JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

An overview of MAS: Basics, Development tools, Applications and Challenges

[1]Prakhar Shukla, [2]Somesh Sharma

[1] MTech. Research Scholar, [2] Associate Professor
[1] Mechanical Engineering Department,
[1] National Institute of Technology, Hamirpur, India

Abstract : The objective of this exploratory study is to present an overview of multi agent technology development tools, as well as their applications and limitations in the manufacturing area. For this, a complete literature review on multi agent technology (MAT) was conducted which suggests that limited research was performed on this area thus in this direction our investigation is on development tools, applications, and issues related to multi agent technology. This piece of work is a modest effort to investigate this subject, in which we evaluated four important areas of MAT in manufacturing application as Production design and development, Production planning and control, Manufacturing Process Control, and Quality Control. According to the detailed analysis, if the multi agent system is implemented properly in any area, it can produce an autonomous system to improve efficiency and performance.

***IndexTerms* - Agent,communication protocol,multi agent system,manufacturing applications,programming languages.**

I. Introduction

Over the last few decades, the tremendous growth of the science and technology industry has had a range of repercussions on human society. As a result, consumer demand has evolved away from mass production and toward mass customization, compelling businesses to deliver highly customized, high-quality products in shorter time frames. As a result, they must deploy more flexible, robust, adaptable, and responsive systems in order to meet client demand and maintain a competitive market position [23,40]. A solution to this challenge is to employ multiagent systems (MAS) to decentralize control functions over a network of autonomous and cooperative entities to provide modularity, autonomy, flexibility, robustness, and adaptability [8]. The intrinsic capabilities of MAS systems to adjust quickly to changing conditions without external intervention set them apart from traditional solutions [8]. Multi-agent technology (MAT), in particular, has the potential to be critical in the implementation of the cyber-physical systems (CPSs) paradigm, which is a key component of the Industrie 4.0 plan. CPSs are a network of embedded systems that integrate sensing, actuation, computation, and physical processes and are made up of real-world physical processes and computing systems that form an extended network of embedded systems [25].

MAS or agent-based technology are studied in detailed in literature and a significant number of application-based articles are present in numerous domain and manufacturing is no exception. Although those articles do not discuss MAS in detail as they are mostly lacking in aspects, of insight they provide on MAS, challenges involved in building a MAS and tools that can be used. This article focuses on those who are new to the field and do not have much understanding of it. A comprehensive survey of existing MAS and related work in literature from various domains is performed and an overview of MAS, its application and challenges and tools that can be used are discussed. The rest of the paper is organized as follows. In section 2 Introduction to MAS, agent, environment and agent communication is given. In section 3 a discussion about agent-oriented programming is there. In section 4 applications of MAS in manufacturing discussed. In section 5 challenge to MAS is discussed and at last section 6 is conclusion.

II. Introduction to MAS

A multi-agent system (MAS) is a coalition of fully or semi-autonomous individual contributors that band together to achieve a synergy of their individual goals and the federations or the engaged set of agents' overall goals. Since a single agent's skills are limited, increasingly complicated real-world situations necessitate the combined and cooperative efforts of a group of agents to address the problem at hand. They rely on communication, collaboration, negotiation, and responsibility delegation to succeed, all of which are predicated on the involved agents' individual rationality and social intelligence.

2.1MAS

A multi-agent system is comprised of multiple autonomous (intelligent) agents, each pursuing its own set objectives while engaging in a shared environment, communicating and potentially coordinating their actions [40]. Each agent may have its objective but they work together like a well-structured organization to achieve a global objective. An effective MAS has a set of properties that very well characterize the system which are displayed in Fig.1

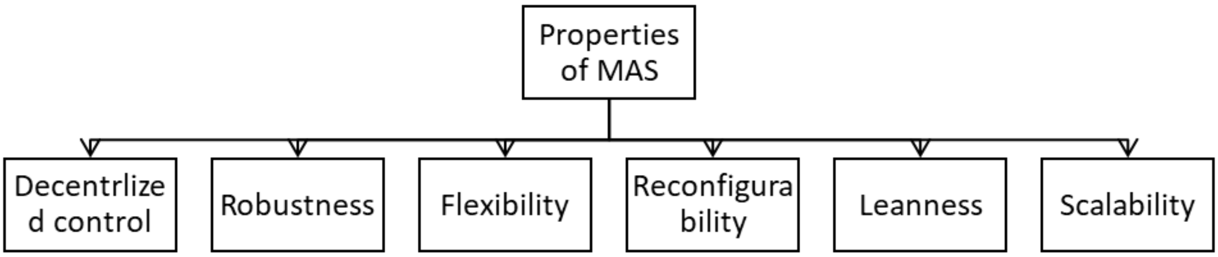

Fig 1: Properties of MAS

2.1.1Decentralized Control

Due to the decomposition of a task into well-defined sub-task control function is distributed over a network of autonomous and cooperative entities but we cannot accentuate on decentralization of control too much because of the fact the individual entities we are talking about themselves work in a well-structured way with centralized control.

