On the open job-shop scheduling problem: a decentralized multi-agent approach for the manufacturing system performance optimization

Article *in* Procedia CIRP · January 2019

DOI: 10.1016/j.procir.2019.02.045

4 authors, including:

Guido Guizzi
University of Naples Federico II

Roberto Revetria
Università degli Studi di Genova

Silvestro Vespoli
University of Naples Federico II

Available online at www.sciencedirect.com

ScienceDirect

Procedia CIRP 79 (2019) 192–197

www.elsevier.com/locate/procedia

12th CIRP Conference on Intelligent Computation in Manufacturing Engineering, 18-20 July 2018, Gulf of Naples, Italy

On the open job-shop scheduling problem: a decentralized multi-agent approach for the manufacturing system performance optimization

Guido Guizzi[a,*], Roberto Revetria[b], Gianluigi Vanacore[a], Silvestro Vespoli[a]

[a]Università degli Studi di Napoli "Federico II", Dipartimento di Ingegneria Meccanica, dei Materiali e della Produzione Industriale, Piazzale Tecchio, 80 – 80125, Napoli (NA)
[b]Università degli Studi di Genova, Dipartimento di Ingegneria Meccanica, Energetica, Gestionale e dei Trasporti, Via dell'Opera Pia, 15 – 16145, Genova (GE)

* Corresponding author. Tel.: +39-081-768-2381. *E-mail address:* guido.guizzi@unina.it

Abstract

This paper investigates a dynamic integration of the process planning and scheduling operations of a typical Open Job-Shop manufacturing system. For this purpose, a modified CNP-based negotiation protocol - through a multi-agent modelling for jobs and operating machines - is proposed. This approach allows the introduction of an agents' hybrid behavior, considering both the own return and the system profit achieving the production performance maximization. Finally, a series of simulation runs are conducted in order to compare the performance of the proposed protocol with a recent optimization approach that uses a simple composite dispatching rule.

Keywords: Simulation; Scheduling; Open job-shop; Multi-agent; Industry 4.0

1. Introduction

The increasing complexity of products Bill Of Materials (BOMs) and technological cycles, combined with a wide variety in mix and quantity of finished products offered to customers, makes the production scheduling more complicated in a "Job Shop" or "Open Job Shop" plant configuration. It follows that the scheduling process nowadays becomes a decisive factor for company competitiveness that cannot admit any kind of inefficiency [21].

The aim of this paper is focused on solving the Open Job Shop Scheduling Problem (OJSSP), which can be considered as a generalization of the Job Shop Scheduling Problem (JSSP) [1]. The OJSSP includes a set of jobs that may be processed by a set of machines without a pre-defined order and, in which every machine can process at most one job at a time. The finding of optimal (or near optimal) solution of this scheduling problem it is a well-known problem with a Non-Polynomial (NP-hard) characteristic [2]. The objective of an OJSSP is represented by a schedule that optimizes a given criterion, such as the waiting time of jobs on the machines, throughput, utilization of machines, workload.

Several authors, in the literature, have tried to solve the OJSSP using both analytical and computational strategies: e.g., some proposed solutions are based on the use of genetic algorithms [3-5], branch and bound algorithms [6], hybrid algorithms [7], particle swarm optimization [8] and simulated annealing algorithm [9].

In this paper, a new highly flexible and innovative technology, based on the Multi-Agent Systems (MAS) [10] is introduced, for the first time, in the Distributed Artificial Intelligence and scheduling fields. In this way, it is possible to create a decentralized and horizontal scheduling model to replace the traditional vertical centralized model. Using a top-down centralized model for task allocation causes rigidity and confines problem-solving ability in the real world [11], although centralization can provide a consistent global view of the state of the system [12]. A decentralized model is a way to address the inflexibility of the hierarchical systems, to improve the reaction to disturbances and to allow parallel computing [13-15].

