Towards an agent-based manufacturing control strategy: a simulation study

Guilherme Cordeiro Lopes
guilherme.cordeiro.lopes@tecnico.ulisboa.pt

Instituto Superior Técnico, Universidade de Lisboa, Lisboa, Portugal

November 2018

Abstract

With the advent of emerging technologies like Big Data, Internet of Things, wireless sensor networks, cloud computing, mobile Internet and Artificial Intelligence, and their introduction to the manufacturing environment, a new paradigm arrived to the industry world: the fourth industrial revolution. With the aim of taking advantage of the new technologies to achieve the smart, flexible and reconfigurable factory of the future, capable of producing customized and small-lot products efficiently and profitably, some advanced manufacturing control systems have been proposed. In this work, a manufacturing system based on intelligent computational agents was designed and tested through simulation with a view to study its applicability as a production system for the factory of the future. A flexible manufacturing system was designed, and its operation was modelled according to a known benchmark. A Multi-Agent System composed of 5 types of reactive agents was designed to control its operation. The agents were modelled using Petri nets and agent communications were defined through the combination of FIPA Interaction Protocols. The system was simulated under the conditions of static and dynamic scenarios, having its performance validated whenever possible by comparison with results from other approaches in the same benchmark. The reactive behaviour performance exhibited by the system was comparable with other approaches, having presented a better performance than other system for one static scenario and a similar performance for the other. The system successfully responded to all dynamic perturbations simulated. Experimental tests were performed to start disclosing the way in terms of hardware integration and agent implementation in a real production system.
Keywords: Agent-based systems, Manufacturing control, Multi-Agent Systems, Agent communication, Flexible manufacturing system.

1. Introduction

Over the times, the industry has been evolving to meet the progressive improvement of life quality that the human society desires. Aiming to provide products with increasingly high-quality and customization, industry has already faced three revolutionary stages known as industrial revolutions. With the advent of emerging technologies like Internet of Things, wireless sensor networks, Big Data, cloud computing, mobile Internet and Artificial Intelligence, and their introduction into the manufacturing world, a new paradigm arrived to the industry world: the fourth industrial revolution [7].

With the aim of achieving a smart, flexible and reconfigurable factory, capable of producing customized and small-lot products efficiently and profitably, some advanced manufacturing control systems have been proposed, taking advantage of the new technologies to design the smart factory of the future [2].

A very representative case of the advanced manufacturing control systems is the agent-based manufacturing control, which is based on Multi-Agent Systems (MAS) technology. These systems consist in an ecosystem of manufacturing resources defined as intelligent, autonomous and cooperative computational entities, known as agents, that can negotiate with each other to implement dynamical reconfiguration and decision-making, in order to achieve their individual goals. In an agent-based manufacturing control system, all the agents are in the same hierarchy level, being organized in a autonomous, distributed and decentralized architecture [2].

Throughout the last decades, several different approaches, architectures and platforms regarding MAS have been introduced and a considerable amount of industrial applications were already implemented and described in the literature. The main fields of application have been smart production, smart electric grids, smart logistics and smart healthcare [3], although some authors also have a prospect of other fields that might benefit from the application of agent technologies, namely traffic control, buildings and home automation, military and network security [4].

The aim of this work is to study the applicability of agent-based manufacturing control as a production control system designed to achieve the smart, flexible and reconfigurable factory. The main focus is the successful design and simulation of a Multi-Agent System used to demonstrate a flexible manufacturing system to be implemented in the Industrial Automation Laboratory (IAL) at Instituto Superior Técnico.

2. Manufacturing System

With a view to implementing an agent-based manufacturing control system, the manufacturing system to be controlled must be properly defined.

If the goal is to control a flexible and reconfigurable production system, there are some required characteristics concerning its physical configuration that need to be met. The conveyor system needs to be flexible, providing more than one path to travel between the same two points, so that the system can provide material-handling flexibility and machine-sequence flexibility. Redundancy is also a key point in this kind of system, being necessary to provide machine flexibility and reconfiguration of the products machine sequence in case of machine breakdown.

2.1. Flexible Manufacturing System Benchmark

The AIP-PRIMECA Flexible Manufacturing System (FMS), located in the AIP-PRIMECA Center at the Universityof Valenciennes, was defined as a benchmark for this thesis, directly influencing the design and operation of the proposed manufacturing system. This flexible production cell, depicted in Figure 1, has been one of the most used for research purposes in the area of distributed agent-related control systems. Its conveyor system configuration allows a really flexible routing of jobs inside the production cell, and the existence of three robots, which provide some operations in common, creates the necessary redundancy for the production. Furthermore, a benchmark was defined from this production cell, aiming to support benchmarking on a physical and real-world system and stimulate benchmarking activities internationally [6].

