Journal of Manufacturing Systems 32 (2013) 801–816

Contents lists available at ScienceDirect

Journal of Manufacturing Systems

journal homepage: www.elsevier.com/locate/jmansys

Technical paper

Dynamic schedule execution in an agent based holonic manufacturing system

Tarun Kanti Jana[a,*], Bipradas Bairagi[b], Soumen Paul[c], Bijan Sarkar[d], Jyotirmoy Saha[d]

[a] Department of Mechanical Engineering, Haldia Institute of Technology, Haldia 721 657, India
[b] Department of Production Engineering, Haldia Institute of Technology, Haldia 721 657, India
[c] Department of Information Technology, Haldia Institute of Technology, Haldia 721 657, India
[d] Department of Production Engineering, Jadavpur University, Kolkata 700 032, India

ARTICLE INFO

Article history:
Received 5 July 2012
Received in revised form 3 July 2013
Accepted 12 July 2013
Available online 22 August 2013

Keywords:
Multi-agent system
Holonic manufacturing system
Fuzzy multi criteria decision making
Contract net protocol
metamorphism
ontology

ABSTRACT

The present paper deals with the negotiation based task allocation to the resources for preparing dynamic scheduling in an agent based holonic control framework. The scheduling priority is developed by Multi Objective Optimization on the basis of Ratio Analysis (MOORA) technique under Fuzzy Multi Criteria Decision Making (FMCDM) environment considering several attributes. The well-known Contract Net Protocol (CNP) is followed for the purpose of task allocation by negotiation and cooperation, where message based communication is accomplished by eXtensible markup language (XML) using J2EE. Different Document Type Definitions (DTDs) are developed for intended applications. Necessary modifications in the scheduling arising out of changes in the volume-mix are made by a distributed cooperative problem-solving algorithm to meet the demand without violating the deadline. The algorithm is implemented using HTML code in front-end with Java Server Page (JSP) through Apache Tomcat 6.02 server. It is advocated that the cooperation based teamwork coupled with higher flexibility and agility is the key to success to remain unperturbed and provide reasonably good solution in the face of disturbances and stands superior to its hierarchical counterpart.

1. Introduction

The manufacturing business fraternity has witnessed dramatic changes in recent years which is characterized by increasingly complex and highly customized product of different volume-mix, smaller batch sizes, increased customer expectations in terms of quality and delivery time. New manufacturing system approaches are required that provide manufacturers with the adaptability and responsiveness to compete in today's market [1]. This establishes the need for a control system and methodology that would allow the system entities to reconfigure, manifest agility, and offer high degree of flexibility to deal with high level of uncertainties and unpredictable situations [2–4].

Holonic manufacturing systems (HMS) relies on the concept of holon [5] to provide a reconfigurable, flexible and decentralized manufacturing architecture to dynamically accommodate changes and meet customer's requirements [2,6–8]. It is a compromise between hierarchical and heterarchical approach that successfully blends their merits to combat the challenges by offering high degrees of flexibility and preserving robustness in view of disturbances [9]. A multi agent based software platform provides distributed intelligent control functions with communication, cooperation and synchronization capabilities. Multi Agent System (MAS) is built on the premise that a complex manufacturing system is divided into a large number of small manageable agents which are autonomous having intelligence to take their own decision, can perceive the environment and respond accordingly. Additionally, the agents can communicate and cooperate with others [10]. In essence, holonic behavior is implemented by agent based systems.

Scheduling in such environments is carried out in dynamic mode to offer flexibility, improve machine utilization and to provide system robustness in view of unexpected system events (such as machine failures, an urgent production order or quantity variations). Furthermore, agents being self-sustainable and intelligent entities, priority of scheduling are not governed by rules like Shortest Processing Time (SPT), Weighted Shortest Processing Time (WSPT), Early Due Date (EDD), Critical ratio (CR), etc. rather, several techno-commercial factors are required to be considered to develop priority for the product scheduling.

