We are IntechOpen, the world's leading publisher of Open Access books
Built by scientists, for scientists

6,900
Open access books available

186,000
International authors and editors

200M
Downloads

Our authors are among the

154
Countries delivered to

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities

CLARIVATE ANALYTICS
BOOK CITATION INDEX
INDEXED

WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Multi agent and holonic manufacturing control

Sugimura Nobuhiro[1], Tehrani Nik Nejad Hossein[2] and Iwamura Koji[1]
[1]Graduate School of Engineering, Osaka Prefecture University, Osaka, Japan
[2]Information and communication Engineering, Toyota Technological Institute, Japan

1. Introduction

The manufacturing industry will continue to be in the future one of the main wealth generators of the world economy (CMV, 1998). In the last decades world has moved towards a global economy, with markets demanding for products with high quality at lower costs, highly customized and with short life cycles, imposing new requirements on manufacturing enterprises, namely in terms of quality, response, agility and flexibility, that are crucial for an enterprise staying in the business.

The traditional manufacturing control systems are not designed to exhibit these capabilities of responsiveness, flexibility, robustness and re-configurability, since they are built upon centralized and hierarchical control structures. They present good production optimization, but a weak response to adopt due to the rigidity and centralization of their control structures. Such centralized hierarchical organization normally leads to situations where the whole system is shutting down by single failures at one point of the system hierarchy (Colombo et al., 2006). The current challenge is to develop collaborative and reconfigurable manufacturing control systems that support efficiently small batches, product diversity, high quality and low costs, by introducing innovative characteristics of adaptation, agility and modularization.

Information and communication technologies, and artificial intelligence techniques, have been used for more than two decades addressing this challenge. Namely, agent-based and holonic manufacturing control seem to be suitable to face these requirements such as modularity, scalability, autonomy and re-usability, since they present decentralization of control over distributed structures. When properly designed and implemented, agent-based control systems result in a performance that is flexible, robust, adaptive and fully tolerant, which are key factors for manufacturing success in the increasingly global marketplace.

In this chapter, we review the different manufacturing control architecture including centralized and distributed. We discuss about intelligent and distributed manufacturing control systems using emerging paradigms, such as multi-agent systems and holonic manufacturing systems (HMSs), and present two case studies about the applications of agent-based manufacturing control systems for process planning and scheduling. The objective of this chapter is to provide an overview about the application of multi-agent systems and holonic manufacturing principles to manufacturing environment, but to focus on the manufacturing control applications.

The chapter is organized as follows. Section 2 reviews the concepts associated with manufacturing control systems, describing the traditional approaches and the distributed

and intelligent ones, namely the agent-based and holonic manufacturing control system. In section 3, we present two case studied including real-time scheduling method for holonic manufacturing system and agent-based dynamic integrated process planning and scheduling in flexible manufacturing system. Finally, we briefly discuss about realizing the agent based manufacturing system by applying the ORiN (Open Robot Interface Network) architecture which recently has been developed for manufacturing automation.

2. Manufacturing control systems

2.1 Traditional approach to manufacturing control problem

The manufacturing control is concerned with managing and controlling the physical activities in the factory aiming to execute the manufacturing plans, provided by the manufacturing planning activity, and to monitor the progress of the product as it is being processed, assembled, moved, and inspected in the factory. Algorithms at this level are used to decide what to produce, how much to produce, when production is to be finished, how and when to use the resources or make them available, when to release jobs into the factory, which jobs to release, job routing, and job/operation sequencing (Baker, 1998).

Due to its complexity, especially the high number of interactions between the different components and the variety of functions executed, manufacturing control systems are traditionally implemented using centralized or hierarchical control approaches, comprising, the following main components: planning, scheduling, execution (i.e. dispatching, monitoring, diagnosis and error recovery) and machine/device control. Each one of these components operates in a specific temporal horizon, ranging from weeks at the strategic level to seconds at the shop floor.

The traditional approach to manufacturing control systems based on centralized or hierarchical control structures, presents good characteristics in terms of productivity, essentially due to its intrinsic optimization capabilities. However, dynamic and adaptive response to change is, currently, the key to competitiveness, and the traditional approaches to manufacturing control typically fall into large monolithic and centralized software packages that are developed and adapted case by case, requiring a huge and expensive effort to implement, maintain or re-configure. In conclusion, they are not adequate because they do not support efficiently the current requirements imposed to manufacturing systems, namely in terms of flexibility, expansibility, agility and re-configurability.

2.2 Agent-based manufacturing control

The multi-agent system paradigm derives from the distributed artificial intelligence (DAI) field, being characterized by decentralization and parallel execution of activities based on autonomous entities, called agents. The definition of agent concept is neither unique nor consensual (Russel & Norvig, 1995; Wooldridge & Jennings, 1995;Wooldridge, 2002). Despite some definitions and interpretations for agents, a suitable definition is: "An autonomous component that represents physical or logical objects in the system, capable to act in order to achieve its goals, and being able to interact with other agents, when it does not possess knowledge and skills to reach alone its objectives". The most important properties of an agent are the autonomy, intelligence, adaptation and co-operation.

There are several agent architectures, ranging from reactive agents, operating in a stimulus–response manner, to deliberative agents characterized by their pro-active reasoning

and goal-oriented behaviour. A well-known deliberative and cognitive agent-type is belief–desire–intention (BDI) architecture, which origin lies in a theory of human practical reasoning, focusing particularly on the role of intentions in practical reasoning (Wooldridge, 2002). In the BDI agents, the decision-making depends on the manipulation of beliefs, desires and intentions of the agents.

A multi-agent system can be defined as a set of agents that represent the objects of a system, capable of interacting, in order to achieve their individual goals, when they have not enough knowledge and/or skills to achieve individually their objectives. Agents organize themselves into a heterarchical structure characterized by the high-level of autonomy and co-operation, being the client–server structure with fixed relations no more applied (Diltis et al., 1991). These features allow a high performance against disturbances, but the global optimization reduced, because the decision-making is local and autonomous, without a global view of the system. The expansibility of the system is easier, and only enough to modify the functioning of some agents or add new agents to the control system.

In the automation and manufacturing domains, an agent can represent physical resources, such as machine tools, robots, auto-guided vehicles (AGVs) and products or logical objects, such as the schedulers and orders (Sepehri & Tehrani, 2005). Using the appropriate distributed control algorithms, individual machines and product agents can make their own manufacturing control decisions relating to resource allocation and coordination, using an automated form of "negotiation". The key benefit of such approach is that if production is disrupted or re-organized in some way, the same negotiation process still takes place, with different machines or products making the decisions, and hence the system is relatively robust to change.

2.3 Holonic manufacturing control

The Holonic Manufacturing System (HMS) is a paradigm that translates into the manufacturing world the concepts developed by Arthur Koestler from living organisms and social organizations. In middle of sixties, Koestler introduced the word holon to describe the basic unit of organization in living organisms and social organizations, based on Herbert Simon theories and on his observations (Koestler, 1969).

The HMS has been proposed and discussed in the HMS consortium, in order to develop a new autonomous distributed architecture of the manufacturing systems, which are applicable to very small batch productions. The HMS consortium has developed the following definitions to help the common understanding of the HMS (Wyns, 1999).
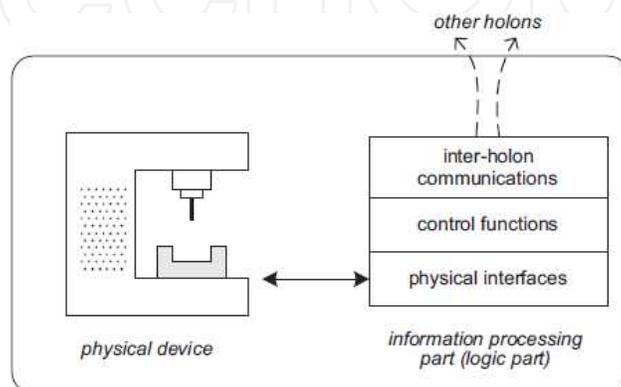

Fig. 1. A physical holon.

- *Holon*: An autonomous and cooperative building block of a manufacturing system for transforming, transporting, storing and/or validating information and physical objects. A holon can represent a physical or logical activity, such as a robot, a machine, an order, a flexible manufacturing system or even an human operator. The holon has information about itself and the environment, containing an information processing part and a physical processing part when the holon represents a physical device (Winkler & Mey, 1994), such as an industrial robot, as illustrated in Fig. 1.
- *Autonomy*: The capability of an entity to create and control the execution of its own plans and strategies.
- *Cooperation*: A process whereby a set of entities develops mutually acceptable plans and executes these plans.
- *Holarchy:* A system of holons that can cooperate to achieve a goal or objective. The holarchy defines the basic rules for cooperation of the holons and thereby limits their autonomy.

Brussel et al. (1998) has proposed PROSA a reference architecture is built around three types of basic holons: order holons, product holons, and resource holons. Each of them is responsible for one aspect of manufacturing control, logistics, technological planning, or resource capabilities respectively. These basic holons are structured using object-oriented concepts like aggregation and specialization. Staff holons can be added to assist the basic holons with expert knowledge. These allow for the use of centralized algorithms and for the incorporation of legacy systems.

