



*Journal of Advances in
Science and Technology*

*Vol. V, No. IX, May-2013,
ISSN 2230-9659*

**AN INVESTIGATION UPON OPTIMIZATION
DEPENDENT HEURISTIC: ISSUES REGARDING
ROBOTIC CELL**

An Investigation upon Optimization Dependent Heuristic: Issues Regarding Robotic Cell

Veerasha A. Sajjanara

Research Scholar, CMJ University, Shillong, Meghalaya

Abstract – This study examines an optimization-based heuristic for the Robotic Cell Problem. This problem emerges in mechanized cells and is an unpredictable flow shop problem with a solitary transportation robot and a blocking obligation. We propose an inexact decomposition algorithm. The proposed methodology breaks the problem into two scheduling problems that are fathomed consecutively: a flow shop problem with extra requirements (blocking and transportation times) and a solitary machine problem with priority constraints, time slacks, and setup times. For each of these problems, we propose a careful extension and-bound algorithm. Likewise, we portray a hereditary algorithm that incorporates, as a change specialist, a neighborhood seek system. We report the outcomes of a computational study that gives prove that the proposed optimization-based methodology conveys superb results and constantly beats the hereditary algorithm. Be that as it may, the hereditary algorithm conveys sensibly great results while obliging altogether shorter CPU times.

INTRODUCTION

The Robotic Cell Problem (RCP) is a generalization of the established stage flow shop problem. It may be detailed as accompanies. Given work set $J = \{1, 2, \dots, n\}$ where each one employment must be transformed nonpreemptively on m machines M_1, M_2, \dots, M_m in a specific order. The processing time of work $(j = 1, \dots, n)$ on machine M_i ($i = 1, \dots, m$) is p_{ij} . At time $t = 0$, all occupations are accessible at an info apparatus meant by M_0 . After consummation, each one employment must be taken from M_m to a yield mechanism that is meant by M_{m+1} (for comfort, we set $p_{m+1,j} = 0, \forall j \in J$). The exchange of an occupation $j \in J$ from M_i to M_{i+1} ($i = 0, \dots, m$) is performed by method of a solitary robot. A vacant or a stacked move of the robot from M_i to M_h ($i, h = 0, \dots, m+1$) takes τ_{ih} units of time. The machines have not include or yield buffering offices. Hence, in the wake of processing work j on machine M_i ($i = 1, \dots, m$), this last remains hindered until the robot gets j and exchanges it to the accompanying machine M_{i+1} . Such a move could only be performed if machine M_{i+1} is free (that is, no employment is constantly handled by or holding up at M_{i+1}). At whatever time, each one machine can prepare at most one employment and each one occupation could be transformed on at most one machine. Besides, the robot can exchange at most one occupation at whenever. The problem is to discover a processing request of the n occupations, the same for each one machine (due to the blocking demand, passing is not conceivable), such that the

time C_{\max} at which all the employments are finished (makespan) is minimized.

The specific instance of the robotic cell problem where the transportation times are negligible lessens to the greatly examined flow shop scheduling problem with blocking (Pinedo, 2008). Lobby and Sriskandarajah (1996) demonstrate that this recent problem is determinedly \mathcal{NP} -hard for $m \geq 3$. Hence, the RCP is firmly \mathcal{NP} -hard for $m \geq 3$ too.

Our cause for the examination of the robotic cell problem stems from its useful pertinence to Flexible Manufacturing Systems (Fmss), which are exceptionally computerized production systems fit for generating a wide assortment of work sorts. As pointed out in Blazewicz et al. (1991), a standout amongst the most troublesome operational problems in Fmss is the advancement of adequate timetables recognizing employments, machines and transportation apparatuses to furnish a fitting coordination of the generation sequencing and time portion of all obliged resources. Blocking requirements emerge for instance in the manufacturing methodology or the chemical industry where a few aspects, (for example, temperature) of the material obliges that each one occupation must tend to the machine before being transformed on the following machine. The RCP is a particular problem in Fmss. Just extremely uncommon cases could be unraveled in polynomial-time and different cases with a solitary robot are as of now \mathcal{NP} -hard (see Knust (1999)).

