Strojniški vestnik - Journal of Mechanical Engineering 58(2012)2, 81-92
DOI: 10.5545/sv-jme.2011.029
Paper received: 2011-02-02, Paper accepted: 2011-11-23

A Dynamic Rescheduling Model with Multi-Agent System and Its Solution Method

Fuqing Zhao[1,2,*] – Jizhe Wang[1] – Junbiao Wang[2] – Jonrinaldi Jonrinaldi[3]
[1] School of Computer and Communication, Lanzhou University of Technology, China
[2] Key Laboratory of Contemporary Design & Integrated Manufacturing Technology, Ministry of Education, Northwestern Polytechnical University, China
[3] School of Engineering, Computer Science and Mathematics, University of Exeter, United Kingdom

Dynamic rescheduling problem is an important issue in modern manufacturing system with the feature of combinatorial computation complexity. A dynamic rescheduling model, which is based on Multi-Agent System (MAS), was proposed. The communication and negotiation mechanism between agents were addressed to support the autonomic decision for each individual agent and the multi-agent system. Furthermore, the simulation results in dynamic scheduling accompanying with its perturbation show that the proposed model and the algorithm are effective to the dynamic scheduling problem in manufacturing system.
Keywords: MAS, agent, dynamic scheduling, Contract Net Protocol, negotiation mechanism, perturbation

0 INTRODUCTION

Today's manufacturing businesses are facing immense pressures to react rapidly and robustly to dynamic fluctuations in demand distributions across products and changing product mix. Traditional manufacturing systems and approaches to production, involving sequential stages from manufacturing systems design, construct, and setup in the preparation phase to production planning, scheduling, and control in the operational phase, can be challenging in satisfying the requirement of the variation. Efficient and practical methods for scheduling and optimization technology are the key to improve the productivity and efficiency of a manufacturing plant [1]. The traditional scheduling and optimization process, which always deals with a clear schedule and a fixed processing time, while for the actual processing problem, there are many uncertain factors, for example, changes in processing time, product demand, delivery, equipment failure, resources and production variations. The dynamic interference of these factors causes that the original dynamic scheduling can not be implemented successfully. Therefore, the rescheduling model and its solution method are of significant importance for the dynamic scheduling problem [2].

Job shop scheduling is to schedule a set of jobs on a set of machines, which is subject to the constraint that each machine can process one job at most at a given time and the fact that each job has a specified processing order through the machines. It is not only a NP-hard problems, it also has the well-earned reputation of being one of the strong combinatorial problems in manufacturing systems. Recently, two single-machine rescheduling problems with linear deteriorating jobs under disruption was studied by Zhao and Tang [3]. Job shop rescheduling problem was being considered as minimizing the total completion time under a limit of the disruption from the original scheduling. However, little information has been given about the autonomic decision mechanism. A reactive scheduling framework based on domain knowledge and constraint programming was proposed by Novasand Henning [4]. An explicit object-oriented domain representation and a constraint programming (CP) approach to the model were utilized to the modeling and realizing method when unforeseen event occurs. A reactive scheduling methodology for job shop, make-to-order industries, by inserting new orders in a predetermined schedule, was introduced to iteratively update the schedules [5]. By applying local rescheduling in response to schedule disruptions was presented to reduce the size of the regarded problems by applying methods of partial rescheduling in literature [6]. Mehta [7] processed the way to absorb the random failure of the disturbance proposed by the appropriate method of inserting new orders in idle time. Kim [8] proposed a flexible production environment, which can handle processing of planning and shop scheduling with symbiotic genetic algorithm. Petrovic [9] used the fuzzy method to study the re-scheduling for the job shop problem facing uncertain disruptions. A genetic algorithm for multi-processor task with resource and timing constraints was put forward to solve the scheduling problem in the manufacturing environment with uncertain disruptions [10]. Wang [11] considered the uncertainty of the impact of events as a set of random changes in the time period for the assembly planning problem which is based on semantic

*Corr. Author's Address: School of Computer and Communication, Lanzhou University of Technology, Lanzhou 730050, China, fzhao2000@hotmail.com

modeling approach. Goncalves [12] proposed a hybrid genetic algorithm for the job shop scheduling problem with reactive scheduling method. Wong et al. [13] introduced multi-agent system as a platform for the dynamic shop floor scheduling problem.

With the aim of enhancing the flexibility of manufacturing systems and achieving optimization of solutions to constantly respond to increasing rates of change in demand patterns and product mixes, rescheduling approaches and production scheduling options should be taken into consideration simultaneously, and evaluated and optimized dynamically. In this way, constrains from both functions can also be fulfilled concurrently and an optimum integrated plan and schedule can then be produced. Mes et al. [14] proposed a distributed agent-based solution to real-time, dynamic transport scheduling problems, which has the advantages of less sensitive to fluctuations in demand or available vehicles than more traditional transportation planning heuristics (Local Control, Serial Scheduling) and provides a lot of flexibility by solving local problems locally. Shen et al. [15] review the research literature on manufacturing process planning, scheduling as well as their integration, particularly on agent-based approaches to the integration of the difficult problems. A class of dynamic scheduling problems characterized by a just-in-time objective was addressed in literature [16], an on-line scheduling heuristic based on a multi-agent architecture was also presented. Kemppainen [17] presented the method of the coordinating power of priority scheduling when customers request different response times and suppliers do their best to fulfill the customer expectations, especially if enforced with different pricing. Pfeiffer et al. [18] presented a simulation-based evaluation technique for the testing, validation and benchmarking of rescheduling methods. Certain stability-oriented evaluations of periodic and hybrid rescheduling methods are described for both single- and multi-machine (job-shop) cases. Cauvin et al. [19] proposed an approach to minimize the impact of disrupting events on the manufacturing scheduling and control system, which is based on a cooperative distributed problem solving approach supported by a multi-agent system framework.

However, few attempts have been done on the universal communication and negotiation mechanism for the dynamic rescheduling problem and corresponding solution approach. We aim to construct a universal dynamic rescheduling model, which is based on Multi-Agent System (MAS), for the job shop scheduling problem in manufacturing systems.

This paper is organized as follows. In Section 1, the rescheduling model which is based on MAS is given. Section 2 describes the detailed functions of agent. The communication and negotiation processes and steps are introduced in Section 3. In Section 4, the proposed model and approach are validated using the popular benchmark functions. Finally, Section 5 concludes the paper with an outlook on future work.