3. Agent and Holonic Case Studies in Manufacturing Systems

3.1 Real-time scheduling method for HMS

In this case study, we discuss about a real-time scheduling method for the manufacturing processes in the HMS. The components in the HMS are basically divided into three classes; they are CNC machine tool (CMT) holons, job holons, and coordination holons. We develop a coordination method for the coordination holon to determine a suitable combination of the CMT holons and the job holons based on the utility values. We verify the effectiveness of the proposed real-time scheduling method from the view point of the objective functions of the individual CMT holons and job holons.
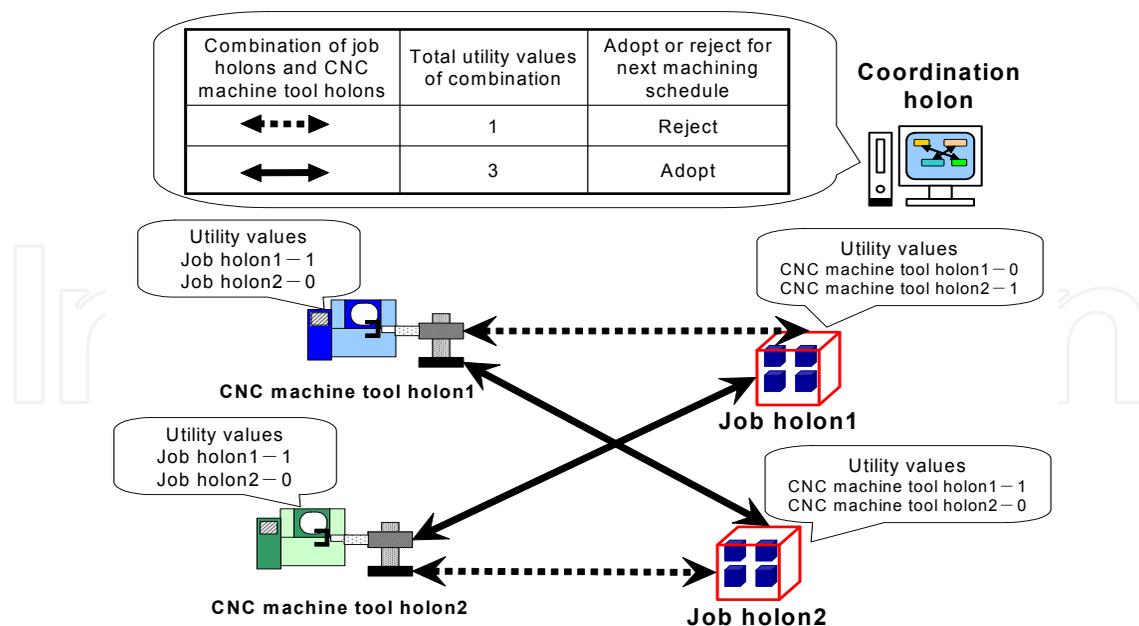
Fig. 2. Real-time scheduling method based on utility values

3.1.1 Basic Architecture of HMS
The holons in the HMS are divided into three classes based on their roles in the manufacturing processes and the scheduling processes.

- <u>CNC machine tool (CMT) holons</u>: They transform the job holons in the manufacturing process. In the scheduling process, they evaluate the utility values for the candidate job holons which carry out the machining process in the next time period.
- <u>Job holons</u>: They are transformed by the CMT holons from the blank materials to the final products in the manufacturing process. In the scheduling process, they evaluate the utility values for the candidate CMT holons which carry out the machining process in the next time period.
- <u>Coordination holon</u>: It carries out the coordination among the holons, and selects a most suitable combination of the CMT holons and the job holons for the machining process in the next time period in the scheduling process.

3.1.2 Real-time scheduling processes based on utility values
It is assumed here that the individual job holons have the following technological information representing the machining process of the jobs.

M_{ik} : k-th machining process of the job holon i. ($i = 1,..,\alpha$),   ($k = 1,..,\beta$).

AC_{ik} : Required machining accuracy of machining process M_{ik}. It is assumed that the machining accuracy is represented by the levels of accuracy indicated by 1, 2, and 3, which mean rough, medium high, and high accuracy, individually.

R_{ikm} : m-th candidate of CMT holon, which can carry out the machining process M_{ik}. ($m = 1,...,\gamma$).

T_{ikm} : Machining time in the case where the CMT holon R_{ikm} carries out the machining process M_{ik}.

W_i : Waiting time until the job holon i becomes idle if it is under machining status.

Objective functions		Objective function values
CMT holon	Efficiency	Σ Machining time / Total time
	Machining accuracy	Σ(Machining accuracy of CMTs – Required machining accuracy of jobs)
Job holon	Flow-time	Σ(Machining time + Waiting time)
	Machining cost	Σ(Machining cost of CMTs)

Table 1. Objective functions of holons

The individual CMT holons have the following technological information representing the machining capability of the resources for the machining process M_{ik}.

MAC_{ikm} : Machining accuracy in the case where the CMT holon R_{ikm} carries out the machining process M_{ik}. MAC_{ikm} is also represented by the level of 1, 2 and 3.

MCO_{ikm} : Machining cost in the case where the CMT holon R_{ikm} carries out the machining process M_{ik}.

W_{ikm} : Waiting time until CMT holon R_{ikm} becomes idle if it is under machining status.

Based on the information above mentioned, a real-time scheduling method based on the utility values shown in Fig. 2 is proposed, to select a suitable combination between the job holons and the CMT holons which carries out the machining process in the next time period. At the time t, all the 'idling' holons have to select their machining schedules in the next time period, as shown in the Fig. 2. The following procedure is proposed for the individual holons to select their machining schedules.

(1) *Retrieval of status data*: The individual 'idling' holons firstly get the status data from the other holons which are 'operating' or 'idling'. The 'idling' holons can start the machining process in the next time period.

(2) *Selection of candidate holons*: The individual 'idling' holons select all the candidate holons for the machining process in the next time period. For instances, the job holon i selects the CMT holons which can carry out the next machining process M_{ik}. On the other hand, the CMT holon j select all the candidate job holons which can be machined by the CMT holon j.

(3) *Determination of utility values*: The individual holons determine the utility values for the individual candidates selected in the second step. For instances, the job holon determines the utility values, based on its own decision criteria for all the candidate CMT holons which can carry out the next machining process.

(4) *Coordination:* All the job holons and the CMT holons send the selected candidates and the utility values of the candidates to the coordination holon. The coordination holon determine a suitable combination of the job holons and the CMT holons which carry out the machining processes in the next time period, based on the utility values. The decision criteria of the coordination holon is to maximize the total sum of the utility values of all the holons.

3.1.3 Evaluation of utility values

It is assumed that the individual holons have one of the objective functions shown in Table 1 for evaluating the utility values.

(1) Efficiency of CMT holons→ ME to be maximized
     ME is the ratio of the total machining time of the CMT holon and the total time after the CMT holon starts the operations. The total time includes both the machining time and the idling time of the CMT holon.
(2) Machining accuracy of CMT holons→MA to be minimized
     MA is the difference between the level of machining accuracy of the CMT holons and the required level of accuracy of the machining process.
(3) Flow-time of job holons→JT to be minimized
     JT is the flow-time of the job holon. *JT* includes the machining time and the idling time of the job holons.
(4) Machining cost of job holons→ JC to be minimized
     JC is the sum of the machining cost of the job holon, which are evaluated from the machining costs of the CMT holons.

The following procedures are provided for the CMT holons to evaluate the utility values. Let us consider a CMT holon j at a time t. It is assumed that $TT_{j \cdot t}$, $ME_{j \cdot t}$, and $MA_{j \cdot t}$ show the total time after the CMT holon j starts its operations, the efficiency, and the evaluated value of machining accuracy of the CMT holon j, respectively. If the CMT holon j selects a candidate job holon i for carrying out the machining process M_{ik}, the efficiency and the evaluated value of the machining accuracy are estimated by the following equations.

$$ME_{j \cdot t+1}(i) = (ME_{j \cdot t} \cdot TT_{j \cdot t} + T_{ikm}) / (TT_{j \cdot t} + T_{ikm} + W_i) \tag{1}$$
$$MA_{j \cdot t+1}(i) = MA_{j \cdot t} + (MAC_{ikm} - AC_{ik}) \tag{2}$$

where, the CMT holon j can carry out the machining process M_{ik} of job holon i ($j = R_{ikm}$).
As regards the job holons, the following equations are applied to evaluate the flow-time and the machining costs, for the case where a job holon i selects a candidate CMT holon j (= R_{ikm}) for carrying out the machining process M_{ik}. It is assumed that $JT_{i \cdot t}$ and $JC_{i \cdot t}$ give the total time after the job holon i is inputted to the HMS and the machining cost, respectively.

$$JT_{i \cdot t+1}(j) = JT_{i \cdot t} + T_{ikm} + W_{ikm} \tag{3}$$
$$JC_{i \cdot t+1}(j) = JC_{i \cdot t} + MCO_{ikm} \tag{4}$$

The objective functions mentioned above have different units. Some of them shall be maximized and others shall be minimized. Therefore, the utility values are normalized from 0 to 1, by applying the following equations.

- **Efficiency of CMT holons**

$$RUV_j(i) = 1 - \{ \max_{i=1, \cdots \tau} [ME_{j \cdot t+1}(i)] - ME_{j \cdot t+1}(i)\} / \{ \max_{i=1, \cdots \tau} [ME_{j \cdot t+1}(i)] - \min_{i=1, \cdots \tau} [ME_{j \cdot t+1}(i)]\} \tag{5}$$

- **Machining accuracy of CMT holons**

$$RUV_j(i) = \{ \max_{i=1, \cdots \tau} [MA_{j \cdot t+1}(i)] - MA_{j \cdot t+1}(i) \} / \{ \max_{i=1, \cdots \tau} [MA_{j \cdot t+1}(i)] - \min_{i=1, \cdots \tau} [MA_{j \cdot t+1}(i)]\} \tag{6}$$

- **Flow-time of job holons**

$$JUV_i(j) = \{ \max_{j=1, \cdots \gamma} [JT_{i \cdot t+1}(j)] - JT_{i \cdot t+1}(j) \} / \{ \max_{j=1, \cdots \gamma} [JT_{i \cdot t+1}(j)] - \min_{j=1, \cdots \gamma} [JT_{i \cdot t+1}(j)]\} \tag{7}$$

- **Machining cost of job holons**

$$JUV_i(j) = \{ \max_{j=1, \cdots \gamma} [JC_{i \cdot t+1}(j)] - JC_{i \cdot t+1}(j) \} / \{ \max_{j=1, \cdots \gamma} [JC_{i \cdot t+1}(j)] - \min_{j=1, \cdots \gamma} [JC_{i \cdot t+1}(j)]\} \tag{8}$$

where, max[$f(x)$] and min[$f(x)$] give the maximum value and the minimum value of $f(x)$

	Resource 1	Resource 2	...	Resource γ
Job 1	a_{11}	a_{12}	...	$a_{1\gamma}$
Job 2	a_{21}	a_{22}	...	$a_{2\gamma}$
...	...	...	...	...
Job τ	$a_{\tau 1}$	$a_{\tau 2}$	...	$a_{\tau\gamma}$

Table 2. Combination of CMT and job holons

evaluated for all candidates x. τ and γ gives the number of the candidate job holons for the CMT holon j, and the number of the candidate CMT holons for the job holon i, respectively.

