



**HAL**  
open science

# Development of a methodology for performance analysis and synthesis of control strategies of multi-robot pick & place applications

Gaël Humbert, Minh Tu Pham, Xavier Brun, Mady Guillemot, Didier Noterman

## ► To cite this version:

Gaël Humbert, Minh Tu Pham, Xavier Brun, Mady Guillemot, Didier Noterman. Development of a methodology for performance analysis and synthesis of control strategies of multi-robot pick & place applications. JCM 2016, Sep 2016, Catania, Italy. hal-01355113

**HAL Id: hal-01355113**

**<https://hal.science/hal-01355113v1>**

Submitted on 16 Apr 2019

**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.

# Development of a methodology for performance analysis and synthesis of control strategies of multi-robot pick & place applications

Gaël Humbert<sup>1</sup>, Minh Tu Pham<sup>1</sup>, Xavier Brun<sup>1</sup>, Mady Guillemot<sup>2</sup>, Didier Noterman<sup>3</sup>

<sup>1</sup> Université de Lyon, INSA-Lyon, Laboratoire Ampère, 20 Avenue Albert Einstein, Villeurbanne 69621, France

<sup>2</sup> Université de Lyon, INSA-Lyon, Laboratoire LAMCOS, Villeurbanne 69621, France

<sup>3</sup> Université de Lyon, INSA-Lyon, Laboratoire DISP, Villeurbanne 69621, France

\* Corresponding author. E-mail address: gael.humbert@@insa-lyon.fr

**Abstract** This paper deals with a new simulation tool for the improvement of multi-robot pick & place applications performance combining behavioral simulation of multiple robots and products flows. A novelty of the proposed work is to take into account in the simulation not only the scheduling rules of each robot, but also the robots collaborative aspect to ensure the desired overall performance for a given task. The transition from simulation to implementation of pick & place strategies is also an issue tackled in this paper. By using a typical example consisting of comparing techniques to optimize the workflow, the utility of the simulation tool is proven. First experimental results validate the simulation results.

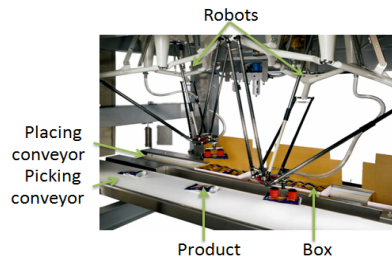
**Keywords:** pick & place application; collaborative strategies; scheduling rules; software tool; experimentation

## 1 Introduction

In recent years, the customer's demand of productivity and flexibility for their production lines has largely increased. This is why robots and robotic pick & place cells are more and more present in some industrial fields such as the food industry. In high-performance applications, typical characteristic of a pick & place robot can reach the following values : velocity 10 m/s, acceleration 100 m/s<sup>2</sup> precision +/- 0.1mm, pick & place cycle 0.40s on average. To improve the performance of these applications it is necessary to improve the design of current production systems (number of robots, performance etc.) whilst also improving the management of flows and workload management when several robots are used.

A pick & place application is usually composed of a series of several robots installed in a line one after the other taking products on a first conveyor and placing them in boxes located on a second conveyor, see Figure 1 [1, 2].

On a multi-robot packaging cell, when there is no workflow optimization system, "pick" instructions are divided equally between the first robots. A final robot is added to try to recover the products that could not be taken by previous robots. Products initially assigned to a robot may not be taken because they finally are out of the robot workspace because of a lack of boxes to fill, for example. In addition there is an unbalanced of the workload between robots.



**Fig. 1.** Robotic cell with delta robot

To the best of our knowledge, in industrial and academic context, there are no simulation tools that take into account the four following aspects: a behavioral simulation of the robots, a simulation of the work environment (product flow, boxes flow), the collaborative work of several robots and finally the possibility to go from simulation to experimentation.

