Enhancing FMS Performance through Multi-Agent Systems in the Context of Industry 4.0

Gastón LEFRANC[1]*, Ismael LOPEZ-JUAREZ[2], Gabriel GATICA[3]

[1] Pontificia Universidad Católica de Valparaíso, 2950 Brasil Valparaíso, 2430000, Chile
gaston.lefranc@pucv.cl *(*Corresponding author*)

[2] CINVESTAV, 1062 Industria Metalúrgica, Parque Industrial Saltillo-Ramos Arizpe,
Ramos Arizpe Coahuila, 25900, México
ismael.lopez@cinvestav.edu.mx

[3] ARTIFICYAN, Industria 4.0, Inteligencia Artificial, Viña del Mar, 2520000, Chile
gabriel.gatica@artificyan.cl.

[3] Universidad Tecnológica Metropolitana, 161 Dieciocho, Santiago, Región Metropolitana, 8330383, Chile
ggatica@utem.cl

Abstract: Flexible Manufacturing Systems (FMSs) offer greater flexibility, efficiency, and automation in manufacturing. However, it remains a challenge to improve, manage and optimize production. This article explores the use of Multi-Agent Systems (MASs) as a tool to address these challenges. A review of recent advances in the application of MAS to FMS and Industry 4.0 is made, including dynamic assignment of tasks to robots based on real-time conditions; coordination of multiple robots, allowing collaboration between them to improve performance; and adaptability to changing environments, allowing the system to adjust to dynamic production demands and unforeseen situations. A case study that demonstrates the application of MAS for cooperative decision-making in a three-robot FMS system is presented. It is appropriate for improving efficiency, flexibility, coordination, and overall decision-making capabilities within a FMS environment. With this, MAS enables production optimization, leading to smarter, more adaptable, and more efficient manufacturing processes.

Keywords: Flexible Manufacturing Systems (FMSs), Industry 4.0, Multi-Agent Systems (MASs), Task Allocation, Multi-Robot Coordination.

1. Introduction

The manufacturing industry faces the challenge of producing a wide variety of products with high efficiency and flexibility. Flexible Manufacturing Systems (FMSs) use robots and other automated equipment to perform production tasks flexibly and adaptively. However, the complexity of FMSs and the need to coordinate multiple resources can make it difficult to optimize production flow.

Multi-Agent Systems (MASs) are computer systems consisting of multiple autonomous agents that interact and cooperate to achieve their objectives. In this way, MAS delivers by distributing tasks, sharing knowledge and coordinating actions. The key characteristics are represented by the fact that agents are autonomous with their objectives and behaviors, allowing for communication and coordination among themselves (Wooldridge, 2009).

MASs are decentralized systems without global control and their behavior arises from the interactions of agents. MASs are applied to various fields such as robotics, manufacturing, logistics and more. In manufacturing, MASs allow decentralized control, dynamic programming and coordination between machines/robots (Leitão, 2004). Holonic manufacturing systems use MAS architectures to achieve flexibility and robustness (Babiceanu & Chen, 2006). Leitão (2009) studies the use of MAS for intelligent manufacturing control to manage dynamic environments through distributed decision-making and presents another holonic MAS architecture for adaptability. MAS has demonstrated benefits in areas such as flexible workshop scheduling and the coordination of multiple robots in manufacturing cells by improving overall responsiveness, efficiency, and collaboration. MAS provides a distributed problem-solving approach suitable for complex and dynamic systems that require autonomy, interactions, and decentralized coordination (Barbosa et al., 2015).

In FMS, agents can represent robots, machines, or even workpieces. The MAS can coordinate agent actions and assign tasks efficiently to optimize system performance. FMS and Industry 4.0 have transformed modern manufacturing, introducing greater flexibility, efficiency and automation into production processes. So, MAS emerges as a powerful tool to optimize the performance of these systems.