1 MAS-BASED RESCHEDULING MODEL

In dynamic shop scheduling environment, the job shop problem can be described as: in a processing unit or system, n jobs need to be processed on m machines, every job J_i ($1 \le i \le n$) has n_i process O_{ij} ($1 \le i \le n$, $1 \le j \le n$) which is needed to processing. Set machine tool with a collection of M, then each process O_{ij} can either be processed by the concentration of machine tools M_{ij} or can be processed in one machine, where $M_{ij} \subseteq M$. If $M_{ij}=M$, the scheduling problem is a completely flexible scheduling problem; if $M_{ij} \subset M$, it is a local scheduling problem with flexible strategy [20].

In scheduling operation, when one machine failure occurs, all machines need to execute the operation of rescheduling of the operation on the predetermined operation processes. As rescheduling model is the corresponding evolution process to the initial scheduling model, the initial problem modeling is as follows.

$$\min \max \left\{ c_{is} \middle| i \in I \right\},$$

S.t.

$$s_{ij+1} \ge s_{ij+1} + p_{ij},\ i \in I,\ J \in \{1, \ldots s-1\}, \tag{1}$$

$$(m_{i_1 j} \ne m_{i_2 j}) \vee (s_{i_1 j} \ge c_{i_1 j} \vee s_{i_2 j} \ge c_{i_2 j}), \tag{2}$$

$$i_1, i_2 \in I,\ i_1 \ne i_2,\ j \in J, \tag{3}$$

$$c_{ij} = s_{ij} + p_{ij},\ i \in I,\ j \in J, \tag{4}$$

$$s_{i_1} \ge 0,\ s_{ij} \ge \sum_{k=1}^{j-1} p_{ik},\ i \in I,\ j \in \{2, \ldots, s\}, \tag{5}$$

$$m_{ij} \in R_j = \left\{ r_{jl}, \ldots, r_{jl_j} \middle| j \in J \right\},\ i \in I, \tag{6}$$

where i is the workpiece number and $i \in I = \{1, \ldots, n\}$, j is the level number and $j \in J = \{1, \ldots, s\}$, r_{jl} is the machine number, s_{ij} is the start time of initial scheduling, m_{ij} is the start machine, p_{ij} is the processing time of the workpiece.

In the above model, Eq. (1) shows that the optimization goal of the scheduling problem is minimum of total process time $C_{\max}$.

Eq. (2) shows the operation of the timing constraints. Eq. (3) shows that if two jobs are processed on the same machine, then they can not be processed at the same time. Eq. (4) shows that once the processing for the workpiece starts, it can not be interrupted until it will be finished. Eqs. (5) and (6) show operation started with variable time window and the variable interval of processing machinery, respectively.

It is supposed that the machine $r_{j_d l_d}$ disruptions at the time interval $[t_b, t_e]$, the initial scheduling begins to adjust in t_b, and the initial scheduling will alter to the dynamic rescheduling:

$$\max\ f = \frac{\sum_{i\in I}\sum_{j\in J} w_{ij}\delta 1_{ij}}{\sum_{i\in I}\sum_{j\in J} w_{ij}} + \frac{\sum_{i\in I}\sum_{j\in J} v_{ij}\delta 2_{ij}}{\sum_{i\in I}\sum_{j\in J} v_{ij}}, \quad (7)$$

S.t.

$$(m_{i_1 j} \neq m_{i_2 j}) \vee (s_{i_1 j} \geq c_{i_1 j} \vee s_{i_2 j} \geq c_{i_2 j}),$$
$$i_1, i_2 \in I, i_1 \neq i_2, j \in J,$$
$$c_{ij} = s_{ij} + p_{ij},\ i \in I, j \in J, \quad (8)$$
$$m_{ij} \in R_j = \left\{r_{j_1}, ..., r_{jl_l} \,\middle|\, j \in J\right\}, \mathrm{i} \in I,$$

$$(m'_{ij} \neq r_{j_d l_d}) \vee (s'_{ij} \geq t_e), i \in I, j \in J,$$
$$(s'_{ij} \geq t_b), i \in I, j \in J. \quad (9)$$

Eq. (7) shows that the objective of the rescheduling is to maximize the time for the adjustment arrangement.

$\delta 1_{ij}$ is the rescheduling operation time of the corresponding alternative scheduling operation for workpiece o_{ij} :

$$\delta 1_{ij} = \frac{\max\left\{\min\left\{c'_{ij}, c_{ij}\right\} - \max\left\{s'_{ij}, s_{ij}\right\}, 0\right\}}{p_{ij}},$$

where $\delta 2_{ij}$ is the dynamic rescheduling decision varible when selecting the alternative for workpiece o_{ij} :

$$\delta 2_{ij} = \begin{cases} 1, & m'_{ij} \in M_{ij} \\ 0, & other \end{cases}.$$

Eq. (8) shows the new constraints when mechanical failures occur. Eq. (9) shows the beginning of the operation of the new interval for the start time of process $o_{ij} \cdot o_{ij}$ means the process step j of workpiece i, w_{ij}, v_{ij} are operating weight of the workpiece and the machine time consistency of weight, respectively. s'_{ij} means the starting time of the operating parts in rescheduling, m'_{ij} is the operating machine for workpiece in rescheduling.

For the rescheduling problem, the structure of the agent can be expressed as: *agent = def* <*Id, Goal, Act, Rule, L*>.

Agent *Id* is the identifier in multi-agent system. Different agent has different agent *Id*.

Goal is the objective of the agent. The goal is keeping the optimization objective of job sequence as optimum or near optimum after inserting a new job to the agent when rescheduling occurs. The goal can be expressed as: $Goal_i = (C^i_{\max}, J^i_1 \to J^i_2 ... \to J^i_{ni})$, where $\left\{J^i_1, J^i_2 ... J^i_{ni}\right\}$ is the union of the current operating jobs and the current order queue $J^i_1 \to J^i_2 ... \to J^i_{ni}$ for machine M_i. $J^i_1 \to J^i_2 ... \to J^i_{ni}$ is the optimum priority sequence or near optimum order of machine M_i. C^i_{max} is the optimal objecive of the machine M_i or near optimal objective value.

Act is the action set of agent in the form of $Act = \{act_1, act_2, ..., act_n\}$, which represents the operation it can be accomplished. Each agent has capabilities of communication and collaboration.

Rule represents the cooperation criterion for the communication between agent and its corresponding agent.