LITERATURE REVIEW

The stunningly fast spread of robotic cells in manufacturing systems that happened throughout the most recent two decades has aroused the development of a lot of people new scheduling problems. We allude to the excellent book of Dawande et al. (2007) for a breakthrough and exhaustive audit of sequencing and scheduling problems emerging in robotic cells. Really, practically all long ago explored robotic cell scheduling problems bargain with cyclic scheduling problems with machines generating a group of comparable parts, in an enduring state. We allude to Dawande et al. (2005) for a characterization plan of these testing problems. Notwithstanding, to the best of our information, the different part-sort robotic cell problem that is tended to in this paper has never been researched previously in the previous exceptionally general structure. For sure, different part sort problems that have been tended to so far are frequently characterized as takes after. We are given an insignificant part set (MPS) that incorporates p diverse part-sorts to be handled. Each one sort ($k = 1, \dots, p$) incorporates d_k comparative parts. The MPS requires to be planned redundantly utilizing p redundancies of an one-unit robot move cycle with the goal of expanding the throughput rate, or equally, minimizing the process duration (for instance see Sriskandarajah et al. (1998) Besides, despite the fact that there is a limitless literature relating to mixed work and vehicle scheduling (see Ganesharajah et al. (1998)), there are not many papers which think about flow shop problem with hindering to minimize makespan (see Hall and Sriskandarajah (1996)). Mascis and Pacciarelli (2002) study the employment shop problem with blocking. They figure it by method of a generalization of the disjunctive chart of Roy and Sussman (1964) and they demonstrate that numerous lands used to plan heuristic strategies don't hold in the blocking case. Reddi and Ramamoorthy (1972) show that the problem $F2 \setminus block \setminus C_{max}$ could be decreased to an unique instance of the voyaging businessperson problem that might be illuminated in polynomial time by the Gilmore and Gomory (1964) algorithm. Dutta and Cunningham (1975) propose element modifying systems to tackle this problem with constrained buffer. Levner (1969) presents limb and-bound algorithms for taking care of the flow shop problem with blocking. Mc Cormick et al. (1989) study a flow shop scheduling problem in a sequential construction system where the problem with constrained buffers could be examined as an obstructing one (all machines have no middle of the road buffers). Abadi et al. (2000) propose a heuristic for minimizing the process duration keeping in mind the end goal to create an association between the no-hold up flow Shop problem and flow shop with blocking. This methodology has been utilized via Caraffa et al. (2001) for computing the quality of makespan for a given succession of the occupations. Ronconi (2005) proposes an extension and-bound algorithm utilizing a lower bound that adventures the blocking characteristic. The creator demonstrates that the acquired bounding plan outflanks the lower bounds

proposed in Ronconi and Armentano (2001). As of late, Grabowski and Pempera (2007) propose two new heuristic algorithms to minimize the makespan in a flow shop problem with blocking dependent upon a tabu hunt approach. Lee and Chen (2001) and Hurink and Knust (2001) concentrated on flow shop scheduling with unequivocal transportation limit and transportation times. Additionally, they expected a boundless buffer space between the machines and negligible unfilled moving times. The literature relating to other scheduling models with transportation perspectives is outlined in Crama et al. (2000). Moreover, Agnetis (2000) and Agnetis and Pacciarelli (2000) examine the unpredictability of a no-hold up flow shop problem in which one robot is utilized to move the parts from a machine to the following, and additionally between the machines and the input/output apparatuses. In this model, occupations are not permitted to hold up not on a machine or on the robot.

Not surprisingly, most robotic cell scheduling problems are intractable. In particular, Hall and Sriskandarajah (1996) prove that the robotic cell problem is strongly \mathcal{NP} -hard for $m \geq 3$. However, an $O(n^4)$ algorithm that solves this problem in two-machine cells is provided in Hall et al. (1997). Aneja and Kamoun (1999) improve this complexity to $O(n \log n)$.

SCHEDULING OF THE ROBOT MOVES

In this section, we are concerned with scheduling the robot moves provided that the processing order of the jobs on the m machines is given. For the sake of alleviating the notation we shall assume that the jobs are indexed according to their processing order.

It is interesting to view the robot sequencing problem as a single machine scheduling problem with side constraints, where the machine represents the robot and the operations correspond to the robot *loaded* moves, respectively. More precisely, we assume that we have a set of operations

$\Sigma = \{O_{ij} : i = 0, \dots, m, j = 1, \dots, n\}$ to be processed nonpreemptively on a single machine. An operation $O_{ij} \in \Sigma$ corresponds to a transfer of job j from M_i to M_{i+1} . The processing time of O_{ij} is $\tau_{i,i+1}$. In addition, there are two types of precedence constraints:

- Type 1: Because of the flow shop constraints, we have $O_{i-1,j} \prec O_{i,j}$ for $i = 1, \dots, m$ and $j = 1, \dots, n$ (where the notation " $u \prec v$ " means that operation u precedes operation v). This constraint expresses that

a job j cannot be transferred to machine M_{i+1} before being transferred to machine M_i .