3.1.4 Coordination among holons

After evaluating the utility values, all the 'idling' holons send all the candidates and their utility values to the coordination holon, and the coordination holon select a most suitable combination of the CMT holons and the job holons, which execute the machining processes in the next time period. The coordination process is summarized in the following, for the case where the coordination holon determines a suitable combination of all the candidates of the job holons i (i = 1,2,..,τ) and the CMT holons j (j =1,2,.., γ).

The utility value δ_{ij} of the combination of job holon i and CMT holons j is given by the following equation

$$\delta_{ij} = RUV_j(i) + JUV_i(j) \tag{9}$$

The problem to be solved by the coordination holon is to select a combination of job holons and CMT holons which maximize the total of the utility value, as shown in the following equation.

$$\underset{a_{ij} \in A}{\text{maximize}} \left(\sum_{i=1}^{\tau} \sum_{j=1}^{\gamma} a_{ij} \cdot \delta_{ij} \right) \tag{10}$$

where, a_{ij} (= 0 or 1) are the decision parameters, as shown in Table 2. If a_{ij} =1, the job holon i is machined by the CMT holon j in the next time period. Otherwise, job holon i is not machined by the CMT holon j. Only one job holon is machined by one CMT holon, therefore, A is a set of a_{ij} which satisfy the following equation.

$$\sum_{i=1}^{\tau} a_{ij} \leq 1, \quad \sum_{j=1}^{\gamma} a_{ij} \leq 1 \tag{11}$$

3.1.5 Case study

Some case studies have been carried out to verify the effectiveness of the proposed methods. The HMS model consisting of 10 CMT holons is considered for the case study. The individual CMT holons have the different objective functions and the different machining capacities, such as the machining time T_{ikm}, the machining accuracy MAC_{ikm}, and the machining cost MCO_{ikm}.

As regards the job holons, 12 job holons are considered in the case study, which have the different objective functions and the machining process. 12 cases are considered in the case study by changing the machining capacities of the individual CMT holons. Fig. 3 summarizes the comparison between the proposed scheduling method based on utility values and the

rule-based method, from the viewpoint of the objective function values of the individual holons. In the Fig. 3, the horizontal axis gives the type of the objective functions of the individual holons, and the vertical axis shows the average number of holons λ, the objective function values of which are improved by the utility values based methods. The λ is calculated by the following equation.
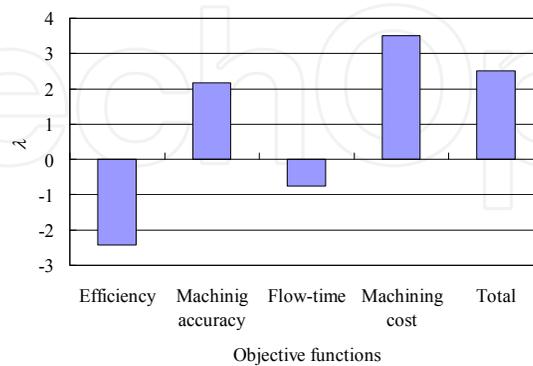

Fig. 3. Results of case study

$$\lambda = \sum_{g=1}^{12} (a_g - b_g) \, / \, 12 \tag{12}$$

a_g: number of holons, the objective function values of which are improved by the proposed method in the case g,
b_g: number of holons, the objective function values of which are deteriorated by the proposed method in the case g.
As shown in Fig. 3. the proposed scheduling method based on utility values is effective to improve the objective function values of the individual holons from the view point of total number of holons. However, the dispatching rules applied to the rule-based method is very effective to reduce the total make span, therefore some holons with the objective functions of the efficiency or the flow-time do not improve their objective function values by the proposed method.

3.2 Agent-based dynamic integrated process planning and scheduling
Process planning and scheduling are important manufacturing planning activities which deal with resource utilization and time span of manufacturing operations. The process planning and scheduling tasks are very complicated and time consuming, if it is applied to the dynamically changing FMSs (Flexible Manufacturing Systems).
PROSA has already proposed a reference architecture for developing distributed manufacturing system including three types of basic holons: resource holons, product holons and order holons (Brussel et al., 1998). Leitao & Restivo (2006) also proposed a control architecture, designated by ADAptive holonic Control aRchitecture (ADACOR) for FMS, intends to contribute to the improvement of the manufacturing control systems performance in term of agile reaction to disturbances and change. Although, there are general architecture and far more detail than is needed for practical applications. We use the elements of PROSA and ADACOR architectures to develop a multi agent architecture for real time integrated process planning and scheduling system in the FMSs, which generate

suitable process plans and schedules based on the status of the FMSs. A comprehensive design and implementation have been done to show the effectiveness of the multi agent approach for dynamic integrated process planning and scheduling of mechanical parts. The methods proposed in the literatures deal mainly with the process planning and scheduling tasks in the static environment in which the jobs specifications and the manufacturing system status are stable (Leitao, 2009). However, it is now required to develop an integrated process planning and scheduling systems applicable to the dynamic environment in which some unforeseen disturbances may occur. In this case study, we propose a multi-agent based integrated system for process planning and scheduling in the dynamic environment, in order to cope with the jobs specification changes and the unforeseen disruptions, such as the malfunction of the machine tools and through the simulation, we are going to illustrate how the agents are able to real timely handle disturbances effectively.

In the literature of agent based manufacturing system, many researches apply simple algorithms such as dispatching rules which are applicable for real time decision making. These methods are simple and applicable, but they do not guarantee the effectiveness for complex problems in the manufacturing systems. As the efficiency becomes more important in the agent based manufacturing (Shen et al., 2006, Wang et al., 2006), we apply coordination agent and a mathematical model for assigning the jobs to the machine tools. The developed model is enough fast to be applicable for the real time agent based manufacturing system when we face unforeseen disturbances such as machine tool breakdown and job specification changes.

3.2.1 Multi Agent Architecture

A multi-agent architecture is proposed to carry out the integrated process planning and scheduling. We mainly use the PROSA and ADACOR reference architectures to define the necessary elements such as agents but we have tailored and customized them according to our specific problem. We also define new agents such as machining process agents and production engineering agents to increase the performance of the architecture for disturbance handling and dynamically generating alternative process plans. In the following sections, the system architectures proposed here are discussed from the viewpoints of the agent definitions, negotiation protocols, and coordination mechanism among the agents.

3.2.2 Agents Definition

Two types of agents are considered in this research to develop the integrated process planning and scheduling systems. They are, physical agents and information agents. The physical agents represent jobs, the manufacturing resources and the machining processes in the FMSs and they are similar to the product, order and resource holons defined in PROSA and ADACOR reference architectures although the contents of the agents and their communications have been customized. The manufacturing resources considered here are the machine tools, the preparation stations, the AGVs (Automated Guided Vehicles), the fixtures and the cutting tools. The information agents are virtual agents which are responsible for governing the negotiation protocol and decision-making.

3.2.2.1 Physical agents

The UML class diagram of the physical agents including their attributes and relationships
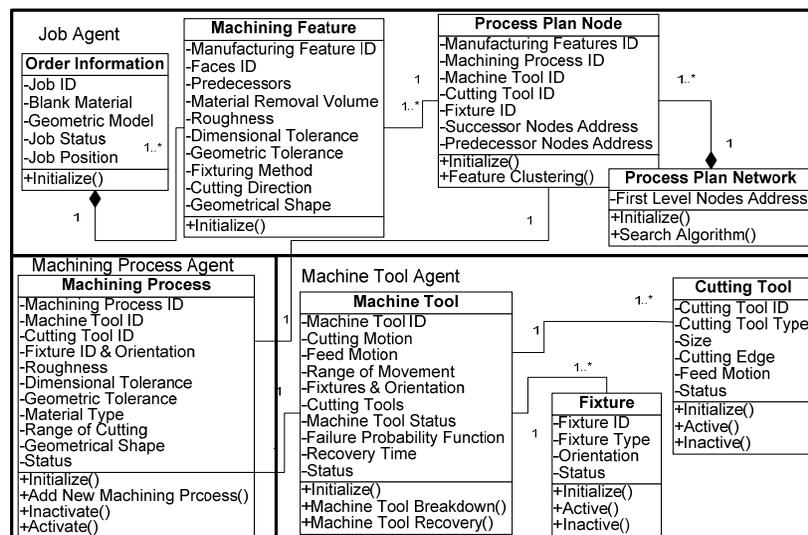
Fig. 4. UML Class diagram of physical agents and their relations.

are summarized in Fig.4. The attributes of the physical agents according to the Fig. 4 are summarized in the following.

Job agents. The job agents represent the jobs to be manufactured in the FMSs. The role of the job agents in the process planning is to certify the correct machining processes of the jobs. It is quite similar of product holon in PROSA architecture although we encapsulate the process plan networks in order to dynamically generate alternative process plans which is the key factor for integrating the process planning and scheduling and disturbance handling. The process plan network has essential role for improving the efficiency of the total system and handling the unforeseen disturbances. Generating alternative process plans has not been discussed clearly in the previous architectures and we present a reliable method based on the process plan network to generate process plans according to available manufacturing processes. The job agents include the following information to describe the orders and the machining features.

Job information:

The job information section describes the order information, the locations and the progresses of the machining processes of the jobs.

Machining features:

The machining feature section gives the machining features of the jobs and their technical data such as the types, the tolerances and the roughness. These technical data are required to select appropriate machining processes.

Process plan networks:

The process plan networks represent the generated process plans in non-linear and hierarchical ways. It includes all the alternative process plans that satisfy the technological requirements of the jobs. Although the mechanical parts are complicated but the size of process plan network in practice will remain limited as we are able to group the manufacturing features. Generally in FMSs, the machining centers which include wide range of cutting tools are able to carry out many manufacturing processes of the job at the same time.

