Management and Production Engineering Review
Volume 13 • Number 1 • March 2022 • pp. 44–51
DOI: 10.24425/mper.2022.140875

An Intelligent Method for the Scheduling of Cyber Physical Production Systems

Hassan KHADIRI[1], Souhail SEKKAT[2], Brahim HERROU[3]

[1] *Sidi Mohamed Ben Abdellah University, Laboratory of Industrial Technologies, Morocco*
[2] *Moulay Ismail University, ENSAM-Meknes, Morocco*
[3] *Sidi Mohamed Ben Abdellah University, Superior School of Technology, Morocco*

Received: 17 November 2020
Accepted: 09 February 2022

Abstract
The new industrial era, industry 4.0, leans on Cyber Physical Systems CPS. It is an emergent approach of Production System design that consists of the intimate integration between physical processes and information computation and communication systems. The CPSs redefine the decision-making process in shop floor level to reach an intelligent shop floor control. The scheduling is one of the most important shop floor control functions. In this paper, we propose a cooperative scheduling based on multi-agents modelling for Cyber Physical Production Systems. To validate this approach, we describe a use case in which we implement a scheduling module within a flexible machining cell control tool.

Keywords
Cyber Physical Production Systems, Scheduling, Multi-Agent Systems, Artificial Intelligence.

Introduction

To survive the industrial competition, a company must optimize the control of its manufacturing process. The Cyber Physical Production Systems allow achieving these challenges. They must be flexible, integrated, and intelligent. Scheduling is one of the most important shop floor control functions. Therefore, CPSs scheduling must be flexible and intelligent. In this paper we propose an intelligent Scheduling approach of Cyber Physical Production Systems. This approach is based on agent's technology and digital twin concept. The rest of this paper is organized as follow. Section 2 presents Cyber Physical Production Systems. Section 3 introduces scheduling problems in general and dynamic scheduling, in particular. In section 4, we describe Multi Agent technology and Digital Twin concept, their use in production scheduling, and propose an intelligent scheduling based on these technologies. Section 5 closes with a practical example implementing a scheduling algorithm within a flexible machining cell control tool.

Corresponding author: *Hassan Khadiri – Sidi Mohamed Ben Abdellah University, Laboratory of Industrial Technologies, Faculty of Sciences and Technologies, B.P. 2202 – Imouzzer Road, Fez, Morocco, phone: +(212) 66 318 9518, e-mail: hassan_khadiri@yahoo.fr*

Cyber Physical Production Systems

The new industrial revolution termed Industry 4.0; consist in building an intelligent factory, thanks to the new internet and web technologies. Cyber physical system CPS is a basic concept on which lean industry 4.0. The concept of Cyber Physical Systems has been defined as a system in which the physical processes (physical space) are intimately integrated with information, calculation, communication, and control systems (cyber space) (Cardin, 2016) (Fig. 1).

To better understand this concept, it is necessary to consider the evolution of production systems de-

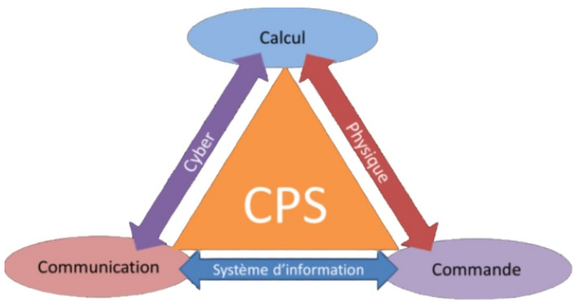

Fig. 1. Cyber Physical System Components (Cardin, 2016)