The objective of the present work is dynamic task allocation through negotiation and preparation of local schedules for

* Corresponding author. Tel.: +913224252900; fax: +913224252800.
E-mail addresses: tarun.jana2000@gmail.com (T.K. Jana), bipradas_bairagi@yahoo.co.in (B. Bairagi), s.paul.it@gmail.com (S. Paul), bijon_sarkar@email.com (B. Sarkar), jmoysaha@yahoo.co.in (J. Saha).

0278-6125/$ – see front matter
http://dx.doi.org/10.1016/j.jmsy.2013.07.004

execution of customer order in a multi agent based holonic manufacturing system. The products are prioritized according to a merit list prepared by MOORA technique under FMCDM paradigm considering several attributes such as revenue generation, urgency, complexity, etc. The well known Contract Net Protocol (CNP) is followed for the purpose of negotiation and bidding. Necessary modifications in scheduling are taken care of by cooperation amongst the resources to deal with the disturbances.

The rest of the paper is organized as follows. Section 2 discusses multi agent based holonic manufacturing scheduling and earlier work pertaining to negotiation based scheduling. Section 3 deals with the proposed holonic manufacturing system. Priority rule along with the scheduling strategy and schedule execution is presented in Section 4. Schedule modifications under disturbances are discussed in Section 5. Results and performances are analyzed in Section 6. Finally, Section 7 concludes the paper.

2. Multi agent based holonic manufacturing and scheduling

Holonic Manufacturing System (HMS) is a subset of Intelligent Manufacturing System (IMS) [6,7] that comprises various functional entities like machine tools, robots, products, AGVs, etc. which are generically called holons. Holons are characterized by *autonomous* and *cooperative* properties. These holons are intelligent, highly agile and manifest very high degrees of flexibility by adopting negotiation and cooperation based work. A system of holons that are engaged to achieve a common goal is called a holarchy. The holonic concept utilizes hybrid control structure by combining the best features of hierarchical and heterarchical organization. It preserves the stability of a hierarchy while providing the dynamic flexibility of a heterarchy. This hierarchy, however, differs from the traditional hierarchical control in the sense that it is loose and flexible having following attributes: (a) holons can belong to multiple hierarchies, (b) holons can form temporary hierarchies, and (c) holons do not rely on the proper operation of each holon in the hierarchy to get their work done [2,6]. A holonic manufacturing system combines high performance with robustness against changes and disturbances. Metamorphism enables the holons to reconfigure in the wake of changing circumstances. Holonic systems utilize mechanism for integration of physical part of an entity with its information processing part as well as with the human decision maker(s) [11]. A holonic system also possesses structural recursivity which implies that a particular holon can be part of several other holons. Successful implementation of holonic concept in manufacturing ensures significant increase in: (i) robustness to disturbances; (ii) adaptability to rapid change; and (iii) efficient use of available resources [12].

The holonic behavior is implemented by MAS, wherein each agent has intelligent properties. An agent is envisaged as 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 [13]. Agents exhibit two contradictory properties: Under normal situation they are highly competitive and vie with each other for lucrative jobs to augment their earnings and credentials. On the contrary, they are cooperative and altruistic when disturbances prevail, implying that under disturbances they offer help to others to solve problems. Agents and holons share many common characteristics such as autonomy, cooperation and openness. A multi-agent system is conglomeration of a set of agents having heterarchical control structure capable of interacting with each other in order to achieve their individual goals, when they do not have adequate knowledge and/or skills to accomplish the same in standalone mode. MAS is fundamentally a software technology where an agent possesses behavioral attributes like *autonomy* (ability to take one's own decision by virtue of proactiveness), *social ability* (interacts with other agents by mutual message-based communication), and *reactivity* (agents can sense and respond to changes in the environment) [14,15]. Although these features help to realize high performance against disturbances, it is difficult to achieve global optimization since the decision-making is local and autonomous without a global view of the system. The expansibility of the system is easier, being only enough to modify the functioning of some agents or add new agents to the control system [13]. Such systems rely on the principle of Distributed Problem Solving (DPS) where a set of modules cooperate dynamically amongst themselves to carry out a specific task. The detailed description of holonic and multi agent based manufacturing and scheduling are discussed adequately by several authors [1,2,6,7,11–13,16–21].

