

A Genetic Algorithm Approach for the Scheduling in a Robotic-Centric Flexible Manufacturing System

Ana I. Pereira^{1,2}, Adriano Ferreira¹, José Barbosa^{1,3}, José Lima^{1,3}, and Paulo Leitão^{1,3,4}

¹ Polytechnic Institute of Bragança,
Campus Sta Apolónia, 5301-857 Bragança, Portugal
{apereira, a.ferreira, jbarbosa, jllima, pleitao}@ipb.pt

² ALGORITMI, University of Minho

Campus Azurém, 4800-058 Guimarães, Portugal

³ INESC TEC - Centre for Robotics in Industry and Intelligent Systems
Rua Dr. Roberto Frias, 4200-465 Porto, Portugal

⁴ LIACC – Artificial Intelligence and Computer Science Laboratory,
Rua Campo Alegre 1021, 4169-007 Porto, Portugal

Abstract. Scheduling assumes a crucial importance in manufacturing systems, optimizing the allocation of operations to the right resources at the most appropriate time. Particularly in the Flexible Manufacturing System (FMS) topology, where the combination of possibilities for this association exponential increases, the scheduling task is even more critical. This paper presents a heuristic scheduling method based on genetic algorithm for a robotic-centric FMS. Real experiments show the effectiveness of the proposed algorithm, ensuring a reliable and optimized scheduling process.

Keywords: Scheduling, Robotic-centric FMS, Genetic Algorithms.

1 Introduction

The manufacturing world is facing a set of constraints, either at an internal level, e.g., due to resource breakdowns or worker absences, or at an external level, e.g., due to rush orders or market fluctuations. In order to overcome these constraints, manufacturing companies must adapt their processes and system configurations, aiming to increase their flexibility and responsiveness. At this point, researchers are proposing new architectures where those topics are being addressed, guided by different governmental initiatives, e.g., Industrie 4.0 and Industrial Internet. Despite this, the proposed architectures must be firstly tested at laboratory environments, where methodologies, mechanisms and algorithms are developed and matured before being at a mature state to be absorbed by industrial companies.

Particularly, the manufacturing scheduling is a complex problem that has a strong impact in the production efficiency. The scheduling problem consists of a

finite set of jobs to be processed on a finite set of machines. Each job must be processed on a given machine and consists of a chain of complex of operations which have to be scheduled in a predetermined given order, a requirement called a precedence constraint [5]. It is not an easy task to schedule a high number of operations in the manufacturing environment. There are several approaches to solve the problem and optimization is one of the most promising way. Examples are Tabu Search, Particle Swarm Optimization, Genetic Algorithm among others.

This paper addresses the scheduling of transfer operations in a robotic centric Flexible Manufacturing System (FMS), installed at a laboratory facility, supporting researchers with a working-bench for research topics. In particular, an optimization scheduling approach based on genetic algorithms is developed and tested in this production system. The achieved results show the effectiveness of the proposed algorithm, ensuring a reliable scheduling process of operations.

The paper is organized as follows. After this brief introduction, Section 2 presents some related work on the scheduling problem. Section 3 describes the flexible manufacturing cell case study. Optimization procedures are presented in Section 4 whereas Section 5 shows the achieved results. Finally, Section 6 rounds up the paper with conclusions and future work.

2 Related work

Generation of the robot program can be decomposed in three main sub-problems: Task Planning, Motion Planning, and Task Scheduling [7].

Task Planning defines and assigns the sequence of operations to be done in order to execute a manufacturing task.

The motion planning describes the desired movement task into discrete motions while avoiding collisions. As example picking several parts in a bin (bin-picking) and planning the motion of the robot or arm.

On the other hand, scheduling, addressed in this paper, is the process of decide (while optimizing) the sequence of operations for each task in a production process or manufacturing process. Examples of scheduling are control air traffic control, airport gate management, allocate plant and machinery resources and plan production processes [8,12].

Nowadays, scheduling plays an important role in manufacturing. It is well addressed by many authors for several decades [2].

The task of scheduling a manipulator robot, one of the main objectives is to find the optimum sequence of the manipulators trip visiting a several points. This problem can be considered a variant of the well known traveling salesman problem (TSP) [10]. Bringing the classical TSP to robotics the optimization focus on optimizing cycle time instead of distance, optimizing the path that ensures the execution of each task in the right order using the less amount of time.

There are several techniques applied to solve the scheduling problems, from deterministic approaches to heuristics procedures. Regarding the Heuristic meth-

ods there are several approaches such as Artificial Intelligence, Simulated Annealing and Evolutionary algorithms [14]. Classical optimization methods can also be applied to the scheduling problem such as the Genetic Algorithm (GA) or Tabu Search [1].