The rationale behind multi-agent systems allows separating an overall target in many local targets. In this way, the decision-making process is distributed at the level of the agents that, working together to reach the overall target, obtains, in sub-

2212-8271 © 2019 The Authors. Published by Elsevier B.V.
Peer-review under responsibility of the scientific committee of the 12th CIRP Conference on Intelligent Computation in Manufacturing Engineering.
10.1016/j.procir.2019.02.045

optimal times, more direct responses to the dynamic variations of the system.

With this regards, the aim of this work is the introduction and the definition of a MAS approach for the OJSSP. In particular, to identify the types and roles of the agents involved in the system and their interaction protocol, two main classes of agents, “Job” and “Resource”, are identified. Following, in order to hold the possible and desirable conflict among the agents, the “FIPA Contract Net Interaction Protocol (FIPA CNIP)” is proposed as interaction protocol. The FIPA CNIP represents an extension of the original CNP protocol proposed by *Smith* in [16] for solving cooperative problem among agents. As a matter of fact, the CNP is useful when the agents are often in a resource-competitive environment, allowing finding the best solutions, acceptable for both parties.

The Agent-Based Simulation Modeling (ABMS), instead, is used as implementation model with the use of AnyLogic® simulation software.

The results obtained from the simulation are then compared with those generated by a recent and different approach in the literature proposed by *Nasiri et al.* in [17], which proposed the same conditions and assumption assumed in the problem statement of this work.

The remainder of this paper is organized as follows. In Section II the follow the formal problem formulation of this work. In Section III the resolution of the introduced MAS and *Nasiri et al.* approach. In Section IV, the results of both approaches are shown, with a particular analysis and comparison about the performance of the proposed strategy versus the *Nasiri et al.* approach. Finally, in Section V follows the Conclusions of this work.

2. Problem Formulations

In this work, it is considered an Open Job Shop Problem consisting of five machines and five different types of jobs characterized by stochastic generation and execution times. Each generated job consists of 5 operations to be performed on each of the five machines without a precedence constraint. Each machine can only manage one job at a time (that must be completed before another can be processed on the same machine). Furthermore, the jobs cannot be interrupted (absence of preemption).

The chosen times between arrivals of successive jobs are presented in Table 1 and reflects the same value chosen from *Nasiri et al.* in [17]. Since each job is different and machines are not identical, every job has a different processing time for every machine. Table 2 presents the time required for processing each job on every machine.

Table 1. Time between arrivals of each type of job

Job Type	Time between arrivals (min.)
1	Uniform (40,60)
2	Triangular (40,51,62)
3	Normal (45,6)
4	Normal (35,3)
5	Exponential (43)

The aim is to achieve optimum scheduling that will minimize the mean waiting time of jobs and maximize the throughput of the production system.

Table 2. Time processing for each part on every machine.

Job	Machine	Processing Time (min)
1	M1	Normal (6,0.2)
1	M2	Uniform (4,6)
1	M3	Triangular (4,5,6)
1	M4	Normal (6,0.4)
1	M5	Normal (4, 0.1)
2	M1	Normal (6, 0.1)
2	M2	Triangular (5,6,7)
2	M3	Normal (8,0.5)
2	M4	Triangular (3,4,5)
2	M5	Triangular (5,6,8)
3	M1	Exponential (7)
3	M2	Exponential (6)
3	M3	Normal (6,0.5)
3	M4	Uniform (8,9)
3	M5	Triangular (7,8,10)
4	M1	Uniform (8,9)
4	M2	Normal (7,0.2)
4	M3	Triangular (6,7,8)
4	M4	Normal (4,0.1)
4	M5	Normal (7,0.5)
5	M1	Exponential (6)
5	M2	Normal (8,0.5)
5	M3	Uniform (6,8)
5	M4	Exponential (4)
5	M5	Uniform (5,9)

3. Resolution Approach

Since the Open Job Shop Scheduling Problem (OJSSP) is an NP-hard problem, the mathematical modeling is not an effective tool to use [17]. With this regards, to determine the optimal dispatching rule, a simulation multi-response optimization approach is predated for both approaches with the trial aim of minimizing the jobs mean waiting time and maximizing the production throughput.

In this section, the two different approaches for solving the open job-shop scheduling problem are presented: the *Nasiri et al.* and the introduced MAS ones.