https://doi.org/10.24846/v33i2y202401

In (Pulikottil et al., 2023), the use of manufacturing agent-based modelling and simulation is explored. A comprehensive review of task allocation methods in manufacturing is presented in (Chen et al., 2019), comparing systems-oriented approaches, using new technologies such as big data and deep learning. A framework for manufacturing system reconfiguration is proposed in (Mo et al., 2023). This framework uses artificial intelligence and digital twins to optimize production based on changing market demands. Another work presents methods to model and schedule tasks using deep learning techniques in an intelligent manufacturing system (Lan & Chen, 2021). In another case study, the same manufacturing process uses machine learning, seeing its strengths, weaknesses, and opportunities through expert evaluations (Lefranc, 2023). Another survey analyses various task assignment techniques used in MAS, for unmanned vehicles. Algorithms are classified based on their techniques and factors such as efficiency and communication are analyzed (Skaltsis et al., 2022). In this work, a consensus algorithm for Multi-Agent Manufacturing systems is proposed. This algorithm helps the system deal with interruptions and maintain efficient operation.

Research presents a data-based decision support system for resource allocation in smart manufacturing (Mezgebe et al., 2019). Another paper uses simulation and machine learning to optimize resource allocation based on historical production data (Mahmoodi et al., 2024). A novel Multi-Agent System for collaborative virtual manufacturing is proposed in the work of Zhang et al. (2019). This system integrates edge computing techniques to facilitate collaboration within a virtual manufacturing environment.

With a comprehensive review, agent-based manufacturing is discussed. The strengths, weaknesses and opportunities of this approach are explored, along with expert evaluations. Additionally, it examines the use of agent-based modelling and simulation in manufacturing (Pulikottil et al., 2023).

The integration of MASs in FMS and Industry 4.0 offers several benefits, such as improved efficiency and productivity, greater flexibility and adaptability, improved coordination and collaboration, and decentralized and robust decisions. MASs play a crucial role in Industry 4.0 by facilitating interconnectivity and communication, enabling decentralized decision-making, and promoting human-machine collaboration. Additionally, MASs enable real-time monitoring and data analysis for optimized operations, enhance supply chain management through improved tracking and coordination, foster innovation and continuous improvement through data-driven insights, improve safety and risk management through predictive maintenance and fault detection, and facilitate mass customization and personalization of products and services.

FMSs offer greater flexibility, efficiency, and automation than traditional manufacturing setups, but face issues such as dynamic task assignment, coordination of multiple robots/machines, and adaptability to changing production demands and environments. The objectives of this work are to leverage the capabilities of MAS as a solution to these challenges.

The contribution of this paper is to provide a review of recent advances in the application of MAS to FMS and Industry 4.0 contexts, covering areas such as dynamic task allocation, multi-robot coordination, collaboration and environment adaptability. Additionally, a case study is presented demonstrating the use of MAS for cooperative decision-making in a FMS setup with three robots, highlighting the potential benefits of increased efficiency, flexibility, coordination and decision-making enabled by MAS in FMS environments.

A FMS is a production system composed of machines, robots and CNC numerical control machines to perform production tasks flexibly and efficiently, with various transportation systems, storage and a central control system that provides advantages such as flexibility to adapt to changing demands, efficiency through optimized resources and reduction cycle times and better-quality compliance. In a FMS, agents can represent robots, machines, or workpieces, and the MAS system coordinates the agents' actions and efficiently assigns tasks to optimize performance.

To better align with the distributed intelligence paradigm of Industry 4.0, multi-agent system implementation should be expanded to give robotic agents more autonomy to negotiate tasks, resolve conflicts, and make decentralized decisions. This involves improving agents' decision-making algorithms and negotiation protocols, allowing them to independently evaluate options, prioritize tasks, and collaboratively reach optimal solutions based on their individual goals and the overall system objectives. The idea is to better leverage the potential of MAS to enable distributed intelligence across multiple entities, resulting in greater adaptability, flexibility and resilience in the manufacturing environment. MAS offers several advantages for FMS optimization. MAS can distribute decision making across multiple agents, improving the flexibility and robustness of the system, as well as adapting to changes in the environment in terms of scalability to accommodate different sizes and complexities in production systems.

This article presents the simulation of a FMS with robots and a MAS system to improve production flow and system performance. The impact of the MAS system on cycle time, robot utilization, and production rate is evaluated. The results demonstrate that the MAS system significantly improves the performance of the FMS, compared to a system without MAS.