Job Status:

> We consider the following status for the job agents.
>> • Idle: The job agent is idle and waiting for the next machining operations.
>> • Machining operation: The job agent is under machining processes on the machine tools.
>> • Transportation and re-fixturing: The job agent is transported and/or re-fixtured for its next machining operations.

Machine tool agents. The machine tool agents represent the machine tools which are quite similar as the resource holon at PROSA and operational holon at ADACOR reference architectures. PROSA considers the machine tool, cutting tools and fixtures as individual holons. At FMSs generally each machine tools include wide range of the cutting tools for doing different machining processes. To avoid the complexity for practical applications in large manufacturing systems, we do not consider separate agents for cutting tools and fixtures to decrease the total number of agents. We are trying to identify the differences between cutting tools and fixtures by using the information from machining process agent. The agents representing such resources as the preparation stations and the AGV are not considered, at present, since only the machining processes are discussed in the present research. The machine tool agents are responsible for generating proposals to the machining processes required from the job agents. The proposals include the machining time, the transportation time and the re-fixturing time needed to carry out the required machining processes of the job agents. The machine tool agents include the following information to represent the machine tools in the FMSs.

1. Machine tool information:

> The machine tool section specifies the shape generation functions, which are represented by the cutting motions, the spindle directions, the feed motions and the maximum product size.

2. Machine tool status:

> We consider the following status for the machine tool agents in the simulation
>> • Idle: The machine tool is idle and negotiating with job agents for next machining operation
>> • Machining operation: The machine tool is machining the job agent
>> • Breakdown: The machine tool has been broken and is under recovery process

3. Cutting tool:

> The characteristics of the cutting tools are described in the cutting tool section, which includes the information about the cutting tool types, the tool sizes and the cutting edge types.

4. Fixture:

> The fixture section describes the fixture types, and the positions of the fixtures against the spindle axis.

Machining process agents. The machining process agents represent the machining processes of machining features of the jobs, which are carried out by the machine tools. It plays a key role for dynamic process planning and disturbance handling by providing the available and suitable machining processes for job agents. This task has mainly handled by the resource holons at previous architectures. In PROSA the resource holon is mainly responsible for

selecting the best process by selecting the appropriate tools and cutting speed. Although in the practical applications, selecting the suitable manufacturing process is mainly is done by using manufacturing standards and tables which we encapsulate this knowledge in the machining process agent. The information related to the available machining processes and their capabilities will be updated from design department. The agents include the following information.

1. Machining process ID which is the combination of the ID of the machine tools, the ID of the fixtures and the ID of the cutting tools.
2. Machining process types and machining features types, which can be generated by the machining processes
3. Surface roughness, tolerances and material removal rate of the machining processes.
4. Machining process status:
   The status represents the dynamic status of the machining processes in the FMSs. The machining process agents have two statuses.
   - inactive: if one of the machine tool, the cutting tool and the fixture related to the machining process are broken-down
   - active: otherwise

3.2.2.2 Information agents

The information agents are virtual agents for governing the negotiation protocol and decision-making.

Production engineering agents. The production engineering agents generate the job agents, the machine tool agents, and the machining process agents to specify the geometric and technological information of the jobs, the machine tools and the machining processes of the FMSs. The agents play a key role for initializing the information of the physical agents. The design department has an important role for initialization and disturbance handling in our architecture. It real timely modifies the job specifications according to the design change orders or customer requests during the manufacturing.

Job order agents. The job order agents represent the manufacturing tasks and it is quite similar to order holon at PROSA and task holon at ADACOR reference architectures. They are information agents, which carry out the negotiation processes between the job agents and the machine tool agents to generate suitable process plans. The agents have crucial influence on the system performance by deploying efficient decision-making mechanism to select the appropriate machine tools for the individual machining features of the jobs.

Coordination agent. The multi agent architecture proposed here is distributed, and the benefits inherited from distributed architecture include flexibility and robustness. However, absence of higher authority, in general, may result in a lower performance compared with hierarchical systems that are able to achieve global optimization. We introduce a coordination agent to improve the performance which is similar to staff holon at PROSA and supervisor holon at ADACOR reference architectures. The coordination agents are proposed to determine a suitable assignment of the job agents to the machine tool agents at each step of the negotiation. The coordination agents are capable to make mathematical models according the information sent from the machine tool and job agents to find a suitable

assignment of the job agents to the machine tool agents at each step of negotiation. There are a few research in the literature about the optimizing the multi agent architecture for integrated process planning and scheduling (Shen et al., 2006). In this case study, we present a method by combining the mathematical modeling and process plan networks to optimize the total system. The optimization method works effectively for real time applications and generates suitable solutions.
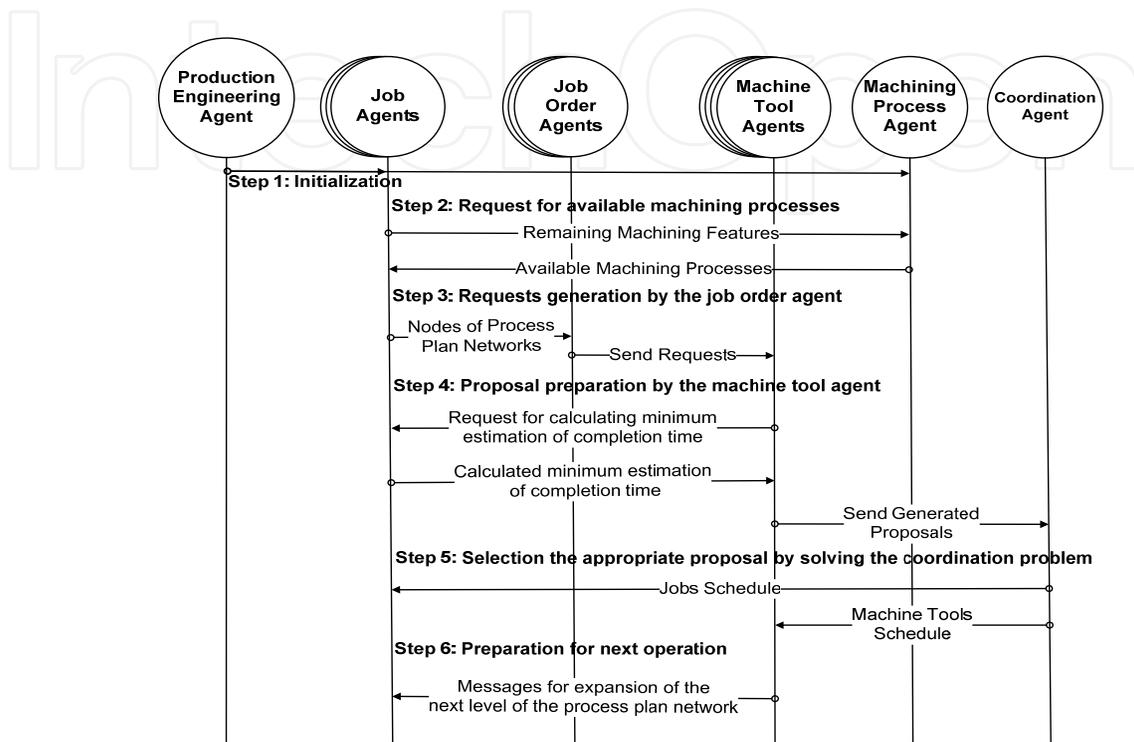

Fig. 5. Negotiation protocol among agents

3.2.3 Negotiation Protocol

A negotiation protocol among the agents is required to coordinate the distributed decisions of the individual agents for solving the complex problems in the integrated process planning and scheduling of the FMSs. Here, we define a negotiation protocol to meet the requirements for integrated process planning and scheduling and also handling the unforeseen disturbances. The problems to be solved here are as follows.

1. Selection of candidate machining processes for individual machining features.
2. Selection of suitable combinations of the machine tools, the cutting tools and the fixtures.
3. Selection of suitable machining sequences of the machining features.

In the protocol proposed here, the individual agents have three types of boards named "request boards", "proposal boards" and "status boards" for the communication. The requests from the other agents are firstly sent to the request boards, and the agents scan and read the requests from the boards, every fixed time intervals named RTIP (reading time interval period). The individual agents secondly generate the proposals to the requests, and store them in the proposal boards. The statuses of the agents are changed and stored in the status board, if necessary. Fig. 5. summarizes the negotiation protocol proposed here to generate suitable process plans and schedules by the distributed

(a) part geometry (b) machining features (c) precedence constraints
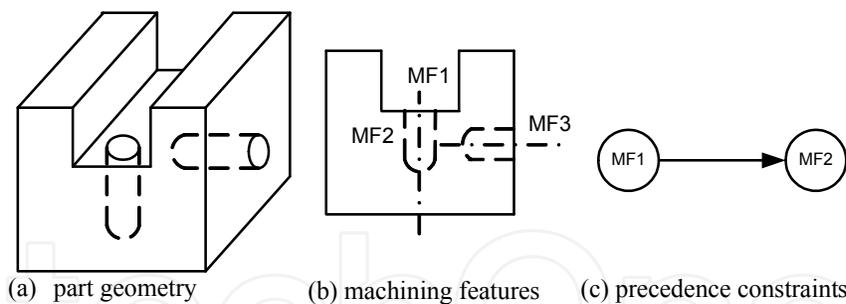
Fig. 6. An example of parts

Machining Features	Machining Process ID	Machine Tool ID	Fixture ID	Cutting Tool ID
MF1	mp_1	mt_1	fi_1	ct_1
MF1	mp_5	mt_2	fi_2	ct_1
MF3	mp_4	mt_1	fi_2	ct_2
MF3	mp_7	mt_2	fi_1	ct_2

mt_1: Vertical machining center, mt_2: Horizontal machining center ct_1: End Milling, ct_2: Drill

Table 3. Alternative machining processes for machining features MF1 and MF3

decision-makings of the individual agents and the negotiations among the agents. The negotiation processes are carried out through the following steps.

