j^{O}	T_j^{B}	U_i
1	15	1	50000	12000	1000
2	10	2	48000	2000	3450
3	15	3	55000	9000	2780
4	10	4	39000	6000	3030
5	25	5	75000	10000	3333
6	10	6	59000	11000	2560
7	10	7	28000	8000	3030
		8	35000	8000	3125
		9	65000	35000	2174
		10	20000	4000	800

Table 5. Problem parameters *AS7*

2.2 Computational Experiment

A series of experiments were carried out on existing instances for determination of the landscape structure. To find a local optimum, we used the local search algorithm LSA. At each iteration, the LSA searches through the neighborhood of radius 1 in metric l_1 around the current solution. If an improving feasible solution in terms of the objective function is found in the neighborhood, then it becomes the new current solution. The process continues as long as an improvement can be found. Starting from any feasible solution, the LSA moves iteratively to a local

Buffers		Machines			
i	d_i	j	T_j^{O}	T_j^{B}	U_i
1	1300	1	87000	27000	23
2	200	2	77000	22000	27
3	0	3	580000	18000	38
4	0	4	410000	12500	30
5	0	5	580000	18000	38
6	2000	6	410000	12500	30
7	0	7	725000	21000	20
		8	550000	14000	40
		9	430000	24000	43
		10	270000	22000	33

Table 6. Problem parameters *AS8*

optimum, i.e. a solution that does not have an improving neighbour within the radius 1 in metric l_1.

In each run, the local search algorithm starts at a randomly generated solution H whose elements h_i are chosen with the uniform distribution between 1 and d_i. This procedure was repeated 300 times to create the necessary sample size. Using on this sample, we calculated the total number of admissible solutions $|\Omega|$ in the minimal ball encompassing all the local optima found. In our case, the ball was chosen in the metric l_1, centered in the best found local optimum.

In Table 7-10 the column $V1$ contains the cardinality $|\Omega|$ for balls containing local optima, the column $V2$ contains cardinality of the entire space of feasible solutions, and the column $V1/V2$ is the ratio of the two. As can be seen from the tables, the second part of the "massif central" / "big valley" conjecture was not confimed and in most of the instances under consideration the optima are scattered throughout the solution space.

instance	$V1$	$V2$	$V1/V2$
1	2,84E+05	5,74E+05	0,49
2	8,88E+11	9,48E+11	0,94
3	8,60E+01	4,85E+03	0,02
4	2,34E+22	2,89E+22	0,81
5	4,17E+07	9,75E+07	0,43
6	1,00E+00	5,23E+08	0,00

Table 7. Results of running a local search multiple times for a series as.1 - as.6

instance	$V1$	$V2$	$V1/V2$
9	7,97E+03	1,00E+04	0,80
10	7,55E+03	1,46E+04	0,52

Table 8. Results of running a local search multiple times for the series vp6.9 - vp6.10

instance	$V1$	$V2$	$V1/V2$
9	6,50E+03	1,00E+04	0,65
10	9,95E+03	1,46E+04	0,68

Table 9. Results of running a local search multiple times for the series vp7.9 - vp7.10

The first part of the "massif central" / "big valley" conjecture is about the correlation $\rho(\varphi(\xi), r(\xi, \xi^*))$ of the value of objective function at local optima $\varphi(\xi)$ to the distance $r(\xi, \xi^*)$ to a global optimum. Our experiments suggest that there is a negative correlation ρ, and all values of the correlation are significantly different from 0 with a confidence level of 95% (Tables 11–12).

instance	$V1$	$V2$	$V1/V2$
1	1,44E+05	1,94E+05	0,74
2	9,96E+03	1,94E+05	0,05
3	9,07E+04	1,94E+05	0,47
4	2,06E+04	1,94E+05	0,11
5	6,95E+04	1,94E+05	0,36
6	6,94E+04	1,94E+05	0,36
7	2,45E+04	1,94E+05	0,13
8	3,42E+03	1,94E+05	0,02
9	6,28E+04	1,94E+05	0,32
10	1,00E+00	1,94E+05	0,00

Table 10. Results of 300 runs of the local search for the series bn5.1 - bn5.10

instance	$\rho(r(\xi, \xi*), \varphi(\xi))$
Series as.1 - as.5	
1	-0,87078206
2	-0,437208613
7	-0,547348483
8	-0,943725905
Series vp6.9 - bn6.10	
9	-0,833583818
10	-0,778237651

Table 11. Results of 300 runs of the local search for series *as* and *vp*6

instance	$\rho(r(\xi, \xi*), \varphi(\xi))$
Series bn5.1 - bn5.9	
1	-0,706451573
2	-0,872602935
3	-0,914939714
4	-0,999969666
5	-0,972555479
6	-0,729655068
7	-0,973827419
8	-0,846807033
9	-0,884916669
Series vp7.9 - bn7.10	
9	-0,907792301
10	-0,889724863

Table 12. Correlation $\rho(\varphi(\xi), r(\xi, \xi^*))$ in series *bn* and *vp*7

It is interesting that in problems as.4, bn5.1, the entire set of local optima splits into clusters, and in each of the clusters, the negative correlation ρ is observed.