A. Nasiri et al. approach

In the approach proposed in [17], since the problem is an open job shop, all five machines in any order must process all jobs. It is assumed that each machine has a queue of jobs waiting to be processed and every new job that enters in the system chooses the machine to be processed on, basing its choice on the length of its queue. Once the job has been processed on a machine, the next one is chosen with the same criterion.

With referring to the jobs in the queue on machines, the execution order is established by a priority value, obtained from a composite dispatching rule. This value is calculated for each job in every machine's queue. For the estimation of this value, the composite dispatching rule considers three information: the number of machines that remain to be processed for the considered job, the remaining working time of the under processing jobs on the considered machine and the remaining processing time of the considered job to be entirely completed.

The composite dispatching rule is presented in Equation (1):

$$S = W_1 * \left(\frac{R_i}{\sum_{j=1}^{5} P_{ij}}\right) + W_2 * \left(\frac{K_i}{5}\right) + W_3 * \left(1 - \left(\frac{P_{ij}}{\sum_{j=1}^{5} P_{ij}}\right)\right) \quad (1)$$

where:

- W_1 is the parameter that weights the remaining working time of the considered job;
- W_2 is the parameter that weights the number of machines that remain to be processed for the considered job;
- W_3 is the parameter that weights the remaining working time of the under processing jobs on the considered machine;
- R_i is the remaining processing time of the part i to be completed;
- P
- P_{ij} is the processing time of part i on machine j;
- K_i is the number of machines left for part i to be processed on;
- S is the composite dispatching rule score.

Each parenthesis in Equation (1) and the weight itself takes a value between 0 and 1.

The model has been implemented in AnyLogic® Version 8.2.3 Professional taking advantage of the allowed combination of the Discrete Event Simulation (DES) and the Agent-Based Simulation (ABS).

In Figure 1 is shown the state-chart we implemented in AnyLogic® for modeling the described approach.

B. Multi Agent System (MAS) approach

The proposed approach relies on the interaction between job agents and resource agents. As said, the interactions among different types of agents are based on a negotiation mechanism that extends the well-known CNP [18]: "FIPA CNIP".

In the FIPA CNIP [18], an agent called "initiator" asks for action the tasks that need to be performed by the various agents

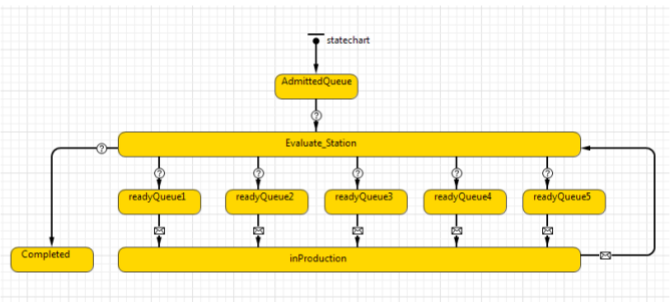

Fig. 1 - Nasiri et al. approach implemented in Anylogic

(one or more). These agents are called "participants". The initiator aims to optimize a proper objective function that needs to be modeled. This function, in turn, may depend on some parameter, such as price or execution time. For a given task, each participant can reply with a proposal or with a refusal. Obviously, the negotiation will continue only with the participants who have submitted a proposal.

The interaction protocol FIPA CNIP [18] consists of four main phases:

a. The initiator sends a Call for Proposals (CFPs) to the participants with a request of executing a task;
b. Each participant analyzes the CFPs and makes a consistent proposal with the specifications of the CFPs, if as it deems appropriate;
c. The initiator chooses the best offer among received, and it assigns the contract to the participant whose bid was successful;
d. The initiator rejects the other proposals received.

The representation of this protocol, based on extensions of the Agent Unified Modeling Language (AUML) proposed in [19], is given in Figure 2.

The Multi-Agent System (MAS) designed results in dynamic scheduling concerning the OJSSP formalized in Section II, using the FIPA CNIP as negotiation protocol with the aim of minimizing the jobs mean waiting time and of maximizing the global system throughput.