2.1.2Robustness

Robustness is closely associated with flexibility as a MAS is capable of self-optimization, self-configuration it can adapt to unforeseen situations and recover itself from various kinds of hardware and software failure. It is closely associated with recovery after a failure has occurred.

2.1.3Flexibility

In MAS flexibility comes from the ability to adapt to sudden changes that happen during the execution phase which are not permanent or intentional. It is a kind of response of the system to an emerging situation that must be dealt with by agents as they agreed upon completion of global objective

2.1.4Reconfigurability

As the process evolves continuously and requirements change every day it is easy to integrate those changes in MAS by changing an agent or incorporating a new one. Unlike the case of flexibility, those changes here are intentional and permanent and do not occur during the execution phase are done to increase the quality and efficiency.

2.1.5Leanness

It is related to the division of tasks among agents such that the task allocated to the agent must be as lean as possible to avoid complexity and agent behavior should be understandable if there is a need to add more responsibility to an individual agent one must consider the further division of task.

2.1.6Scalability

MAS hardware is modular that is agents run on the different computer systems are connected through the internet if needed it is possible to incorporate extra hardware without much effort.

2.2Agent

An agent can be defined as a problem solving, autonomous, goal-driven computational entity with social abilities which is capable of effective and proactive conduct in an open and dynamic environment in the sense that it observes and acts on it to achieve its objectives [23]. In Fig.2 properties of an agents are shown based on those properties an agent's behavior can be described. These are the basic properties that an advance agent is expected to exhibit as suggested by author in [23].

2.2.1Autonomous

An autonomous agent is one which is self-sufficient and can operate without direct external intervention in a sense it can be said that an advance agent has control over its behavior

2.2.2Intelligent

Intelligence is a property which correspond to both autonomy and adaptiveness as those originate from agent's ability to learn from past experiences and environment without external intervention.

2.2.3Responsive

Responsiveness stands for agents' ability to respond to changes environment. An agent is well equipped with sensors and actuators so that it can sense the changes in environment and respond accordingly with help of actuator

2.2.4Goal Oriented and Proactive

An agent continuously works to achieve its goal. In order to achieve its goal, it can take initiative if needed, thus goal orientation is very similar to proactive behavior of agent where agent is eager and complex enough to sense and respond to possible changes in the environment.

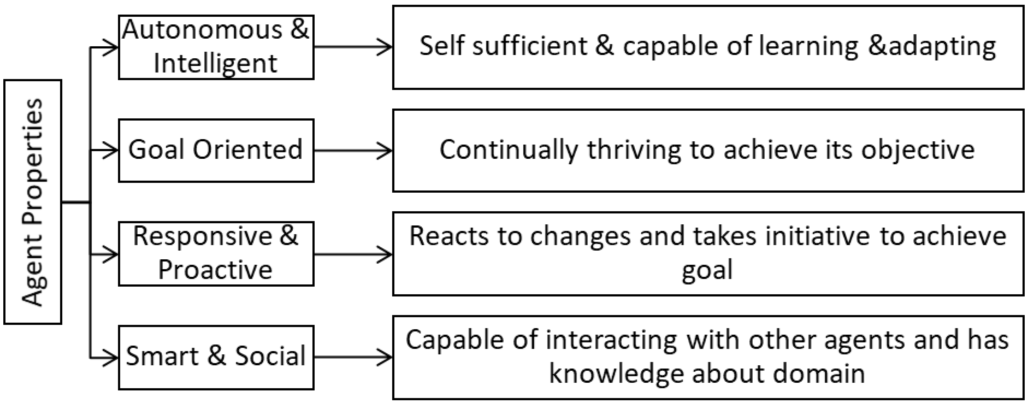

Fig 2: Properties of Agents

2.2.5Smart and Social

Smart behavior of agent originates from its problem-solving ability based on the knowledge that it has about a well-defined area. An agent must have data and knowledge about its domain and environment to display such behavior. In order to achieve its local objective or to contribute towards completion of global objective an agent communicates and shares data with other agents and operator such behavior correspond to social ability of agent.