Figures 4 and 5 show diagrams of local optima for problems as.6, bn5.1, where the ordinate shows the value of objective function of local optimum $\varphi(H)$, and the abscissa is distance in metric l_1 to the best found solution.

Clustering effect is especially well manifested in instances with parallel sections of line. This effect can be justified by the fact that there are several different paths in lines with a parallel structure. from start buffer to end buffer. Thus, if there are two parallel paths that are identical in their network structure, such that one path will have relatively large buffers, and other one will have relatively small buffers, then it is possible to obtain two solutions with close values of the objective function, but located at a large distance from each other.

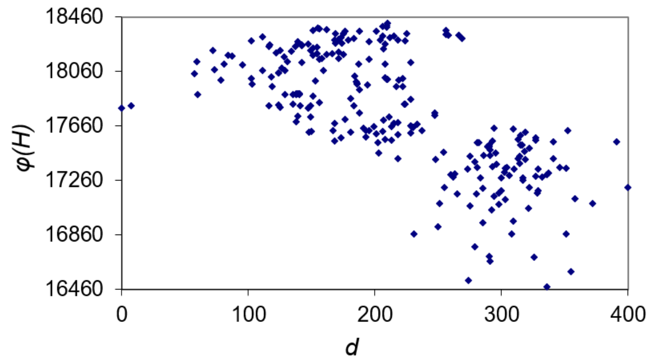

Fig. 4. The set of local optima obtaained in as.4

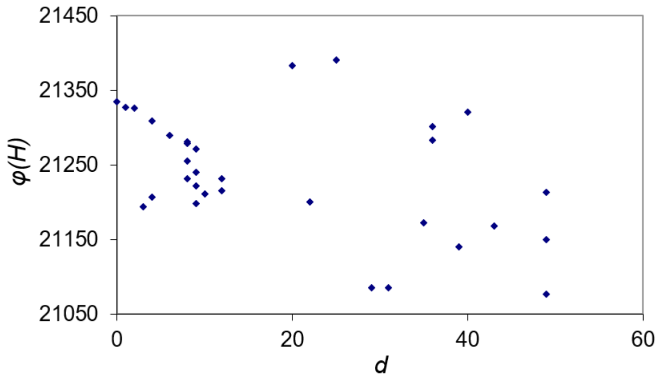

Fig. 5. The set of local optima obtained in bn5.1.

Another reason for clustering effect is the two-machine line symmetry. This property is indicated by Levin and Pasjko (1969) for a serial line, consisting of two machines and a buffer between them. If we swap the parameters of the first and the second machines so that the input buffer of the line will become its output, and the output buffer will become the input, then line performance will not change.

Consider an example of a three-machine serial line t.1 with the parameters: $T_i^{\mathrm{O}} = T_i^{\mathrm{B}} = 1$, $i = 1, \ldots, 3$; $U_1 = 1$, $U_2 = 0.5$, $U_3 = 1$; $d_j = 4$, $c_j = 0$, $j = 1, \ldots, 2$; $T_{am} = 7000$; $J(H) = 50 \cdot (h_1 + h_2)$; if $V(H) < 2570$ then $R(V(H)) = 0.9 \cdot V(H)$, otherwise $R(V(H)) = 2570$.

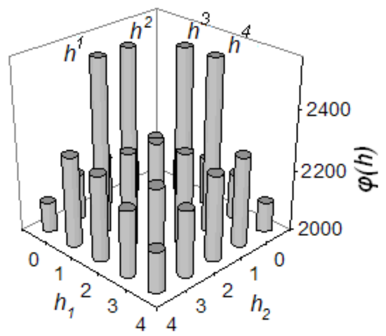

Fig. 6. Values of the objective function on the set D for problem t.1

The set of global optima (see Fig. 6) of this example splits into two clusters. The first cluster contains solutions $H^1 = (1, 2)$ and $H^2 = (1, 3)$, and the second is the solutions $H^3 = (2, 1)$ and $H^4 = (3, 1)$. Clustering of the optima in this example is a consequence of the line symmetry effects in the aggregation algorithm from Dolgui (1993); Dubois and Forestier (1982), which is used to evaluate the line.

In the general case, for lines with $n \geq 3$ buffers, the effect of symmetry can be observed in several internal areas of a line. As a consequence, with this aggregation algorithm, a set of local optima may be divided into several clusters.