sign paradigms. In this paragraph we will, at first, describe this evolution, then we will present cyber physical production systems concept and their proprieties. Indeed, to answer quickly and economically the more and more variable demands, the production system became more and more complex. Manufacturing automation must meet several challenges. The next generation manufacturing systems must be flexible, integrated, interoperable and intelligent. Several enterprise engineering paradigms have been proposed to satisfy these requirements. The first paradigm is the concept of Dedicated Transfer Lines which first goal is productivity. Flexible production system is a second concept proposed to produce small and middle batches. it is based on (CIM) approach, information technology oriented, it aims achieving fully computerized and automated production systems integration around a common information system and is using ERP and MES software package, and products Design and manufacturing software tools (CADM, PLM.). Reconfigurable Manufacturing Systems (RMS) was to answer the inability of flexible Manufacturing Systems to tackle frequent demand changes in diversity and in volumes (Lamrani et al., 2003). They have been proposed by Yoram Koren team at the university of Michigan in 1995 (Koren & Ulsoy, 1997). This paradigm uses the Plug-and-produce concept, which allows Manufacturing Systems components to be added and removed according to demand changes. RMS modularity achievement requires connection interfaces of modules standardization. An RMS must therefore be interoperable.

These last years, with the development of internet and web technologies, Onori (Onori et al., 2009) proposed Evolvable Production Systems paradigm, which must be able to adapt. To address this, EPS must provide systems solutions that outlast several generations of product variants and foresee to accomplish adaptation capacity of Manufacturing Systems at lower levels, where the system is less complex. It is also necessary to endow control system with intelligent decision-making functions: coordination, Scheduling, etc. The Cyber Physical System must, then collect data from the physical system and analyses it to extract a knowledge that will be able to be used to control Manufacturing Systems in autonomous and intelligent manner and to bring it a certain conscience and auto-configuration and auto-organization capacity. Zuehlke (Zuehlke, 2010) has implemented these concepts through Services Oriented Architecture (SOA) within the Factory of Things. To be autonomous and intelligent, decision-making process of CPSs control must be improved. A lot of research is made nowadays to reduce machining times, improve product quality. In maintenance area, to anticipate failures, Lee (Lee et al., 2015) has implemented a Prognostics and health Management (PHM) software for Cyber physical systems.

Scheduling is one of the most important shop floor control functions that must be highlighted in CPPS. Indeed, CPPS must flexibly adapt their behaviour and structure to react to before runtime unknown changing conditions. The traditional way of engineering and operating production systems is based on static information flows. To react to disturbances during operation like varying customer demand, CPPSs have to be able to adjust their behaviour. The traditional way of implementing automation software based on assumptions about static behaviour of production facilities cannot address this requirement anymore (Vogel-Heuser et al., 2015). Recent reviews on scheduling and decision-making in production systems have been published under the new Industry 4.0 paradigm (Rossit et al., 2019). The main issue is whether it is possible to generate tools for controlling the different links of the supply chain, integrating this information to schedules. Ivanov et al. (Ivanov et al., 2018) have focused on the application of control theory to planning and scheduling. In this paper we will propose an intelligent Scheduling approach. Thus, in the following paragraph, we will describe production scheduling problem.

Shop floor scheduling problem

A scheduling problem consists in dispatching job tasks into resources and defining their rooting, in order to optimize criteria or to find trade-offs between several criteria's (delays, inventory reduction, set up time minimization). Scheduling problems have many application areas: shop floor scheduling, scheduling processes and data flows onto computer processors. In this paper, we will explore job shop scheduling problem.

Formalization of scheduling problem

We formalize a job shop scheduling problem by a set of n independent jobs $Ji \in \{J1, J2, \ldots, Jn\}$ and $i = 1, \ldots, n$, each job has to be performed on a set of m machines $Mj \in \{M1, M2, \ldots, Mm\}$ and $j = 1, \ldots, m$. In a shop floor, a production consists in manufacturing a products batch. Jobs are thus work orders, suggested by Material requirements planning. It is also a requirement to produce a determined quantity of product that must be delivered before a due date. The achievement of a job requires several operations executed on machines. The ordering of these operations is defined by a process plan, also called, operation sheet, route sheet, production rooting, etc. The operations will be done on machines. Each Job Ji follow its process plan composed of set of ordered operations Oij that must be executed in order onto machines, this order define precedence constraints between a same job operation.