The fundamental idea for solving this problem is to distribute tasks among many autonomous and intelligent entities that negotiate with each other: the Agents. As in the previous approach for a behavioral uniformity, the model has been implemented in AnyLogic® Version 8.2.3 Professional taking advantage of the allowed combination of the DES and ABS.

Two Agent Types have been defined: "Job Agent", "Resource Agent" (which represents the operating machines).

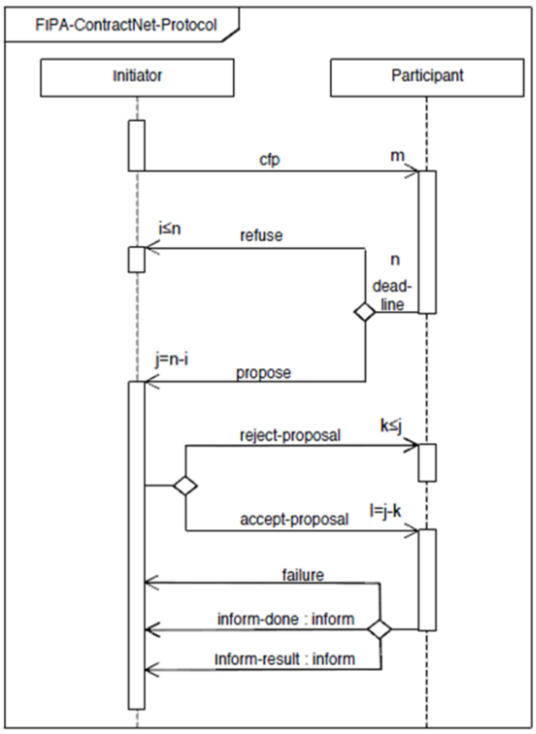

Fig. 2 - FIPA Contract Net Interaction Protocol [18]

The implementation of the CNIP was made by modeling the behavior of the above-mentioned Agent Types, through two statecharts, that shows the possible states of the agents. The transitions between the states are activated by a particular message (i.e., exchanged among agents) or by the occurrence of specific conditions. The statecharts of these Agent Type, are shown in Figure 3 and Figure 4.

In this model, the resource agent has the role of Initiator. As a matter of fact, it is the only agent that can start a Call for Proposals (CFP) addressed to the jobs agents, which assumes, in this case, the Participant role.

Obviously, it is possible to issue a CFP only from a resource in the "Available" state. This resource (that represents an operating machine) independently checks if there are jobs in the available state and when the search gives a positive result, the resource transit from the "Available" state in the "Evaluation_Proposal_and_Acceptance" state. Instead, in the case no jobs in the "Available" state are found, the Resource agent waits for a prefixed period (in our case, fixed to thirty seconds) before starting a new CFP.

In the CFP elaboration state, the resource generates the CFPs which consists of a service request that expresses the availability of the machine to perform one of the available jobs. The number of CFPs sent will be equal to the number of jobs in the "Available" state. At this point, the resource agent goes into "Wait_reply" state in which wait for replies from the job agents. The replies received from the Jobs Agent contain the job type to be processed and the required processing time for this particular resource. For example, if a milling machine agent starts a CFPs, all the jobs in the "Available" state elaborates the call and formulate a reply to the milling machine. However, the milling machine, if sufficiently automated, may perform plenty of operations. In this case, every Jobs Agent replies with the required time for completing its operation on the machine that sent the CFP (i.e., the time to perform a hole may vary if a milling machine is chosen, instead of an automated drill machine).