Step 1: Initialization

The production engineering agents firstly generate all the job agents, the machine tool agents and the machining process agents to initialize the status of the target FMSs. They also define the machining features, which can be generated simultaneously by the same combinations of the machine tools, the cutting tools and the fixtures, and assign them to the job agents.

Step 2: Requests for available machining processes

The job agents select a set of the machining features, which can be machined in the next machining process, based on the precedence constraints among the machining features. For example, let us consider a case shown in Fig.6. This part consists of three machining features (see Fig.6 (b)). They are, one slot MF1, and two holes MF2 and MF3. The precedence constraints for this example is shown in Fig. 6(c). In the first step, the machining features MF1 and MF3 are sent to the machining process agents, and they select a set of available alternative machining processes for the individual machining features, based on the specifications of the machining features, such as the geometries, the sizes, the surface roughness, and the tolerances as shown in the Table 3.

The selected machining processes include the information about the machine tools, the cutting tools and the fixtures. The selected machining processes for the individual machining features are sent back to the job agents. The job agents generate a set of groups of the machining features, which can be machined by the same combinations of the machine tools, the cutting tools and the fixtures. Machining features belonging to the same setup have been grouped together for one machine to minimize the setup/fixture time. This means that the grouped machining features are machined by one machining process concurrently, and the job agents generate the nodes representing all the grouped machining features in the process

RMF = Remaining machining features set
AMF = Available machining features set for the next operation

MT = Manufacturing time
ECT = Estimation of minimum completion time

N_1
$mp_1 = (mt_1, fi_1, ct_1) \rightarrow MF1$
$RMF = \{MF2, MF3\}$
$AMF = \{MF2, MF3\}$
MT = 738
ECT = 2070

N_2
$mp_3 = (mt_1, fi_2, ct_2) \rightarrow MF3$
$RMF = \{MF1, MF2\}$
$AMF = \{MF1\}$
MT = 1397
ECT = 2227

N_3
$mp_4 = (mt_2, fi_2, ct_1) \rightarrow MF1$
$RMF = \{MF2, MF3\}$
$AMF = \{MF2, MF3\}$
MT = 685
ECT = 2525

N_4
$mp_6 = (mt_2, fi_1, ct_2) \rightarrow MF3$
$RMF = \{MF1, MF2\}$
$AMF = \{MF1\}$
MT = 1312
ECT = 2440

N_0
$RMF = \{MF1, MF2, MF3\}$
$AMF = \{MF1, MF3\}$

N_5
$mp_2 = (mt_1, fi_1, ct_2) \rightarrow MF2$
$RMF = \{MF3\}$
$AMF = \{MF3\}$
MT = 435

N_6
$mp_3 = (mt_1, fi_2, ct_2) \rightarrow MF3$
$RMF = \{MF2\}$
$AMF = \{MF2\}$
MT = 897

N_7
$mp_5 = (mt_2, fi_2, ct_2) \rightarrow MF2$
$RMF = \{MF3\}$
$AMF = \{MF3\}$
MT = 943

N_8
$mp_6 = (mt_2, fi_1, ct_2) \rightarrow MF3$
$RMF = \{MF2\}$
$AMF = \{MF2\}$
MT = 1341

N_9
$mp_3 = (mt_1, fi_2, ct_2) \rightarrow MF3$
$RMF = \{\}$
$AMF = \{\}$
MT = 897

N_10
$mp_6 = (mt_2, fi_1, ct_2) \rightarrow MF3$
$RMF = \{\}$
$AMF = \{\}$
MT = 1531

End

First Level   Second Level   Third Level

Fig. 7. Process plan network.

plan networks as shown in the first level nodes N_1 to N_4 in Fig.7. The contents of the process plan networks are described in the previous paper. (Tehrani et al., 2007). The individual nodes in the process plan networks represent a set of the machining features which could be machined by one machining process. An algorithm also generates nodes representing the machining sequences of the machining features based on the precedence constraints. The information of the generated nodes is sent to the job order agents for the negotiations.

Step 3: Request generation by job order agents

The job order agents create requests for the machining process execution for the individual nodes of the process plan networks, which are the groups of the machining features that can be generated by same machine tools. The requests for the job agents are generated according to the process plan network which guarantees that we always generate feasible solutions and different operations of one job agents can not be processed simultaneously. The generated requests are sent to the request boards of the corresponding machine tool agents. The content of request includes the machining features and selected machine tool, cutting tool and fixture. As you can see in Fig. 7, there are four nodes N_1 to N_4 in the first level of the process plan network. For each of them, the requests are generated and sent to the related machine tools MT1 and MT2.

Step 4: Proposal preparation by machine tool agents

The machine tool agents read all the requests from the request boards every RTIP (Reading Time Interval Period) if it is idle and each machine can handle only one job at a time. The machine tool agents analyze the request messages, and generate appropriate proposals to all the requests. We consider a heuristic algorithm to estimate minimum completion time for generating appropriate proposal for each request by the machine tool agents, as shown in the followings.

Minimum completion time estimation

The machine tool agents need to estimate the completion time of the remaining machining features of the job agents. A procedure is developed and given to the job agents to estimate the minimal completion time of the remaining machining features, based on the process plan networks shown in Fig. 7. When a machine tool agent requires a job agent to estimate the minimum completion time, the job agent starts the procedures from the start node which is specified by the machine tool agent, and repeat to generate and to select suitable successive nodes with the minimum machining time. When all the machining features are included in the process plan networks, the job agent find both the machining sequences of the machining features and the estimated minimal completion time. Consider a case where we are at node N_i of the process plan network and we are going to estimate the manufacturing time from the node N_i to the end node. The algorithm for calculating the estimated minimum completion time from node N_i to the end node is summarized in the followings.

Initialization:

- Set RMF and AMF. The RMF is the set of the remaining machining features. The AMF is the set of the available machining features that do not have any preceding machining features and could be done firstly considering the precedence constraints among the nodes of the process plan networks.(AMF $\subseteq$ RMF).

- Put the node N_i in ECTS set. The ECTS is the set of the nodes in the path from node the N_i to the end of the process plan network, which has the minimum manufacturing time.

- Set a initial node N_i. In the N_i, RMF ={set remaining machining features}, AMF={set of machining features without any successors}.

(1) Generate a set of successor nodes $SN = \{N_t | t = 1,2, \ldots |SN|\}$ of the node N_i for all feasible machining processes $mp_r = (mt_i, fx_f, ct_t), r = 1, \ldots, R$, (R = total number of available machining processes) by applying the following algorithm.
   - Cluster all features of the AMF set of the node N_i that could be machined with the machining process mp_r,
   - Generate a new node N_t representing a set of machining features which can be machined by the machining process mp_r and put it in the SN set. The links to the nodes N_t, which are successor nodes, are stored in the node N_i for further processing,
   - Estimate the manufacturing time for node N_t that includes the time of the machining, the transportation and the re-fixturing processes,
   - Update the RMF and AMF sets for the node N_t,

(2) Select a successor node N_k from the SN set which has the minimum machining time for the next step of extension, and move it to the ECTS set.

(3) If RMF set of N_k is not empty consider node N_k as node N_i and go to (1).

(4) If RMF set of N_k is empty, it means that we are in the end of the process plan network. The sum of the manufacturing time for the nodes in ECTS set is the estimation of the minimum completion time from node N_i to the end.

Let us consider a case where we are going to calculate the estimation of minimum completion time for node N_1 at the process plan network shown in Fig. 7. We start with node N_1, and there are four successor nodes N_5, N_6, N_7, N_8 from the node N_1 as shown in Fig. 7. We select the node N_5 which has the minimum manufacturing time, and we put it in the ECTS set. We expand the node N_5 at the next stage of the algorithm and there are two successor nodes N_9, N_{10}. The node N_9 is selected and which has the minimum manufacturing time, we put it in the ECTS set. As you can see in Fig. 7, for the node N_9 the RMF set is empty and the algorithm stops. It is because that there are no remaining machining features in the node N_9. The sum of the manufacturing time for the nodes in ECTS set is the estimation of the completion time from node N_1 until end.

Following this, the job agent returns the estimated completion time to the machine tool agent. As you can see in the Fig.7, the estimation of completion time for all nodes N_1, N_2, N_3, N_4 are calculated and these values are returned to the machine tool agent. This procedure can estimate the completion time of all the remaining machining features, however it requires the additional communications between the machine tool agents and the job agents. The machine tool agents generate proposals for each request based on the minimal completion time of the remaining machining features and send them to the coordination agents.

Step 5: Selection of appropriate proposals by coordination agent

The coordination agents scan all received proposals from the machine tool agents every RTIP, and assign the appropriate machine tool agents to the job agents. At present, we consider only the flow time of the job agents, and our goal is to minimize the average flow time of all the job agents. The flow time considered here includes the machining time, the transportation time, the re-fixturing time and the tool changing time. The constraints of the model are that only one machine tool agent is selected for each job agent and only one job agent has been assigned to each machine tool agent. The followings summarize the formulas representing the optimization problems considered here.

Parameters:

$$MP = \{mp_r : (mt_i, fx_f, ct_t) | r = 1, \dots, R\}, R = |MP|, \tag{13}$$

$$MT = \{mt_j | j = 1, 2, \dots n\}, n = |MT|, \tag{14}$$

$$FI = \{ft_f | f = 1, 2, \dots F\}, F = |FI|, \tag{15}$$

$$CT = \{ct_t | t = 1, 2, \dots T\}, T = |CT| \tag{16}$$

where,

mp_r: ID of machining process, mt_j: ID of machine tools, ft_f: ID of fixtures, ct_t: ID of cutting tools.

$FT^j_{(i,r)}$: Estimation of completion time of job agent i ($i = 1,2,..m$) according to the machining process mp_r ($r = 1,2,..R$) with machine tool agent mt_j ($j = 1,2,..n$).