The smallest elements present in the production cell are the five available components "Axis_comp", "I_comp", "L_comp", "r_comp" and "screw_comp", plus the "Plate" where they are placed. By combination of these components, it is possible to assemble 7 different letters: "B", "E", "L", "T", "A", "I" and "P". The final products proposed to the client are words formed with these jobs and they

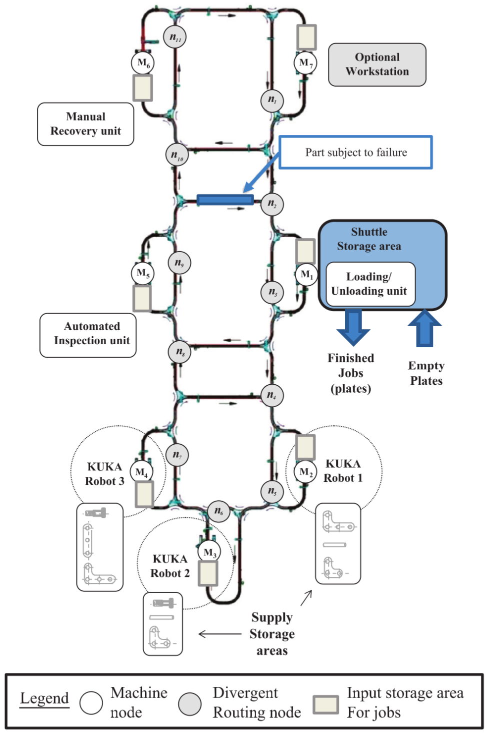

Figure 1: AIP-PRIMECA cell layout [6].

are three: "BELT", "AIP" and "LATE".

Each job has its own production sequence, i.e., an ordered list of elementary manufacturing operations. In this assembly cell, there are eight manufacturing operations: "Plate loading", "Axis mounting", "r_comp mounting", "I_comp mounting", "L_comp mounting", "Screw_comp mounting", "Inspection" and "Plate unloading".

The cell is composed of seven machines, two of which being optional and not used in this work. The machines are represented in Figure 1 with the symbols M_1 to M_7, being:

- M_1: loading/unloading unit.
- M_2, M_3 and M_4: assembly workstations.
- M_5: automatic inspection unit.
- M_6: manual recovery unit (not used).
- M_7: extra assembly workstation (not used).

Table 1 shows the different operations executed by each machine, together with the corresponding manufacturing processing time of each operation.

The conveyor system is composed of a main loop, four transversal sections composing multiple inner loops, several derivations to reach the machines and positioning units in front of machines. The transversal sections are responsible for the material-handling flexibility.

Table 1: Manufacturing operations processing times.

	M_1	M_2	M_3	M_4	M_5
Plate loading	10				
Plate unloading	10				
Axis		20	20		
r_comp		20	20		
I_comp				20	
L_comp		20		20	
Screw_comp			20	20	
Inspection					5

2.2. Proposed IAL Manufacturing System

The proposed flexible production system was designed with the goal of exploiting some existing resources of the IAL at Instituto Superior Técnico.

The laboratory contains 8 stands with several industrial automation equipments in each one of them. Considering that these stands are arranged in groups of 4, the goal was to propose a flexible system with a group of 4 stands as workstations working simultaneously and executing some redundant operations between them. In order to make the system feasible, a manipulator robot would be added to the centre of the four stands, to move the jobs between them, and a conveyor belt would be placed between the two stands on the right, to transport the jobs from the outside to the robot and vice-versa.

Since the benchmark presented in the previous subsection already comprises the required flexibility and complexity for the design of an agent-based manufacturing control system, the IAL Manufacturing System was designed analogously to the AIP-PRIMECA FMS, so that a direct comparison of performance can be carried out. This analogy is explained in Table 2.

Table 2: Analogy between the AIP-PRIMECA FMS and the IAL Manufacturing System.

AIP-PRIMECA FMS	**IAL Manufacturing System**
M_1 - Loading/Unloading Unit	Conveyor belt
M_2 - Assembly robot 1	Workstation B2
M_3 - Assembly robot 2	Workstation B3
M_4 - Assembly robot 3	Workstation B4
M_5 - Automated inspection unit	Workstation B5
Three mounting operations of each robot	Three pneumatic cylinders of each workstation
Inspection	Three cylinders of workstation B1 at the same time
Free workstation	Green light
Operation being executed	Red light
Job input storage area occupied	Yellow light
Flexible Conveyor System	Four conveyor belts + four elevators of the workstations

The flexible conveyor system was modelled using the conveyor belts and the elevators of the workstations as follows:

1. By analysis of the conveyor system layout in Figure 1, it can be stated that there are 4 different paths for a job to go from one machine to another, knowing that while the job is waiting for a free spot it will go around the outer loop.

2. These paths are defined in the following way for the example of the trip between M_2 and M_3: if the job can go directly to machine M_3, it follows the shortest way just passing near node n_5; if it only receives the information that there is a free place in the machine after passing it, it will turn around in the nearest transversal section, which can happen after node n_7, defining the second path, after node n_9, defining the third path, or after any of the following nodes, going for a complete turn around the outer loop and in this way defining the fourth path.

3. These four paths can be defined for all the necessary trips between machines in the same way, with the exception of the trip between machine M_3 and M_2, where the shortest way involves the use of the first transversal section.

4. Having defined these possible paths, the time spent going around the outer loop while waiting for a free place is modelled in the IAL Manufacturing System as a trip in a closed-loop conveyor system composed of the four conveyor belts of the workstations.

5. When the job already knows to which machine is going, the path it uses is modelled in the IAL Manufacturing System by the robot placing the job in one of floors of the elevator of the destination workstation, being each one of the four possible paths represented by a different elevator floor.