We suppose that operations are non-pre-emptive, that machines are disjunctive, and finally, that each operation must be executed by single machine. Scheduling consists in timing and ordering the achievement of these operations, considering timing constraints (delays, ordering etc.) and resource constraints, which consist of the use and availability of resources required by tasks. A solution of scheduling problem is a definition of the start time t_{ij} for each operation Oij. This solution must satisfy all the constraints and optimize the criterions.

Dynamic and distributed scheduling

To solve a shop floor scheduling problem, we must consider random disruptions such as failures or raw material shortages. Static scheduling is constructed beforehand without considering random disruptions. It can be completed by a production re-scheduling each time an important disruption occurs. Dynamical scheduling is an on-line approach where the schedule is generated automatically.

To develop a scheduling software in a company we must rely on the IEC 62264 standard. This standard can provide a framework for an automated interface between production facilities and control systems and gives a representation of how information can be modelled and used (International Electrotechnical Commission, 2016). It organises the levels of decision-making hierarchically. Level 0 is associated to the physical production process. At this level we use sensors and actuators. Level 1 involves the intelligent devices that control the physical process. At this level we use Programmable Logic Controllers (PLC) and Distributed Control Systems (DCS). Level 2 represents the control and supervision of production facilities. Systems acting on Level 2 are Supervisory Control And Data Acquisition (SCADA) software's. Level 3 is associated to the production workflow. At this level we use manufacturing execution/operations management systems (MES/ MOMS). Finally, level 4 involves the business activity of the company. Systems acting on this level are ERP software's. Within a company an ERP Software proposes, in general, a Static scheduling.

In flexible shop floor, dynamic scheduling is done by SCADA Software. In this Software process plan can be programmed. This process plan completed by simple Priority Rules can constitute dynamic scheduling for simple cases and when the machine number and products diversity is low. However, when the products diversity is high, an MES software can be used to; firstly, transform Work Orders WO suggested by an ERP Software into closed WO for each shop floor resource, then, by following the process plan of considered product, he will send production instructions to shop floor machines and perform in real time, the launched WO follow up (Fig. 2). CPPS change the way in which decisions are made in the realm of industrial scheduling. Indeed, given the computing power of CPPS they should be able to launches operations on physical devices (level 1) plan and supervise the operations of the process (level 2) and handle the disturbance of the production process (level 3). Several control structures can be used for production scheduling (Giebels, 2000): centralized, hierarchical, coordinated, distributed, supervised distributed, etc. Current control systems are dominated by centralization, which is not adequate for handling production interruptions. To control Cyber Physical System, there has been a trend towards introducing autonomy and flexibility. The distributed architecture allows having flexible organization able to react in real time to random disruptions. Nevertheless, an initial centralized scheduling will be necessary to control the production. Therefore, we propose to adopt a distributed supervised architecture.