The first contribution of this paper is the development of a software interface that represents the robotic cell part in a 3D environment. The developed software is able to simulate realistic product and box flows, generate the trajectory of the end effector, and propose several collaboration strategies between robots.

The second contribution is to propose a tool that includes experimental aspects in order to directly go from simulation to implementation. Simulation must be done in such a manner that the translation is as easy as possible to have a fast implementation in-situ, simple language and similar controller architecture are used in simulation and practice.

The third contribution is to show a comparative study of the simulation of different pick & place strategies for several robots. An experimental validation is also presented. The results show that simulation and experimentation results are closed.

Section 2 presents a new simulation tool dedicated to pick & place applications, the software environment and pick & place strategies are shown. Section 3 shows a comparative study of the simulation of different pick & place strategies for several robots. An experimental validation is also presented. The results show that simulation and experimentation results are closed.

## 2 A new simulation tool dedicated to pick & place applications

### 2.1 Software tool

In the literature, there are several works dedicated to robotic pick & place simulation, which are only used for visualization, to verify the kinematics and dynamics. They are also used for robot design to validate its behavior, its movements and its interaction with the environment (collision detection). Johari *et al.* [3] have used Workspace5 to visualize an entire robotic application system in order to detect collision between robots and the environment. Sam *et al.* [4] have designed a pick & place robotic system using SolidWorks Softmotion software to study the motion of the modeled articulated robot.

To improve the productivity of a pick & place multi-robots application, the flow management has to be improved. There are several programs that are able to simulate this. Mirzapourrezaei *et al.* [5] have used Witness to evaluate various aspects of manufacturing systems. The objective was to escalate the productivity and efficiency of the line. Hindle *et al.* [6] have used Simul8 to answer the complex scheduling problem of sequencing part requirements through a composites manufacturing center. Nikakhtar *et al.* [7] have compared two simulation tools: Arena and Witness. However, these programs are dedicated to flow simulation. Visualization is very basic, and mainly focused on the flows. 3D visualization does not exist and it is difficult to represent the kinematic and dynamic behavior of the robots.

Unlike other works, the environment software used allows the creation of a virtual machine, in 3D and in real time. This simulated robotic cell will have the same kinematics and the same dynamics as the real one, its environments can also be simulated: products arriving on a conveyor etc. Scenarios can be implemented to verify its behavior. A high-level layer can be used to implement a products pick & place strategy and a collaboration strategy between several robots if necessary. This software is also modular, it can configure the production system (robot, conveyor etc.), its environment (products, boxes etc.) and the different scenarios.

The pick & place application creation consists of several steps, first the definition of the simulation model is carried out. At this stage the graphic objects and the kinematic behaviors of the application objects are defined (robots, conveyors etc.). The second step is the development of collaborative strategies between robots in simulation, see sub-section 2.2. Finally, the simulations could be run, see Figure 2, to test the model behavior and analyze the results of the different strategies, see sub-section 3.1. Once the parameters of simulation are optimized, experimentation in-situ is carried out, see sub-section 3.2. Experimental tests can be done to check the algorithms' operation and test their performance.

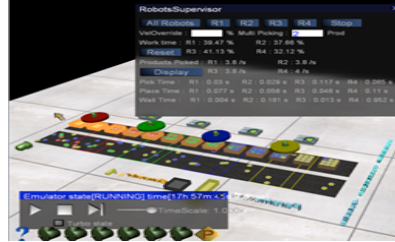


Fig. 2. Simulation example with four robots and two co-current conveyors.

## 2.2 Pick & place strategies

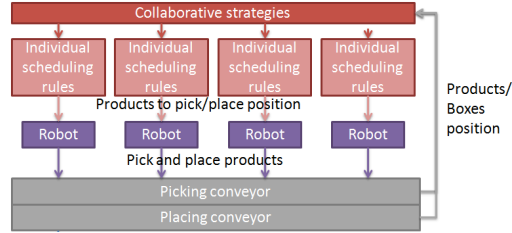
When a single robot is used a queue or a basic sort direction is sufficient. Matton et al. [8] have proposed innovative online scheduling rules based on queue.