After falling into one of the clusters, a sequence of points generated by an algorithm, based on the local search principles, usually remains in the cluster until the end of the calculations. For example, for a tabu-search algorithm, transition between clusters is unlikely, which also makes such instances difficult for it. The GA with local optimization heuristic from Dolgui et al. (2007) turned out to be more efficient than the Tabu Search. As experiments have shown, the population of the pure GA, and that of the GA with local search, contain individuals from different clusters, although there is a competition between different clusters for representation in the population.

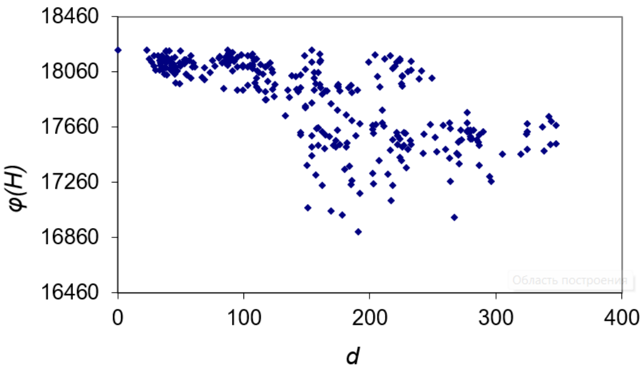

Fig. 7. Final population of the pure GA on instance as.4

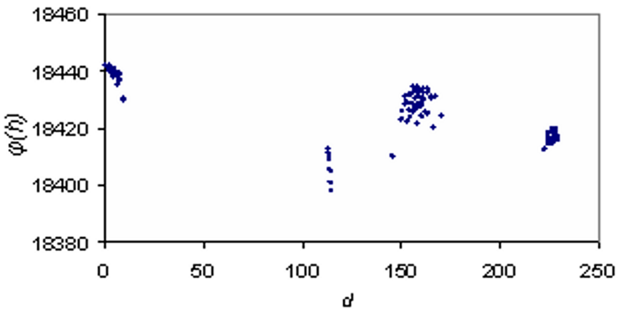

Fig. 8. Final population of the GA with local optimization heuristic on instance as.4 (note that here the scale of both axes is different from that in Figures 4 and 7)

Problem as4 can serve as an illustrative example. Fig. 4 shows the structure of the set of local optima obtained by the multistart of the local search. Figures 7 and 8 show the final populations of the pure GA and the GA with local optimization heuristic. Multiple clusters are clearly present in both cases.

We expect that the GA could be further improved if each pair of parent solutions were chosen from the same cluster (then the crossover could have a similar effect as e.g. in Dang et al. (2016)). Excluding the equivalent solutions in view of the problem symmetries may reduce the number of clusters in the GA population. This should have the same effect as the usage of non-degenerate solution encodings (Reeves and Dai, 1999) and may be the subject for further research as well.

REFERENCES

Ancelin, B. and Semery, A. (1987). Calcul de la productivité d'une ligne integrée de fabrication. *RAIRO-Autom. Prod. Inf.*, 21, 209–238.

Boese, K., Kahng, A., and Muddu, S. (1993). On the big valley and adaptive multi-start for discrete global optimizations. *Technical report, UCLA CS Department, TR-930015.*

Coillard, P. and Proth, J. (1984). Effet des stocks tampons dans une fabrication en ligne. *Revue Belge de Statistique, d'Informatique et de Recherche Opéationnelle*, 24 (2), 3–27.

Dallery, Y., David, R., and Xie, X. (1989). Approximate analysis of transfer lines with unreliable machines and finite buffers. *IEEE Transactions on Automatic Control*, 34, 943–953.

Dallery, Y. and Gershwin, S. (1992). Manufacturing flow line systems: a review of models and analytical results. *Queueing Syst.*, 12(1-2), 3–94.

Dang, D.C., Friedrich, T., Kötzing, T., Krejca, M.S., Lehre, P.K., Oliveto, P.S., Sudholt, D., and Sutton, A.M. (2016). Escaping local optima with diversity mechanisms and crossover. In *Proc. of the 2016 Genetic and Evolutionary Computation Conference (GECCO 2016)*, 645–652. ACM.

De Koster, M. (1987). Estimation of line efficiency by aggregation. *Int. J. Prod. Res.*, 25, 615–626.

Dolgui, A. (1993). Analyse de performances d'un atelier de production discontinue: méthode et logiciel. *Research Report INRIA*, 1949.

Dolgui, A., Eremeev, A., Kolokolov, A., and Sigaev, V. (2000). A genetic algorithm for buffer allocation in production line with unreliable machines. In *Proc. of the International Workshop Discrete Optimization Methods in Scheduling and Computer-Aided Design*, 26–31.