- Type 2: Because of the blocking constraints, we have $O_{ij} \prec O_{i-1,j+1}$ for $i = 1, \dots, m$ and $j = 1, \dots, n - 1$. This constraint enforces that job $j + 1$ cannot be transferred to machine M_i before transferring job j from M_i to M_{i+1} :

Moreover, there is a minimal time lag $l_{i-1,j} \equiv p_{ij}$ between the completion time of operation $O_{i-1,j}$ and the start time of operation O_{ij} ($i = 1, \dots, m$ and $j = 1, \dots, n$). This time lag corresponds to the processing time of job j on machine M_{i+1} ($i = 0, \dots, m$).

Finally, if two operations $O_{i-1,j}$ and O_{hk} are processed consecutively (with $j = k$) then there is a nonnegative setup time $\gamma_{ih} \equiv \tau_{ih}$ between these two operations. This setup time corresponds to an empty positioning move from M_i to M_h . Thus, optimizing the robot moves requires minimizing the makespan on a single machine with precedence constraints, time-lags, and setup times. Hurink and Knust (2002) investigated this problem in the case where the precedence constraints stem from a job shop environment. They proved that the problem is NP-hard and proposed a tabu search algorithm. However, the complexity of the problem when the precedence constraints stem from a flow shop environment remains an open question. In the sequel, we describe an exact branch-and-bound algorithm for solving this latter problem.

A GENETIC ALGORITHM

During the last two decades, metaheuristics have become powerful tools for the approximate solution of intractable combinatorial optimization problems. In particular, genetic algorithms have been implemented for providing high-quality solutions to a wide variety of challenging flow shop problems (see for example Ruiz et al. (2006) and Rajkumar et al. (2009), to quote just a few). In this section, we investigate the ability of a genetic algorithm (hereafter, referred to by GA1) to effectively solve the robotic cell problem. Next, the performance of GA1 will be compared against the performance of the proposed approximate decomposition algorithm. The main features of GA1 are the following :

- Solution encoding : An ordered list (i.e. permutation) $\sigma = (\sigma(1), \dots, \sigma(n))$ of the n jobs is used to represent a chromosome.
- Fitness computation : Given a chromosome $\sigma = (\sigma(1), \dots, \sigma(n))$, the fitness of the

corresponding solution is computed as follows. First, the m machines are sequenced according the ordering that is specified by the chromosome. Second, the sequence of the robot moves is obtained using the afore-mentioned list scheduling algorithm: at each iteration, we schedule the robot operation whose predecessors have been already scheduled and whose starting time is minimal. In so doing, a feasible schedule is obtained. Clearly, we might use a more sophisticated approach for scheduling the robot moves, but this would require longer CPU times.

- The crossover operator : We implemented a powerful crossover operator that was recently introduced by Ruiz et al. (2006) for solving the (standard) permutation flow shop problem. This operator is called *Similar Job Order Crossover* (or, SJOX for short) and is

described as follows. Given two parent chromosomes, a crossing point is randomly selected along the length of the first parent, and an offspring is produced by copying the identical jobs that are located at the same positions in both parents into the corresponding positions of it. Then missing jobs before the crossing point are inherited from the first parent. Lastly, the missing jobs are copied in the relative order of the second parent. Clearly, if no identical jobs are at the same position in the parents, the crossover operator will behave like the one-point crossover.

- The mutation operator: The performance of a GA might be significantly improved when it is hybridized with a local search procedure. In our GA, instead of using a classical mutation operator, we implemented the following interchange procedure. First, a chromosome is chosen randomly and is mutated with a specified probability. Then, we perform a pairwise interchange of jobs (the interchanged jobs are not necessarily adjacent). Obviously, if this interchange leads to a reduction of the makespan then the chromosome is replaced with the improved one. The process is continued until no improvement could be achieved.

CONCLUSION

In this paper, we have investigated the robotic cell problem. We have proposed an approximate decomposition algorithm. To the best of our knowledge, this is the first attempt to solve this complex scheduling problem. The proposed approach breaks the problem into two scheduling problems: a flow shop problem with blocking and transportation times and a single machine problem with precedence constraints, time lags, and setup times. We have proposed for each of these two problems an exact branch-and-bound algorithm. In order to assess the performance of the proposed

approach, we have proposed a genetic algorithm that includes a powerful crossover operator as well as a local search-based mutation operator. We reported the results of a computational study that provide evidence that the proposed two-phase approach is robust, consistently delivers high-quality solutions, and outperforms the genetic algorithm. However, we found that the genetic algorithm requires significantly shorter CPU times and proves useful for efficiently solving large-scale instances.