Design variables:

$x^j_{(i,r)} = \begin{cases} 1: \text{if the machine tool agent } mt_j \text{ is selected for job agent } i \text{ according to the} \\ \quad \text{machining process } mp_r \\ 0: \text{otherwise.} \end{cases}$

Mathematical Model:

$$\text{Minimize} \quad Z = \sum_{j=1}^{n} \sum_{i=1}^{m} \sum_{r=1}^{R} x_{(i,r)}^{j} \, FT_{(i,r)}^{j} \tag{17}$$

$$\sum_{j=1}^{n} \sum_{r=1}^{R} x_{(i,r)}^{j} + Dummy_i = 1, \quad i = 1,2,..m \tag{18}$$

$$\sum_{i=1}^{m} \sum_{r=1}^{R} x_{(i,r)}^{j} + Dummy_j = 1, \quad j = 1,2,..n \tag{19}$$

$$x_{(i,r)}^{j} = 0,1 \tag{20}$$

We add dummy variables to equations (18) and (19) to change the constraints of sets of equations. Equation (17) is the objective function that is the total of the estimated flow time of all the job agents. Equation (18) is a constraint that only one machine tool agent is selected for each job agent. Equation (19) is a constraint that only one job agent has been assigned to each machine tool agent. The model described in equations (17)-(20) is an assignment problem and can be solved as a linear programming model. We can release the equation (20) from the model and apply linear techniques and the optimal solution will be integer. We can use other objective functions such as minimizing the manufacturing costs and minimizing the average of tardiness of all jobs with the above model.

After solving the above model, the coordination agents inform both the job agents and the machine tool agents that the machining features sent from the job agents shall be machined by the selected machine tools. This means that the coordination agents dynamically generate the process plans and the production schedules of the job agents and the machine tool agents. The job agents and the machine tool agents selected here carry out the requested machining processes in the next step. Therefore, the statuses of these agents are changed, and the status data are stored in the status boards. All the agents monitor the status data if necessary.

Step 6: Preparation for next operation

When the machine tool agents complete the machining operations of the job agents, the job agents modify their process plan networks. That is, the job agents delete the corresponding nodes representing the group of the machining features which was completed by the machine tool agents. New nodes of the process plan networks are generated to specify the groups of the machining features to be machined in the next step. The procedures presented in Steps 2 to 6 are repeated until the job agents do not have any remaining machining features.

3.2.4 Synchronization

The synchronization of negotiation between different agents is important issue for developing the multi agent architecture. The Petri nets (Proth & Xie 1996) are used, in the case study, for synchronizing the messages and the negotiation protocols between the different agents. This Petri nets control both the sequence and the timing of the interaction and the messages between the agents. Each Petri net represents one agent or interacting agents. Fig. 8 shows an example of the interaction between the agents for generating and sending the requests to the request board of the machine tool agents and generating the proposals by the machine tool agents. These Petri nets are linked with each other with global transition (transitions, $t_2, t_4, t_8, t_{14}, t_{17}$ in Fig. 8).
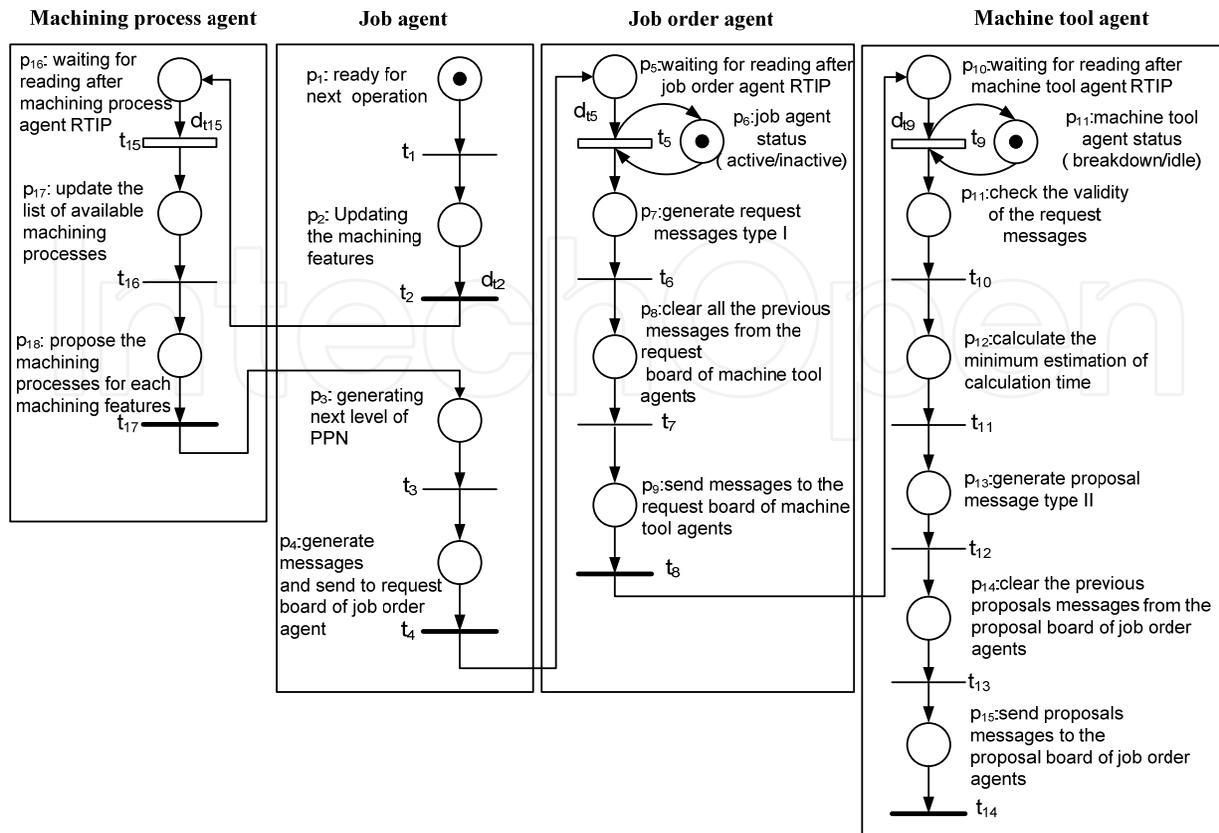
Fig. 8. Synchronizing agents for generating requests and proposals

3.2.4 Simulation Software and Experimental Results

A prototype of the agent based integrated process planning and scheduling system and the graphical presentation system have been developed for the case studies. The system developed here is able to simulate the distributed decision makings of the agents, the negotiation processes among the agents, and also the manufacturing processes in the FMS. The coordination agent use ILOG CPLEX optimization engine for solving the integer programming model of the coordination and for assigning the job agents to machine tool agents. Some case studies have been carried out to verify the applicability and the effectiveness of the proposed system to the integrated process planning and scheduling problems in the FMSs. The FMS considered here includes 7 machine tools and 4 job types. Fig. 9 shows the geometries of the job agents and their manufacturing features including cylinder and box type shape for the case studies. The detailed information of the machining features and the machining resources of the case studies are brought in the previous paper (Tehrani et al., 2007). The RTIP in the simulation is set to be 2 sec. for the machine tool agents, 3 sec for coordination agents and 4 sec. for the job agents.

3.2.4.1 Efficiency of the proposed architecture

Two case studies have been done to evaluate the impact of introducing the coordination agents in multi agent systems. We compare the results with the dispatching rules which the job agents applying SPT dispatching rules for selecting the machine tools for their manufacturing operations without assisting from the coordination agents.
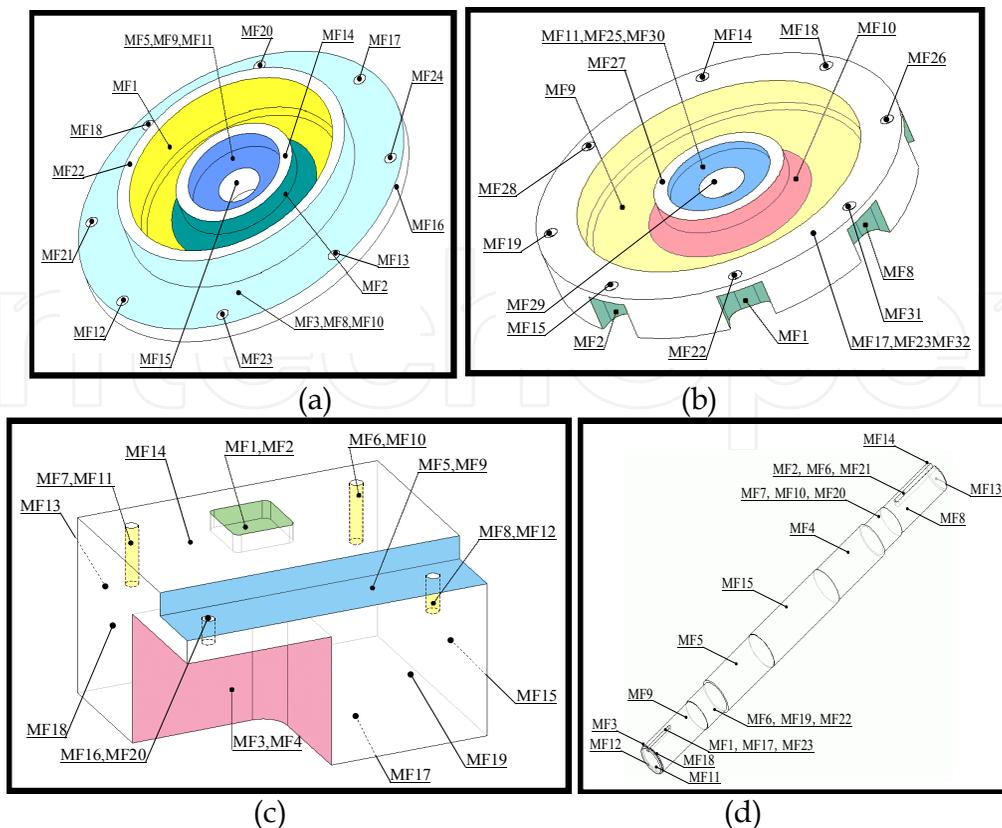
Fig. 9. Jobs considered in case studies.

Fig. 10 summarizes the comparison of the proposed architecture and the previous method from the view points of the average flow time of all the job agents and the calculation time for coordination. In the Fig. 10 the vertical axis gives the flow time of the individual job agents and the horizontal axis shows the individual job agents and their types.