If several robots are used, more complicated algorithms than queue are necessary. The aim is to manage robots towards products, boxes and conveyors. To do this, it is better to use optimization algorithms. Research works are related to optimization algorithms used in some robotic applications. Huang *et al.* [9] have utilized the greedy randomized adaptive search procedure to search for the optimal combination of part dispatching rules. Slim *et al.* [10] have compared three metaheuristics: ant colony optimization, genetic algorithm and particle swarm optimization. The aim is to maximize the throughput rate taking of a pick & place robotic system into account the execution time. Fujimito *et al.* [11] have used a genetic algorithm to seek the best combination of dispatching rules in order to obtain an appropriate production schedule. In these works, only simulation is used. Experimentation the translation from simulation to experimentation is not tackled.

In the literature, a few patents are related to pick & place strategies. Izumi *et al.* [12] have filed a patent about conveyors sharing in order to share the robots workload.

The tool developed incorporates two levels of strategies shown in Figure 3. Simple individual scheduling rules for a single robot can be:

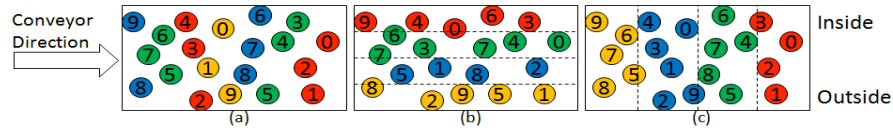
- FIFO: First In First Out. The robot picks the first product in its workspace.
- LIFO: Last In First Out. The robot picks the last product in its workspace.
- SPT: Shortest Processing Time. The robot picks the nearest products of its end effector.



**Fig. 3.** Simulation architecture with two levels of strategies.

There are also collaborative strategies that assign the products to the robots before they arrive in their workspace. An example where four robots are used is given in figure 4. The products 1 are assigned to the robot 1, the products 2 are assigned to robot 2...

- DownToUp: Assign to the robots the products one by one from the downstream to upstream of the conveyor (Figure 4.a).
- Horizontal: Assign to the robots a horizontally area corresponding to its number (Figure 4.b).
- Vertical: Share Assign to the robots a vertically area corresponding to its number (Figure 4.c).



**Fig. 4.** Example of different collaborative strategies.

### 3 Simulation and experimentation results

#### 3.1 Simulation results

Tests were conducted with the algorithms explained in section 2.2. The performance of different algorithms can be assessed using several indicators: number of picked products by each robot, the total number of picked products, average picking-placing time and finally the workload percentage which is defined by the following equation (1) with  $T_{Pick}$ ,  $T_{Place}$  and  $T_{Wait}$  respectively picking, placing and waiting time in seconds.

$$Workload = \frac{T_{Pick} + T_{Place}}{T_{Pick} + T_{Place} + T_{Wait}} \quad (1)$$

The simulations are performed with the following arbitrary parameters:

3 robots: speed 10 m/s, acceleration 100 m/s<sup>2</sup>; linear movement; conveyors speed: 0.15m/s in co-current; 5.5 products per second with random position; simulation time: 30 min.

**Table 1.** Results of the individual scheduling rules in steady state.

Picking / Placing rules	Product picked	Workload (%)	Average pick-place time (s)
	R1 - R2 - R3 - Total	R1 - R2 - R3	R1 - R2 - R3
FIFO / FIFO	4242 / 4316 / 1344 / 9902	92.2 / 92.2 / 30.4	0.414 / 0.407 / 0.431
FIFO / LIFO	3884 / 3974 / 2042 / 9900	94.3 / 94.2 / 43.6	0.455 / 0.442 / 0.403
SPT / SPT	4605 / 4080 / 1229 / 9914	92.1 / 91.6 / 29.1	0.379 / 0.429 / 0.451

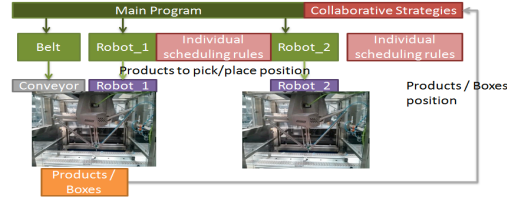
**Table 2.** Results of the collaborative strategies in steady state with FIFO rule.

