



**HAL**  
open science

# Bi-Objective Indirect Optimization of Robotic Transportation Task Assignment Based on Auction Mechanism

Souleymane Moussa Goumeye, M'Hammed Sahnoun, Fabrice Duval,  
Abdelaziz Bensrhair

► **To cite this version:**

Souleymane Moussa Goumeye, M'Hammed Sahnoun, Fabrice Duval, Abdelaziz Bensrhair. Bi-Objective Indirect Optimization of Robotic Transportation Task Assignment Based on Auction Mechanism. IEEE Electron Devices Technology and Manufacturing Conference (EDTM), Apr 2020, Penang, Malaysia. pp.1-4, 10.1109/EDTM47692.2020.9117862 . hal-04082880

**HAL Id: hal-04082880**

**<https://hal.science/hal-04082880v1>**

Submitted on 10 May 2023

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Bi-Objective Indirect Optimization of Robotic Transportation Task Assignment Based on Auction Mechanism

Moussa G. Souleymane<sup>1,2</sup>, Sahnoun M'hammed<sup>1</sup>, Duval Fabrice<sup>1</sup>, Benshrhair Abdelaziz<sup>2</sup>

<sup>1</sup>Cesi Lineact, Research and Innovation Laboratory, Rouen, France

<sup>2</sup>Litis Insa of Rouen, Computer Science and Information Processing, Rouen, France  
e-mail:smoussa@cesi.fr

**Abstract**—Flexible manufacturing systems (FMS) are used to ensure the production of several products using the same set of production tools. Transportation tasks using robots are introduced in order to increase the routing flexibility of the FMS. Several researchers are interested in the optimization of the production planning but few of them considered the transportation planning. On another hand, each optimization algorithm needs an adapted control architecture to be executed. In this paper, we propose the use of a hybrid architecture control to implement the bi-objective optimization algorithm based on Multi-Objective Genetic Algorithm (MOGA). Results showing the efficiency of the proposed approach are presented.

**Keywords**— *Task Assignment, Optimization, Auction, Robots, Transportation tasks*

## I. INTRODUCTION

The concept of Flexible Manufacturing Systems (FMS) was introduced to deal with market requirements affected by the unpredictable customer demands. FMS are characterized by their ability to adapt to new situations in the production systems and consist in integrating smart elements to allow the production of several kind of products. In addition, the organisation of production systems has evolved from production lines to production cells in order to provide more flexibility to the production systems [1], which has led to the emergence of new kind of tasks such as the transportation of products between production machines. The robotization of such tasks using autonomous mobile robots is necessary due to their drudgery and repeatability. However, it increases the complexity of FMS.

Thus, it is important to find a way to assign task to these autonomous entities, while reducing the transportation cost and optimising the overall performance of the factory. Many researchers focused on the way to allocate tasks to these autonomous and smart elements in FMS separately or with integration of machines production planning [2] [3]. Several criteria are used of evaluate the system such as the makespan, the cost, the delivery time.... Due to the complexity of this kind of problem, two main methods are proposed in the literature to resolve the allocation problem. The first one is based on a central algorithm that can propose optimal solution. The drawback of these methods is the lack of flexibility. The second one try to increase the flexibility of the system by making a distributed decision-making, where every component of the system make its own decision. The drawback of these methods is the non-optimality of the global solution. In order to overcome the inconvenient of these two approaches, a third solution based on indirect optimization has been developed. These methods seek to define a set of optimal rules that draw the framework of decisions and actions made by the smart resources.

In this paper, we seek to use an indirect optimization mechanism to develop an algorithm able to provide optimal

resource assignment. It is based on the use of a bi-objective genetic algorithm that optimises both the total cost and completion time of each job performed by smart element such as mobile robots and/or robotic arms. The proposed rules are a set of virtual prices and profits that can generate an assignment and scheduling of the smart resources.

The reminder of this paper is as follow: The next section presents a state of the art on task assignment algorithms in FMS context. We then turn our attention, to detailing the proposed approach. An experimentation is performed on a simulated flexible workshop for the research validation. The paper closes with conclusion and plans for future works.

## II. LITERATURE REVIEW

### A. Flexible manufacturing system

Over the last few decades, there have a number of significant changes in customer demands for manufacturing products. While in the past, price sensitivity led to more uniform products, nowadays, many high-quality and often customized products at reasonable prices and prompt delivery are required [4]. These market demands force the factories to adapt to the pace that is constantly growing. The integration of flexible manufacturing system (FMS) is a solution to cope with these changes.