The structure of the paper includes an introduction, in the present section, covering the concepts of FMS and MAS, namely a review of the recent applications and implementations of MAS in FMS/Industry 4.0, and of how MAS can enable manufacturing processes to be optimized. Section 2 presents the methodology using the Anylogic™ software, employed in this research. Section 3 describes the simulation of MAS and offers some considerations in this regard, while section 4 presents the proposed case study and describes the simulation process. Section 5 presents the obtained results, and, finally, the conclusions are given in section 6.

2. Methodology

A simulation of a FMS with robots and a MAS system is carried out to optimize the production flow. The simulation was performed using AnyLogic™ software. System components are modeled, including robots, worktables, workpieces, and the MAS system. Random data on order processing time, workpiece arrival rate, and robot availability are generated.

AnyLogic™ is a powerful simulation modelling tool that supports agent-based modelling, enabling the creation and analysis of multi-agent systems. It allows modelling of individual agents with unique behaviors, attributes, decision-making rules, and interactions and relationships between agents (Grigoryev et al., 2013).

In terms of Python integration, AnyLogic™ provides seamless connectivity to leverage external Python code within simulation models. Users can develop custom simulation algorithms, data analysis routines, or machine learning models in Python and integrate them into their AnyLogic™ models. This allows the user to extend AnyLogic's built-in functionality with custom logic written in Python, enabling more complex and specialized simulations (van Rossum, 2007).

The interaction between AnyLogic™ and Python is facilitated through the AnyLogic™ Analytics Cloud libraries, which enable two-way communication and data exchange between the two environments. Python code can be run from AnyLogic™ models, passing data and parameters, while simulation results can be accessed and processed in Python for further analysis or visualization. The integration leverages the strengths of both platforms, combining AnyLogic's easy-to-use multi-agent modelling environment with Python libraries and tools for data manipulation, analysis, and advanced algorithms.

Two scenarios were evaluated:

Scenario 1: FMS system without MAS

Scenario 2: FMS system with MAS

Typically, the FMS operates in a basic manner, without intelligent decision-making or coordination to optimize operations. It can rely on predefined rules or fixed schedules without dynamically adapting to changes.

For each scenario, simulations are carried out ten times and the following performance indicators are calculated:

- Cycle time: The average time it takes to complete a project from its arrival to its departure. "Project" refers to a work order to be completed by the FMS. Cycle time is the total duration required for the FMS to accomplish all tasks and operations;
- Robot utilization: the percentage of time that robots are busy performing tasks;
- Production rate: the number of pieces produced.

The FMS model in the simulation includes:

- Robots: They were modelled with specific capabilities to perform tasks such as cutting, inspection, and assembly;
- Worktables: They were modelled for temporary storage of parts and as workstations for robots;
- Parts: Different types of parts with specific processing requirements were defined.

3. MAS Simulation

The MAS system in the simulation was composed of four agents:

- Planning agent: it oversees the entire assembly process;
- Coordinating agent: it is responsible for assigning tasks to robots considering factors such as capacity, location of robots, and status of workpieces;
- Robot agents: they represent physical robots and execute the tasks assigned by the coordinating agent;
- Quality control agent: it monitors part quality throughout the process;
- Scheduling agent: it reassigns the process to an available robot. It prioritizes urgent tasks like the correction of defects to maintain final product quality.

A task assignment algorithm was implemented based on the capacity and location of the robots. The algorithm considers the processing time of each task, the availability of the robots and their current location to assign the tasks efficiently.

The coordinating agent used a task assignment algorithm based on the robots' capacity and location. The algorithm considered the processing time of each task, the availability of the robots and their current location to assign the tasks efficiently.

In general, Multi-Agent Systems are best suited for tasks that require collaboration and coordination between multiple entities. These intelligent agents are best suited for tasks that require autonomous decision making and individual intelligence.

The choice of one or the other will depend on the specific characteristics of the problem to be solved. It is important to highlight that both technologies are areas of active research, and their use is expected to increase in the future.

Additional considerations:

- Communication: Communication between agents is a fundamental aspect of the design and implementation of a Multi-Agent System. It is important to define an adequate communication protocol that allows agents to exchange information efficiently and reliably;
- Coordination: The coordination of the actions of the agents is another important aspect to guarantee the correct functioning of the Multi-Agent System. Different coordination mechanisms can be used, such as negotiation, auction, or centralized planning;
- Security: Security is an important factor to consider in the development of Multi-Agent Systems, especially in critical systems. Security measures need to be implemented to protect the system from attacks and intrusions.