The final IAL Manufacturing System layout is depicted in Figure 2. Here, the robot stands in the centre of the four workstations B2, B3, B4 and B5. In each workstation, the conveyor belt and the elevator are positioned closer to the robot, since they are used to model the operation of the flexible conveyor system from the benchmark rather than the operation of the workstations itself. Apart from these two automation elements, the workstations have three pneumatic cylinders each, with the indication of the operation they perform, and three lights to exhibit their internal state. The conveyor belt placed between workstations B2 and B3 also contains the same three lights so that its state can be exhibited similarly to the workstations.

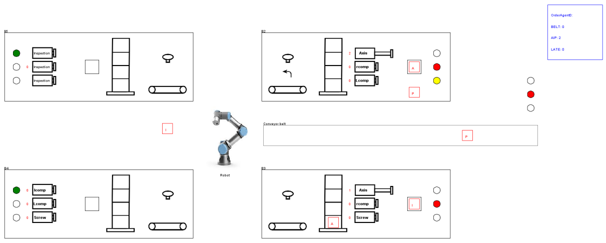

Figure 2: IAL Manufacturing System model.

3. Multi-Agent System Architecture

For the control level of the manufacturing system introduced in the previous chapter, a MAS was designed. The application of this system involves a set of distributed, autonomous and cooperative agents representing each one of the elements of the manufacturing system.

The design of the MAS was performed according to the Foundation for Intelligent Physical Agents (FIPA) Specifications, in order to take advantage of the standardized communication protocols, message transport and agent management. One of these protocols is the FIPA-Contract-Net (Figure 3), which is specially preponderant for its wide applicability in agent negotiation. Along with the FIPA-Propose and the FIPA-Request Protocols, these three Interaction Protocols (IP) will have a relevant role in the agent communication of the proposed agent-based system [1].

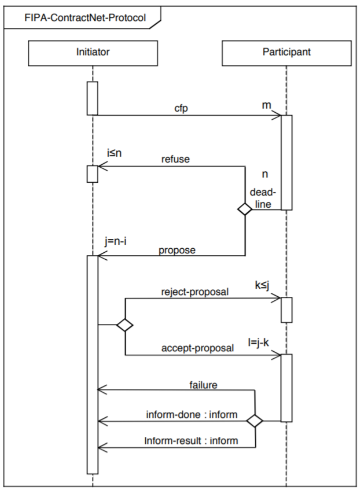

Figure 3: FIPA Contract Net Interaction Protocol [1].

The Contract Net Protocol (CNP) is an IP used by an Initiator that wants to explore the best proposal to make a contract. It starts with a Call-For-Proposals (CFP) message from the Initiator to the participants, who answer with a Refuse or Propose messages and have their proposals being accepted or rejected. The Propose Protocol is a very simple IP that only includes a Propose message from one Initiator to one participant that accepts or rejects the proposal. The Request Protocol is an IP used by an Initiator to get the participant to perform a specific action by sending a Request message specifying the action and receiving an Agree or Refuse message as answer.

3.1. Agents

The agent-based model presented in this work was built according to the physical mapping method, by which different agents are used to represent different real physical entities. Taking this into consideration, the designed MAS is composed of five different types of reactive agents: (1) Order Agent; (2) Job Agent; (3) Workstation Agent; (4) Robot Agent; (5) Conveyor Agent. The Order Agents represent the orders submitted by the clients. The Job Agents represent the jobs that are loaded into the production system, necessary to complete the orders. The Workstation Agents represent the four workstations B2, B3, B4 and B5. Lastly, the Robot Agent and the Conveyor Agent represent the central robot and the conveyor belt, respectively.

The behaviour of each type of agent was modelled by using the Petri nets formalism, which is a tool fit to model and to analyse the behaviour of complex event-driven systems. The Petri net behavioural model of the Job Agent can be seen in Figure 4.

When the Job Agent is created, its job is being loaded in the conveyor belt. When it finishes the loading, the agent starts the CNP with the workstations and analyses the proposals, in order to find the workstation that will execute the next operation in its production sequence. If no proposal is received, the job will travel in the closed-loop conveyor system and the Job Agent will periodically initiate the CNP until a Propose message is received. The Propose messages contain the state of the workstation, which can be "Free" or "CanWait", and the Job Agent prioritizes the proposals of "Free" workstations. As soon as the Job Agent finishes the negotiation with the workstations, it starts the Request protocol with the Robot Agent, requesting to be moved from its current location to the right floor of the elevator of the destination workstation. When it arrives at the workstation, The Job Agent initiates the Request Protocol with the workstation to start the operation and repeats it as soon as it ends, to start the next one. In the case of positive answers, the job continues in that workstation completing the following production steps, in case of a negative answer, the Job Agent goes back to starting the CNP with the other workstations. After completing all the operations except the unloading, the Job Agent requests the robot to move the job from the current workstation to the conveyor belt. When