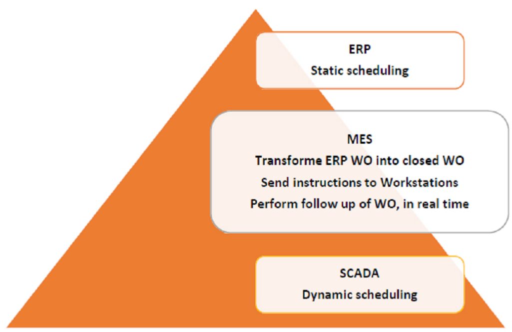

Fig. 2. Software tools used in shop floor scheduling

Dynamic scheduling approaches

According to the approach used for developing the scheduling system, dynamic scheduling can be divided into several categories: In completely reactive scheduling, no pre-schedule is generated, and scheduling is done in real-time (Bouhalouan et al., 2009). In robust pro-active scheduling, the schedule is generated beforehand anticipating the impact of disturbances in the manufacturing system. In hybrid approaches (reactive and pro-active) a schedule based on the proactive approach is generated beforehand and according to reactive approach, rescheduling is considered for responding to real-time disturbances. Finally, a dynamic scheduling, using a distributed and supervised architecture and which is based on a hybrid approach (reactive and proactive) is the most suitable for an intelligent scheduling of Cyber Physical Production Systems. The first step of implementing scheduling method is manufacturing system modelling. We use then an agent-based modelling to design a workshop scheduling tool. In the next paragraph we will describe this approach.

Multi agent approach

Research in Multi Agent Systems area are very active and operational scheduling systems have already been developed (Bouhalouan et al., 2009) (Hafri & Najid, 2001). This paper concerns multi-agent approach use to propose distributed architectures that promote cooperation between agents. In this paragraph we will first introduce agent concept and multi agent systems architecture, and then we will propose intelligent scheduling method based on multi-agent approach.

Multi agent systems architecture

Intelligent agents are programs implemented within a system that operates in constantly evolving environment. It is an IT entity that is integrated into an open computer system and can communicate with other agents. It is driven by a set of goals and has own resources. An agent is better defined by its properties (Kirn et al., 2006). The properties of an agent are, therefore, autonomy, reactivity, pro-activity, and sociability. These properties differentiate an agent from a simple software object. Moreover, when an agent has knowledge, it is said cognitive. Cognitive agents usually have goals to achieve. Indeed, they have explicit plans, as well as a knowledge base. On the other hand, they must interact with each other to achieve the goals. To do so agents, must have the ability to communicate across a set of protocols. A Multi-Agent System (MAS) is a computerized system composed of multiple interacting intelligent agents who are able to act on their environment and to collaborate with each other. MAS inherits from distributed artificial intelligence communication modes between agents and from System Theory the ideas of emergent behaviour expressed by the fact that a system can be more than the sum of its parts (Wooldridge, 2002).

Digital twin (DT) is another, promising concept, widely applied in the manufacturing field, especially in prognostics, scheduling, quality management etc. A popular model for the DT is the five-dimension DT, which includes a physical entity, an evolving virtual entity, DT data, on-demand services, as well as connections (Tao et al., 2019). The Digital Twin and Multi Agents concepts can be combined to develop scheduling software tool, each device can be equipped with a software agent, it constitutes its Digital Twin. Moreover, CPPS may be composed to higher level CPPS. In some works, also the term cyber-physical component is used (Feldmann et al., 2013). Consequently, a Cell or a Workshop DT can be developed based devices Digital Twins. Since this DT model primarily places emphasis on fusing data from both physical and virtual spaces, it is considered as an efficient way to realize cyber-physical convergence (Zhang et al., 2021).

To develop a multi-agent system, we must define an agent-oriented development process (Padgham & Winikoff, 2004). This process should allow the identification of basic functions of the multi-agent system, through the development of use cases scenarios and by considering the dynamics to which the system must be able to react. Then, he must determine kind of agent that the system should contain and how these agents will interact. To achieve the communication between system agents, an Integration Framework must be developed. The architecture of this Framework must include several agents that will implement different collaboration and negotiation communication services. To implement MAS, it is necessary to detail the internal structure of each agent and how it will execute the tasks assigned to him according to the interaction mechanism provided by multi-agent Framework. We therefore propose using multi-agent approach for workshop Scheduling software tool implementation, we develop in the following a distributed scheduling architecture based on intelligent agents.

Proposal of multi agent architecture for workshop scheduling

Multi agent approach is suitable for modular and distributed applications. Indeed, a distributed scheduling allows actors involved in the decision-making process (agents) to interact so that the scheduling is better implemented.