2.1. Dynamic scheduling

Scheduling in manufacturing is an activity of allocating jobs to resources with respect to a time frame that considers SPT, EDD, critical ratio, etc. as criteria for establishing priority of the jobs for processing in static environments in a centralized manner and considered as N-P hard type of problem. This problem becomes more complicated in an open, distributed and dynamic platform [22], where each entity prepares its own schedule and having only partial information of a bigger problem, global optimization is difficult to achieve. Further, the occurrence of a variety of unexpected events continually forces reconsideration and revision of pre-established schedules. Thus approaches developed to solve the problem of static scheduling becomes little significant during implementation [23]. But, the need of the hour is quick responsiveness to the changes, utilization of one's own computational and decision making power and if required, dynamic collaboration with others to find any possible solution. In most of the real-world environments, scheduling is therefore envisaged as an ongoing reactive process that demands amendments in response to the changes in the environment. Techniques like Lagrangian relaxation [24], constraints satisfaction [25], heuristics [26], reinforcement learning [27], neural networks and inductive learning [28,29], genetic algorithm [30] are found to deal with dynamic scheduling problem. However, the aforementioned techniques essentially being centralized and based on simple theoretical model, fail miserably in real-life situations [18,31,32].

2.2. Multi agent based dynamic scheduling

The MAS paradigm provides a very promising approach to dynamic scheduling problem and has gained increasing attention [33]. It has several distinguished advantages [18]: (i) the platform creates opportunities for parallel and decentralized computing, (ii) individual resources tries to improve local performance that eventually leads to attain global objective, (iii) possibility of dynamic integration of planning and scheduling. The integration of process planning and scheduling helps to optimize both simultaneously as a single problem considering the constraints of both domains [32,34–36], (iv) physical part of resource agents are connected directly with the information processing part that ensures rescheduling easier and faster, and (v) schedules are achieved by negotiation rather than search that improve performance. Each holonic agent by virtue of its autonomy makes its own scheduling to achieve a common system goal. During scheduling, a holon uses the knowledge and information about itself and its surroundings.

2.3. Negotiation as scheduling mechanism

Unlike traditional systems, jobs are not assigned to the resources in an agent based system; rather resources being governed by their autonomy, negotiate on several criteria/attributes and finalize the agreement for execution. Negotiation therefore serves the basis for scheduling.

According to Davis and Smith [37], *negotiation* is envisaged as key mechanism to realize distributed problem solving, which has three distinct components: (a) there is a two-way exchange of information, (b) each party to the negotiation evaluates the information from its own perspective, and (c) final agreement is achieved by mutual selection.