It is understood, from Fig. 10(a) and (b), that the multi-agent systems with the coordination agents generate more suitable process plans and schedules from the viewpoint of the average flow time of the all the job agents. As you can see, the average flow time has been improved 10.9% and 10.39% for the cases (a) and (b) of Fig. 10, respectively. It is because that the mathematical programming methods applied here are suitable to reduce the average flow time of the job agents of the job shop process planning and scheduling problems. The calculation time for coordination is enough short and the proposed method   is suitable for the real time application, when we have enormous number of job agents and machine tool agents.

3.2.4.2 Robustness of the proposed architecture

An additional experiment is also carried out to assess the robustness of the proposed architecture against the malfunction of the machine tools. The original process plans and schedules are shown for 10 job agents in the Gantt chart of Fig. 11 (a). In the experiment, the machine tool "MT14" is broken down at simulation time 4811 sec. and the recovery time is assumed to be 5000 sec. As you can see in the Gantt chart of Fig. 11 (b), the proposed architecture can dynamically generate alternative process and schedule to cope with the malfunctions of the machine tools. The job agents can be dynamically allocated to another manufacturing route in the process plan networks and new process plans for jobs 7,6,4,3 and job 2 has been generated dynamically.
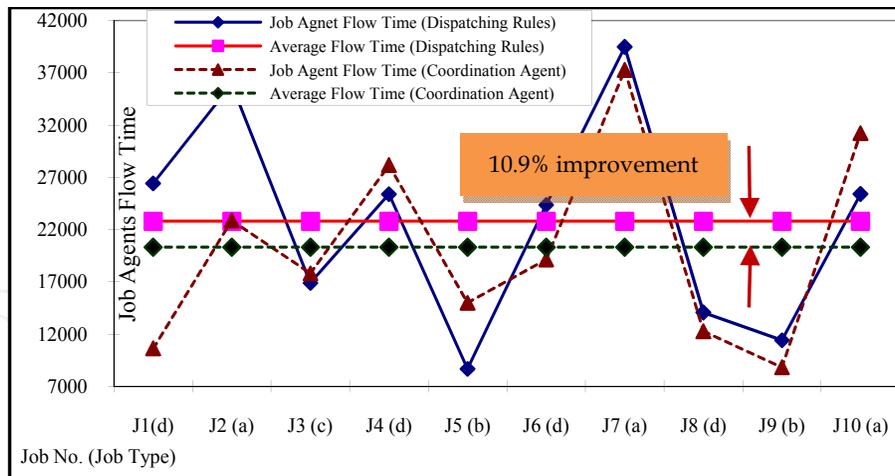
(a) Case study with 10 job agents
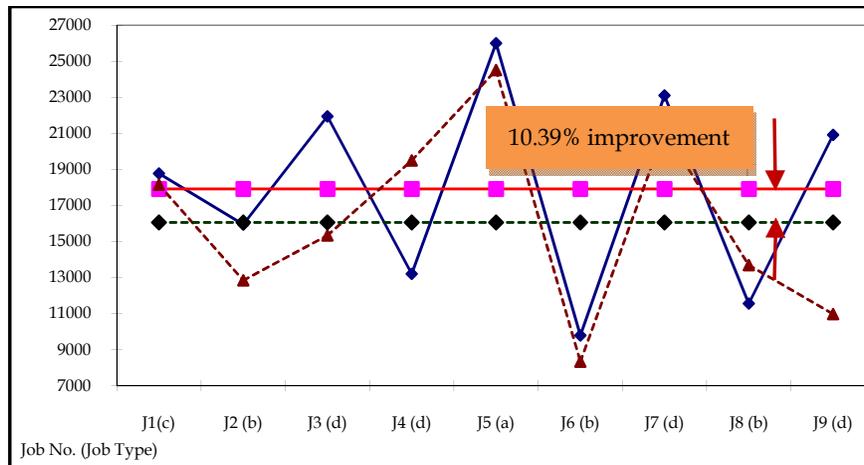
(b) Case study with 9 job agents.

Fig. 10. Case study and comparison with previous result.

In the other experiments, the following unforeseen changes have been considered in the job specifications.

1. Change the roughness of the machining features
   - Job 03, MF16 at simulation time 3000
   - Job 10, MF18 at simulation time 10000
2. Add a new machining feature to the job
   - Job 02, MF21 at simulation time 7000
   - Job 04, MF24 at simulation time 5000
   - Job 05, MF25 at simulation time 2900
3. Change the size of machining feature
   - Job 10, MF16 at simulation time 10000
   - Job 03, MF21 at simulation time 6500

The results are shown the Gantt chart of Fig. 11 (c). As shown in Gantt chart Fig. 11 (c), the proposed architecture can dynamically generate updated process plans and schedules to cope with the changes of job specifications.
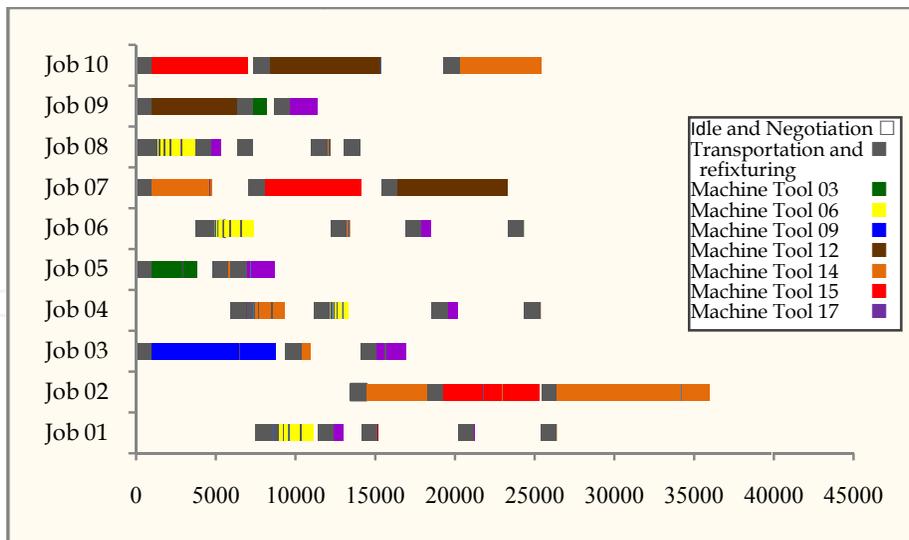
(a) Original schedule without unforeseen changes
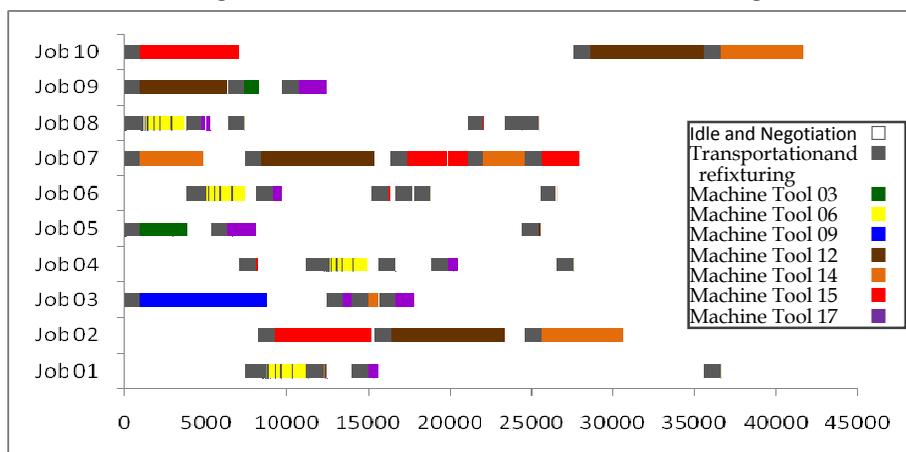
(b) Modified schedule for malfunction of machine tool "MT14"
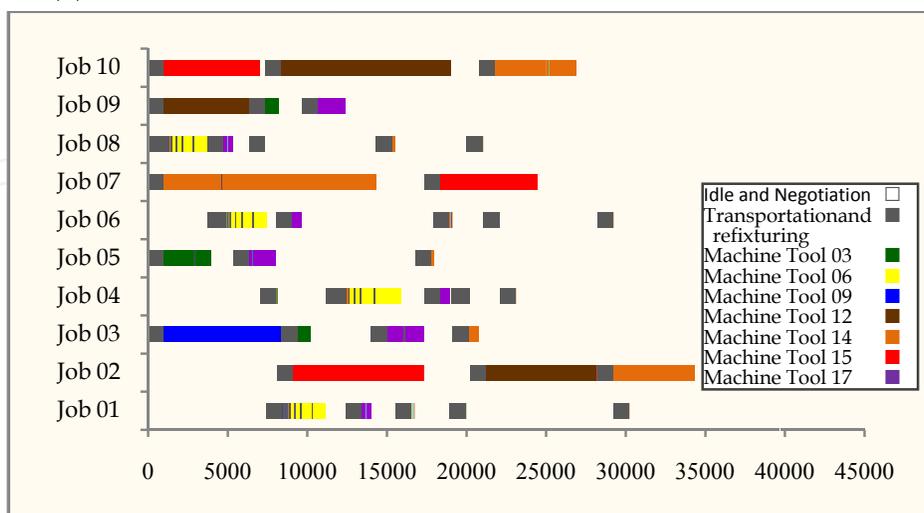
(c) Modified schedule for job specification changes