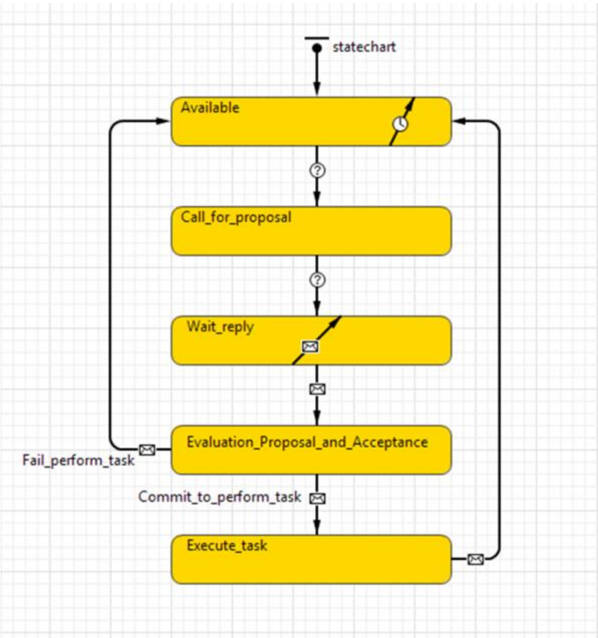

Fig. 3 - Resources Agent Statechart

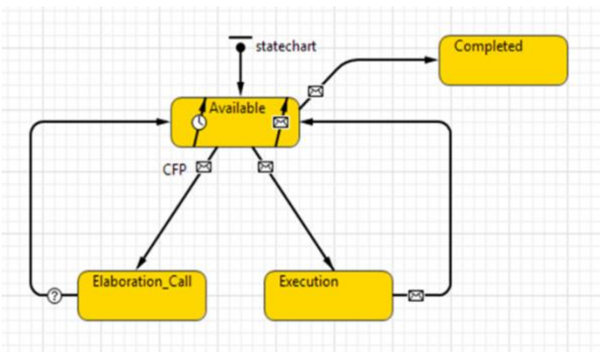

Fig. 4 - Jobs Agent Statechart

As said, when the Resource agent receives the proposals, it moves in the "Evaluation_Proposal_and_Acceptance" state. In this state, the resource agent evaluates each proposal received from the various job agents, calculating the value of a newer composite dispatching rule presented in Equation (2). Each function group in Equation (2) can assume a value between zero and one. Then, the composite dispatching value (S) will be between zero and three.

Finally, the resource agent accepts the Job Agent that owns the proposal with the highest score. This acceptance is communicated to Job Agent with an acceptance message; at the same time, the Resource Agent communicates to the other Job Agent the refusal and removes from the list of the proposals that have not been accepted.

The composite dispatching rule presented in Equation (2), represents the objective function to be evaluated by the initiator (resource) agent for every proposal received from the Job Agent.

This composite dispatching rule takes into account the remaining processing time of the job, the estimated processing time of the job on the chosen machine and the waiting time of the Job before it can be processed on this machine.

Once that this rule is defined, it is important to find the best proportion of the various components using three parameters that weights the importance of every aspect in the function. The choice of this value is demanded to an optimization algorithm that, simulating different work scenarios, choose the best combination of this parameter, minimizing the jobs mean waiting time and maximizing the system throughput.

$$S = X_1 * \left(\frac{t_{p_i} - t_{r_i}}{t_{p_i}}\right) + X_2 * \left(\frac{t_{p_{i,m}}}{t_{p_{\max i,m}}}\right) + X_3 * \left(\frac{t_{w_{i,m}}}{t_{w_{\max i,m}}}\right) \quad (2)$$

where:

- $t_{p_i} = \sum_{m=1}^{5} t_{p_{i,m}}$;
- $t_{r_i} = \sum_{m=1}^{5} t_{r_{i,m}}$;
- m, is the m-th resource;
- i, is the i-th job type;
- t_{p_i}presents the total processing time of the job i;
- $t_{p_{i,m}}$represents the processing time of job i on the resource m;
- t_{r_i}represents the total residual processing time of the job i;
- $t_{r_{i,m}}$represents the residual processing time of job i on the resource m;
- $t_{p_{\max i,m}}$represents the max processing time of the job i on the resource m;

- $t_{w_{i,m}}$ represents the waiting time of the job i on the resource m;
- $t_{w_{\max i,m}}$ represents the maximum waiting time of job i on the resource m;
- X_1 is the parameter that weights the remaining processing time of job i on the remaining resources $(0 \leq X_1 \leq 1)$;
- X_2 is the parameter that weights the processing time of job i on resource m $(0 \leq X_2 \leq 1)$;
- X_3 is the parameter that weights the waiting time of job i on resource m in $(0 \leq X_3 \leq 1)$;
- S is the composite dispatching rule score $(0 \leq S \leq 3)$.