Dolgui, A., Eremeev, A., and Sigaev, V. (2007). HBBA: hybrid algorithm for buffer allocation in tandem production lines. *J. Intell. Manuf.*, 18(3), 411–420.

Dolgui, A. and Svirin, Y. (1995). Models of evaluation of probabilistic productivity of automated technological complexes. *Vesti Akademii Navuk Belarusi: phisikatechnichnie navuki*, 1, 59–67.

Dolgui, A., Eremeev, A., Kovalyov, M., and Sigaev, V. (2018). Complexity of Bi-objective Buffer Allocation Problem in Systems with Simple Structure. In *Optimization Problems and Their Applications*, volume 871 of *Communications in Computer and Information Science*, 278–287. Springer.

Dolgui, A., Eremeev, A., Kovalyov, M.Y., and Sigaev, V. (2013). Complexity of Buffer Capacity Allocation Problems for Production Lines with Unreliable Machines. *Journal of Mathematical Modelling and Algorithms in Operations Research*, Volume 12(Issue 2), pp 155–165.

Dubois, D. and Forestier, J. (1982). Productive et encours moyens d'un ensemble de deux machines separees par une zone de stockage. *RAIRO Automatique*, 16(12), 105–132.

Gershwin, S. (1987). An efficient decomposition method for the approximate evaluation of tandem queues with finite storage space and blocking. *Operations Research*, 35 (2), 291–305.

Gershwin, S. (1993). *Manufacturing Systems Engineering.* Prentice Hall.

Gershwin, S. and Schor, J. (2000). Efficient algorithms for buffer space allocation. *Annals of Operations Research*, 93, 117–144.

Hains, D., Whitley, L., and Howe, A. (2011). Revisiting the big valley search space structure in the TSP. *Oper. Res. Soc. 62*, 305–312.

Heavey, C., Papadopoulos, H., and Browne, J. (1993). The throughput rate of multistation unreliable production lines. *Europ. J. Oper. Res.*, 68, 69—-89.

Kassoul, K., Cheikhrouhou, N., and Zufferey, N. (2021). Buffer allocation design for unreliable production lines using genetic algorithm and finite perturbation analysis. *Int. J. Prod. Res.* doi:10.1080/00207543.2021.1909169. Published online.

Levin, A. and Pasjko, N. (1969). Calculating the output of transfer lines. *Stanki i Instrument*, (8), 8–10.

Li, J. (2005). Overlapping decomposition: a system-theoretic method for modeling and analysis of complex manufacturing systems. *IEEE Transactions on Automation Science and Engineering*, 2 (1), 40–53.

Li, J. and Meerkov, S. (2009). *Production Systems Engineering.* Springer US.

Reeves, C.R. and Dai, P. (1999). The effect of degenerate coding on genetic algorithms. In *Artificial Neural Nets and Genetic Algorithms*, 208–213. Springer Vienna, Vienna.

Reeves, C. (1999). Landscapes, operators and heuristic search. *Ann. Oper. Res.*, 86(0), 473–490.

Sachkov, V. (1982). *Introduction to combinatorial methods of discrete mathematics.* M.: Science.

Sevast'yanov, B. (1962). The problem of how bunker capacity influences averages idle time for an automated line of machines. *Teor. Veroyat. Primen.*, 7(4), 438–447.

Shi, C. and Gershwin, S. (2009). An efficient buffer design algorithm for production line profit maximization original research. *International Journal of Production Economics*, 122 (2), 725–740.

Smith, J. and Daskalaki, S. (1988). Buffer space-allocation in automated assembly lines. *Operations Research*, 36, 343–358.

So, K. (1997). Optimal buffer allocation strategy for minimizing work-in-process inventory in unpaced production lines. *lIE Transactions*, 29, 81–88.

Sörensen, K. and Janssens, G. (2004). A petri net model of a continuous flow transfer line with unreliable machines. *Eur. J. Oper. Res.*, 152, 248–262.

Stuart, K. and Simon, L. (1987). Towards a general theory of adaptive walks on rugged landscapes. *Journal of Theoretical Biology*, 128(1), 11–45.

Tan, B. and Gershwin, S. (2009). Analysis of a general markovian two-stage continuous-flow production system with a finite buffer. *International Journal of Production Economics*, 120 (2), 327–339.

Terracol, C. and David, R. (1987). Performance d'une ligne composée de machines et de stocks intermédiaires. *RAIRO-Autom. Prod. Inf.*, 21, 239–262.

Vouros, G. and Papadopoulos, H. (1998). Buffer allocation in unreliable production lines using a knowledge based system. *Computers Ops. Res.*, 25(12), 883–891.