4. Case Study

MASs are considered a core technology for Industry 4.0 due to their ability to enable decentralized, autonomous and adaptive systems. This paper presents a real-world case study that demonstrates the practical application of MAS in the management of a FMS.

The case study has two purposes: first, it shows how MAS principles can be implemented in a real manufacturing environment, to address challenges such as task allocation, coordination, and decision making. Second, it provides empirical evidence of the potential benefits of using MAS, such as increased efficiency, adaptability, and overall system performance.

By offering a concrete real-world implementation and its results, the paper strengthens the case for adopting MAS as a viable solution for smart and flexible manufacturing aligned with Industry 4.0 goals. The case study acts as a proof of concept and highlights the practical value of MAS in complex manufacturing contexts.

This case study explores the application of MAS for decision-making in a FMS with three robots. The study investigates the potential of MAS to optimize production flow, manage resource allocation and improve overall system performance in the context of Industry 4.0.

The FMS consists of three robots responsible for assembling three different parts (A, B and C). The assembly process consists of four stages (see the visual representation of the FMS from Figure 1 and the block diagram of its assembly process from Figure 2):

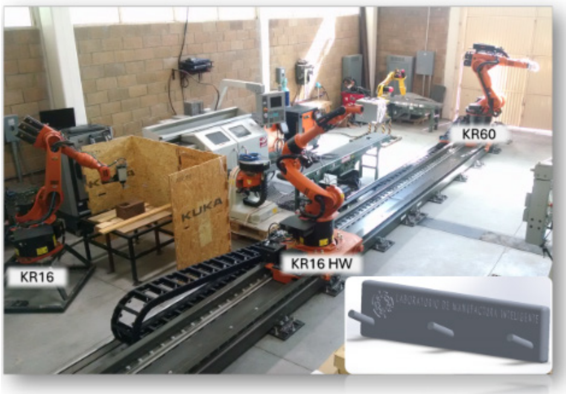

Figure 1. Flexible Manufacturing System (FMS)

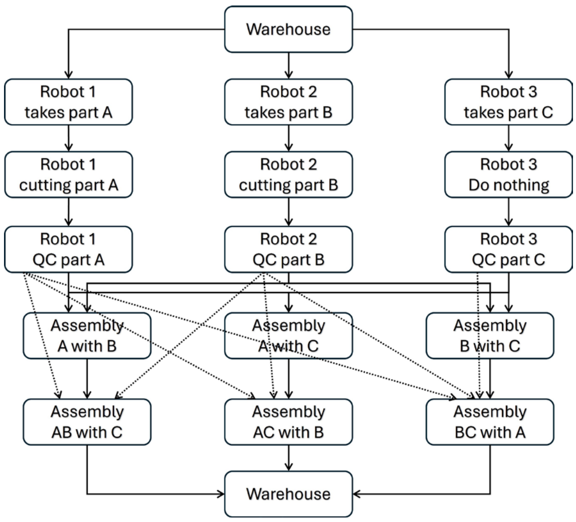

Figure 2. Assembly process block diagram

Stage 1. Taking and Cutting Parts: Each robot utilizes sensors and grippers to retrieve designated parts (A, B, or C) from the warehouse. Robot 1 performs cutting on part A, while Robot 2 cuts part B. Part C requires no cutting;

Stage 2. Quality Control: Robots 1, 2, and 3 utilize vision and measurement sensors for independent quality verification of their respective parts (A, B, and C);

Stage 3. Assembly: Parts A and B require collaborative assembly by two robots (combinations of Robots 1, 2, or 3);

Stage 4. Final Assembly: The remaining two robots collaborate to assemble part C with the assembly AB, forming the final product.

The MAS system makes the decision to choose which robots assemble parts A and B and parts C with AB, according to their availability at the time of assembly.