The FMS control systems must be responsive and adapt quickly to external changes. Be tolerant of breakdowns and be able to minimize their consequences. To get these features (fault tolerances, adaptability and reactivity), they should be as decentralized as possible and as centralized as necessary [5].

In FMS, three kind of interaction between elements can be notified : hierarchical interaction, heterarchical one and hybrid approach [4]. In the hierarchical organization decisions are taken in the high levels. The main advantages of this approach are their robustness, fast response time, and their reduced complexity of implementation [6]. However, hierarchical architectures suffer from a lack of flexibility and high rigidity. This makes it difficult to add new elements (machines, robots,...) in the system [4]. Heterarchical architectures are represented as sets of resources, and each set is controlled by a single controller. Since there is no hierarchy in heterarchical systems, we can only have one level in the composition of the system. Their main advantages are a high flexibility and a local intelligence to deal with unexpected events. Despite these advantages, the heterarchical control system, have some drawbacks. In fact, the local decisions lead to have myopia of entities since there is no global decision entity [7].

In order to benefit from the overall optimization of the hierarchical approach and the responsiveness of the heterarchical approach without suffering the myopia of

agents, the hybrid approaches have been proposed. The hybrid architectures use a centralized approach when the objective is optimization, and a more heterarchical approach in the presence of unexpected events and modifications [5].

Various architectures and task allocation algorithms have been structured using these three classes (hierarchical, heterarchical and hybrid) to benefit from their advantages.

### B. Task assignment in FMS

Reference [3] developed an algorithm for assigning tasks to a multi-robot system, based on a heterarchical auction mechanism by taking into account several constraints (Deadline, robot capacity,...). Each robot has a payoff when performing a task. This gain constitutes its bid and allows the submission to its neighbors when a task is proposed. Reference [2] experimented a fully distributed architecture to deal with the assignment of tasks to Automatic Guided vehicles (AGV). Biddings are used to assign tasks, where the auctioneers are the machines. This makes the approach decentralized because each machine manages the assignment of tasks according to the products it needs or its neighbor machine needs. The AGV able to propose the minimal transportation time wins the bids. These type of approaches suffer from the same shortcomings as totally heterarchical control systems: absence of a global decision. An architecture composed of three levels for FMS scheduling and control has been developed in [8]. The highest level is in charge of an advanced scheduling, which it carries out according to the returns of the second level, periodically or in the event of disruptions. Data from the highest levels are taken as advices and not orders by the bottom levels, which considers the state of the workshop in real time and reschedule if necessary. This method allows a multi-level scheduling, in which the final decision is made by the entities at the lowest level.

These multi-level scheduling methods allow a fair workloads among available resources resilient to disruptions. However, they require good synchronization between the different decision levels to detect when a rescheduling should be performed by whose and at which level. Also, since the final decision always comes down to one level, it may happen that scheduling of other levels could be removed. Reference [9] proposed a generic algorithm for assigning tasks through an auction process. In this approach, virtual prices are associated to operations and virtual minimum profits are set for resources. These virtual values (prices and profits) will be the guidelines of the solution. The Figure 1, shows the difference between the classical approaches and the indirect one. The classical approaches are fully distributed, where the auction is won by the highest bidder among the candidates without a real intervention of the central auctioneer, which has a global vision of all the resources. This type of configuration can then lead to a poor distribution of tasks. In indirect approaches, the central auctioneer acts on the bidding by attributing the virtual profits to resources and virtual prices to operations. These virtual values guide and influence the selected choice.

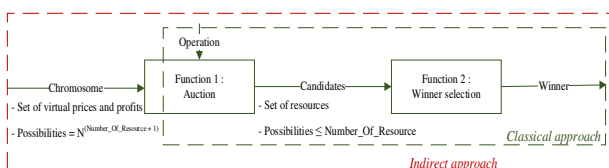


Fig. 1. Auctions procedures.

The approach proposed in [9] does not take into account the assignment of multiple product to resources, which represent its main drawback. The approach we have proposed in the rest of this paper aims to improve this method, by the introduction of many improvements and differences that will be detailed.