2.3Environment

Environment is the limited surrounding that an agent is concerned with and is existing in, it can be physical space or logical positioning of agent in a network. An agent continuously senses its environment to support decision making process. In Fig.3 properties of an environment are shown based on those properties an environment's behavior can be described as suggested by author in [40]. Depending upon these properties' environment is broadly categories as:

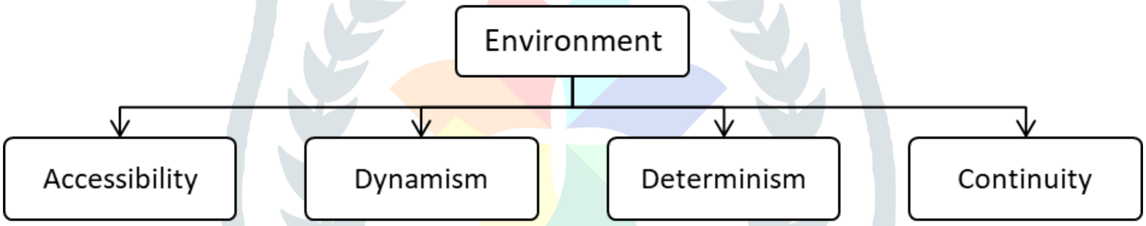

Fig 3: Properties of Environment

2.3.1Accessible or Inaccessible

In an accessible environment agent can obtain up-to-date data information with accuracy where in an inaccessible environment that's not possible.

2.3.2Static or Dynamic

A static environment is one in which changes occurring in environment are only function of agent's action where as in dynamic environment changes occur due to various other factors including agent's action.

2.3.3Deterministic or Non-Deterministic

If the results of agent's action are predictable regarding environment's state, then it's a deterministic environment or else vice versa.

2.3.4Continuous or Discrete

` In a continuous environment an agent's state is affected by a continuous function, that is there are infinite number of actions and precepts possible.

2.4Agent Communication Network

As we have already discussed about agent's we know that they are collaborative and cooperative entities and in order to do so they need a communication channel or network so that they could share information. There are two types of agent communication, direct communication and indirect communication [23].

Direct communication is like a formal letter enclosed in an envelope and written in structured way to provide information in a comprehensive way. Here envelope is the syntax and messages are actual data, although syntax is standardized but messages are not. There are several communication languages such as ACL or Agent Communication Language which was given by FIPA, Knowledge Query and Manipulation Language (KQML) given by DARPA and etc.

Indirect communication is like an auction where an item to sold is placed and information regarding it is conveyed to bidders present in auction hall who bid for it and highest bidder gets the item. Usually used where there a number of agents of similar capability are involved such drone swarming.

Table 1 Agent Communication Protocols and Languages

Communication Protocols and Languages	Description	References
Contract Net Protocol	It is a communication protocol appropriate for process with clear task distribution where agents communicate in three stages that are bids, announcement and award message.	[33, 41]
KQML	It is a high-level language based on speech act theory and regulates the run time exchange of information between the agents. It is composed of three layers that are content, message and communication layers.	[10, 22]
FIPA-ACL	Agent Communication Language was developed by the Foundation for Intelligent Physical Agents. This ACL appears to be similar to KQML in that it defines an 'outer' language for messages, as well as 20 performatives for describing the intended interpretation of messages.	[12, 19]
Belief-Desire-Intention Based Negotiation Protocol	The belief-desire-intention (BDI) architecture incorporates data structures that are clearly represented and loosely correspond to these mental processes. This protocol could be used to resolve conflicts between agents using a game theory architecture.	[2, 9]

III. Agent Development Platforms

Agent oriented programming is quite new concept given by Yoav Shoham in 1993 during his study of artificial intelligence and it is built on the concept of effectively coding agents in terms of subjective concepts developed by agent scholars to represent the features of agents [3, 23].

Agent oriented programming is closely associated to object-oriented programming, as agents are the building blocks in an agent-based programming language, and agents are developed by programming their behavior patterns same as objects in object-oriented programming [40].

Table 2 Agent Development Platforms

APL	Agent Model	Implementation Platform	References
ASTRA	Belief-Desire-Intention	JAVA	[11]
AUML	FIPA and OMG	UML	[43]
Chromar	Rule based	Haskell	[20]
JaCaMo	Belief-Desire-Intention	JAVA	[6]
JADE	FIPA	JAVA	[5, 35]
JADEL	Domain specific language	JAVA or JADE	[4,5]
JADEX	Belief-Desire-Intention and object-oriented programming	JAVA	[30]
Jason	Belief-Desire-Intention	JAVA	[17]
RMAS	Cyber physical system and database centric	MATLAB or SQL lite	[7]
SARL	Domain specific language	JAVA	[18]

JADE is an open-source platform which is FIPA (Foundation for Intelligent Physical Agents) compliant and was introduced by Telecom Italia under the Library Gnu Public License, it includes the agent communication channel, agent management system, and directory facilitator, which are all required by FIPA to manage the agents' infrastructure [25, 23]. It is also one of the most used agent programming tools as being complaint to FIPA standard it has various advantages over others and is based on Java programming language which adds advantage of cross-platform through Java Virtual Machine