To facilitate decentralized decision-making within the FMS, a MAS architecture is implemented, comprising the following agents:

- Planning Agent: It oversees the entire assembly process, including task sequencing; robot assignment to tasks; system resource management; production quantity determination; and communication with the order management system for demand information;

- Robot Agent: Each robot possesses an individual agent for: action control and communication with other agents; executing tasks assigned by the planning agent; and sharing status updates (location, parts owned, task capabilities);

- Coordination Agent: It manages actions of the two robots collaborating during A/B and C assembly, ensuring synchronization and task completion;

- Quality Control Agent: It monitors part quality throughout the process, notifying the planning agent of any defects. This agent can also request part recutting, if necessary;

- Scheduling Agent: It reassigns the process to an available robot and prioritizes urgent tasks, like defect correction, to maintain final product quality.

Decentralized Decision-Making with MAS:

Agents interact to make crucial decisions regarding planning, execution, and coordination of the assembly process. Key decision areas include:

- Robot Selection: The planning agent considers location, task capabilities, and execution time to select two available robots for each assembly stage;

- Task Assignment: Tasks like cutting, quality control, and assembly are assigned to the chosen robots by the planning agent;

- Task Sequencing: Considering task dependencies, execution times, and robot synchronization needs, the planning agent determines the task execution order;

- Robot Coordination: The coordination agent ensures precise and efficient assembly by synchronizing the actions of collaborating robots;

- Exception Handling: Agents can identify and address potential issues such as missing parts, defect detection, robot failure, or part replacement needs;

- Task Reassignment: In case of robot failure or inability to complete a task, the scheduling agent reassigns it to an available robot;

- Task Prioritization: The scheduling agent prioritizes urgent tasks like defect correction to maintain final product quality;

- Adaptation to Changes: The MAS adapts to environmental changes (new orders, robot availability), by updating planning and allocation decisions.

This case study demonstrates the potential of MAS in optimizing decision-making within a FMS. By leveraging decentralized collaboration and real-time communication, MAS facilitates scheduling agents in processes.

4.1 Simulating a MAS for FMS

The flowchart from Figure 3 illustrates the simulation process of a MAS-based FMS, highlighting the interactions between various components and the overall flow of the simulation.

1. Initialization Phase:

   a. Start (A): The simulation starts with initializing system parameters;

   b. Initialization (B): Robots, parts, tools, and the environment are initialized, establishing the initial state of the system.

2. Task Planning, Assignment, and Execution:

   a. Task Planning (C): The MAS system analyzes the current state of the system and plans the tasks that need to be performed by the robots;

   b. Task Assignment (D): The MAS system assigns the planned tasks to the available robots, considering their capabilities and current positions;

   c. Task Execution (E): The assigned robots execute the tasks, manipulating parts, using tools, and interacting with the environment according to the task specifications.

3. State Monitoring and Data Collection:

   a. State Update (F): The system's state is updated after each task execution, reflecting the changes in robot positions, part status, and the overall assembly process progress;

   b. Data Collection (G): Simulation data is collected, including assembly time, waiting time, resource consumption, and other relevant metrics.

4. Quality Check and Simulation Termination:

   a. Quality Check (H): The quality of the assembled products is verified according to predefined standards, to ensure they meet the required specifications;

   b. End (I): The simulation terminates when either all products have been assembled or the maximum simulation time has been reached.

5. Environment Simulation (J): The behavior of the environment is simulated, including the arrival of new products, the availability of resources, and any dynamic changes that affect the system.

6. Agent Creation (K): Upon initialization, autonomous agents representing robots, parts, tools, or other system components are created.

The provided flowchart serves as a general representation of the simulation process. The actual flowchart may vary in complexity, depending on the specific characteristics of the FMS being simulated.

The MAS plays a crucial role in coordinating task planning, assignment, and execution, ensuring efficient and optimized production within the FMS.

The simulation process enables the evaluation of different MAS strategies and system configurations, to identify the most effective approach for optimizing FMS performance.