During the evaluation and the acceptance process, it might happen that, for some reason, an error occurs creating a wrong the communication between agents. In this situation, a failure message that deletes the CFP and proposal in question from the interaction cycle has been envisaged.

Once the execution is finished, a completion message takes the Resource Agents in the "Available" state, making it ready to starts a new CFP and a consequent cycle of interaction.

The Job Agent, instead, has the Participant role. It replies with a proposal to the various CFPs issued by resources. Also, the Job Agent is in the "Available" at the beginning, and it obviously remains in this state until it receives a CFP message from Resource Agents.

The arrival of a CFP message triggers the transition that allows the job agent to transit into "Elaboration_Call" state. In this state, the Job Agent identifies the Resource Agent that own the CFP evaluates the CFP itself and elaborates a proposal in which it specifies the processing time and the operation typology it wants to perform on that specific resource. The proposal is transmitted through a message to the resource agent. At this point, the Job Agent switch into the "Available" state, awaiting a possible message of acceptance from the resource agent and, eventually ready for the answer to other CFPs. When an acceptance message is received, the Job Agent switch into "Execution" state sending, at the same moment, an execute message to the selected Resource Agent. At the end of the execution, the proposal is removed, and the Job Agent returns to the "Available" state.

Once that the job has been processed on each of the five different resources, a completion message switches the Job Agent in "Completed" state. Finally, this represents the points in which the total job processing time and its queue total time is calculated.

4. Simulation results and analysis

The simulation allows to model and analyzes complex systems in many real situations where it is extremely difficult to identify the best strategy among different alternatives [20-25]. Using the simulation, it may be possible to estimate the system behavior in some conditions and parameters scenario, evaluating many performance indicators and estimating the most appropriate solution.

In this paper, all the simulation scenario refers to an 8 hours work shift. The measured and examined performance have been respectively the "mean waiting time" of jobs before their acceptance on a machine and the production system "throughput" which measure the rate at which the entire system complete the assigned jobs per unit of time. While the mean waiting time of jobs must be minimized, the throughput of the production system must be maximized.

Using the OptQuest optimization algorithm from OptTEK®, two optimizations have been performed for each approach, allowing to obtain the value of the optimal parameter for both models.

In addition, the optimizations provide the corresponding estimated values for the jobs mean waiting time and the throughput of the production system. These values are shown in Table 3.

Table 3. Key Performance value obtained from the optimization with the optimal parameter value founded

Key Perfomance	MAS Approach	*Nasiri et al.* Approach
Mean Waiting Time (min.)	11.6196	14.1320
Throughput (pcs.)	58.948	57.904
Work In Progress (pcs.)	7.16	5.44

Using the Mean Square Pure Error (MSPE), fully explained in [21], we estimated the number of replications to be performed in order to remove the simulator variance. The sufficient number of replications we observed for both approaches is 250.

With this regard, using 250 simulation results for each parameters combination, an Analysis of Variance (ANOVA), was executed for the performances comparison between both models. In this tests, we considered as a null hypothesis: "The means of the two models are equal", with a confidence level of "α = 0.95". Obviously, the alternative hypothesis is: "The means of the two models are different".

The table 5 shows that null hypothesis for the Mean Waiting Time may be refused with a level of confidence of 99,9% while the Table 6 shows that the null hypothesis for throughput may not be refused with a level of confidence of 80,3%.

We may conclude, with a very high-level of confidence, that the MAS approach decreases significantly (about the 18 %) the Mean Waiting Time of jobs [Figure 5], but we may not say the same thing for the increase of throughput [Figure 6] because the null hypothesis may not be refused.

Table 4. One-way ANOVA Mean Waiting Time between MAS and *Nasiri et al.* approach.