We then propose a system based on local agents that cooperate with a central planning system. Supervised distributed architecture is a trade-off between distributed and supervised approaches. This choice is motivated by several advantages which are: first supervision level provides a global vision and decision-making capacity and second the distributed level provides a dynamic allocation process that allow responsiveness and flexibility.

This proposed multi-agent architecture includes a supervisor agent who supervise the workshop and several machine agents each one associated with a machine workstation. These agents cooperate and interact, by message exchange, according to a decentralized approach to converge towards a global solution. This architecture analyses data collected in real-time, predicts delays, and supports the process of dynamically requesting rescheduling given the occurrence of irregularities by connecting information systems including the SCADA, MES, and ERP systems.

We assume that an Agent is a set of programs grouped in a module and that all these programs achieve the same function or concern. In addition, in a computer control system we have two levels: High-Level Control (HLC) allow decision making and achieve device coordination. Low-Level Control (LLC) allow carrying out actions requested by HLC Control level. However, high-level decisions depend on real-time data collected by LLCs. HLCs are transactional software and make decisions over a longer period of time and usually have a certain lead time. The five dimensions DT model is used to combine the speed of LLC level and the reasoning carried out at HLC level to make the right decisions as quickly as possible. It therefore allows meeting Cyber Physical Systems challenge, namely, having cyber part intimately linked to physical part of manufacturing system.

The first phase, of our approach consists of building-up prediction models and response action models, that should be performed prior to production. This model is used to develop a Knowledge Base KB of each agent. The second phase is to use this KB and compare the past reference data based on real-time data after the start of production and actively changing the schedule when a problem occurs. The proposed scheduling is based on the OPT Optimized Production Technology method, which consists of controlling the workshop from the bottleneck. On the other hand, we propose to implement a Kanban system between the bottleneck and the upstream workstation (Klein, 2008). The supervisor agent identifies workshop bottleneck and establishes a production plan based on process plan constraints. It then sends production instructions as messages to bottleneck machine, machine C in our example (Fig. 3). Locally, each agent defines a schedule by using priority rules (as FIFO, SPT, etc.) to minimize machine vacancy time. Scheduling solution is the result of system agent's interactions. The interaction in the multi-agent system consists of a message exchange that link dynamically agents. This dynamic relationship leads to convergence towards the desired solutions. During the scheduling process, each Machine agent cooperates with the machines between which it has precedence relation for the establishment of its local plan. These machine agents are identified in its knowledge base. Whenever supervisor or bottleneck agent sends it a message, it informs in his turn via a message each one of his neighbour agents of the new operation to be executed. He also consults his message queue to know the operations he must perform.

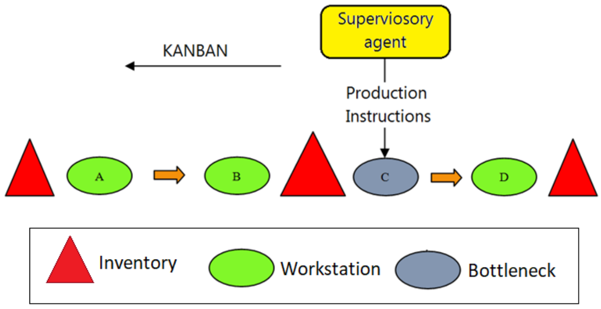

Fig. 3. Proposed Multi Agent Scheduling Architecture

Design of a scheduling software component

The proposed approach is applied to the flexible machining cell at the High School of engineering ENSAM. We will develop a scheduling software component for its control. In this section, we will describe the ENSAM machining cell, then, we will present, the specification and the implementation of the scheduling software component.

At the analysis of the current situation step we follow an architecture which consists of the three views (resources, information, and organization).

Resource view: the manufacturing system (Fig. 4), is organized around Transport system which allows a routing of work-piece carriers, toward a 3 axes CNC milling machine (MILL 55). A5 axes, jointed-arm robot (RV-2AJ), is used to load and unload parts from CNC machine.