L is the Agent communication language. Different agents communicate with each other with *L*. ACL (Agent communication language) with FIPA rules and modified contract net protocol, which are used in this paper.

2 MAS BASED MODEL FOR DYNAMIC SCHEDULING SYSTEM

Currently, there is a wide range of either commercial or open source agent development tools, called agent platforms, that are compliant with the FIPA specifications. For example JADE, FIPA-OS, ZEUS, GrassHopper and MAST [21], Manufacturing Agent Simulation Tool, which was developed by the Rockwell, Automation Research Center in Prague. The initial idea was to propose and implement the agent based solution of some typical manufacturing task. The material-handling domain has been chosen, especially the task of the transportation of products between manufacturing cells using different means of transport, for instance, the conveyor belts or the AGVs (Automated Guided Vehicles). Munir Merdan et al. [22] use MAST, which has been validated with real-world hardware to strengthen the external validity of the simulation results.

Basically, each of these agent platforms provides a user with a library of JAVA classes (since all of them are programmed in JAVA) that allow to create agents with application-specific attributes and behaviors and with the capabilities to send and receive messages following different FIPA interaction protocols. The vital part for agent platform is the runtime environment. which provides a space for agents to live. The running environment consists of white pages services registering existing agents and their contact addresses, yellow pages services used to register and locate services offered by agents and finally the message dispatching mechanism ensuring the inter-agent communication within the platform as well as among agents hosted at different platforms.

2.1 Function Design of Agent

Traditional rescheduling is generally obtained manually and/or is computer aided in accordance with certain reallocation algorithm [23]. We use the MAS-based intelligent scheduling systems, which collaborate with each other to guarantee the intelligence of machines that utlized MAS as the software of control unit. Therefore, jobs for rescheduling in manufacturing shops can achieve the automation and optimization. The basic structure of improved contract net model consists of Management Agent (MA), Resource Agent, Supervision Agent (SA) and Workpiece Agent.

MA is the core of the scheduling system. İt is mainly responsible for evaluating and scheduling the task entered into the rescheduling system. The information of the task is composed of the host information and the degree of emergency for the concrete task. MA transmits the information to the Resource Agent for processing. Communication between MA and other Agents is shown in Fig. 1.

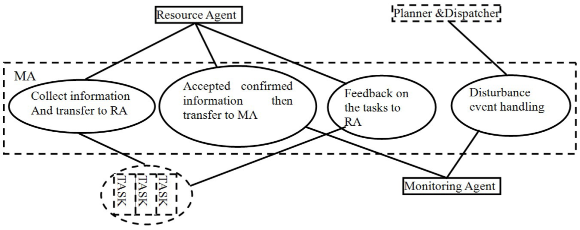

Fig. 1. *Description of Management Agent*

Resource Agent (RA) is responsible for receiving and processing production tasks entering into the shops. And in accordance with the current processing capacity, RA determines whether to perform the task or not. After the decomposition of tasks, the tender will be distributed to Equipment Agent (EA). According to the rules of the agreement, RA lays out a concrete processing planning, then submits it to the SA. In addition, guides the production for EA after obtaining the feedback from SA. The internal schematic for Resource Agent is shown in Fig. 2.

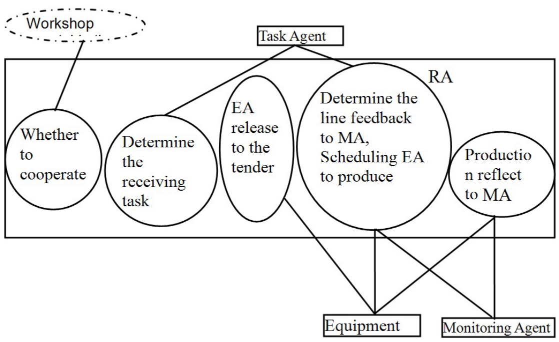

Fig. 2. *Resource Agent internal schematic*

SA is responsible for the simulation for the candidate production planning, which is returned back from the MA. Moreover, SA selects processing route and forwards it to the MA to be performed. Furthermore, SA is responsible for the supervision of Agent equipment failure, the entering of new equipments and the arrival of other emergency tasks. Fig. 3 shows the internal schematic of SA.

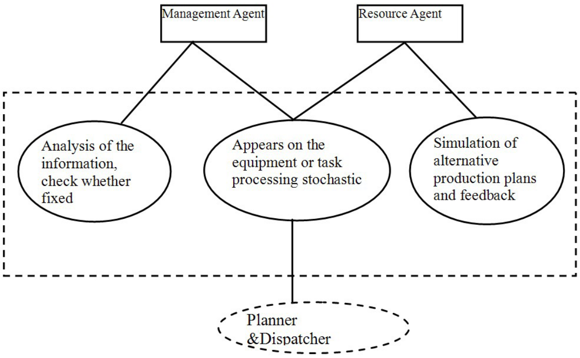

Fig. 3. *Supervision Agent internal schematic*

EA can be considered as a manufacturing unit. Each process unit is administrated by one EA, which is responsible for the corresponding operation management, command transmission for equipment, and collection of processing information, etc. After receiving the information returned from RA, EA evaluates the corresponding equipment and decides whether to tender or not. If EA tenders for a task, it sends its bid according to the operation situation of the equipment. In addition, EA sends the capability of itself to RA. Internal schematic of EA is shown in Fig. 4.

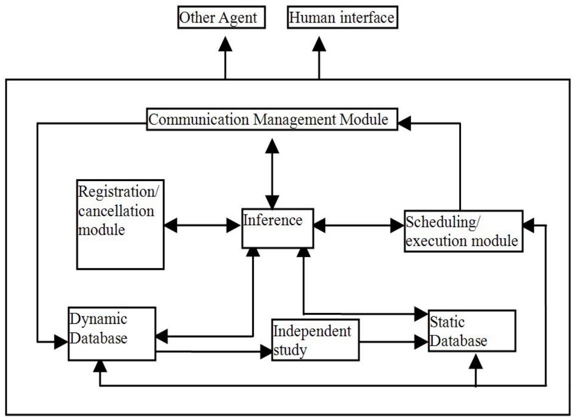

Fig. 4. *Equipment Agent Internal Schematic*

Afterwards, the MA sends a message to Workpiece Agent with the accepted information. Communication primitives in the process can be expressed as:

:Sender(managerAgent@abc:1099/jade)
:Receiver(Equipment@abc:1099/jade)
:Ontology AMS-ontology
:Protocol FIPA-contract-net
:Language FIFA-KQML
:Content "((Issue (taskid(01),surface
Type(plane),machining Type(drilling),number(8),tolerance
(geometic Tol:02dimensional tol:01roughness:02),
deadline(2010.12.01/21:10)))"

RA selects processing tasks in sequence from the waiting tasks to be processed. RA sends the process information as a proposition to EA. The communication primitives can be expressed as:

(CFP
:Sender(Agent-identifier:name resource@abc:abc:1099/jade)
:Receiver(Agent-identifier:name equipment@abc:abc:1099/jade)
:Content(action issues:issuebook:taskli\task01:working
procedure\01\)
:task ready time"2010-12-01 21:20"
:surface roughness 4:dimensional tolerance
time\"60"\:deadline\"2010-12-01 21:20"\)
:Reply-with CFP1
:in Reply-with PROPOSE1
:Language FIPA-KQML
:Ontology scheduling ontology
:Protocol fipa-contract-net
)

Workpiece Agent analyzes the tender received according to the capacity of itself and status of the request, then replies with proposed tender in given deadline. Tender request primitives for Workpiece Agent can be expressed as:

(PROPOSE
:Sender(Agent-identifier:name equipment@abc:1099/jade)
:Receiver(Agent-identifier:name resource@abc:1099/jade)
:Content"((action(bidbook(bidbook
:finishtime\2010-12-01\21:30\)):cost:10:equipment
(Agent-identifier:name equipment@abc:1099/jade)))"
:Reply-with CFP1
:in Reply-with PROPOSE1
:Language FIPA-KQML
:Ontology scheduling ontology
:Protocol fipa-contract-net
)

The proposed communication and the interaction process are implemented on the Java Agent Development Environment (JADE) platform. JADE is a multi-agent system platforms, which conforms strictly to FIPA criteria. The JADE programmer can use JAVA to exploit the system when the agent is built (Administrator Guide, Programmer Guide). Meanwhile, because JADE simplifies the communication process between agents by delivering messages which abide by FIPA criteria (FIPA), the message can also be inserted into the sequenced object to realize the standardization parameter delivery. Furthermore, the yellow function can be directly used because of DF function provided by JADE to guarantee the register for customer system. With AMS and Sniffer tools provided by JADE, users can debug the implementation platform and easily achieve the total functioning of the system. The startup interface is shown in Figs. 5 and 6.

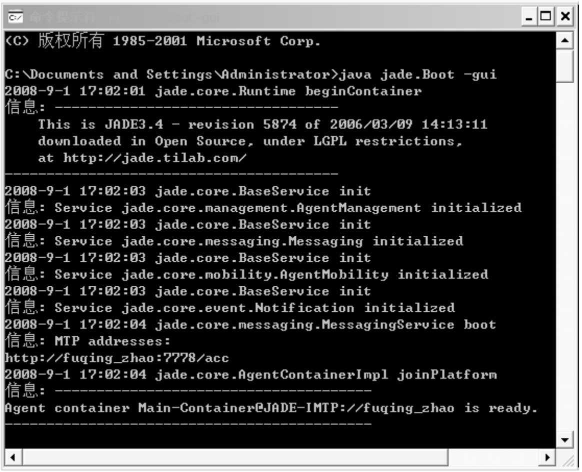

Fig. 5. *Startup interface of JADE platform*

2.2 The Rescheduling Process Based on Contract Net Protocol

MA generates the appropriate contract according to the task order in the task allocation model. The final task allocation is determined by the bidding mechanism through the contract net protocol [24].

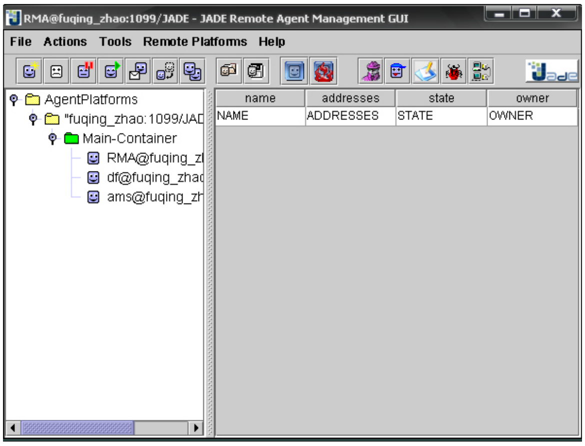

Fig. 6. *GUI Startup interface of JADE platform*

However, given the consultation efficiency and frequent dynamic scheduling in the workshop, in order to improve efficiency, a two-way consultation mechanism for global scheduling is utilized. By introducing the two-way consultation mechanism, the workshop no longer needs to accept bidding information from MA passively. It can take the initiative to inform the MA of rescheduling information for RA and EA in free time to shorten the scheduling time needed. Meanwhile, RA no longer needs to send bidding information to the workshop with broadcast mode unconditionally. In contrast, RA inspects whether there are request submitted from other agent, afterwards, invites bidding from the workshops which have submitted the bid previously. Therefore, the communication between the agent system decreases obviously, along with the improvement of negotiation by the two-way consultation mechanism. Fig. 7 shows the multi agent based dynamic scheduling model with the two-way consultation mechanism.

In multi-agent dynamic scheduling system, any agent in the agent society may be involved in more than one cluster. With on-going clustering and agents becoming involved in multiple compositions, a multi-dimensional cluster negotiation process is illustrated in Fig. 8. Four kinds of agents, such as scheduling agent, RA and SA, etc. are involved in the agent cluster. The interaction can be traced in the JADE platform.

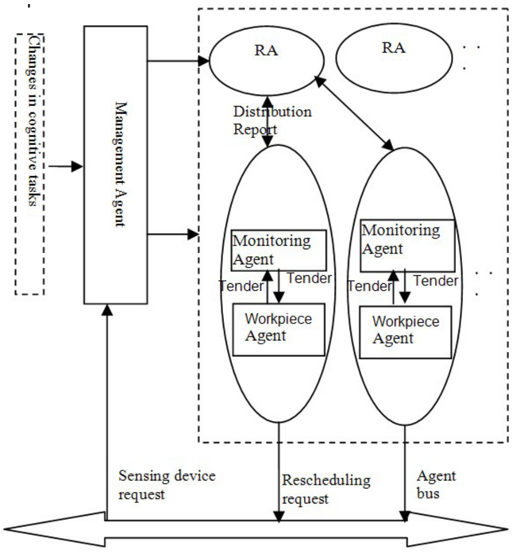

Fig. 7. *Multi Agent Based Dynamic Scheduling Model*

Autonomous negotiation strategy is used in the local Scheduling, which focuses primarily on the negotiation process for one single operating. When the task administration agent obtains the process needed to be perform, it selects a certain agent in the machine queues with specific status to forward notice according to the time of task obtaining.