CNP is a systematic procedure that establishes communication between nodes in a distributed manner for the purpose of dynamic task contracting and natural load balancing [38]. Following CNP, the negotiation between two different categories of agents take place by message based two-way communication considering product as manager and resource as contractor. Rabelo and Camarinha-Matos [39] described negotiation as a process that makes agents talk about a task until one of them is selected for its execution. In their work, negotiation takes place between business processes (BPs) and enterprise activities (EAs) following procedure rule sets (PRSs) that specifies the precedence interrelationships between BPs and EAs. Rabelo et al. [33] adopted CNP coordination mechanism to support the task assignments to agents and the negotiation method to overcome conflicts taking place during scheduling in a multi agent framework called HOLOS. Markus et al. [40] followed a market based approach to solve dynamic order processing and scheduling problem. Kanchanasevee et al. [41] followed CNP based scheduling approach to implement production plans for individual workstations of an induction motor assembly plant in a multi agent based holonic system. A 'negotiation holon' is responsible for negotiation processes between part and machine holons. Yang and Lin [42] proposed a hybrid hierarchical/heterarchical shop floor control system in job dispatching using multiple criteria – the production price and the utilization rate of each cell. A task is awarded to a cell that can execute it at minimum cost, provided the loading rate of the cell is below a threshold value. Sousa and Ramos [43] adapted CNP as a part of negotiation process to dynamically assign operations to the resources of a HMS that broadly comprises production planning and process planning holon. The work advocates for a renegotiation phase to avoid conflicts and to take care of dynamic changes in the system. Huang et al. [9] considered CNP based negotiation mechanism to offer tasks to the resources using the criteria of low-award/high-penalty in a holonic control virtual enterprise. Kotak et al. [11] opined that communication and negotiation are the means to achieve the system goal and followed the same in a material transport system where the selection of optimal path is governed by the shortest distance. Suesut [44] used multi level CNP as a means of negotiation process in an automatic warehouse system that relies on holonic philosophy. Leitao and Restivo [22] introduced a credit system as an agreement (part of negotiation) between task holons and operational holons to carry out dynamic scheduling following holonic approach. Wong et al. [36] developed an agent-based approach for the dynamic integration of the process planning and scheduling where selection of a schedule and task allocation was achieved by negotiation between part agent and machine agents. The negotiation procedure followed a hybrid CNP considering a fictitious cost as criteria. Two algorithms are developed to monitor whether the agent's individual decision are compatible with the global objectives. Lai [45] used negotiation to accommodate a rush order by offering reward/penalty scheme. Wang et al. [46] proposed an ontology based knowledge representation approach to provide a semantic interoperable environment to realize automatic negotiations in a virtual enterprise.

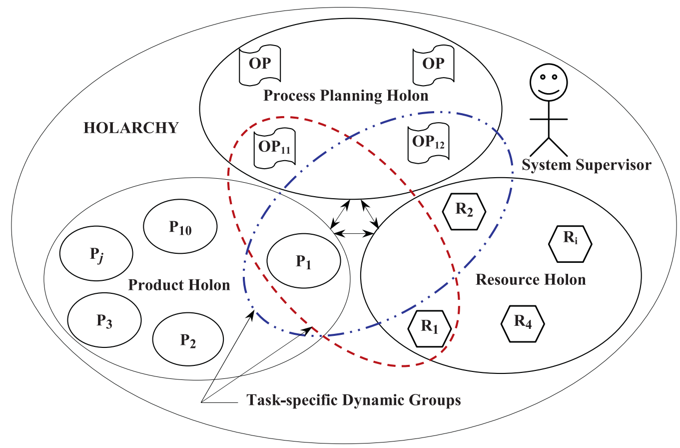

Fig. 1. Proposed holonic manufacturing system.

Hsieh and Chiang [8] used a two layer contract net protocol to describe the negotiation between order holons, product holons and resource holons in a HMS. Adam et al. [47] proposed a holonic multi-agent system (HoloMAS) to provide an adaptive control mechanism for manufacturing systems, where the local objective of a HA is accomplished by interactions with others following CNP. Renna [48] followed a multi-agent based dynamic scheduling in manufacturing cells where a part agent negotiates with the machine agent for task allocation using the criteria of improving the resource efficiency according to the manufacturing cell objectives.

3. Proposed holonic system

In the present work, a manufacturing system holarchy is developed to execute production order from customer. The proposed system comprises product holon, resource holon and process planning holon along with a system supervisor as coordinator to ensure integration of human elements in the system as shown in Fig. 1. The modus operandi of the system along with the association among different entities is depicted by UML (class) diagram in Fig. 2. The customer order is launched by the system supervisor and products are created and consequently the information are passed to the resource holon. System supervisor establishes harmony amongst them in pursuit of the goal of the organization. Any modification in the customer order, say quantity changes are also conveyed to the resources to make necessary adjustments. Different holons in turn reciprocate with the action taken reports so as to adhere bottom-up approach. However, in view of disturbances, the holons may deny the system supervisor and can take their own decision by virtue of autonomy and cooperation.