## III. PROPOSED MECHANISM

The task allocation is based on offline scheduling. The approach is composed of ten steps, expanded on four main algorithms as shown in Figure 2. The step 0, has the sole objective of generating a random chromosome. The chromosome value are chosen randomly, within a wide range (between 0 and highest operations cost N).

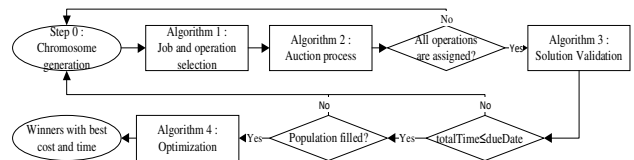


Fig. 2. Approach overview.

### A. Assumptions and Constraints

#### 1) Assumptions :

- a) Each job is associated to a product (e.g  $j = 2$ , means the second product).
- b) Each transportation resource (robot) has its own performances to achieve an operation (time, cost), and all the robots are able to execute the transport operations.
- c) Each production machine and transportation resource only processes one product at time, and the production planing is guided by the transportation scheduling. We consider a job shop scheduling problem.

#### 2) Constraints

- a) The jobs must be executed in parallel in order to reduce the makespan. A precedence constraints existes between operation of a job.
- b) The selection of operation for biddings is constrained by the availability of picking machine and destination one, since the production planing is guided by the transportation scheduling.

TABLE I. LISTE OF SYMBOLS

Symbol	Notation
J	Set of jobs
R	Set of robots
W	Set of robots winners
j	A job, $j \in J$
r	A robot, $r \in R$
unitCost <sub>r</sub>	Unit cost per time associated to robot r
C <sub>roj</sub> ; T <sub>roj</sub>	Cost / Time associated to robot r for operation op of job j
dt <sub>roj</sub>	Delivery time of product associated to job j during operation op by robot r
T <sub>max</sub>	Time at which both the robot and the product are available
dest <sub>jop</sub>	Destination of product associated to job j during operation op

<i>Symbol</i>	<i>Notation</i>
$dep_{jop}$	Picking machine of product associated to job $j$ during operation $op$
$P_{opj}$	Virtual price associated to operation $op$ of job $j$
$F_{minr}$	Minimum virtual profit associated to resource $r$
$ch$	A chromosome
$pt$	Machine processing time
$nbrMachine$	Number of machines in the workshop
$busyMachine$	Number of occupied machine
$bufferProduct$	Time buffer contening the processing time of all products
$bufferRobot_r$	Time buffer containing the activity of robot $r$

## B. Algorithms

Step 0: Chromosome generation

$$Ch = [P_{11}, P_{12}, \dots, P_{nk}, F_{min1}, \dots, F_{minr}]$$

### Algorithm 1: Job and operation select

Step 1: Choice of job and operation

$$j = \text{index}(\text{Min}(\text{bufferProduct}))$$

$$op = op_{-1} + 1$$

Step 2 : Machine availability

**If**  $dest_{jop}$  and  $dep_{jop}$  **not** busy:

“The machines are free”

$$dest_{jop} = \text{busy}$$

**Else :**

“The machine is busy”

Goto(step 1)

**End if**

**Return**  $j, op$

**End if**

In *algorithm 1*, we select a job and an operation to be sent to robots for bidding based on two criteria. First, the product associated to the job must have the lowest buffer time (Step 1). This condition guarantees a low makespan, because all the parallel jobs will almost end at the same time. Second, the machines of destination and picking of the product must be available (Step 2). While the machines are not available, the operation can not be performed for the moment and another operation have to be chosen. The output of the algorithm 1, is a feasible operation of a job.

### Algorithm 2: Auction process

Step 3: Resource performance estimation

“Operation cost”

$$C_{ropj} = T_{ropj} * \text{unitCost}_r$$

“Availability time of robot and product”

$$T_{max} = \text{Max}(\text{bufferProduct}_j, \text{bufferRobot}_r)$$

“Product delivery time”

$$dt_{ropj} = T_{max} + T_{ropj}$$

Step 4 : Potential candidates for the operation

**For**  $r$  in  $R$  :

**If**  $P_{opj} - C_{ropj} \geq F_{minr}$  :

$candidates_{opj} \leftarrow r$

**End if**

**End for**

**If**  $candidates_{opj} == \text{None}$  :

“The operation  $op$  of job  $j$  has no candidate”

Goto(Step 0)