And it authorizes it to appropriate work piece agent by negotiation. If MA obtains a task at the same time, it starts negotiation randomly. When the machine and the authorized work piece agent accomplish the task which was assigned, RA informs the MA, which is responsible for the current task, of the finished status. In addition, RA updates status of itself, and, transfers to the queues which are waiting for the tasks in the next time intervals. EA alter to idle status at the same time.

2.3 Rescheduling for Emergency Orders

Due to the market fluctuating frequently, new orders arrive from time to time. Therefore, it is of significant importance to arrange new orders efficiently. İn the agent clusters, RA informs MA with conventional methods of starting negotiation mechanism. İf the new order can not be inserted to the scheduling, a certain amount of production planning is to be released by RA. However, the deadline of the order, which is released by RA, must be guaranteed. By iteratively releasing process, until emergency orders are to be rescheduled successfully. The flow chart is shown on the left in Fig. 9.

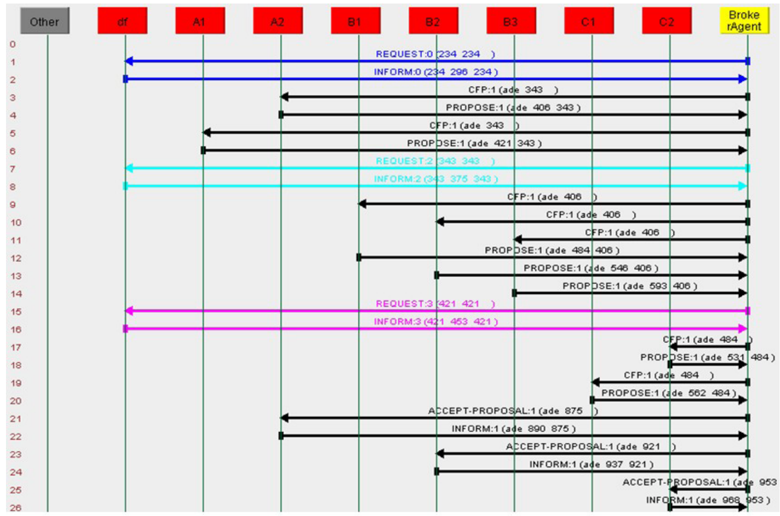

Fig. 8. *Multi-agent negotiation and interaction process based CNP (simulated result)*

2.4 Failure of the Machine

For the machine in breakdown, the agents system terminate its running operation immediately, and then issue a notice of repairment. Equipment Agent sends the current processing status to RA. RA records the current status, takes the processing task back and examines the alternative EA available. If there is an alternative EA, RA dispatches the process to continue the task. Whereas, if there are no altenatives available, the recalled task is to be bid and reschedule again. The flow chart is shown in middle of Fig. 9.

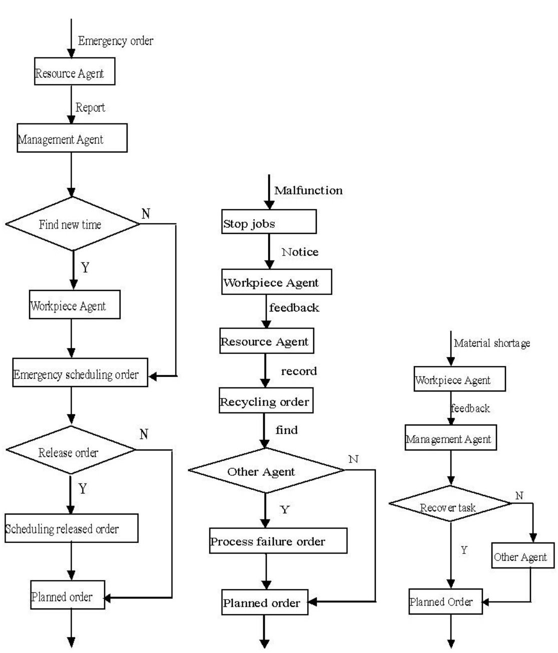

Fig. 9. *Dynamic scheduling process*

For other exceptions: such as the shortage of raw materials, the task can not be completed before deadline. MA is to recall the corresponding tasks in order to bring the rest rear scheduling task in production planning ahead. The unfinished task is to be scheduled until the process constraints satisfy the scheduling request. The flow chart is shown on the right of Fig. 9.

3 SCHEDULING ALGORITHMS FOR DYNAMIC RESCHEDULING MODEL

The process for dynamic rescheduling, which is based on MAS, consists of multi-stage local scheduling. The local scheduling of each stage is carried out under CNP model. The basic algorithm is as follows:

Step 1: MA releases initial bid price PR_i after it receives scheduling information from other agents.

It can be defined as $PR_i=(t_i|T_a/B_a/M_a)$, where t_i is the deadline of answering a bid from other agents. T_a is the time constraints to complete the task. B_a is the spatial constraints. M_a is material relationship constraints.

For an emergency order, once the original work piece delays, the total delay time should be shortened as much as possible. So, the time constraint can be expressed as:

$$T_a = \min[(T_s+t_i),(T_d+T_p)], \qquad (10)$$

$$c_{ik} - p_{ik} + M(1-a_{ihk}) \geq c_{ih},$$

S.t.

$$i = 1, 2, ..., n;\ h, k = 1, 2, ..., m,$$

$$c_{jk} - c_{ik} + M(1 - x_{ijk}) \geq p_{jk},$$

$$i, j = 1, 2, ..., n; k = 1, 2, ..., m,$$

$$c_{ik} \geq 0,$$

$$i, = 1, 2, ..., n; k = 1, 2, ..., m,$$

$$x_{ijk} = 0 \text{ or } 1,$$

$$i, j = 1, 2, ..., n; k = 1, 2, ..., m,$$

where Eq. (10) indicates time constraints. T_s is the time that MA offers the initial bid. T_d is the latest time that all the jobs have been accomplished. T_p is the average expectant operating time. c_{ik} is the finished time for the work piece i processed by machine k. p_{ik} is the machining time for workpiece i processed by machine k. a_{ihk} and x_{ijk} are coefficient and the decision variable, respectively.