IV. Application of MAS in Manufacturing

MAT concepts are strictly computer science but is not limited to it, as its scope in various key domains is already addressed in literature and considerable industrial applications have surfaced since origin of MAS

4.1Product Design and Development

As system designs are becoming more complex and resilient, and the essence of global business has evolved, practitioners and researchers have sought to use multi agent systems to tackle a growing number of difficulties. As time-based competition becomes more intense, product development cycles shrink, demanding more concurrent design between several designers, or among design professional and retailers or among vendors and market analysts. Several industrial design and supply firms have sought to build collaborative relationships, each with its own functional impairment. The intricacy of industrial development efforts has compelled developers to take on a more networked or collaborative role. Efficient collaborative network can significantly reduce product development time and identify any manufacturing and assembly difficulties, such as the need for development adjustments on the production floor.

A multi-agent model of truly collaborative operations that captures collective decision behavior and thus provides a realistic framework for capturing the transition from sequential to a concurrent design process could aid in the support of these procedures. The unit of analysis in this area is influenced by the magnitude and character of the event being studied: when studying collective design co - ordination, systems are likely to model a network, whereas when examining communication across multiple operational boundaries such as production, marketing, and sales, systems tend to model a network. In order to address such problem author of [34] describes a distributed multi-agent approach to product design and manufacturability analysis with the goal of developing a distributed concurrent engineering system that allows geographically distant entities to collaborate toward overall system goals. Similar problem is also addressed in article [28] where using digitalized information, an architecture for a web-based MAS was recommended to promote joint product development and testing among geographically distributed functional agents an in article [38] offered a collaborative design framework based on agents to reduce the time and expense of developing new airplane components.

4.2Production Planning and Control

On the shop-floor stage in manufacturing, MAS designs aim to mimic systems' capacity to deal with unpredictable events. Frequent changes in the production plan as a result of production disturbances, such as machinery failure or the late or uncertain arrival of component components, can produce disruptions in manufacturing and scheduling, with a 'ripple effect' on other solid functional boundaries. Dynamical agent interactions are proposed in MAS-based production systems and assembly line designs to fulfil local and global objectives. In addition, this section of the MAS research discusses two more significant issues.

4.2.1Job assignment on the shop floor and dynamic scheduling

Manufacturing scheduling refers to the systematic and sensible allocation of resources to certain jobs. Suppliers and clients typically make arrangements detailing the timing and quantity of commodities to be delivered; however, variations in availability, production and processing functions, and other factors might disrupt such expected patterns of cooperation. Agents may be limited by the lack of time and resources, or by the failure to complete tasks on schedule in an earlier step of the production process. However, on parallel design shop floors, a number of agents are good at conducting the same operation at the same time, allowing for quicker execution by allocating jobs among dispersed entities or functions. The MAS model has been used to simulate a broad range of scheduling scenarios involving both human and machine workers. In [37] a decentralized production scheduling framework was presented which included a networked production floor control structure, dynamic distributed scheduling algorithms, MAS modelling of work cells, and service-oriented shop floor integration. In [16] presents an event-driven factory floor work-in-progress management platform that intends to monitor and regulate variable production and material handling through tracking and visualization of shop floor manufacturing processes facilitated by radio frequency identification.

4.2.2Integration of Production Planning and Scheduling (IPPS)

Many previous shop floors functioned completely in sequence prior to the advent of parallel processing, which meant that barriers like machine malfunction and arriving materials delays quickly put best possible outcome out of reach for real system performance. Manufacturers responded by creating various forms of variable production control aimed at coping with variations in incoming amounts. In various applications MAS has been used to integrate production planning and scheduling processes as such in article [39] a two-stage ant colony optimization algorithm is implemented in a MAS to accomplish IPPS in the job shop type flexible manufacturing environments where first step is to identify the preferred set of operations and machines and second is to identify the most favorable sequence and schedule. Similarly in article [29] a multi-agent architecture for an integrated and dynamic system for process planning and scheduling for numerous jobs, as well as a negotiation protocol based on PROSA and ADACOR architecture, were described.

4.3Manufacturing Process Control

In today's manufacturing business, process control has become an essential component of quality management. One of the goals of production quality control is to ensure that items are created without faults and according to parameters without adding unduly to production time and expense. Over-controlling a process may be just as damaging to a business as under-controlling it. In industry, operators frequently rely on their instincts and personal knowledge to determine where process verification should be implemented and where it should be tightened when processes fail to meet requirements. However, for effective and efficient control of manufacturing processes a few problems must be addressed.