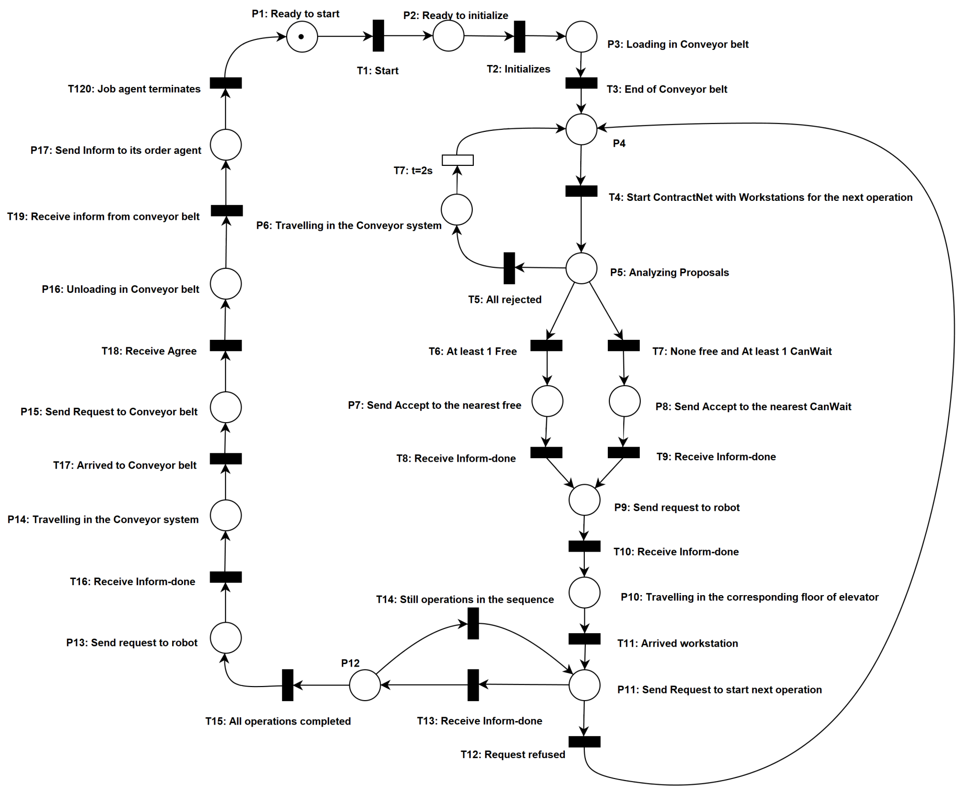

Figure 4: Behavioural model of the Job Agent.

it arrives, asks the Conveyor Agent to exit the production system, and, when it finishes the unloading, sends the Inform-done message to its corresponding Order Agent and terminates.

3.2. Agent Communication

The use of standardized FIPA IPs provides substantial help in the design and implementation of MAS, sparing extra modelling efforts. Furthermore, some agent-related platforms and simulators have been developed in accordance with the FIPA specifications, exhibiting tools to implement FIPA protocols in a faster and more efficient way.

The Propose Protocol is used as a means of negotiation between Order Agents, so that they can internally define the order in which they will be produced. They send Propose messages to other Order Agents, with its Due date as content, and if they only receive Accept-Proposals, they proceed to sending Request messages to the Conveyor Agent. The other Order Agents repeat the protocol until they all sent the Request messages for the jobs they need to load in the system. The interaction diagram for this negotiation is presented in Figure 5.

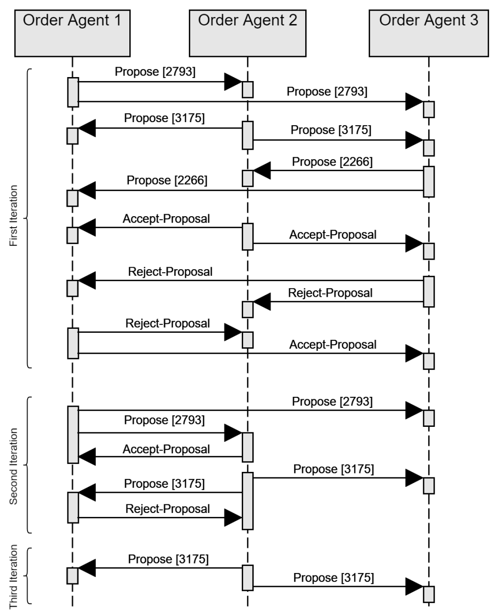

Figure 5: Interaction diagram for the Order Agents negotiation.

The use of the CNP by the Job Agents has a central role in the agent communications. This is due to the fact that this protocol allows the workstation allocation for each operation the jobs have in their production sequence. A typical example of the interactions happening in the case where job "B" needs to start its production by finding a workstation that can execute operation "Axis mounting" is shown in Figure 6.

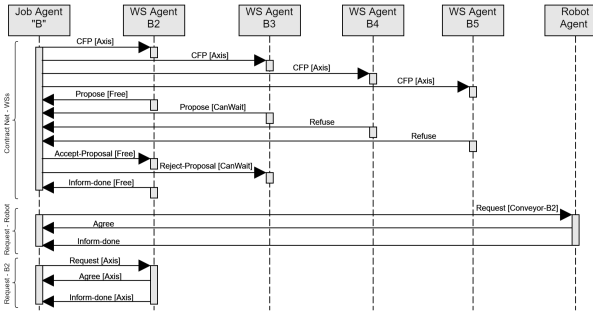

Figure 6: Interaction diagram for the example of operation "Axis mouting".

3.3. Decision Rules

Two different decision rule bases were defined in this work.

The first relates with the fact that the orders load their jobs by Due Date order and the jobs choose the proposal from the first workstation that proposes in the Contract Net, which coincides with the nearest. For instance, workstation B2 is chosen over B3 and B4, and workstation B3 is chosen over B4.

The second rule base was defined to have jobs being loaded by the order in which their orders are submitted and to balance workstation utilization by having jobs choosing for the least used workstations whenever they have a choice, which translates to choosing workstation B4 over workstation B2 or B3 whenever they present the same conditions to execute an operation.