Step 2: EA answers a bid from another agent and a bid for the tender. EA evaluates whether it can first satisfy the resource constraint or not. Then, give evaluated bid price″ $PR_j=(a_j | T_c, M_c)$, where a_j is the wait time committed by itself. T_c is the earliest beginning time which is produced by the EA after assessment. If the EA can not satisfy T_a and B_a, or occupied by one process constraints, then EA gives up bidding. If T_c is being the idle status, then EA answers the tender to RA actively and schedules in idle time to save scheduling time.

Step 3: RA assesses the bid from EA and then authorize to the EA outperformed. MA evaluates all bids returned from all the bids with $\min[(T_c + T_p), M_c]$. Select the best EA to authorize, namely, considering the earliest start time, the workpiece capacity and efficiency of the EA.

Step 4: Perform the operation process. The EA which was authorized is to perform job tasks. During the process, certain failures are possible to occur. However, if the system works well, MA calculates the total process time, and then sends the information collected by it to EA. If there are certain failure occurs, EA reports the status to SA, terminates the operation, and transfers the task needed to be rescheduled to the fault repairment negotiation process.

4 SIMULATION RESULTS AND DISCUSSION

In this paper, the actual production line simulation model was used to testify the efficiency of the method which is proposed in our paper. The system is running on Intel Pentium (R) 4 CPU 2.93 GHz processor, 512 MB of memory, the operating system was Windows 2003 server, JADE version is 3.9.

In order to testify the efficiency of the rescheduling model and the scheduling algorithms, one machine shop is utilized as a test case. The workshop we selected is composed of 5 parts, 4 processes and 8 machines. The parts and machines were mapped as resource agent and workpiece agent in MAS model as Fig. 2. The UML functional structure of Resource Agent in the model is shown as Fig. 10. The model and the communication process of SA, EA, MA and Workpiece Agent (WA) were modeled as in section 2.

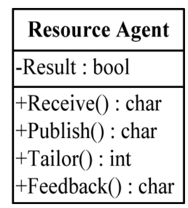

Fig. 10. *Resource Agent*

The workshop has the capacity for planing, milling, turning, drilling and other processes. In addition, there are multi-functional machines, two different specification planers, two different specifications lathes, one milling machine and one multi-function machine in the workshop. A multi-function machine can accomplish processes for drilling, planing operations. The relationship between a process for machines is shown in Table 1 (in the table, 1 indicates that the machine can complete the process, otherwise value is 0). Process sets of the workpiece are shown in Table 2. The processing time for each process is shown in Table 3. Two cases were simulated: one case is the deadline for all parts which have no strict requirement. The other case is that all the parts ordered have tense deadline. In the experiment, for the first case, a deadline is set as the average processing cycle under FIFO scheduling rules. For the second case, the set deadline scale as 1:1.2.

The obtained results by our MAS based model and algorithm were compared with those obtained under the rules of FIFO and EDD. The performance comparison results are shown in Table 4.

In Table 1, A to H represents 8 machines. Among these machines, A and B are Planer, C is a Milling machine, D, E and G are Lathe, the others are Multi-function machine. ‘1’ means the process can be completed, and ‘0’ means it can not be completed. The processing sequence of the workpiece in the simulation model is described in detail in Table 2. In Table 3, the machine is the concrete machine equipment used for the processing sequences. *ID* is the number of the job in Table 2. The corresponding figures signify the process time in corresponding machines. The simulation results in the normal production line and emergency production line are shown in Table 4.

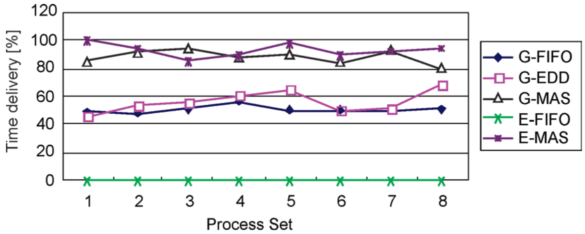

Fig. 11. *Time delivery comparison in normal state*

In Figs. 11 to 14, FIFO means first input first output. EDD means earliest due date. G-FIFO is FIFO in General status; G-EDD is EDD in General status; G-MAS means in General MAS status; E-FIFO means in Emergency FIFO status; E-MAS means in Emergency MAS status; G-F-FIFO means in General Failure FIFO status; G-F-MAS means in General Failure MAS status; E-F-FIFO means in Emergency and Failure FIFO status; E-F-MAS means in Emergency and Failure MAS status.

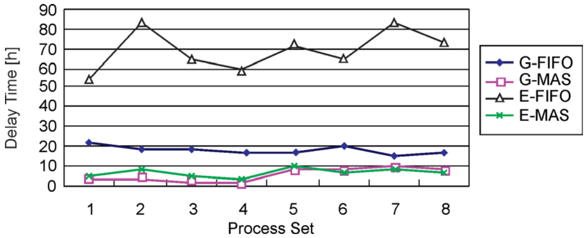

Fig. 12. *Delay time comparison in normal state*

As can be seen from Table 4, Figs. 11 and 12, the proposed consultation mechanism can reduce the weighted average delay in delivery of products. Moreover, it can shorten product delivery time. Hence, the model and the algorithms utilized are effective to the rescheduling problem.

For the problem of equipment failure, it is assumed that there is one machine with a daily failure of 12 h and simulation time is one month. Other test conditions are the same with previously tested cases. FIFO rules is not be used to process equipment failure, while MAS based consultation and negotiation mechanism is to reschedule the rest processes. Simulation and performance results are shown in Table 5, Figs. 13 and 14.

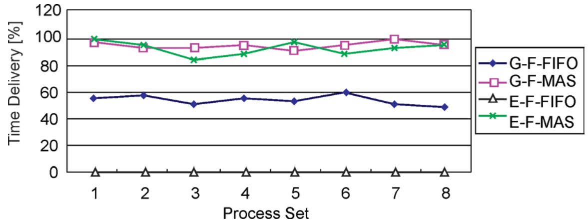

Fig. 13. *Time delivery comparison in failure model*

Under the circumstance of equipment failure, results indicate that local autonomic negotiation mechanism we utilized can alleviate the effection of equipment failure for the production line in manufacturing shops.