3 Flexible Manufacturing Cell Description

The experimental case study considered in this work is a real small-scale production system composed by one IRB 1400 ABB robot, two punching machines and two indexed lines, as illustrated in Figure 1. The machines are supplied by Fischertechnik and constitute a hardware platform that provides the necessary experimental environment.

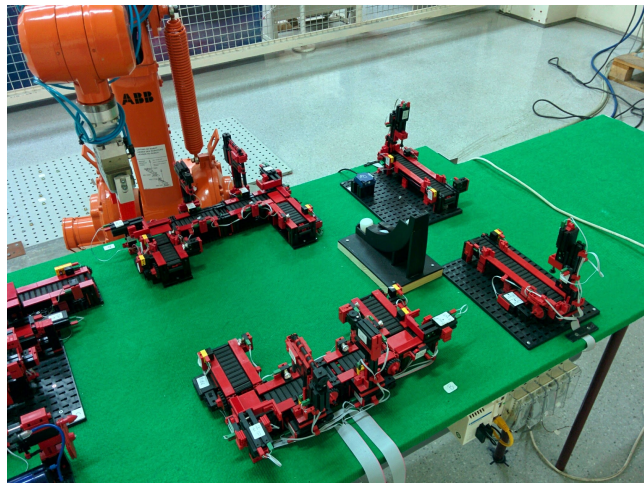


Fig. 1. The Fischertechnik cell

The punching machines are composed of two infrared sensors to detect the parts in the beginning of the conveyor and in the punching position, and two switch sensors to detect the end of the movement of the punching device. The conveyor and the punching are moving through two 24V DC motors. The capacity of each cell is to process one part at each time. The low-level logic control of the punching machine A is implemented as an IEC 61313-3 program running in a Modicon M340 PLC (accessed by Modbus) and the punching machine B is controlled by Java program running in a Raspberry Pi.

The indexed lines are composed of two workstations interconnected by several conveyors disposed in U shape, allowing to process four parts simultaneously. Each workstation, composed by the conveyors and machines, use eight 24V DC motors (four for the conveyors, 2 for the machines and 2 for the piston-like movement). Four switch sensors are used to determine the range of movement of

the piston-like movement and five infrared sensors are used to detect the presence of the parts in the conveyors and in the processing positions. The indexed line A is controlled by an IEC 61313-3 program running in a Modicon M340 PLC (accessed by Modbus) while the indexed line B is controlled using a Omron CPM1 PLC (accessed by a RS232 asynchronous line).

The industrial manipulator robot executes the transfer operations between the machines using proper RAPID programs and is accessible through the ABB S4 DDE Server, wrapped in a OPC server.

Finally, a human operator performs visual inspection operations to verify if the processing operations are according with the specification. The human operator interacts with the system through a Human-Machine Interface (HMI) from Omron connected to the system using the Omron PLC C200HG PLC.

Two different parts types can circulate in the system, each one having a particular process plan as illustrated in Table 1. The circulation of parts within the flexible production system is tracked by a radio-frequency identification (RFID) reader, which allows to uniquely identify each part. Each transportation task is executed using, the shared, the IRB 1400 robot.

Table 1. Process plan for the catalog of parts

Sequence	Part "A"	Part "B"
#1	punch	drill.1
#2	drill.1	drill.2
#3	drill.2	punch
#4	inspection	inspection

The available resources at the small-scale production system possess a set of skills as illustrated in Table 2.

Table 2. Process plan for the catalog of parts

Resource	Skill	Machine number	time
Punching	punch	{1, 2}	{8.2, 8.4}
Indexed 1	drill.1	{3, 5}	{7.4, 8.4}
Indexed 1	drill.2	{4, 6}	{8.4,7.6}
Inspector	inspection	7	7.6

Having this, a control system manages the production in the system, comprising heterogeneous automation control devices by introducing agility and reactivity to the occurrence of unexpected disturbances, such as new products and machine failures, namely an ADACOR based holonic system is deployed in this small-scale system [11].

A crucial aspect, addressed in this paper, is the cell production scheduling that must address responsiveness issues, particularly allowing a swift (re)scheduling according to the system production needs and/or disturbances.

4 Optimization procedures

In the present work, the genetic algorithm (GA) was used, which is an optimization procedure based on the biological analogy of "survival of the fittest", oriented to solve constrained or unconstrained optimization problems [4].

The GA works with a population of individual, where each individual represents a given solution. The GA repeatedly modifies the population trying to improve the quality of the population over several generations. At each generation (iteration), it selects, randomly, individuals from the previous population to apply crossover and mutation procedures. In crossover procedure two random individuals are selected and exchange genes to obtain a different individual, whereas in mutation procedure only one individual is randomly selected and new genes are introduced [3].