4. Simulation

In other to conduct the simulation study, a suitable agent simulation software must be selected. The decisive aspects for the selection are the communication and compatibility, characteristics of utmost importance in the implementation of agents. Considering that the agent communication in the designed MAS employs FIPA IPs with standardized messages, the selected simulation software is the GAMA Platform, due to its FIPA compatibility and ability to represent FIPA Agent Communication Language (ACL) messages in GAML, the agent-based language used in GAMA.

In the context of the GAMA environment, a simulation of the IAL Manufacturing system was created, where all the simulation tests under the conditions of static and dynamic scenarios took place. Thus, the five types of agents were defined in GAML as five different species, specifying the actions and behaviours of each one in accordance with the Petri nets behavioural models.

Regarding the operation of the IAL Manufacturing System model, shown in Figure 2, the jobs appear and are loaded in the conveyor belt, from right to left. Then, they enter in the closed-loop conveyor system composed by the four conveyor belts of the workstations. After successfully completing the CNP, the jobs move to one of the elevator floors of the destination workstation, where they stay until the corresponding transportation time for that path has passed. If the workstation is free, the job proceeds to the operation area, if an operation is already being executed to another job, the job assumes its place in the job input storage area. In between workstations the jobs can return to the closed-loop conveyor system and, after concluding the operations in the production sequence, they are unloaded in the conveyor belt, from left to right.

From the static and dynamic scenarios defined in [6], two static scenarios and four dynamic scenarios were selected to be simulated in this work.

In Table 3, the number of shuttles allowed in the production system and the specification of the orders corresponding to each static scenario are provided. Table 4 presents the description of the four dynamic scenarios. Static scenario C0 will be the main reference for the operation of the manufacturing system. This happens since the dynamic scenarios will be simulated under C0 conditions and the differences of performance will be measured in comparison to C0.

Table 3: Static Scenarios data.

Scenario	No. of shuttles	Order #	BELT	AIP	LATE	Due date
C0	4	#1	1	-	-	382
		#2	-	1	-	238
B0	10	#1	-	2	-	327

Table 4: Dynamic Scenarios data.

Scenario	Description	Parameters
#PS5	At a given time, a rush order appears.	Type of order: AIP. Arrival Time: just after the end of the production of the fourth job in the cell. Due date: ASAP.
#PS7	At a given time, a part of the conveyor system is due for maintenance in a given time window.	Start time: just after the fourth job is unloaded. The conveyor must no longer accept shuttles, and as soon as it is empty, the maintenance starts. Duration (seconds): 25 × Total number of jobs.
#PS9	At a given time, one of the redundant machines will go down in a given time window.	Machine: M2. Start time: just after the departure of the first shuttle from M2. Duration (seconds): 25 × Total number of jobs.
#PS10	At a given time, one of the critical machines will go down in a given time window.	Machine: M4. Start time: just after the departure of the second shuttle from M4. Duration (seconds): 25 × Total number of jobs.

4.1. Static Scenarios

The Gantt charts portraying the operation of the system for static scenario C0 with rule base 1 and rule base 2 are presented in Figures 7 and 8.

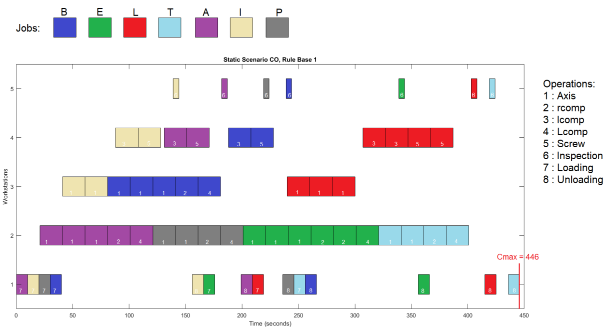

Figure 7: Gantt chart for Static Scenario C0 with rule base 1. ($C_{max} = 446s$)

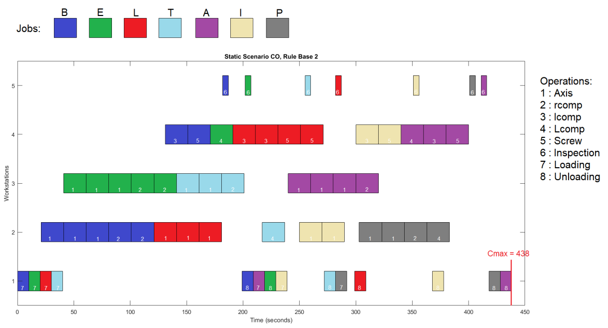

Figure 8: Gantt chart for Static Scenario C0 with rule base 2. ($C_{max} = 438s$)

In the second chart, a more balanced workstation utilization is evident, with workstation B2 performing 15 operations against 12 of the other two workstations instead of the 19 operations against 10 displayed in the first chart. The overall performance of rule base 2 is better than rule base 1 ($C_{max} = 438s < C_{max} = 446s$), and no obvious sign of bad allocation in this configuration is visible.

To assess the performance of the proposed agent-based system, some results introduced in [6], regarding simulation and real experiments using the potential fields approach in the AIP-PRIMECA FMS, were used for comparison. The potential fields approach [8], like the CNP, is also a reactive approach used in heterarchical control architectures.