The case-study utilized is a real industrial problem aiming at evaluating reschedules in a large job-shop environment with MAS simulation platform. The simulation architecture presented in the previous sections constituted the stochastic evaluation environment in which absolute evaluation of static schedules was performed. The case study elaborated concerned a factory that produces mechanical products by using machining and welding resources, assembly and inspection stations and some highly specialized machines. Production is performed in a make-to-order manner where deadline is an absolute must, even regarding unpredicted orders. Since quality assurance is a key issue, tests may result in extra adjustment operations. The process for dynamic rescheduling, which is based on MAS, consists of multi-stage local scheduling. The local scheduling of each stage is carried out under the CNP model. And the Multi-agent negotiation and interaction process based on CNP is in full-duplex communication manner, which is a distinct difference between our approach and the methods in [25].

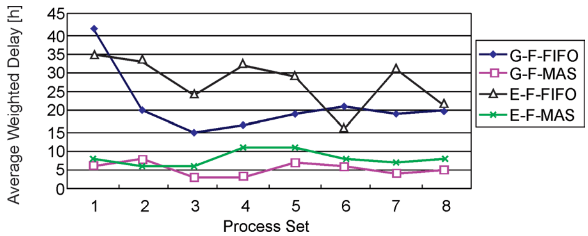

Fig. 14. *Average weighted delay in failure model*

From Table 4, Figs. 11 and 12, in the two statuses of Normal production line and Emergency production line, the scheduling accurate delivery of MAS have advantages over those obtained under the rules of FIFO and EDD with less average weighted time.

Table 1. *Relationship between the machine and process*

Process	Machine							
	A (Planer1)	B (Planer2)	C (Milling)	D (Lathe1)	E (Lathe2)	F (Multi-function)	G (Lathe)	H (Multi-function)
P1	1	1	0	1	0	1	1	0
P2	0	0	1	0	0	0	0	1
P3	0	0	0	1	1	0	1	0
P4	0	0	1	0	0	1	1	0

Table 2. *Equipment and process*

Piece ID number	1 procedure	2 procedure	3 procedure	4 procedure
j1	Plane(id:1)	Milling (id:2)	Diamond (id:4)	Car (id:3)
j2	Car (id:3)	Diamond (id:4)	Milling (id:2)	Plane (id:1)
j3	Milling (id:2)	Plane (id:1)	Car (id:3)	Diamond (id:4)
j4	Diamond (id:4)	Plane (id:1)	Milling (id:2)	Car (id:3)
j5	Milling (id:2)	Diamond (id:4)	Plane (id:1)	Car (id:3)

Table 3. *Procedure processing time*

ID	Machine								
	Plane			Milling		Car		Diamond	
	A	B	C	C	G	D	E	F	H
j1	3	4	6	10	8	2	3	5	9
j2	7	8	8	5	6	6	7	6	4
j3	2	3	3	3	5	4	5	3	3
j4	8	9	9	2	7	11	12	7	7
j5	3	5	6	5	3	5	6	8	6

Table 4. *Simulation results under emergency by MAS model*

Production Line Status	Scheduling rules	Time delivery [%]								Weighted average delay [h]							
		A	B	C	D	E	F	G	H	A	B	C	D	E	F	G	H
General	FIFO	49	48	51	57	50	50	50	51	21	18	18	16	16	20	15	16
	EDD	46	54	55	61	65	50	52	68	/	/	/	/	/	/	/	/
	MAS	85	91	94	88	90	84	93	80	3	3	1	1	8	7	9	8
Emergency	FIFO	0	0	0	0	0	0	0	0	55	84	65	59	71	65	84	74
	MAS	100	94	85	89	98	89	92	94	5	7	5	3	10	6	8	6

Table 5. *Simulation results under equipment failure by MAS model*

Production Line Status	Scheduling rules	Time delivery [%]								Weighted average delay [h]							
		A	B	C	D	E	F	G	H	A	B	C	D	E	F	G	H
General Failure	FIFO	55	58	51	55	54	59	50	49	41	20	15	17	19	21	19	20
	MAS	96	92	92	95	90	95	99	94	6	8	3	3	7	6	4	5
Emergency Failure	FIFO	0	0	0	0	0	0	0	0	35	33	24	32	29	16	31	121
	MAS	99	94	88	89	98	89	92	94	8	6	6	11	11	8	7	8

The Simulation results under equipment failure by MAS model was shown in Table 5. In Figs. 13 and 14, it is seen that the MAS method we proposed in this paper shows a high efficiency in adjusting the jobs to other machines available at the required time with the less delay and delivery time. In contrast, the adjusted rules adopted by FIFO can not be worked in the fault status.

5 CONCLUSION

Dynamic rescheduling method is widely used in the modern production plant. In this paper, the Contract Net Protocol, which is based on MAS, is introduced to the rescheduling of the workshop environment. It is a new way of solving the communication and negotiation problem in this field. After fully considering the effection of the equipment failure and repairment in the process of production, the complex dynamic rescheduling process is to be divided into the communication and negotiation processes of multi-agents. Therefore, the capability of autonomic decision for tackling the unexpected events, which occur in the production, is extended. By simulation in the actual production workshop, the model and algorithm, which are based on MAS, were identified as effective to the rescheduling problem in the manufacturing system.

It is worth pointing out that the test cases studied in this work are not very many. We will explore the efficiency of our model and approach on those problems with a larger number of decision variables in the future. The future work should also includes studies on the process specific interaction between agents in multi agent area and how to extend our model and algorithm based on MAS to solve constrained or discrete multiobjective optimization problems.

6 ACKNOWLEDGEMENT

This work is financially supported by the National Natural Science Foundation of China under Grant No. 61064011. And it was also supported by Scientific research funds in Gansu Universities,Science Foundation for the Excellent Youth Scholars of Lanzhou University of Technology, Educational Commission of Gansu Province of China, Natural Science Foundation of Gansu Province, and Returned Overseas Scholars Fund under Grant No. 1114ZTC139, 1014ZCX017, 1014ZTC090, 1114ZSB091, and 1014ZSB115, respectively.

7 REFERENCES

[1] Ayton, H., Lawley, M., Mckay, K. (2005). Executing production schedules in the face of uncertainties: a review and some future direction. *European Journal of Operation Research*, vol. 161, no. 1, p. 86-110, DOI:10.1016/j.ejor.2003.08.027.