**End if**

Step 5 : Winner of operation between the candidates

**For**  $r$  in  $candidates_{opj}$  :

$$winner_{opj} \leftarrow \text{Min}(dt_{ropj})$$

**End for**

*Algorithm 2*, receives as input the operation selected in algorithm 1. In this part, each transportation resource

estimates its performances (cost and delivery time of product) related to the operation (step 4). The delivery time ( $dt_{ropj}$ ) of product is obtained by adding to the time needed by the robot to perform the operation and the time at which both the robot and the product are available ( $T_{max}$ ). After bids reception the auctioneer selects the potential candidates and winner to the operation using the same criteria as in [9]. All the robots generating a payoff, are candidates (step 5) and the robot minimizing the delivery time between candidates wins the bidding (step 6).

### Algorithm 3: Solution validation

Step 6: Total time and cost associated

“Total Cost”

$$\text{totalCost} = \sum_{r \in W} \sum_{j \in J} \sum_{op \in J} C_{ropj}$$

“Total Time”

$$\text{totalTime} = \sum_{r \in W} \sum_{j \in J} \sum_{op \in J} dt_{ropj}$$

Step 7: Chromosome validation

**If**  $\text{totalTime} \leq \text{dueDate}$  :

population  $\leftarrow ch$

fitnessTime  $\leftarrow \text{totalTime}$

fitnessCost  $\leftarrow \text{totalCost}$

**Else :**

“Generate a new chromosome”

Goto(step0)

**End if**

Once, all operations are assigned, the chromosome (generated at step 0) is validated through the *algorithm 3*. This algorithm provides a population of chromosomes to *Algorithm 4* in charge of optimization.

### Algorithm 4: Optimization

Step 8: Initial pool

**For**  $i$  in number\_population :

do (step0, algorithm1, algorithm2, algorithm3)

**End for**

Step 9: Fitness evaluation

sort(fitnessTime)

sort(fitnessCost)

**For**  $ch$  in population

$$\text{fitness} \leftarrow \text{rank}_{ch}(\text{fitnessCost}) + \text{rank}_{ch}(\text{fitnessTime})$$

**End for**

**sort**(fitness)

The optimization process in *Algorithm 4*, is based on genetic algorithm with typical tools (Initial population, mutation and crossover). The fitness function is a set of rank associated to chromosome in the population. The rank of a chromosome is obtained by adding its rank regarding the total costs fitness (fitnessCost) and its rank regarding the total times fitness (fitnessTime). More details about the fitness evaluation are provided in [10]. Our bi-objective function minimizes both the total cost and time of all the operations :

$$\text{Min}(\text{totalCost}, \text{totalTime})$$

In the approach proposed in [9], only the total cost is optimized. The different constraints such as the precedence between operations, machines availability and multiple jobs assignment are also not taken into account.

## IV. EXPERIMENT

### A. Description

The experience we made is a workshop composed of four production machines, four products, three mobile robots for products transportation. A job is associated to each product composed of four operations. The whole operations have to

meet a due Date of 300 seconds. The Figure 4 shows an example of product sequence.

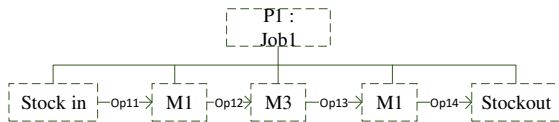


Fig. 3. Sequence of job 1 associated to product 1.

### B. Results

The Figure 4, shows the result of task assignment to robots that considers all the constraints (Precedence of operation in a job, parallel job execution...). The result provides as well a sequence between jobs. The robots buffer containing the operations won and the date where these operations are processed, testify the interest of the indirect approach of optimization. Thanks to the intervention of the central auctioneer through virtual prices and profits, the operation was distributed among the robots. Since robots have different performances (costs and times of operations), a direct optimization approach would have led to having the robots with better performance as winners of operations. This leads to a poor distribution of tasks among resources. The poor distribution of tasks is responsible of resources overexploitation, which leads to a rapid decrease in their performance.