4.3.1Manufacturing Disturbance Handling

Manufacturing systems are dynamic, non-linear, and often chaotic settings that are prone to unforeseen disruptions that cause deviations from initial plans and affect system performance. Disturbance management has always been done in a centralized manner, with a fail-fix approach with only one type of disturbance in mind: machine failure. MAS application in this specific domain can help to cover a broader perspective from just machine failure to overall disturbance handling due to various factors. A multi-agent architecture is proposed in [29] to improve the architecture's performance for agile and responsive disturbance handling. The author of [24] addresses this problem by proposing a holonic disturbance management architecture based on the ADACOR principles. The disturbance handling functions are distributed, giving the re-scheduling problem a predictive dimension and taking into account the most common types of shop floor disruptions.

4.3.2Process Cycle Time Reduction

The duration it taken to finish a single manufacturing operation on one entity, or several in one go, from beginning to end, is referred to as the process cycle time. It is made up of different time elements like actual production time, wait period, travel times, inspection time, and so on. However, only processing time is beneficial for manufacturing the rest is wasted due to the time spent moving and placing components on the machine. To address such a problem, a MAS application can significantly reduce lead time. Inspection time which is the time required to examine a product to ensure that it is free of defects. Inspections can be built into the production process, so that no separate inspection function is needed as in [25] author proposed customization of inspection plan varying from product to product to reduce redundant test and inspection time. Transportation time which contributes to lead time manufacturing which is due to moving work in process inventory from one work station to another, for reduction in such wastage of industrial resources in [36] author describes research on the use of heuristics to enhance the productivity of an multi agent based automated guided vehicle control (AGV) through optimization of the AGV guide path, the agent-based controller was shown to maintain a deadlock-free movement of AGVs in a free space world model.

4.4Quality Control

When it comes to any product or service, quality is critical. Quality has become the market distinguishing factor almost for all goods and services due to the high level of competition. It also serves as the foundation for a productive manufacturing system that reduces waste. The quality of the finished product is a direct indicator of the manufacturing system's operational reliability. The primary goal of smart manufacturing system operation and maintenance is to ensure the predictability and reliability of finished products. There have been several applications of MAT in the domain of quality control each focusing on a specific aspect of quality control although the major factors or problems regarding quality control that should be addressed.

4.4.1Quality of conformance

The potential of a good or service or a process to fulfil its design specifications through production with a tolerable deviation in design specifications is known as quality of conformance. According to the author of [32], who developed a fuzzy Q learning MAS for process control of continuous chemical production, the quality of product conformance can be maintained by continuously monitoring the production process through data acquisition and validation with benchmarked data, and any deviation found in data will indicate a deviation in the production process, which can be corrected in time to avoid major losses. According to [27], an individual agent responsible for the execution of quality control tools can be deployed in different departments of a plant. Quality experts determine which quality control tools are appropriate for each department based on the condition of the departments, the type of input data, and the importance of checking quality.

4.4.2Integrating quality and maintenance

Quality of a manufactured good is a direct function of machines state on which it is being processed any slight deviation in any functional characteristic of machine can result in huge deviation of manufactured product thus necessitating the need of integrated manufacturing and maintenance of machine. In [1] presents a prescriptive maintenance model is organized into four layers: data management, predictive data analytic toolbox, recommender and decision support dashboard, and an overarching layer for semantic-based learning and reasoning to improve two core abilities: first efficiently processing large amounts of multi-modal and diverse data collected from multidimensional data sources, and second to effectively producing decision support metrics and suggestions.

V. CHALLENGES TO MAS

Although MAS and agent-based technology have numerous advantages over conventional technology such as automation, decentralization of control and enhancing systems overall performance yet the number of existing industrial application of MAS are limited and is majorly due to challenges regarding MAS. Through study of present work few of those challenges are identified and those are: -

5.1Consensus and Coordination

The MAS consensus challenge is concerned with achieving global agreement on a specific attribute of concern. In recent years the consensus problem has received a lot attention from scholars and a survey of solutions present for different type of MAS are summarized in [31]. In [26], the author investigates event-triggered linear MAS consensus control and provides robust consensus conditions and a feedback matrix method for achieving consensus with a desired disturbance absorption ability.

Whereas coordination problem of MAS is concerned with collaborative behavior of agents to achieve global objectives. In [44] the problem of coordination control was identified and a general approach to design convergent coordination control laws for multiagent systems subject to motion constraints was presented.

5.2Formation and Connectivity

The problem of formation and connectivity are a major concern for the flocking and swarming field of unmanned aerial vehicles as addressed by authors in [15, 21].

The author of [21] addresses the formation control problem of multiple unmanned aerial vehicle systems and suggests a new formation control method that accomplishes three-dimensional formation tracking by incorporating smooth distributed consensus control protocols into the geometric pattern model.