Fig. 11. Gantt chart for case study of robustness
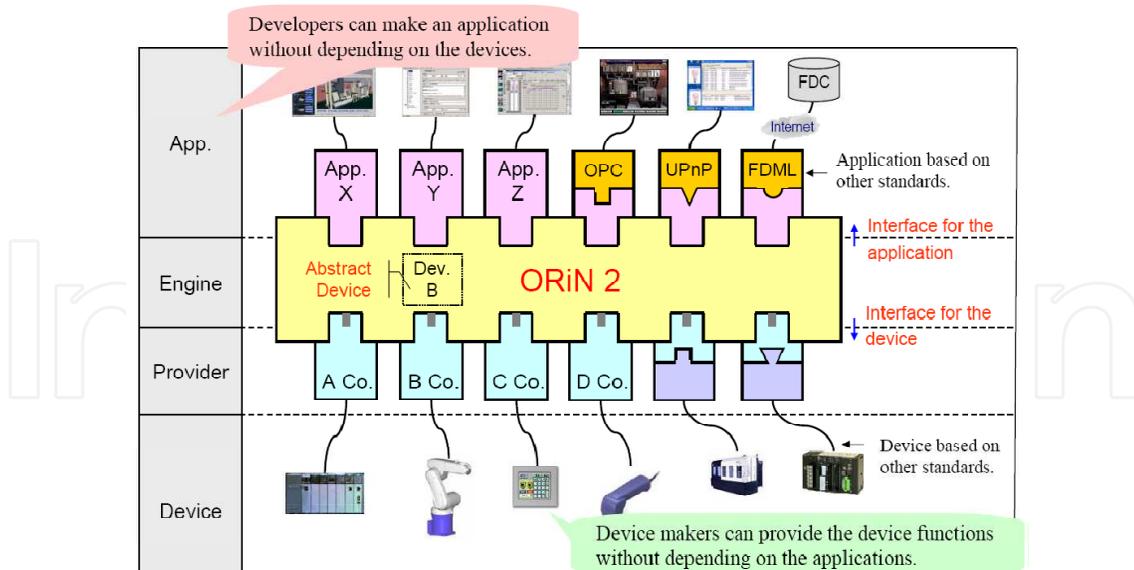
Fig. 12. Two layers of ORIN architecture

4. Realizing the agent manufacturing system

In spite of the promising perspective of these emergent distributed and intelligent approaches, until now the industrial applications of control systems developed in the context of reconfigurable manufacturing systems are extremely rare and the implemented functionalities are normally restrict, being very slow the adoption of these concepts by industry (Marik & McFarlane 2005).

We have collaboration with DENSO Wave Co. for realizing the agent manufacturing system through the ORIN architecture. ORIN 2.0 (Open Robot Interface for Network) provides integrated interface to access to the devices on the network (Hibino et al., 2006). You can easily access the data inside the devices from application software by using ORIN regardless of the manufacturers, devices or specifications of communication protocols. ORIN is a Distributed Real Manufacturing Simulation Environment (DRMSE) that consists of two layers; engine layer and provider layer as shown in the Fig. 12. The provider layer has a function to absorb a difference of controller equipment types and emulators. The engine layer provides interfaces for manufacturing applications.

ORIN proposes a hardware and software architecture for realizing the agent based manufacturing system. The agents would be software modules that communicate with the real hardware in the manufacturing system through the ORIN platform. The communication between agents for making decision and handling the negotiation protocol could been done and synchronized through the communication channels provided by ORIN platform. The job agents and corresponding physical part would be recognized and traced through the manufacturing by using bar code or RFID. The machine tools and robots could be connected directly through their controller and we can also define and re-program PLCs and different controller of the manufacturing systems.

In our research, we have successfully integrated our agent based simulation program with ORIN architecture. A barcode reader (DENSO AT10Q-SM) and a bar code generator (DENSO QRdraw Ad) have been connected to the agents through the ORIN architecture. The job agent receives the information from kanban by barcode reader. The bar code generator

has been applied for generating the kanban cards including the job agent information, the disturbances and the job specification changes. The job agents and the machine tool agents can communicate and exchange data real timely through the ORIN architecture with the corresponding hardware in the manufacturing system.

5. Conclusion

Manufacturing companies at the beginning of 21th century have to face a dynamic environment where economical, technological and customer trends change rapidly, requiring the increase of flexibility and agility to react to unexpected disturbances, maintaining the productivity and quality parameters. The traditional manufacturing control systems are adapted on a case-by- case basis, requiring an expensive and huge time-consuming effort to develop, maintain or re-configure. The missing re- configurability is derived from the lack of agility to support emergency (change and unexpected disturbances). The challenge is to develop innovative, agile and reconfigurable architectures for distributed manufacturing control systems, using emergent paradigms and technologies. Multi-agent systems and HMSs are two promising paradigms to build this new class of distributed and intelligent manufacturing control systems. In this chapter, the manufacturing control systems, especially using artificial intelligence techniques to develop it, namely multi-agent systems and HMSs, was reviewed. Two case studies have been discussed in detail and their contributions, results and benefits of applying agent and holonic manufacturing control have been reviewed.

In first case study, a new real-time scheduling methods for the HMS are proposed to select a suitable combination of the CNC machine tool (CMT) holons and the job holons which carry out the machining process. A distributed decision-making procedure is proposed to select a suitable combination of the CMT holons and the job holons for the next machining processes, based on the utility values for the candidates. Some case studies of the real-time scheduling have been carried out to verify the effectiveness of the proposed methods. It was shown, through case studies, that the proposed methods are effective to improve the objective functions of the individual holons. In the second case study, a multi-agent system was proposed for the integrated process planning and scheduling systems for the FMSs. A systematic procedure was proposed to generate suitable process plans of the jobs and suitable schedules of the machine tools. The proposed method is able to solve the process planning and scheduling problems concurrently and dynamically, with use of the mathematical optimization methods and search algorithms of the process plan networks. Some case studies have been carried out to verify the applicability of the proposed method to the integrated process planning and scheduling problems in the FMSs including 7 machine tools and 10 jobs. It was shown, through the case studies, that the proposed multi-agent architecture is capable to generate appropriate process plans and schedules. It was also shown that the proposed architecture generates alternative process plans dynamically, to cope with the malfunctions of the machine tools and unforeseen job specification changes.

In the future research, we are trying to expand the architecture for other objective functions and multi objective integrated process planning and scheduling. We also are trying to develop general agents according to DCOM technology and defining interfaces for them that make agents possible to connect directly to ORIN to communicate with manufacturing hardware, real timely.

6. References

CMV, (1998). *Visionary Manufacturing Challenges for 2020, Committee on Visionary Manufacturing*. National Academic Press, Washington, DC, USA.

Baker, A. (1998). A survey of factory control algorithms which can be implemented in a multi-agent heterarchy: dispatching, scheduling and pull. Journal of Manufacturing Systems, Vol. 17, No. 4, pp. 297–320.

Brussel, H.V., Wyns, J., Valckenaers, P. & Bongaerts, L. (1998). Reference architecture for holonic manufacturing systems: PROSA. *Computers in Industry*, Vol. 37, No. 3, pp. 255–274.

Colombo, A., Schoop, R. & Neubert, R. (2006). An agent-based intelligent control platform for industrial holonic manufacturing systems. IEEE Transactions on Industrial Electronics, Vol. 53, No. 1, pp. 322–337.

Diltis, D., Boyd, N. & Whorms, H. (1991). The evolution of control architectures for automated manufacturing systems. *Journal of Manufacturing Systems*, Vol. 10, No. 1, pp. 63–79.

Hibino, H., Inukai, T. & Fukuda, Y. (2006). Efficient Manufacturing system implementation based on combination between real and virtual factory, *International Journal of Production Research*, Vol. 44, No. 18-19, pp. 3897-3915.

Koestler, A. (1969). *The Ghost in the Machine*. Arkana Books, London.

Leitao, P. (2009). Agent-based distributed manufacturing control: A state-of-the-art-survey. *Engineering Applications of Artificial Intelligence*, Vol. 22, pp. 979-991.

Leitao, P. & Restivo, F. (2006). ADACOR: a holonic architecture for agile and adaptive manufacturing control. *Computers in Industry*, Vol. 57, No. 2, pp. 121–130.

Marik, V. & McFarlane, D. (2005). Industrial adoption of agent-based technologies. *IEEE Intelligent Systems*, Vol. 20, No. 1, pp. 27–35.

Proth, M. & Xie, J., X. (1996). *Petri Net a Tool for Designing and Management of Manufacturing System*, John Willey and Sons.

Russel, S. & Norvig, P. (1995) Artificial Intelligence, A Modern Approach. Prentice- Hall, New Jersey.

Sepehri, M., M & Tehrani, H. (2005). Dynamic scheduling architecture for AGVs and machines in holonic manufacturing system with Petri nets, *International Journal of Industrial Engineering-Theory Applications And Practice*, Vol. 12, No. 2, pp. 132-142.

Shen, W. M., Wang, L. & Hao, Q. (2006). Agent-Based Distributed Manufacturing Process Planning and Scheduling: A State-of-the-Art Survey, *IEEE Transaction on System, Man, and Cybernetics-Part C: Application and Reviews*, Vol. 36, No. 4, pp. 563-577.

Tehrani, H., Sugimura, N., Tanimizu Y. & Iwamura, K. (2007). A Search Algorithm for Generating Alternative Process Plans in Flexible Manufacturing System, *Journal of Advanced Mechanical Design, System, and Manufacturing*, Vol. 1, No. 5, pp. 706-716.

Wang, L. H., Shen, W. M. & Hao, Q. (2006). An Overview of Distributed Process Planning and Its Integration with Scheduling. *International Journal of Computer Applications in Technology*, Vol. 26, No. 1-2, pp. 3-14.

Winkler, M. & Mey, M. (1994). Holonic manufacturing systems. European Production Engineering.

Wooldridge, M. (2002). *An Introduction to Multi-Agent Systems*. Wiley, New York.

Wooldridge, M. & Jennings, N. (1995). Intelligent agents: theory and practice. *The Knowledge Engineering Review*, Vol. 10, No. 2, pp. 115–152.

Wyns, J. (1999). Reference Architecture for Holonic Manufacturing Systems–The Key to Support Evolution and Reconfiguration, PhD Dissertation, *Department of Mechanical Engineering, Katholieke Universiteit Leuven*.

Future Manufacturing Systems

Edited by Tauseef Aized

This book is a collection of articles aimed at finding new ways of manufacturing systems developments. The articles included in this volume comprise of current and new directions of manufacturing systems which I believe can lead to the development of more comprehensive and efficient future manufacturing systems. People from diverse background like academia, industry, research and others can take advantage of this volume and can shape future directions of manufacturing systems.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

INTECH
open science | open minds