The first of these results comprises the performance of this approach in the simulation of static scenario C0 and is presented in the Gantt chart of Figure 9.

Analysing the chart and comparing with scenario C0 with rule base 2, it is observable that both systems executed the exact same workstation allocation. Furthermore, the makespan of the benchmark is higher ($C_{max} = 448s$), which would mean an improvement percentage of approximately 2% of the

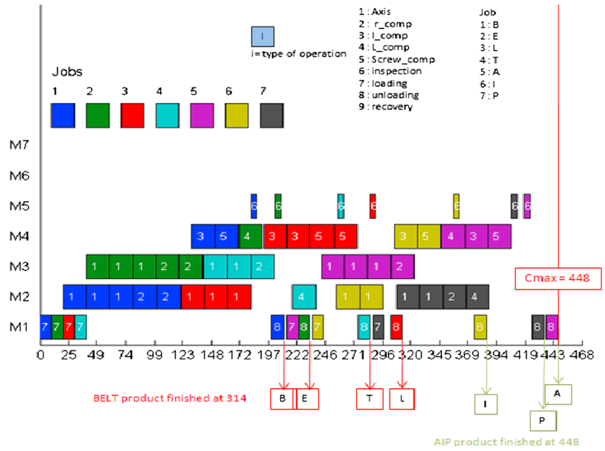

Figure 9: Gantt chart for Static Scenario C0 from benchmark [6]. ($C_{max} = 448s$)

proposed control system over the potential fields approach.

However, this difference in the makespan is caused by the not documented small time intervals between the unloading of a job and the loading of a new one, visible after the unloading of job "B" (in blue), job "E" (in purple) and job "T" (in cyan). For this reason, the performance of the proposed control system is considered validated, once it achieves a workstation allocation similar to the one of a demonstrated distributed control system, but the improvement percentage over the given system is not to be considered.

In addition to scenario C0, the proposed manufacturing control system was also tested under the conditions of static scenario B0 and the Gantt chart can be observed in Figure 10. This happened for the main purpose of establishing a comparison not only with experiments using the potential fields approach once more, but also with the performance of a reactive and optimized hybrid manufacturing control architecture called ORCA (dynamic Architecture for an Optimized and Reactive Control), introduced in [5], from which the reactive part is also based in the potential fields approach and the simulation and experimental studies were also executed in the AIP-PRIMECA FMS.

The system has a good performance, with the maximum makespan of $C_{max} = 326s$. In terms of workstation allocation, the rule base of the proposed MAS control approach is followed and the only drawback that can be observed is the excess of utilization of workstation B2 compared to B3.

Table 5 establishes the comparison of the proposed control system with the above mentioned control approaches, using results introduced in [5]. The table presents results for the potential fields approach and for the two hybridization levels of

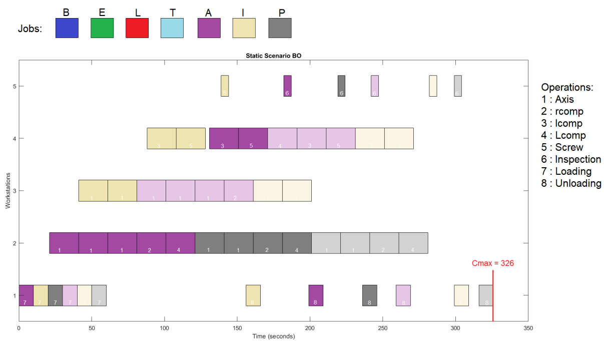

Figure 10: Gantt chart for Static Scenario B0. ($C_{max} = 326s$)

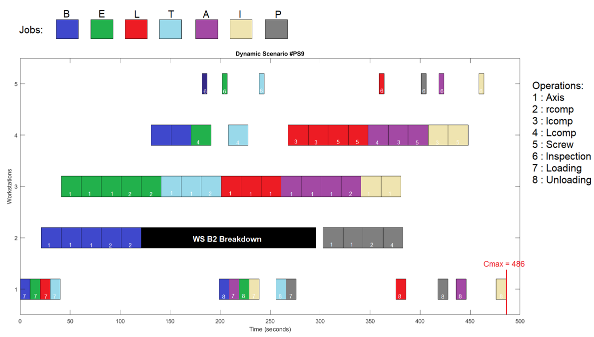

Figure 11: Gantt chart for Dynamic Scenario #PS9. ($C_{max} = 486s$)

ORCA: (1) with the ILP (Integer Linear Programming) solver, only providing the job order (ORCA-FMS 1) and (2) providing also the machine sequence to be followed (ORCA-FMS 2).

Table 5: Makespan for several control approaches under B0 conditions.

Production Scenario	C_{max} (seconds)			
	Potential Fields	ORCA-FMS 1	ORCA-FMS 2	Proposed MAS
B0 (2×AIP)	345	323	314	326

By analysing the given results, the proposed MAS control system achieved a better performance in this scenario than the potential fields approach ($C_{max} = 326s < C_{max} = 345s$), with a reduction of makespan in the order of 5.5%. However, as expected, both hybridization levels of the ORCA architecture demonstrated better performances, due to the presence of the ILP optimization.