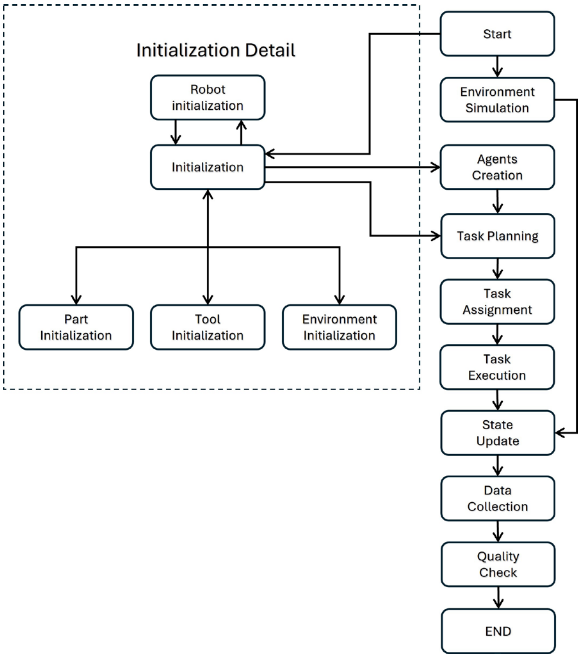

Figure 3. Flowchart of a MAS-based FMS simulation

4.2 Simulation Algorithms

Negotiation between robots allows for more decentralized, adaptive, fault-tolerant and scalable decision-making. However, it may not guarantee global optimality as robots make decisions based on local knowledge.

The choice depends on the specific FMS requirements. Centralized planning suits stable, predictable environments where global optimization is crucial. Negotiation is better for dynamic, complex environments needing adaptability, fault tolerance and scalability, even if not fully optimal.

Aligning the decision-making regime with FMS goals around optimality, responsiveness, scalability and fault tolerance is the key to achieving the desired performance levels.

The simulation is carried out in Python. This requires a set of algorithms that work together to simulate the robots' behavior, the assembly process, and the system environment. The main algorithms of system simulation in Python are:

1. Main simulation algorithm: This algorithm controls the overall flow of the simulation. It is responsible for: initializing the system (robots, parts, tools, etc.), running a simulation cycle that includes updating the system status (position of robots, status of parts, etc.), making decisions for robots (task assignment, movement, etc.), simulating the assembly process, by collecting simulation data (assembly time, waiting time, etc.), and finishing the simulation when the conditions are met;

2. Decision-making algorithm for robots: This algorithm determines what action each robot should perform in each simulation cycle. Actions may include moving to a new position, pick up a piece, leave a piece, assemble a piece, and wait for a part or tool to become available. The decision-making algorithm can be based on different strategies, such as: centralized planning, negotiation between robots, and others;

3. Assembly process simulation algorithm: This algorithm simulates the assembly process of a product. The algorithm considers: the assembly sequence of the different parts, the time needed to assemble each piece, the tools, and the resources necessary for each tool (cutting, inspection, assembly). The algorithm utilizes the following:

- Worktables: They represent the stations where the robots perform part-processing tasks;
- Workpieces: Represented as entities that flow through the system and require processing;
- MAS System: Represented as a coordinating agent in charge of assigning tasks to the robots and monitoring the state of the system.

The MAS uses a task assignment algorithm based on the robots' capacity and location.

The algorithm considers the following factors:

- Processing time which is the time required for a robot to complete a specific task on a workpiece;
- Robot availability which considers the status of the robots (occupied or unoccupied);
- Current location of the robot, which considers the position of each robot in the system.

The algorithm selects the most suitable robot for each task, considering these factors to minimize cycle time and maximize robot utilization.

The simulation was run for both scenarios: FMS system with and without MAS and the results obtained for the performance indicators were analyzed.

Ten simulation replicates were run for each scenario, with the same random parameters. Data on cycle time, robot utilization, and production rate were recorded for each replica. The results obtained were subsequently analyzed.

4.3 Communications Protocol

A simple communication protocol was defined for MAS agents to exchange information. The protocol included messages to: task requesting from the coordinating agent; reporting the completion of a task *m*; and reporting the status of the robot (occupied or unoccupied).

5. Results

The simulation researched the performance of a three-robot Flexible Manufacturing System (FMS) assembling ten parts with five tools. Assembly time per piece was 10 seconds, tool waiting time was 5 seconds, and new product arrival occurred every minute.

The system successfully assembled 100 products in 100 minutes, with an average assembly time of 60 seconds per product. Notably, the low average waiting time of 30 seconds indicates well-synchronized robots and minimal bottlenecks. Additionally, the utilization rates of 80% for robots and of 60% for tools, respectively, suggest efficient resource allocation (Table 1).