The GA can be applied to solve a variety of optimization problems, including problems in which the objective function is discontinuous, non-differentiable, stochastic, or highly nonlinear [9].

The genetic algorithm applied in this work is summarized by the following algorithm.

Algorithm 1 : Genetic Algorithm

- 1: Generates a randomly population of individuals, P , with dimension N
 - 2: **while** Stopping criterion is not met **do**
 - 3: $BP =$ Best 50% individuals from population P
 - 4: $BP' =$ Apply crossover procedure in population BP
 - 5: $BP'' =$ Apply mutation procedure in population BP
 - 6: $P = N$ best individuals of $\{P \cup BP' \cup BP''\}$
-

The initial population, P consists of N individuals, where each of one represents a feasible schedule (all constraints are satisfied).

The iterative procedure terminates after a maximum number of iterations (number of generations) or after a maximum number of function evaluations.

5 Numerical Results

The case study considers different scenarios, constructing different amounts of both parts A (n_A) and B (n_B), considering four scenarios namely: 10 parts of A and 5 parts of B; 5 parts of A and 10 parts of B; 10 parts of A and 10 parts of B; 20 parts of A and 10 parts of B.

In this work, it was consider $x = (x^1, \dots, x^{n_A+n_B})$ where x^i represents the part to construct ("A" or "B") and each part needs four jobs, defined in Table 1, so x_j^i represents the machine that do the job j of the part x^i , for $j = 1, \dots, 4$ and $i = 1, \dots, n_A + n_B$. For each x is possible to define the schedule of the n_A parts of "A" and n_B parts of "B", defined as $S(x)$. The optimization problem is defined as $\min S(x)$.

The GA algorithm described in the previous section was used to perform the identification of the best possible scheduled for the presented case studies. For the maximum number of iterations it was considered the value 100 and 5000 for the maximum number of function evaluations.

Since the genetic algorithm is a stochastic procedure the algorithm has performed 100 runs, from which was retrieved the minimum value in all runs, resulting in the sequence "BBBBAAAAAAAAAAB", which needs 137.4 seconds to construct all parts.

Table 3. Schedule obtained using genetic algorithm.

	Jobs													
	B	B	B	B	A	A	A	A	A	A	A	A	A	B
Machines	3	5	3	3	1	1	2	2	2	1	2	2	1	5
	6	4	4	4	5	3	3	3	5	3	3	3	3	4
	2	1	1	2	4	6	4	4	6	4	4	4	4	2
	7	7	7	7	7	7	7	7	7	7	7	7	7	7

The allocation sequence is represented in the gantt chart presented in Figure 2, where the colored columns represent the parts to be executed (one color per part) and in each line is the corresponding machines that the part has to visit in order to be built.

Due to the fact that there is no dependency between the parts, the algorithm leads to many global solutions to solve this flexible manufacturing problem. This means that the execution order of the parts order can be different and still have an optimal minimum value to performed all different parts "A" and "B".

The table 4 presents the results for the four scenarios, illustrating the minimum value of the objective function $S(x)$, S_{min} ; the average value of the found minimum, S_{avg} ; the number of possibles results found with the minimum value, Nx_{min} ; and the time, in seconds, that the GA needs to solve the problem $Time$.

Table 4. Schedule obtained using genetic algorithm for different scenarios.

Scenarios	S_{min}	S_{avg}	Nx_{min}	Time (s)
10 A and 05 B	1.372×10^2	1.373×10^2	57	2.45×10^2
05 A and 10 B	1.372×10^2	1.373×10^2	47	2.32×10^2
10 A and 10 B	1.752×10^2	1.753×10^2	57	3.91×10^2
20 A and 10 B	2.512×10^2	2.513×10^2	65	7.91×10^2