4.2. Dynamic Scenarios

One of the dynamic scenarios tested is #PS9, which simulates a very usual situation of a breakdown in one workstation. In this case, the breakdown happens in workstation B2, one of the redundant workstations. With the workstation down, the Workstation Agent refuses all CFP and Requests messages and the jobs must wait in the closed-loop conveyor system for a place in the other available workstations. The performance of the control system under this breakdown situation is presented in Figure 11.

Similarly to what was introduced for the simulation of reference scenario C0, a Gantt chart with the simulation performance of the potential fields approach under the conditions of scenario #PS9 in the AIP-PRIMECA is also available in [6].

The performance is portrayed in Figure 12, from where it can be concluded that the proposed control system had a similar behaviour to the potential fields control approach. This fact is important to, once more, validate the operation of the designed agent-based manufacturing control system.

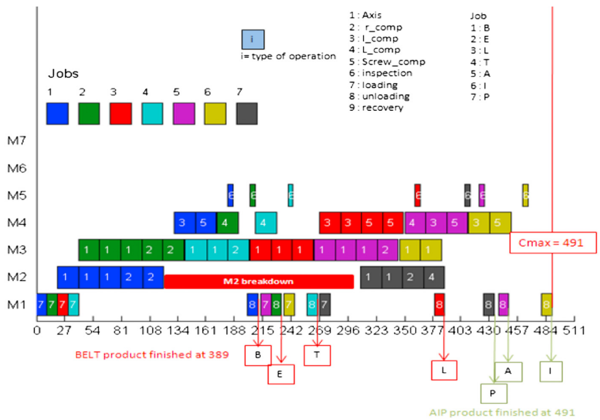

Figure 12: Gantt chart for Dynamic Scenario #PS9 from benchmark [6]. ($C_{max} = 491s$)

5. Experimental Implementation

With a view to implementing the agent-based system previously introduced and simulated, some experiences were conducted so that the conditions under which the system would be implemented could be understood.

In order to address the integration of agents and hardware, a simulation with hardware-in-the-loop was conducted. Next, an illustrative experience on how the agents would be implemented in the experimental facility using a suitable agent framework was performed. These experiments were performed in computers from the IAL at Instituto Superior Técnico, which are Intel Core i7-4790 CPU @ 3.60GHz with 8,00 GB of RAM and were running Windows 10.

5.1. Connection to PLC

The integration of agents and hardware addressed in this work was the connection between one of the simulated agents and one Programmable Logic Controller (PLC) of the IAL. The goal was to have one of the simulated Workstation Agents communicating in real time with the PLC from one of the automation stands.

In order to perform this experience, a group of PLC variables designated "Flags" was created to define the communication channels between the agent and the PLC. Furthermore, the PLC was programmed in a way that the activation of each variable would originate the execution of the required task autonomously.

An intermediate software was used and an alternative communication method was defined between this software and the simulation (Figure 13). The selected intermediary was a MATLAB Script. The alternative communication method consists in having GAMA and MATLAB reading and writing from the same data file, which is a .txt file containing as content the state of the PLC variables. The scheme of this interface is displayed in Figure 13.

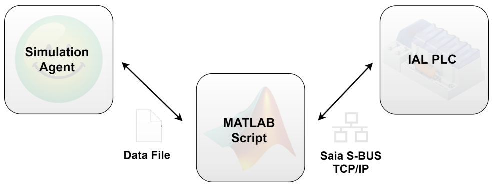

Figure 13: Interface between a simulation agent and a PLC.

During the experiment, the computer showed not having enough processing power to run the simulation in real world time, and so the simulated time was running slower than the PLC time. In order to evaluate if, in case the simulation was running at the same real world time of the PLC, the performance of the system with the connection to the PLC would be the same as the reference simulation of scenario C0 with rule base 1, the timers in the PLC program were adjusted and the simulation was run again with a view to having the PLC working in simulation time. The result of this experiment is depicted in the Gantt chart of Figure 14.

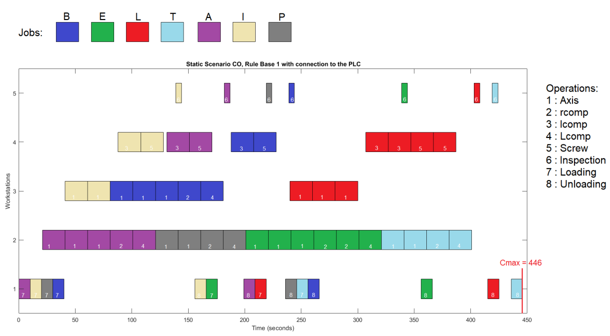

Figure 14: Gantt chart for Static Scenario C0 in connection to the PLC. ($C_{max} = 446s$)

As it can be seen, once the PLC operations were adjusted to the simulation time, the workstation allocation was similar to the one presented in the simulation of scenario C0 with rule base 1 (Figure 7). This shows that is possible to integrate hardware and achieve the same workstation allocation as in the pure simulation. Nevertheless, no conclusions can be drawn regarding the communications induced delays, since no in-depth study to assess those delays was conducted.

5.2. Contract Net Protocol Example

For the implementation of agents in a real industrial facility, an agent framework must be thoroughly selected according to predefined criteria.

JADE was chosen as the agent framework, since it was the one that better fitted the required criteria. It is an open source platform for agent based applications that aims to simplify the development of MAS by providing a set of system services and agents in compliance with the FIPA specification.