Source	DF	Adj SS	Adj MS	F-Value	P-Value
Method	1	856.4	856.44	26.96	0.000
Error	498	15818.9	31.76		
Total	499	16675.3			

Table 5. One-way ANOVA Throughput between MAS and *Nasiri et al.* approach.

Source	DF	Adj SS	Adj MS	F-Value	P-Value
Method	1	34.30	34.32	1.67	0.197
Error	498	10241.20	20.56		
Total	499	10275.50			

5. Conclusions

This work, through the use of FIPA-CNIP protocol and the Multi-Agent System (MAS), presents an innovative way to solve the NP-Hard Open Job Shop Scheduling Problem (OJSSP) under uncertainty. The contributions of the introduced model are dual: first of all, the MAS architecture allow to divide the unique scheduling problem of a typical Open Job-Shop into a series of simpler problem. Secondly, the introduced composite dispatching-rule is more realistic and effective of the common scheduling technique, allowing to use the three introduced weight parameters as adjustment knobs of the scheduler, in respect of the expected job typology mix.

In addition, as it is possible to observe from the comparison between the MAS approach and the *Nasiri et al*. [17] ones, the introduced approach showed a consistent decrease (about 18 %) of the jobs mean waiting time with an outstanding confidence level. At the same time, it is not possible to say the same about the Production Throughput index, in which the ANOVA results does not allow to register a consistent and stable improvement of the proposed approach. In any case, the advantages of the introduced dispatching rule are yet sufficient to justify future research investment in the MAS approach to the scheduling. In this sense, it is necessary to further investigate about the meaningfulness of the weight parameters inside the introduced dispatching rules, investigating the effect of each parameter on the priority dispatching value. In addition, would be interesting and useful deeply investigate the resilience of the MAS approach in respect to the *Nasiri et al.* [17] ones and also to other method established in literature.

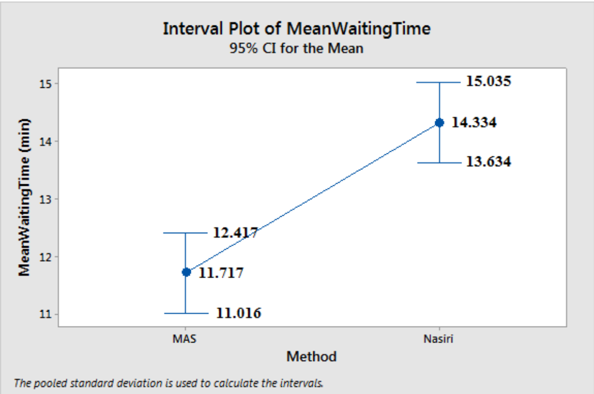

Fig.5 – Mean Waiting Time Interval Plot

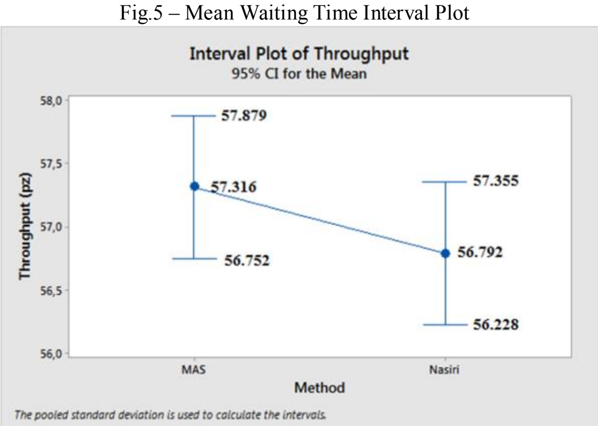

Fig.6 – Throughput Interval Plot

References

[1] Witkowski T, Antczak P, Antczak A. Solving the Flexible Open-Job Shop Scheduling Problem with GRASP and Simulated Annealing. Int Conf on Artificial Intelligence and Computational Intelligence; 2010.

[2] Garey MR, Johnson DS, Sethi R. The complexity of flow shop and job shop scheduling. Mathematics of Operations Research; 1976;1:117–129.

[3] Liaw C. A hybrid genetic algorithm for the open shop scheduling problem. European Journal for Operational Research; 2000;124:28–42.