Informational view: control architecture of the system includes a SCADA software (CIROS Supervision) in the cell level of control system. A network is connecting the machine controllers to the SCADA; the CNC machine is equipped with a DNC link based on

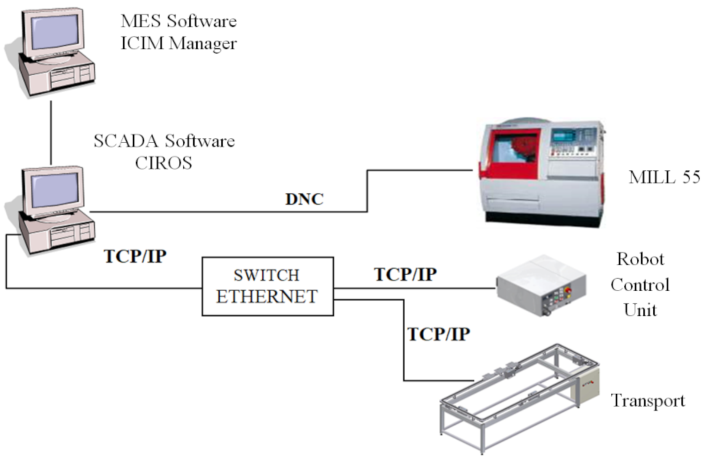

Fig. 4. Flexible machining cell at the ENSAM School

RS-232, the robot and Transport system with Ethernet extension. The shop floor control is achieved by MES software, ICIM Production Manager.

Organizational view: The production process is organized as follows: the cell production manager declares the Work Order WO list within ICIM Manager Software. He then, Store the Raw material and launch the execution of WO list into CIROS Software. During production process, the CIROS software displays supervision screens allowing production manager to know production process state and react if abnormal events occur. When WO list execution is finished CIROS update the stock content within ICIM Manager Database.

ICIM Production Manager achieves basic functions of MES Software. We will design a software component for scheduling function. Let us consider a work order sent by the ICIM Manager to the cell controller CIROS. The cell controller will select the process plan of the considered part from the database. It will determine adequate material and start the tasks. ICIM Production Manager Server communicates with CIROS through ODBC driver to collect data. The SQL language is used for data exchange between CIROS and other applications. CIROS communicates with the robot through the TCP IP driver. Programs based on server and client protocols, allow this communication, they are using the send and receive instructions.

To develop the integration framework of work cell control tool, a SCADA software CIROS Supervision (Festo Didactic, 2008), was used. This is a real-time, development environment. CIROS Software, allows developing programs and storing them in a library. The libraries are used for a central management of shared project components. They may be linked into several projects. The changes which are made also have an effect on the projects which include components of the library. In the integration framework, there is a hierarchy of programs. The high-level programs allow launching a production order at the cell level of CIM Pyramid. It is the Product Order level. PenHolder (42104) is, for instance, the program that allows launching the production of the product referred as 42104. A high-level program launches the macro process task level programs. They are macro tasks involving more than one machine. The Prog operation or task level consists in a task carried out by a single machine. ExecProg (MP, 15, 1, 0) is, for example, used to launch the pick and place robot program that moves on the palette from position 15 toward position 1. The device command level is the lowest level; it enables to implement the communication protocol between the supervisor and the machines.

The CIROS Project, that controls the machining cell, contains a module named Prod_OrderList() that reads the Work Order WO launched by ICIM Manager and starts the execution of its operations into workstations. The Prod_Strategie() module launches the operations of this Work Order on workstations. It represents the supervisor part of the scheduling architecture; it deals with precedence constraints and develops a global production plan. This program is the Cell Digital Twin of the Multi Agent architecture. The ExecProg() modules associated with each machine constitute the distributed part of the architecture; they allow dealing locally with the resource constraints. They form a queue, at this machine and implement the FIFO priority rule, using the semaphore concept. This concept use the CLAIM instruction to reserve the machine and RELEASE instruction to release it (Fig. 5).This figure show ExecProg (LOAD-