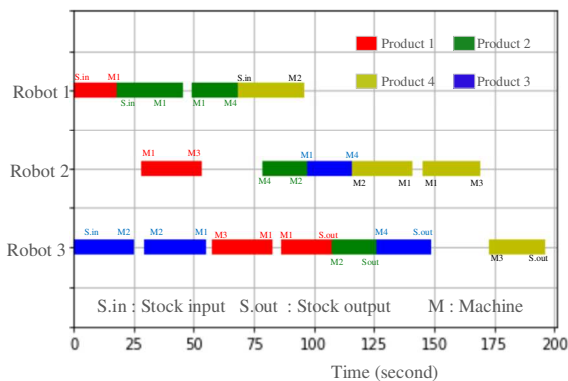


Fig. 4. Transport and production planning found

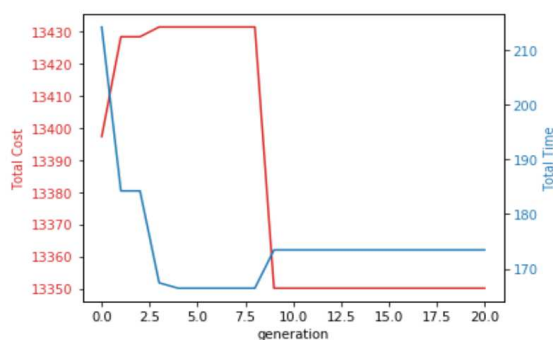


Fig. 5. Total cost and time evolution per generation

The results of the optimization (Figure 5) demonstrate the effectiveness of the bi-objective function. For the first generations, the total cost and time are inversely proportional (One of the values is high and the other low). After 8

generations, both of the value total cost and time are minimized.

## V. CONCLUSION

To summarize, in this paper we presented a new approach for assigning tasks to resources in an industrial context for the transportation of products. In general, tasks assignment provide optimal or optimized solution consisting in a timed list tasks for each resource. In addition, the task assignment based on auction are highly distributed in the sense that the choice of winner only depends on robot performances. This distribution leads to the loss of the overall objectives of the system. The presented approach provides hybrid mechanism for the control of FMS based on auction, in which the main auctioneer influences the selection of winner. In fact, an optimization process provides optimal virtual values associated to operations and resources, which lead to an optimal allocation tasks solution.

The presented approach provides an advanced scheduling of both production and transportation process through the establishment of a set of rules (virtual price and profits). The next step is to test the proposed solution with disturbances and introduce a dynamic allocation process, in order to implement approach on the industrial platform of the LINEACT laboratory.

## ACKNOWLEDGMENT

This research was made possible thanks to €2.6 million financial support from the European Regional Development Fund provided by the Interreg V France Channel England Programme in context of CoRoT Project.

## REFERENCES

- [1] Y. Koren et M. Shpitalni, "Design of reconfigurable manufacturing systems," *Journal of manufacturing systems*, vol. 29, pp. 130-141, 2010.
- [2] M. H. F. B. M. Fauadi, W.-L. Li et T. Murata, "Combinatorial auction method for decentralized task assignment of multiple-loading capacity AGV based on intelligent agent architecture," 2011.
- [3] L. Luo, N. Chakraborty et K. Sycara, "Distributed Algorithms for Multirobot Task Assignment With Task Deadline Constraints," *IEEE Transactions on Automation Science and Engineering*, vol. 12, pp. 1-13, 7 2015.
- [4] M. Peschl, "An architecture for flexible manufacturing systems based on task-driven agents," 2014.
- [5] P. Leitão et F. Restivo, "ADACOR: A holonic architecture for agile and adaptive manufacturing control," *Computers in industry*, vol. 57, pp. 121-130, 2006.
- [6] N. A. Duffie et V. V. Prabhu, "Real-time distributed scheduling of heterarchical manufacturing systems," *Journal of Manufacturing Systems*, vol. 13, pp. 94-107, 1994.
- [7] C. Pach, « ORCA: Architecture hybride pour le contrôle de la myopie dans le cadre du pilotage des Systèmes Flexibles de Production,» 2013.
- [8] L. Bongaerts, «Integration of Scheduling and Control in Holonic Manufacturing Systems (PhD Thesis),» 1988.
- [9] N. He, "Agent-based Hierarchical Planning and Scheduling Control in Dynamically Integrated Manufacturing System," 2011.
- [10] M. Sahnoun, B. Bettayeb, S.-J. Bassetto et M. Tollenaere, "Simulation-based optimization of sampling plans to reduce inspections while mastering the risk exposure in semiconductor manufacturing," *Journal of Intelligent Manufacturing*, vol. 27, pp. 1335-1349, 2016.