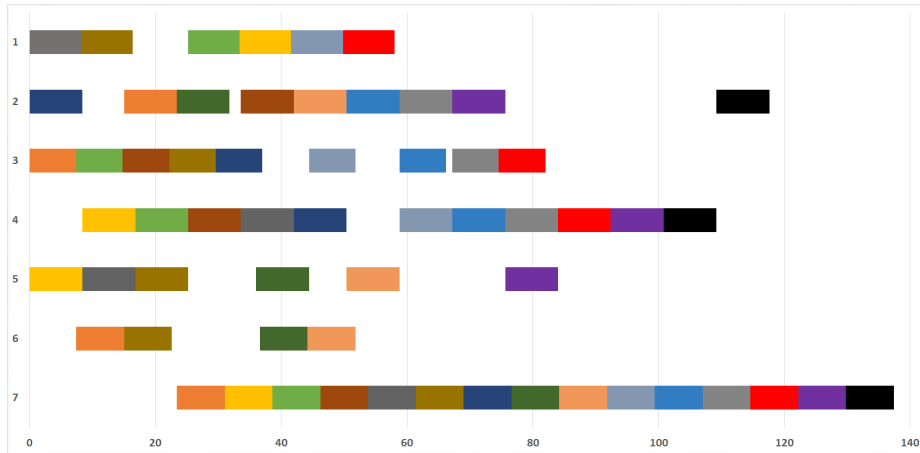


Fig. 2. Gantt chart

Notice that in each scenario the genetic algorithm has $(n_A + n_B)!$ possible schedules to solve the problem. Is possible to verify that the time needed to solve the problem does not increases proportionally with the increase in the number of parts to be built.

6 Conclusion and Future work

This paper has presented a flexible manufacturing system (FMS) composed by a set of resources that is able to produce two product types, having each one a given production sequence. The paper addresses the problem of selecting the most appropriate product production sequence. This topic assumes a crucial importance in the way that the transportation of the parts to be produced in the FMS is accomplished by a shared robot, making its usage critical.

A genetic algorithm approach was developed to, having a given production batch, select the most appropriate production sequence. Results have shown that the used algorithm is able to successfully reach an optimized solution.

Future work will be devoted to further enhance the GA algorithm, making it more efficient, particularly in computation time. Additionally, the algorithm will be embedded into the control layer existent in the FMS, namely the multi-agent system where the IRB 1400 robot tasks are managed by a dedicated agent.

Acknowledgments

This work was funded by the ERDF European Regional Development Fund through the COMPETE Programme (operational programme for competitiveness) and by National Funds through the FCT within projects SAICTPAC/0034/2015, POCI-01-0145-FEDER-016418, POCI-01-0145-FEDER-007043 and UID/CEC/00319/2013.

References

1. Baar, T., Brucker, P., Knust, S., *Tabu-search algorithms and lower bounds for the resource-constrained project scheduling problem*, Meta-heuristics: Advances and trends in local search paradigms for optimisation, 18, 1998.
2. Biskup, D. *A state-of-the-art review on scheduling with learning effects*, European Journal of Operational Research, Volume 188, Issue 2, pp. 315329, 2008.
3. Guevara, G., Pereira, A. I., Ferreira, A., Barbosa, J., Leitão, P. *Genetic algorithm for flexible job shop scheduling problem - a case study*. In Proceedings of the International Conference on Numerical Analysis and Applied Mathematics 2014, Book Series: AIP Conference Proceedings, n. 1648, pp. 140006.1-140006.4, 2015.
4. Holland, J. H. *Adaptation in Natural and Artificial Systems*, Ann Arbor, The University of Michigan Press, 1975.
5. Jain, A. S., Meeran, S. *A State-Of-The-Art Review Of Job-Shop Scheduling Techniques*, 1998.
6. Kennedy, J., and Eberhart, R. *Particle swarm optimization* Proceedings of ICNN95 - International Conference on Neural Networks, 4, 19421948, 1995.
7. Kolakowska, E., Smith, S. and Kristiansen, M. *A Constraint optimization model of a scheduling problem for a robotic arm in automatic systems*, International journal of Robotics and Autonomous Systems, pp. 267-280, vol. 62, n.2, 2014.
8. Kuhn, K., Loth, S., *Airport Service Vehicle Scheduling*, Eighth USA/Europe Air Traffic Management Research and Development Seminar (ATM2009), 2009.
9. Kumar, M., Husian, M., Upreti, N., Gupta, D. *Genetic Algorithm: Review and Application*, International Journal of Information Technology and Knowledge Management, 2(2), pp. 451-454, 2010.
10. Lawer, E., Lenstra, J., Rinnooy Kan, A., Shmoys, D. *The travelling salesman problem*. Chichester, UK: Wiley; 1985.
11. Leitão, P., Restivo, F. *ADACOR: A Holonic Architecture for Agile and Adaptive Manufacturing Control* Computers in Industry, vol. 57 (2), pp. 121130, 2006.
12. Lindholm, P. Giselsson, N-H. Quttineh, H. Lidestam, C. Johnsson, K. Forsman, *Production Scheduling in the Process Industry*, 22nd International Conference on Production Research, 2013.
13. Storer, R. H., Wu, S. D., Vaccari, R. *New search spaces for sequencing problems with application to job shop scheduling*, Manage Sci 38, 14951509, 1992.
14. Wall, M. B, *A Genetic Algorithm for Resource-Constrained Scheduling*, PhD, Massachusetts Institute of Technology, 1996.