To evaluate the effectiveness of Multi-Agent Systems (MASs), ten simulation replicates were performed with and without MAS for a week-long period (equivalent to real production). Compared to the non-MAS scenario, the MAS system achieved a significantly shorter average cycle time, highlighting a more agile production flow.

This improvement is attributed to the optimized task assignment and coordination facilitated by the MAS, which minimizes robot idle time and maximizes resource utilization. Furthermore, robot usage increased by 9.1%, demonstrating better resource allocation. This, coupled with the 20.3% reduction in cycle time, led to a notable production rate increase of 18.2% (see Table 2).

The simulation results unequivocally demonstrate the effectiveness of MAS in optimizing FMS performance. By enabling efficient task coordination and resource allocation, MAS minimizes waiting times and maximizes resource utilization, leading to significant production improvements. These findings strongly support the implementation of MAS for enhanced FMS performance within an Industry 4.0 context.

6. Conclusion

This paper has presented the integration and the application of Multi-Agent Systems (MASs) into Flexible Manufacturing Systems (FMSs), within a simulated environment, focusing on its impact on production flow and system performance, in the context of Industry 4.0. The findings demonstrate the compelling potential of MAS for optimizing complex manufacturing processes.

Table 1. Simulation values

Performance Indicators Scenario	Without MAS (%)	With MAS (%)	Improvement (%)
Cycle time (seconds)	25.2	20.1	20.3 ↓
Use of robots (%)	78.3	85.4	9.1 ↑
Production rate (pieces/hour)	14.3	16.9	18.2 ↓

Table 2. Performance indicators

Measure	Value
Total simulation time	100 minutes
Number of assembled products	100
Average assembly time per product	60 seconds
Average waiting time per product	30 seconds
Robot utilization rate	80%
Tool utilization rate	60%

The simulation results revealed significant improvements across key performance indicators, when employing a MAS within the FMS. A 20.3% reduction in cycle time means greater efficiency of the production flow. The 9.1% increase in robot utilization indicates better resource management. Moreover, the production rate rose by 18.2%, highlighting the system's improved productivity. These quantitative findings provide strong evidence for the effectiveness of MAS in optimizing FMS performance.

The simulation results demonstrate that integrating a MAS system into a FMS can significantly improve system performance. The MAS system optimizes task allocation, which reduces cycle time, increases robot utilization, and, consequently, increases production rate. The results show that the simulation is sensitive to the data used and the system configuration. It is important to carry out adequate validation and calibration of the simulation model. Different task allocation algorithms and coordination strategies can be explored within the MAS system to analyze their impact on the performance of the FMS. Integrating the MAS system with production planning and inventory control systems can further improve manufacturing system optimization.

This study opens interesting avenues for future research efforts. Evaluating the performance of alternative task allocation algorithms, such as dynamic programming or genetic algorithms, within the MAS framework is a promising direction. Additionally, exploring more intricate scenarios with a higher number of robots, varied production demands, and unforeseen disruptions can provide valuable insights for real-world implementation.

Furthermore, integrating the MAS system with production planning and inventory control systems holds significant promise for further optimization within an Industry 4.0 context. This holistic approach could lead to the development of even smarter and more responsive manufacturing processes, fostering greater adaptability and efficiency in response to dynamic market demands.

In conclusion, the integration of MAS offers a transformative pathway towards achieving Industry 4.0 goals. This research not only demonstrates the effectiveness of MAS in optimizing FMS performance, but also paves the way for further advancements that can revolutionize modern manufacturing.

REFERENCES

Babiceanu, R. F. & Chen, F. F. (2006) Development and Applications of Holonic Manufacturing Systems: A Survey. *Journal of Intelligent Manufacturing*. 17, 111–131. doi: 10.1007/s10845-005-5516-y.

Barbosa, J., Leitão, P., Adam, E. & Trentesaux, D. (2015) Dynamic self-organization in holonic multi-agent manufacturing systems: The ADACOR evolution. *Computers in Industry*. 66, 99-111. doi: 10.1016/j.compind.2014.10.011.

Chen, Y., Sun, F., Zhang, Y. & Tao, F. (2019) Task allocation in manufacturing: A review. *Journal of Industrial Information Integration*. 15, 207-218. doi: 10.1016/j.jii.2018.08.001.