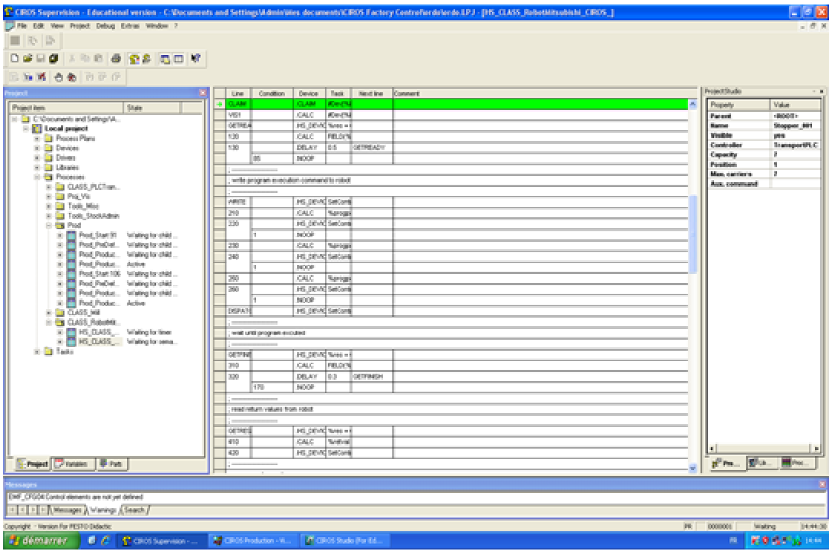

Fig. 5. ExecProg modules running with the semaphore concept

MILL, 1, 2) program running, it is called a process. This process is stopped at CLAIM instruction (marked green). It is waiting for semaphore.

Indeed, several programs form a queue, upstream the Robot. When a program is using the Robot, it reserves it using CLAIM instruction, so other programs must wait until it releases it using RELEASE instruction. When the robot become free the next program at the queue can then use it. Thus, our ExecProg (LOADMILL, 1, 2) process can continue and go to the following instruction.

The current scheduling architecture is only valid for simple cases. Indeed, the current cell contains only one workstation composed of two machines. If we add two other workstations (milling and turning workstations for example) the current scheduling programs will lead to a chaotic situation with overloaded inventory at the bottleneck workstation.

Furthermore, each workstation cannot accept a buffer stock greater than 3 work pieces. The strategy of the scheduling method implementation was to use the SCADA software CIROS Supervision (Festo Didactic, 2008), which is a real-time, development environment. At first, we will update a supervisor agent program within the ICIM Manager software. This agent will start by identifying the bottleneck machine and establish a schedule for this machine. In this way the production orders will not be launched at once from the ICIM Manager database. Then, locally, for some machines, we will program the SPT priority rule, within the ExecProg() program launched by the ProdStrategie() program to reduce the idle time of this machine

Conclusion and perspectives

In this paper we have implemented a dynamic and distributed scheduling software component within a Cyber Production Systems control tool. We have relied on multi-agent concepts to deal with production system complexity. This scheduling software component is able to react in real time to random disruptions, and it is suitable for an intelligent scheduling of Cyber Physical Production Systems. To achieve the communication between system agents, we used the SCADA software CIROS Supervision. It is ad hoc solution. A perspective for the future is to use Agent UML for system modelling and JADE for the implementation. Another interesting future research direction is to use machine learning algorithms to improve workshop scheduling strategy. We will also address, in the future, the automation of other shop floor control functions to design smarter production systems.

References

Bouhalouan D., Aissani N., and Beldjilali B. (2009), *Proposal of a model for Scheduling an Automated Production System Applications of hybrid genetic algorithms*, International Conference on Computer Science and its Applications CIIA.