Collaborative strategies	Product picked	Workload (%)	Average pick-place time (s)
	R1 - R2 - R3 - Total	R1 - R2 - R3	R1 - R2 - R3
DownToUp	3302 / 3299 / 3300 / 9901	76.8 / 76.4 / 75.8	0.442 / 0.439 / 0.437
Horizontal	3662 / 3263 / 2899 / 9824	73.9 / 67.7 / 67.1	0.396 / 0.421 / 0.451
Vertical	3310 / 3312 / 3279 / 9901	74.3 / 73.5 / 75.8	0.425 / 0.420 / 0.42

The Table 1 shows the results in steady state of a simulation where only one individual scheduling rule is applied. First it appears without any collaborative strategies, the workloads of the robots are unbalanced for all the scheduling rules. The first robots pick the maximum of products while the last one picks the remaining products. SPT rule increase the unbalance between robots. In FIFO/FIFO rule the robot remains in one side of its space while in FIFO/LIFO rule the robot will move in a larger area. This is why picking and placing time in FIFO/FIFO rule are smaller than FIFO/LIFO rule. The Tables 2 gathers the results in steady state of a simulation where the individual scheduling rules FIFO/FIFO but with additional collaborative strategies between the robots. The workloads of the robots are balance because the products are distributed equally. It is noteworthy that horizontal strategy is not a good assignment method because the picking and placing time increases from the first robot to the last robot. The main reason is that the assigned areas to the robots are increasingly far of the robot centers. It is clear that the best one is the vertical strategy, the workload is equal and with a reduced picking and placing time.

### 3.2 Experimentation validation

After the simulations, algorithms and strategies translation in PLC language is carried out. To facilitate this translation, the programs are written with the simplest possible functions, which also reduce the execution time. In addition, the controller software uses an object-based language similarly to the simulation software. The architecture of the Figure 5 is the implementation of that of Figure 3. It is composed of the same scheduling rules and collaborative strategy blocks than for the simulation.



**Fig. 5.** Controller architecture with two levels of strategies.

A demonstrator is used to check the operation of the program architecture, algorithms and strategies translated. It is composed of a Schneider Electric P4 delta robot to the pick & place, a ring conveyor, with a sensor for detecting the products and a vacuum gripper for taking the objects and is commanded by a Schneider Electric controller LMC400C and program by SoMachine Motion. The results of the Tables 3 are obtained with the following nominal conditions: Conveyor speed: 0.1 m/s in counter-current, end effector nominal speed: 1.2 m/s, end effector acceleration: 20 m/s<sup>2</sup>, linear movement, products every 50 mm, boxes with two places every 200 mm, picking logic: FIFO, placing logic: LIFO, time: 10 min. The different tests are done with the following conditions:

1. Nominal condition.
2. Nominal condition with picking logic: LIFO.
3. Nominal condition with picking logic: SPT.

**Table 3.** Simulation and experimentation results.

	Test condition	Average pick-place time (s)	Products picked
Simulation / Experimentation	1	1.65 / 1.64	298 / 296
	2	1.82 / 1.84	284 / 282
	3	1.51 / 1.51	319 / 298

Table 3 shows a comparison between simulation and experimentation results. For all the simulation conditions, the picking and placing time and the number of taken products are similar, the error is very low. The experimental results are in accordance with those presented in Table 1. The results of LIFO/FIFO and LIFO/LIFO rules test are reversed because in the simulation test the conveyors are in the same direction while they are in the opposite direction in practice. This shows the interest of the simulation tool to test and improve the pick & place multi-robots performance.