The problem of leader following synchronization with connectivity preservation for a multiple single-integrator system, in which the leader system could be any linear autonomous system, was investigated in [15], and a distributed state feedback control protocol was proposed that can preserve the system's connectivity while also accomplishing asymptotic tracking among all followers to the leader system's output.

5.3Security and Fault tolerance

The security and fault tolerance problem are associated with unforeseen physical faults and cyber-attacks on a single agent which can spreads quickly to other agents via information interaction, leading to severe drop in system performance and even MAS destruction as suggested by author in [42] along with a complete survey on Security and Fault tolerance of MAS.

False data injection attacks in bilateral teleoperation systems are studied in [14], where the attacker can inject false data into the states being exchanged between the master and slave robots, and a physics-based detection scheme with an encoding–decoding structure was proposed to detect any general false data injection attack.

In [13], the authors examine the distributed resilient observer-based fault-tolerance control problem for heterogeneous linear MAS in the presence of actuator faults, and a novel fault-tolerant controller is developed to compensate for actuator faults without provoking chattering behaviors.

5.4Standards

While going through literature regarding MAS application and development methodologies the lack of standards regarding agent programming and simulation was found which can be said to major drawback regarding MAS development and industrial application. Although there are a lot of agent programming languages in literature but their application is limited leaving behind a few such as JADE or AUML.

VI. Conclusion

This study provided an overview of multi agent applications in manufacturing domain. It described the concept of MAS and explained the agent, environment, and communication protocol, as well as brief overview of the numerous agent programming languages that are being used. The study then provided a brief categorization of multi agent applications in the domain of manufacturing, as well as the issues that arise during the development and operation of these systems. The detailed study of multi agent system concludes that: Traditional manufacturing processes and quality control systems are incapable of meeting the demand for mass customization in production systems that are designed to handle only mass manufacturing. Companies competing in the global market are confronted with the problem of constantly changing consumer demand, forcing them to minimize product development time and implement the notion of concurrent engineering into their manufacturing system. For facing such challenges multi agent system are considered to be the best solution by various researchers and industrial experts alike even considering them to be essential for devolvement of cyber-physical system in manufacturing domain.

References

[1] Ansari, F., Glawar, R., & Nemeth, T. (2019). PriMa: a prescriptive maintenance model for cyber-physical production systems. Https://Doi.Org/10.1080/0951192X.2019.1571236, 32(4–5), 482–503. https://doi.org/10.1080/0951192X.2019.1571236

[2] Archibald, B., Calder, M., Sevegnani, M., & Xu, M. (2022). Modelling and verifying BDI agents with bigraphs. Science of Computer Programming, 215, 102760. https://doi.org/10.1016/J.SCICO.2021.102760

[3] Bellifemine, F. Luigi., Caire, Giovanni., & Greenwood, Dominic. (2007). Developing multi-agent systems with JADE. 286.

[4] Bergenti, F. (2014). An Introduction to the JADEL Programming Language. Proceedings - International Conference on Tools with Artificial Intelligence, ICTAI, 2014-December, 974–978. https://doi.org/10.1109/ICTAI.2014.147

[5] Bergenti, F., Iotti, E., Monica, S., & Poggi, A. (2017). Agent-oriented model-driven development for JADE with the JADEL programming language. Computer Languages, Systems and Structures, 50, 142–158. https://doi.org/10.1016/j.cl.2017.06.001

[6] Boissier, O., Bordini, R. H., Hübner, J. F., Ricci, A., & Santi, A. (2013). Multi-agent oriented programming with JaCaMo. Science of Computer Programming, 78(6), 747–761. https://doi.org/10.1016/J.SCICO.2011.10.004

[7] Bonci, A., Longhi, S., & Pirani, M. (2019). RMAS Architecture for Autonomic Computing in Cyber-Physical Systems. IECON Proceedings (Industrial Electronics Conference), 2019-October, 2996–3003. https://doi.org/10.1109/IECON.2019.8926991

[8] Bond, A. H., & Gasser, L. G. (1988). Readings in distributed artificial intelligence. M. Kaufmann.

[9] Cao, L., Cao, J., Li, M., & Tang, F. (2008). Method of using BDI agents to implement service-oriented workflow mapping in AGWMS. Https://Doi.Org/10.1080/17445760701659290, 23(2), 171–195. https://doi.org/10.1080/17445760701659290

[10] Choi, J., Kim, Y., Park, Y. T., & Kang, S. H. (2010). Agent-based product-support logistics system using XML and RDF. Http://Dx.Doi.Org/10.1080/00207720210133660, 33(6), 467–484. https://doi.org/10.1080/00207720210133660

[11] Collier, R. W., Russell, S., & Lillis, D. (2015). Reflecting on agent programming with agentspeak(L). Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 9387, 351–366. https://doi.org/10.1007/978-3-319-25524-8_22