Cardin O. (2016), *Contribution to the design, evaluation and implementation of cyber-physical production systems*, Habilitation to Direct Research, University of Nantes, IRCCyN Laboratory UMR CNRS 6597, IUT of Nantes.

Feldmann S., Rösch S., Schütz D., and Vogel-Heuser B. (2013), Model-Driven Engineering and Semantic Technologies for the Design of Cyber-Physical Systems, chez 11th IFAC Workshop on Intelligent Manufacturing Systems – The International Federation of Automatic Control, São Paulo, Brazil.

Festo Didactic (2008), CIROS Supervisions – User Manual.

Giebels M. (2000), *EtoPlan: a Concept for Concurrent Manufacturing Planning and Control – building holarchies for manufacture-to-order environments*, Université de Twente, Netherlands.

Hafri Y. and Najid N.M. (2001), *Use of the multi-agent approach for real-time control of production systems*, 3rd Francophone Conference on Modeling and Simulation, Design, Analysis and Management of Industrial Systems, MOSIM'01, Troyes (France).

International Electrotechnical Commission (IEC 62264-3) (2016), *Enterprise-control system integration – Part 3: Activity models of manufacturing operations management*, ICS : 25.040.40.

Ivanov D., Sethi S., Dolgui A., and Sokolov B. (2018), A survey on control theory applications to operational systems, supply chain management, and Industry 4.0, *Annual Reviews in Control*, Vol. 46, pp. 134–147.

Kirn S., Herzog O., Lockemann P. and Spaniol O. (2006), Multi-agent Engineering: theory and applications in enterprises, *International Handbooks on Information Systems*, Springer Science & Business Media.

Klein T. (2008), Active kanban to ensure centralized/distributed decision-making interoperability Application to a furniture industry, *Engineering sciences*. Henri Poincaré University – Nancy.

Koren Y. and Ulsoy A.G. (1997), Reconfigurable manufacturing systems, *Engineering Research Center for Reconfigurable Machining Systems*, ERC/RMS report #1. Ann Arbor.

Lamrani S., Hasan R. and Martin P. (2003), *Design process for Reconfigurable Manufacturing Systems (RMS): Approach using Petri Nets (RP)*, rev. 5th International Conference Design and Integrated Production. CPI2003, No. 62, ENSAM Meknes.

Lee J., Bagheri B. and Kao H.A. (2015), A Cyber-Physical Systems architecture for Industry 4.0-based manufacturing systems, *Manufacturing Letters 3*, pp. 18–23. DOI: 10.1016/j.mfglet.2014.12.001

Onori M., Semere D. and Lindberg B. (2009), Evolvable systems: An approach to Self-X Production, *Proceedings of the DET/CIRP conference on Digital Enterprise Technology*, Hong Kong.

Padgham L. and Winikoff M. (2004), *Developing Intelligent Agent Systems: A Practical Guide*, Chichester:

John Wiley & Sons. .

Rossit D.A., Tohmé F. and Frutos M. (2019), Production planning and scheduling in Cyber-Physical Production Systems: a review, *International Journal of Computer Integrated Manufacturing*, Vol. 32, No. 14–5, pp. 385–395.

Tao F., Zhang M. and Nee A. (2019), *Digital twin driven smart manufacturing*, 1st ed. Elsevier.

Vogel-Heuser B., Lee J. and Leitão P. (2015), Agents enabling cyber-physical production systems, *Automatisierungstechnik*, De Gruyter Oldenbourg, Vol. 63, No. 110, pp. 777–789.

Wooldridge M.J. (2002), *An Introduction to Multiagent Systems*, New York: John Wiley & Sons.

Zhang M., Tao F. and Ayc N. (2021), Digital Twin Enhanced Dynamic Job-Shop Scheduling, *Journal of Manufacturing Systems*, Vol. 58, pp. 146–156.

Zuehlke D. (2010), SmartFactory – Towards a factory-of-things, *Annual Reviews in Control*, Vol. 34, pp. 129–138.