[2] Rossi, A., Boschi, E. (2009). A hybrid heuristic to solve the parallel machines job-shop scheduling problem. *International Journal of Production Research,* vol. 40, no. 2, p. 118-127.

[3] Zhao, C.L., Tang, H.Y. (2010). Rescheduling problems with deteriorating jobs under disruptions. *Applied Mathematical Modeling,* vol. 34, no. 1, p. 238-243, DOI:10.1016/j.apm.2009.03.037.

[4] Novas, J.M., Henning, G.P. (2010). Reactive scheduling framework based on domain knowledge and constraint programming. *Computers & Chemical Engineering,* vol. 34, no. 12, p. 2129-2148, DOI:10.1016/j.compchemeng.2010.07.011.

[5] Gomes, M.C., Barbosa-Povoa, A.P., Novais, A.Q. (2010). A discrete time reactive scheduling model for new order insertion in job shop, make-to-order industries. *International Journal of Production Research,* vol. 48, no. 24, p. 7395-7422, DOI:10.1080/00207540903433858.

[6] Kuster, J., Jannach, D., Friedrich, G. (2010). Applying local rescheduling in response to schedule disruptions. *Annals of Operations Research,* vol. 180, no. 1, p. 265-282, DOI:10.1007/s10479-008-0488-x.

[7] Mehta, S.V., Uzsoy, R.M. (1998). Predictable scheduling of a job shop subject to breakdowns. *IEEE Transactions on Robotics and Automation,* vol. 14, no. 3, p. 365-378, DOI:10.1109/70.678447.

[8] Kim, Y.K., Park, K., Ko, J. (2003). A symbiotic evolutionary algorithm for the integration of process planning and job shop scheduling. *Computers and Operations Research,* vol. 30, no. 8, p. 1151-1171, DOI:10.1016/S0305-0548(02)00063-1.

[9] Petrovic, D., Duenas, A. (2006). A fuzzy logic based production scheduling/rescheduling in the presence of uncertain disruptions. *Fuzzy Sets and Systems*, vol. 157, no. 16, p. 2273-2285, DOI:10.1016/j.fss.2006.04.009.

[10] Cheng, S.C., Shiau, D.F., Huang, Y.M. (2009). Dynamic hard-real-time scheduling using genetic algorithm for multiprocessor task with resource and timing constraints. *Expert System with Application*, vol. 36, no. 1, p. 852-860, DOI:10.1016/j.eswa.2007.10.037.

[11] Wang, H., Xiang, D., Duan, G.H. (2009). Assembly planning based on semantic modeling appoach. *Computer in Industry*, vol. 58, no. 3, p. 227-239.

[12] Goncalves, J.F., Mendes, J.J., Resende, M.G. (2005). A hybrid genetic algorithm for the job shop scheduling problem. *European Journal of Operational Research,* vol. 167, no. 1, p. 77-95, DOI:10.1016/j.ejor.2004.03.012.

[13] Wong, T.N. Leung, C.W., Mak, K.L., Fung, R.Y.K. (2006). Dynamic shop floor scheduling in multi-agent manufacturing system. *Expert System with Applications,* vol. 31, no. 3, p. 486-494, DOI:10.1016/j.eswa.2005.09.073.

[14] Mes, M., van der Heijden, M., van Harten, A. (2007). Comparison of agent-based scheduling to look-ahead heuristics for real-time transportation problems. *European Journal of Operational Research*, vol. 181, p. 59-75, DOI:10.1016/j.ejor.2006.02.051.

[15] Shen, W.M. (2006). Agent-based distributed manufacturing process planning and scheduling: a state-of-the-art survey. *IEEE transactions on systems,*

man, and cybernetics-part C: Applications and reviews, vol. 36, no. 4.

[16] Boccalatte, A., Gozzi, A., Paolucci, M., Queirolo, V., Tamoglia, M. (2004). A multi-agent system for dynamic just-in-time manufacturing production scheduling. *Systems, Man and Cybernetics, IEEE International Conference*, vol. 6, p. 5548-5553.

[17] Kemppainen, K. (2007). *Priority scheduling revisited-dominant rules, open protocols, and integrated order management.* Katariina kemppainen and Helsinki school of Economics. A-264, p. 18-28.

[18] Pfeiffer, A., Kadar, B., Monostori, L. (2007). Stability-oriented evaluation of rescheduling strategies, by using simulation. *Computers in Industry,* vol. 58, p. 630-643, DOI:10.1016/j.compind.2007.05.009.

[19] Cauvin, A.C.A., Ferrarini, A.F.A., Tranvouez, E.T.E. (2009). Disruption management in distributed enterprises: A multi-agent modelling and simulation of cooperative recovery behaviours. *International Journal Production Economics*, vol. 122, p. 429-439, DOI:10.1016/j.ijpe.2009.06.014.

[20] Kacem, I., Hammadi, S., Borne, P. (2002). Pareto-optimality approach for flexible job-shop scheduling problems: Hybridization of evolutionary algorithms and fuzzy logic. *Mathematics and Computer in Simulation*, vol. 60, no. 3-5, p. 245-276, DOI:10.1016/S0378-4754(02)00019-8.

[21] Vrba, P. (2003). MAST: Manufacturing Agent Simulation Tool. *Proceedings of IEEE Conference on Emerging Technologies and Factory Automation*, vol. 1, p. 282-287.

[22] Merdan, M. Moser, T., Wahyudin, D., Biffl, S., Vrba, P. (2008). Simulation of workflow scheduling strategies using the MAST test management system. *Control, Automation, Robotics and Vision*, *ICARCV 10th International Conference,* vol. 17-20, p. 1172-1177.

[23] Jiaa, H.Z., Fuh, J.Y.H., Neea, A.Y.C., Zhang, Y.F. (2007). Integration of genetic algorithm and Gantt chart for job shop scheduling in distributed manufacturing systems. *Computer and Industrial Engineering*, vol. 53, no. 2, p. 313-320, DOI:10.1016/j.cie.2007.06.024.

[24] Lim, M.K., Zhang, Z. (2003). A multi-agent based manufacturing control strategy for responsive manufacturing. *Journal of Materials Processing Technology*, vol. 139, no. 1-3, p. 379-38, DOI:10.1016/S0924-0136(03)00535-1.

[25] Baran, M.E., El-Markabi, I.M. (2007). A multiagent-based dispatching scheme for distributed generators for voltage support on distribution feeders. *Power Systems, IEEE Transactions*, vol. 22, no. 2, p. 52-59.