[12] Cucurull, J., Martí, R., Navarro-Arribas, G., Robles, S., Borrell, J., & Suades, G. (2010). Fragment Transfer Protocol: An IEEE-FIPA based efficient transfer protocol for mobile agents. Computer Communications, 33(18), 2203–2214. https://doi.org/10.1016/J.COMCOM.2010.08.005

[13] Deng, C., & Wen, C. (2020). Distributed Resilient Observer-Based Fault-Tolerant Control for Heterogeneous Multiagent Systems under Actuator Faults and DoS Attacks. IEEE Transactions on Control of Network Systems, 7(3), 1308–1318. https://doi.org/10.1109/TCNS.2020.2972601

[14] Dong, Y., Gupta, N., & Chopra, N. (2020). False Data Injection Attacks in Bilateral Teleoperation Systems. IEEE Transactions on Control Systems Technology, 28(3), 1168–1176. https://doi.org/10.1109/TCST.2019.2903446

[15] Dong, Y., & Huang, J. (2014). Leader-following rendezvous with connectivity preservation of single-integrator multi-agent systems. IEEE/CAA Journal of Automatica Sinica, 1(1), 19–23. https://doi.org/10.1109/JAS.2014.7004615

[16] Fang, J., Huang, G. Q., & Li, Z. (2012). Event-driven multi-agent ubiquitous manufacturing execution platform for shop floor work-in-progress management. Http://Dx.Doi.Org/10.1080/00207543.2012.693644, 51(4), 1168–1185. https://doi.org/10.1080/00207543.2012.693644

[17] Fernández-Díaz, Benac-Earle, C., & Fredlund, L. (2015). Adding distribution and fault tolerance to Jason. Science of Computer Programming, 98(P2), 205–232. https://doi.org/10.1016/J.SCICO.2014.01.007

[18] Galland, S., Rodriguez, S., & Gaud, N. (2020). Run-time environment for the SARL agent-programming language: the example of the Janus platform. Future Generation Computer Systems, 107, 1105–1115. https://doi.org/10.1016/J.FUTURE.2017.10.020

[19] H Zhou, B., Feng Xi, L., -Y Tao, L., Zhou, B., Feng, L. X., & Tao, L. (2007). A framework of order evaluation and negotiation for SMMEs in networked manufacturing environments. Https://Doi.Org/10.1080/09511920601020730, 20(2–3), 199–210. https://doi.org/10.1080/09511920601020730

[20] Honorato-Zimmer, R., Millar, A. J., Plotkin, G. D., & Zardilis, A. (2019). Chromar, a language of parameterised agents. Theoretical Computer Science, 765, 97–119. https://doi.org/10.1016/j.tcs.2017.07.034

[21] Kada, B., Khalid, M., & Shaikh, M. S. (2020). Distributed cooperative control of autonomous multi-Agent UAV systems using smooth control. Journal of Systems Engineering and Electronics, 31(6), 1297–1307. https://doi.org/10.23919/JSEE.2020.000100

[22] Kim, Y., Kang, S. H., Lee, S. H., & Yoo, S. B. (2010). A distributed, open, intelligent product data management system. Https://Doi.Org/10.1080/09511920150216341, 14(2), 224–235. https://doi.org/10.1080/09511920150216341

[23] Leitão, P., & Karnouskos, S. (2015). Industrial agents : emerging applications of software agents in industry (first).

[24] Leitão, P., & Leitaõ, P. (2010). A holonic disturbance management architecture for flexible manufacturing systems. Https://Doi.Org/10.1080/00207543.2010.518735, 49(5), 1269–1284. https://doi.org/10.1080/00207543.2010.518735

[25] Leitao, P., Rodrigues, N., Turrin, C., & Pagani, A. (2015). Multiagent System Integrating Process and Quality Control in a Factory Producing Laundry Washing Machines. IEEE Transactions on Industrial Informatics, 11(4), 879–886. https://doi.org/10.1109/TII.2015.2431232

[26] Liu, Y., & Jia, Y. (2018). Event-triggered consensus control for uncertain multi-agent systems with external disturbance. Https://Doi.Org/10.1080/00207721.2018.1543733, 50(1), 130–140. https://doi.org/10.1080/00207721.2018.1543733

[27] Mahdavi, I., Shirazi, B., Ghorbani, N., & Sahebjamnia, N. (2013). IMAQCS: Design and implementation of an intelligent multi-agent system for monitoring and controlling quality of cement production processes. Computers in Industry, 64(3), 290–298. https://doi.org/10.1016/J.COMPIND.2012.11.005

[28] Mahesh, M., Ong, S. K., & Nee, A. Y. C. (2007). A web-based multi-agent system for distributed digital manufacturing. Https://Doi.Org/10.1080/09511920600710927, 20(1), 11–27. https://doi.org/10.1080/09511920600710927