[4] Wang B, Yang X, Li Q. Genetic simulated annealing algorithm for robust job shop scheduling. Fuzzy info and Eng; 2009:817–827.

[5] Braglia M, Grassi A. A new heuristic for the flowshop scheduling problem to minimize makespan and maximum tardiness. Int J of Production Research; 2009;47:273-288.

[6] Brucker P, Hurink J, Jurisch B,Wöstmann B. A branch & bound algorithm for the open-shop problem. Discrete Applied Mathematics; 1997;76:43-59.

[7] Low C, Yeh Y. Genetic algorithm-based heuristics for an open shop scheduling problem with setup, processing, and removal times separated. Robotics and computer integrated manufacturing; 2009;25:314-322.

[8] Sha DY, Hsu C Y. A new particle swarm optimization for the open shop scheduling problem. Computers and Operations Research; 2008;35, 3243–3261.

[9] Zhang R, Wu C. A simulated annealing algorithm based on bottleneck jobs for the open shop scheduling problem. Proc of the 7th World Congress on Intelligent Control and automation; 2008.

[10] Wooldridge M. An Introduction to MultiAgent Systems. 2nd Edition: Wiley Publishing; 2009.

[11] Shen W, Wang L, Hao Q. Agent-based distributed manufacturing process planning and scheduling: A state-of-the-art survey. IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.; 2006;36:563-577.

[12] Ouelhadj D, Petrovic S. A survey of dynamic scheduling in manufacturing systems. J. Scheduling; 2009;12:417-431.

[13] Leitao P, Restivo F. A holonic approach to dynamic manufacturing scheduling. Comput.-Integr. Manuf.; 2008;24:625-634.

[14] Guizzi G, Vespoli S, Santini S. On the architecture scheduling problem of Industry 4.0. Proceedings of the 3rd INCOSE Italia Conference on Systems Engineering; 2017;2010:94-100.

[15] Improta G, Triassi M, Guizzi G, Santillo LC, Revetria R, Catania A, Cassettari L. An innovative contribution to health technology assessment. Studies in Computational Intelligence; 2012;431:127-131.

[16] Smith RG. The Contract Net Protocol: High-Level Communication and Control in a Distributed Problem Solver. IEEE Transaction on Computer; 1980;205C:1104-1113.

[17] Nasiri MM, Yazdanparast R, Jolai F. A simulation optimization approach for real-time scheduling in an open shop environment using a composite dispatching rule. Inter J of Comp Int Manuf; 2017;30:1239-1252.

[18] SC00029 FIPA Contract Net Interaction Protocol Specification. http://www.fipa.org/specs/fipa00029/XC00029G.html

[19] Huget MP, Odell J. Representing agent interaction protocols with agent UML. Lecture Notes in Computer Science; 2015;3382:16-30.

[20] Di Bona G, Falcone D, Silvestri A, Forcina A. IFM Target 2.0: an innovative method to define reliability target for prototype systems. Int J of Adv Manuf Tech; 2018;95:3349-3367.

[21] Mosca R, Cassettari L, Giribone. MSPE e Monte Carlo Pricing Method: tecniche di controllo della convergenza nei modelli finanziari. AIFIRM Magazine; 2010;5:13-26.

[22] Baruah S, Koren G, Mao D, Mishra B, Raghunathan A, Rosier L, Shasha D, Wang F. On the competitiveness of on-line real-time task scheduling. Real-Time Systems;1992;4:125-144.

[23] Revetria R, Guizzi G, Giribone P. Renewable energy source integrated with electric power supply in manufacturing companies: A management model. Int J of Power and Energy Systems; 2015;35:161-166.

[24] Romano E, Guizzi G, Chiocca D. A decision support tool, implemented in a system dynamics model, to improve the effectiveness in the hospital emergency department. Int J of Procurement Manag; 2015;8:141-168.

[25] Di Bona G, Forcina G, Petrillo A, De Felice F, Silvestri A. A-IFM reliability allocation model based on multicriteria approach; 2016;33:676-698.