Scheduling robotic cells constitute a challenging class of scheduling problems, and despite its pertinence to the fast growing topic of FMSs, its in-depth investigation is still in its infancy. We believe that the development of exact as well as tailored sophisticated metaheuristics for this important problem class is a promising avenue for future research.

REFERENCES

- Dutta, S.K., and Cunningham, A.A. (1975). Sequencing two-machine flowshops with finite intermediate storage. *Management Science* 21, 989-996.
- Abadi, K., Hall, N. and Sriskandarajah, C. (2000). Minimizing cycle time in a blocking flowshop. *Operations Research* 48, 177-180.
- Crama, Y. Kats, V. Van de Klundert, J. and Levner E. (2000). Cyclic scheduling in robotic flowshops. *Annals of Operations Research* 96, 97-124.
- Caraffa, V. Ianes, S. Bagchi, T.P. and Sriskandarajah, C. (2001). Minimizing makespan in a blocking flowshop using genetic algorithms. *International Journal of Production Economics* 70, 101-115.
- Blazewicz, J. Eiselt, H.A. Finke, G. Laporte, G. and Weglarz, J. (1991). Scheduling tasks and vehicles in a flexible manufacturing system. *International Journal of Flexible Manufacturing Systems* 4, 5-16.
- Desrochers, M. and Laporte, G. (1991). Improvements and extensions to the Miller- Trucker-Zemlin subtour elimination constraints. *Operations Research Letters* 10, 27-36.
- Agnetis, A. (2000). Scheduling no-wait robotic cells with two and three machines. *European Journal of Operational Research* 123, 303-314.
- Dawande, M.W., Geismar, H.N., Sethi, S.P., Sriskandarajah, C. (2007). *Throughput Optimization in Robotic Cells*, Springer.
- Ganesharajah, T., Hall, N.G., and Sriskandarajah, C. (1998). Design and operational issues in AGV-served manufacturing systems. *Annals of Operations Research* 76 (1), 109-154.
- Grabowski, J. and Pempera, J. (2007). The permutation flow shop problem with blocking, A tabu search approach. *Omega* 35, 302-311.
- Hall, N.G., Kamoun H., and Sriskandarajah, C. (1997). Scheduling in robotic cells: classification, two and three machine cells. *Operations Research* 45, 421-439.
- Dawande, M.W, Geismar, H.N., Sethi, S.P., and Sriskandarajah, C. (2005). Sequencing and scheduling in robotic cells: Recent developments. *Journal of Scheduling* 8, 387-426.
- Lee, C.-Y. and Chen, Z.-L. (2001). Machine scheduling with transportation considerations. *Journal of Scheduling* 4, 3-24.
- Knust, S. (1999). Shop-scheduling problems with transportation. PhD thesis, Fachbere- ich Mathematik/Informatik, University of Osnabruck.
- Mc Cormick, S.T., Pinedo, M.L. Shenker, S. and Wolf, B. (1989). Sequencing in an assembly line with blocking to minimize cycle time. *Operations Research* 37, 925-935.
- Hurink, J. and Knust, S. (2002). A tabu search algorithm for scheduling a single robot in a job-shop environment. *Discrete Applied Mathematics* 119, 181-203.
- Reddi, S.S. and Ramamoorthy, C.V. (1972). On the flowshop sequencing problems with no-wait in process. *Operational Research Quarterly* 23, 323-330.
- Lawler, E.L., Lenstra, J.K., Rinnooy Kan, A.H.G., and Shmoys, D. (1993). Sequencing and Scheduling: Algorithms and Complexity. *Handbooks in Operations Research and Management Science* 4, S.C. Graves, A.H.G. Rinnooy Kan, P. Zipkin (eds.), 445-522.
- Miller, C., Tucker, A., and Zemlin, R. (1960). Integer programming formulations and travelling salesman problems. *J. ACM* 7, 326-329.
- Levner, E.M. (1969). Optimal planning of parts machining on a number of machines. *Automation and Remote Control* 12, 1972-1978.
- Mascis, A. and Pacciarelli, D. (2002). Job-shop scheduling with blocking and no-wait constraints. *European Journal of Operational Research* 143, 498-517.
- Rajkumar, R., Shahabudeen, P. (2009). An improved genetic algorithm for the flowshop scheduling problem. *International Journal of Production Research* 47, 233-249.

- Ruiz, R., Maroto, C., and Alcaraz, J. (2006). Two new robust genetic algorithms for the flowshop scheduling problem. *Omega* 34, 461-476.