## 4 Conclusion

In this paper, we proposed a new tool to improve the performance of multi-robot pick and place applications. This tool is based on the real-time 3D simulation of the robot tool and of its environment, allowing also the implementation of individual and collaborative control strategies. Several tests have been done, first-



ly, a simulation test to compare different individual and collaborative control strategies. Then, a comparison between simulation and experimental results is conducted and show that the simulation is very close to reality. This tool also allows a fast translation of algorithms from simulation to implementation. One of the interests of this tool is to test different algorithms for the robots before implementation in-situ, to check if they operate properly and to know what is best. This avoids stopping a production line for these tests or to save time if the line is in development. Another interest is to take into account the four following aspects: a behavioral simulation of the robots, a simulation of the work environment, the collaborative work of several robots and finally the possibility to go from simulation to experimentation. Future work is to develop other multi-robot collaboration algorithms using this tool before a validation and a performance analysis on a test bench composed of three robots and two independent conveyors.

**Acknowledgment** The research work reported here was made possible by Schneider Electric with the CIFRE 158/2013.

## References

1. Schubert, R. (2000). Process and apparatus for introducing products into containers. Patent US 6122895 A.
2. Sahin, H. (2005). Design of a secondary packaging robotic system. PhD thesis, Middle est technical university.
3. Johari, N., Haron, H., and Jaya, A. (2007). Robotic modeling and simulation of palletizer robot using workspace5. In 4th International Conference on Computer Graphics, Imaging and Visualization (CGIV 2007), pages 217 \_ 222. IEEE.
4. Sam, R., Arri\_n, K., and Buniyamin, N. (2012). Simulation of pick and place robotics system using solidworks softmotion. In International Conference on System Engineering and Technology (ICSET), pages 1 \_ 6. IEEE.
5. Mirzapourrezaei, S., Lalmazloumian, M., Dargi, A., and Wong, K. Y. (2011). Simulation of a manufacturing assembly line based on witness. In Third International Conference on Computational Intelligence, Communication Systems and Networks (CICSyN), pages 132 \_ 137.
6. Hindle, K. and Du\_n, M. (2006). Simul8-planner for composites manufacturing. In Proceedings of the Winter Simulation Conference, 2006. WSC 06., pages 1779 \_ 1784. IEEE.
7. Nikakhtar, A., Wong, K. Y., Zarei, M., and Memari, A. (2011). Comparison of two simulation software for modeling a construction process. In Third International Conference on Computational Intelligence, Modelling and Simulation (CIMSiM), pages 200 \_ 205. IEEE.
8. Mattone, R., Adduci, L., and Wolf, A. (1998). Online scheduling algorithms for improving performance of pick-and-place operations on a moving conveyor belt. In Proceedings of the IEEE International Conference on Robotics and Automation, ICRA-98, pages 2099 \_ 2105.
9. Huang, Y., Chiba, R., Arai, T., Ueyama, T., and Ota, J. (2012). Part dispatching rule-based multi-robot coordination in pick-and-place task. In 2012 IEEE International Conference on Robotics and Biomimetics (ROBIO), pages 1887 \_ 1892. IEEE.
10. Daoud, S., Hicham, C., Farouk, Y., and Lionel, A. (2014b). E\_cient metaheuristics for pick and place robotic systems optimization. *Journal of Intelligent Manufacturing*, 25 :27\_41.
11. Fujimoto, H., Tanigawa, I., Yasuda, K., and Iwahashi, K. (1995). Applications of genetic algorithm and simulation to dispatching rule-based fms scheduling. In Proceedings 1995 IEEE International Conference on Robotics and Automation , volume 1, pages 190\_195. IEEE.
12. Izumi, T., Koyanagi, K., Matsukuma, K., and Hashiguchi, Y. (2013). Robot system. Patent US 8606400 B2.