[29] Nejad, H. T. N., Sugimura, N., & Iwamura, K. (2010). Agent-based dynamic integrated process planning and scheduling in flexible manufacturing systems. Https://Doi.Org/10.1080/00207543.2010.518741, 49(5), 1373–1389. https://doi.org/10.1080/00207543.2010.518741

[30] Piunti, M., Ricci, A., Braubach, L., & Pokahr, A. (2008). Goal-directed interactions in artifact-based MAS: Jadex agents playing in CARTAGO environments. Proceedings - 2008 IEEE/WIC/ACM International Conference on Intelligent Agent Technology, IAT 2008, 207–213. https://doi.org/10.1109/WIIAT.2008.349

[31] Qin, J., Ma, Q., Shi, Y., & Wang, L. (2017). Recent advances in consensus of multi-agent systems: A brief survey. IEEE Transactions on Industrial Electronics, 64(6), 4972–4983. https://doi.org/10.1109/TIE.2016.2636810

[32] Sahebjamnia, N., Tavakkoli-Moghaddam, R., & Ghorbani, N. (2016). Designing a fuzzy Q-learning multi-agent quality control system for a continuous chemical production line – A case study. Computers & Industrial Engineering, 93, 215–226. https://doi.org/10.1016/J.CIE.2016.01.004

[33] Smith, R. G. (1980). The Contract Net Protocol: High-Level Communication and Control in a Distributed Problem Solver. IEEE Transactions on Computers, C–29(12), 1104–1113. https://doi.org/10.1109/TC.1980.1675516

[34] Sun, J., Zhang, Y. F., Nee, A. Y. C., Su, J., Zh An, Y. F., & Ee{, A. Y. C. N. (2010). A distributed multi-agent environment for product design and manufacturing planning. Http://Dx.Doi.Org/10.1080/00207540010004340, 39(4), 625–645. https://doi.org/10.1080/00207540010004340

[35] Trappey, C. v., Trappey, A. J. C., Huang, C. J., & Ku, C. C. (2009). The design of a JADE-based autonomous workflow management system for collaborative SoC design. Expert Systems with Applications, 36(2), 2659–2669. https://doi.org/10.1016/J.ESWA.2008.01.064

[36] Wallace, A. (2006). Multi-agent negotiation strategies utilizing heuristics for the flow of AGVs. Https://Doi.Org/10.1080/00207540600726752, 45(2), 309–322. https://doi.org/10.1080/00207540600726752

[37] Wang, C., Ghenniwa, H., & Shen, W. (2008). Real time distributed shop floor scheduling using an agent-based service-oriented architecture. Https://Doi.Org/10.1080/00207540701738052, 46(9), 2433–2452. https://doi.org/10.1080/00207540701738052

[38] Wang, W., Li, Y., Li, H., & Liu, C. (2012). An agent-based collaborative design framework for feature-based design of aircraft structural parts. Https://Doi.Org/10.1080/0951192X.2011.597779, 25(10), 888–900. https://doi.org/10.1080/0951192X.2011.597779

[39] Wong, T. N., Zhang, S., Wang, G., & Zhang, L. (2012). Integrated process planning and scheduling – multi-agent system with two-stage ant colony optimisation algorithm. Https://Doi.Org/10.1080/00207543.2012.720393, 50(21), 6188–6201. https://doi.org/10.1080/00207543.2012.720393

[40] Woolidge, Michael. (2013). An introduction to multiagent systems.

[41] Yeung, W. L. (2018). Efficiency of task allocation based on contract net protocol with audience restriction in a manufacturing control application. Https://Doi.Org/10.1080/0951192X.2018.1493227, 31(10), 1005–1017. https://doi.org/10.1080/0951192X.2018.1493227

[42] Zhang, D., Feng, G., Shi, Y., & Srinivasan, D. (2021). Physical Safety and Cyber Security Analysis of Multi-Agent Systems: A Survey of Recent Advances. IEEE/CAA Journal of Automatica Sinica, 8(2), 319–333. https://doi.org/10.1109/JAS.2021.1003820

[43] Zhang, H. H., & Ma, R. Z. (2011). A modeling methodology for MAS based on AUML. Proceedings of the International Conference on Uncertainty Reasoning and Knowledge Engineering, URKE 2011, 2, 185–188. https://doi.org/10.1109/URKE.2011.6007940

[44] Zhao, S., Dimarogonas, D. v., Sun, Z., & Bauso, D. (2018). A General Approach to Coordination Control of Mobile Agents with Motion Constraints. IEEE Transactions on Automatic Control, 63(5), 1509–1516. https://doi.org/10.1109/TAC.2017.2750924