An illustrative experience on how the agents would be implemented in the experimental facility was performed, using JADE and two networked computers. The experience portrays the application of the CNP in the allocation of workstations for each operation contained in the production sequence of a job. Illustrative parts of both the Job Agent and the Workstation Agent were implemented in JADE.

To conduct the experience, JADE was launched in one of the computers, creating the Agent Platform, the Main Container and starting the Remote Agent Management GUI. Three Workstation Agents ("WS2", "WS3" and "WS4") were created in the "Main-Container". Then, using the other computer, a second container called "Container-1" was remotely launched in a way it would belong to the same Agent Platform and the Job Agent "B" was created there.

One section of the exchanged messages between the agents is presented in Figure 15, where it is possible to identify the pattern: before initiating the CNP with the workstations, marked by the CFP messages being sent, the Job Agent "B" sends a Request message to the Directory Facilitator (yellow pages service of JADE) to query the list of workstations that provide the next operation in the sequence.

More than the result itself, it is important to highlight the ability of JADE to allow easy and fast development of agent-based applications to work efficiently in real time and distributed across different machines.

6. Conclusions

In this work, a Multi-Agent System was designed to demonstrate a flexible production system and a simulation study was conducted to test the system

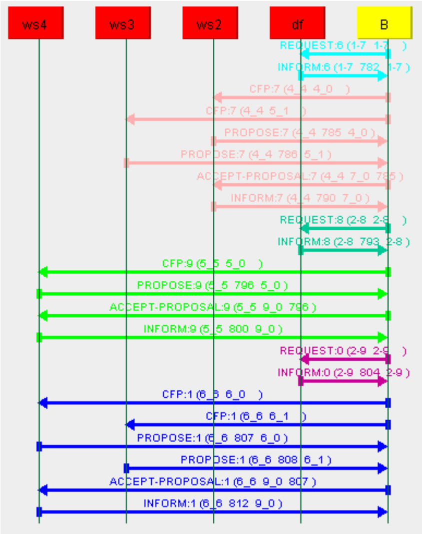

Figure 15: Exchanged messages with Job Agent "B".

through static and dynamic scenarios. Moreover, the preliminary steps into an experimental implementation were taken.

In order to validate the behaviour of the designed control system, a comparison was established with the performance of other reactive distributed control architectures. For static scenario C0, the performance of the system was compared with the Potential Fields approach under simulation in the benchmark FMS and both systems presented a similar behaviour, validating the performance of the proposed control system. For scenario B0, the proposed control system achieved a better global performance than the Potential Fields approach and, as expected, had a worse performance than the hybrid manufacturing control architecture ORCA. Nevertheless, it showed very promising results, considering that ORCA uses an optimization algorithm.

The system successfully responded to all dynamic scenarios. In #PS9, the performance of the proposed system was also like the expected and a comparison of behaviour with the Potential Fields approach through simulation or experimentation in the AIP-PRIMECA was possible, resulting in similar workstation allocation.

Regarding the experimental tests, a successful connection was established between one simulated agent and one PLC, disclosing the way in terms of hardware integration. However, no conclusion could be drawn concerning the real time communication induced delays. As to the JADE example, the way in which the agents would be implemented and would operate in real time was successfully illustrated.

The control system presented a reactive behaviour performance comparable with other distributed control architectures introduced in the literature. Yet, a lack of long term vision over the behaviour of the distributed MAS control architecture can be observed throughout the simulations in the local decision making of the agents, resulting in workstation allocations far from optimal.

An agent-based system is a promising solution for the control of flexible and reconfigurable systems that experience dynamic environments and require a very reactive behaviour. However, the optimal solution of an *a priori* optimization can always be considered as a way of reducing the lack of horizon and improving agent decision-making. Reactive and optimized hybrid control architectures are likely to be composing the control systems for the factories of the future.

References

[1] FIPA Web Site: http://www.fipa.org/.

[2] P. Leitão. Agent-based distributed manufacturing control: A state-of-the-art survey. *Engineering Applications of Artificial Intelligence*, 22(7):979–991, 2009.

[3] P. Leitão, S. Karnouskos, L. Ribeiro, J. Lee, T. Strasser, and A. W. Colombo. Smart Agents in Industrial Cyber-Physical Systems. In *Proceedings of the IEEE*, volume 104, pages 1086–1101, 2016.

[4] P. Leitão, V. Marik, and P. Vrba. Past, Present, and Future of Industrial Agent Applications. *IEEE Transactions on Industrial Informatics*, 9(4):2360–2372, nov 2013.

[5] C. Pach, T. Berger, T. Bonte, and D. Trentesaux. ORCA-FMS: A dynamic architecture for the optimized and reactive control of flexible manufacturing scheduling. *Computers in Industry*, 65(4):706–720, 2014.

[6] D. Trentesaux, C. Pach, A. Bekrar, Y. Sallez, T. Berger, T. Bonte, P. Leitão, and J. Barbosa. Benchmarking flexible job-shop scheduling and control systems. *Control Engineering Practice*, 21(9):1204–1225, 2013.

[7] S. Wang, J. Wan, D. Li, and C. Zhang. Implementing Smart Factory of Industrie 4.0: An Outlook. *International Journal of Distributed Sensor Networks*, 2016.

[8] N. Zbib, C. Pach, Y. Sallez, and D. Trentesaux. Heterarchical production control in manufacturing systems using the potential fields concept. *Journal of Intelligent Manufacturing*, 23(5):1649–1670, 2012.