Grigoryev, I., Borshchev, A. & Alexei A. (2013) AnyLogic: A Multimethod Simulation Tool. In: *Proceedings of the 2013 IEEE Winter Simulation Conference, 08-11 December 2013, Washington, DC, USA*. New Jersey, USA Institute for Electrical and Electronics Engineers (IEEE). doi: 10.1109/WSC.2013.6721529

Lan, X. & Chen, H. (2021) Research on Modeling and Scheduling Methods of an Intelligent Manufacturing System Based on Deep Learning. *Wireless Communications and Mobile Computing*. 2021, 4586518. doi: 10.1155/2021/4586518.

Lefranc, G. (2023) Multirobot Allocation In A Flexible Manufacturing System, Using Reinforcement Learning For Decision-Making, Case of Study. *Procedia Computer Science*. 221(5), 41-48. doi: 10.1016/j.procs.2023.07.006.

Leitão, P. (2009) Agent-based distributed manufacturing control: A state-of-the-art survey. *Engineering Applications of Artificial Intelligence*. 22(7), 979-991. doi: 10.1016/j.engappai.2008.09.005.

Leitão, P. J. P. (2004) *An agile and adaptive holonic architecture for manufacturing control*. PhD thesis. Department of Electrotechnical Engineering Polytechnic Institute of Bragan, Portugal.

Mahmoodi, E., Fathi, M., Tavana, M., Ghobakhloo, M. & Ng, A. H. C. (2024) Data-driven simulation-based decision support system for resource allocation in industry 4.0 and smart manufacturing. *Journal of Manufacturing Systems*. 72, 287-307. doi: 10.1016/j.jmsy.2023.11.019.

Mezgebe, T. T., Demesure, G., Bril El Haouzi, H., Pannequin, R. & Thomas, A. (2019) CoMM: A consensus algorithm for a multi-agent-based manufacturing system to deal with perturbation. *International Journal of Advanced Manufacturing Technology*. 105(9), 3911-3926. doi: 10.1007/s00170-019-03820-0. hal-02126766.

Mo, F., Chaplin, J. C., Sanderson, D., Ur Rehman, H., Monetti, F. M., Maffei, A. & Ratchev, S. (2023) A Framework for Manufacturing System Reconfiguration Based on Artificial Intelligence and Digital Twin. In: Kim, K. Y., Monplaisir, L. & Rickli, J. (eds.) *Flexible Automation and Intelligent Manufacturing: The Human-Data-Technology Nexus*. *FAIM 2022*. Lecture Notes in Mechanical Engineering. Cham, Switzerland, Springer, pp. 361–373. doi: 10.1007/978-3-031-18326-3_35.

Pulikottil, T., Estrada-Jimenez, L. A., Ur Rehman, H., Mo, F., Nikghadam-Hojjati, S. & Barata, J. (2023) Agent-based manufacturing – review and expert evaluation. *The International Journal of Advanced Manufacturing Technology*. 127, 2151–2180. doi: 10.1007/s00170-023-11517-8.

Skaltsis, G. M., Shin, H.-S. & Tsourdos, A. (2022) A survey of task allocation techniques in MAS. In: *Proceedings of the 2021 International Conference on Unmanned Aircraft Systems (ICUAS), 15-18 June 2022, Athens, Greece*. New Jersey, USA Institute for Electrical and Electronics Engineers (IEEE). pp. 488-497. doi: 10.1109/ICUAS51884.2021.9476736.

van Rossum, G. (2007) Python programming language. In: Chase, J. & Seshan, S. (eds.) *ATC›07: 2007 USENIX Annual Technical Conference on Proceedings of the USENIX Annual Technical Conference*, 17-22 June 2007, Santa Clara, CA, USA. 41(1). Berkeley, CA, USA, United States, USENIX Association. pp. 1-36.

Wooldridge, M. (2009) *An Introduction to MultiAgent System*. 2nd ed. New Jersey, USA, John Wiley & Sons.

Zhang, X., Tang, S., Liu, X., Malekian, R. & Li, Z. (2019) A Novel Multi-Agent-Based Collaborative Virtual Manufacturing Environment Integrated with Edge Computing Technique. *Energies*. 12(14), 2815. doi: 10.3390/en12142815.