Product holon takes the initiative for necessary execution of the products by the resources. The activities of the product holon includes getting the list of operations from process planning holon, negotiation with the resources for getting the work done, and monitoring the progress of conversion. It also interacts with other resources, if required, under changing situations (disturbances). Ten different products ($P_1 \ldots P_{10}$) are considered for manufacturing.

Resource holon consists of four Turning Centers ($R_1 \ldots R_4$) of varying capacity and specification. A resource holon controls the machine tool (physical part) and monitors the entire range of activities of the machine. The information processing part is responsible for assessing its technical abilities in response to a task; computation of machining time based on operation plan, preparing agenda of work, monitoring current process status in regard to scheduling, and looking into the availability for future requirements. Further,

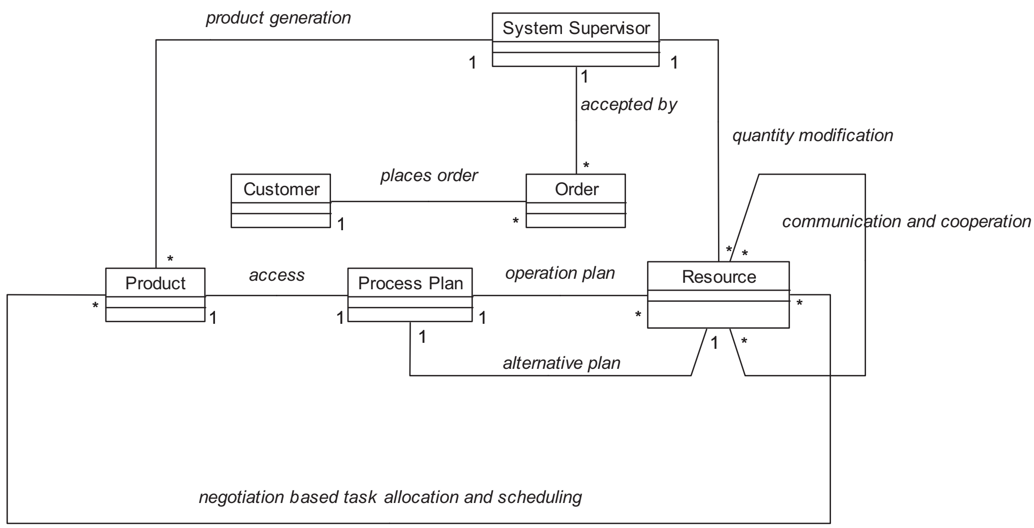

Fig. 2. Class diagram showing association among different entities.

resources interact with each other to achieve cooperation based work under changing circumstances.

Process planning holon generates the required operation plan considering raw material quality and size of the blank, facilities and tooling available for a particular resource. Once the product is launched and identified by its code, this holon prepares the operation plan and issues the same to the product as well as to the resources for necessary negotiation based task allocation. The alternative operation plan is also developed in view of modifications in facilities (e.g. changes in the cutting tool call for modifications of the operational parameters). The proposed holonic control manifests that a coherent cluster is formed mutually by holons belonging to different groups to achieve a specific goal.

4. Scheduling technique

This section presents formulation of priority rule for the product scheduling, negotiation protocol for task allocation, the means for communication, generic schedule strategy and its implementation in the proposed system. The schedule is accomplished by a two-tier procedure: (i) evolving a priority rule from multiple perspectives and (ii) developing a scheduling strategy.

4.1. Evolving priority rule

In the proposed system, each product negotiates with the resources one by one and bargains on several attributes through multiple iterations and finalizes the agreement for execution. The products belong to a particular customer order and are generated at a time with an expectation that these are to be completed within a single deadline. The simultaneous creation of products gives rise to a conflicting situation since everyone tries to be processed at the earliest. Further, the potential resources also vouch for processing attractive and sophisticated products. To resolve this issue, a product ranking strategy based on their merits is adopted so that the products would be given preference for negotiation in the descending order of merit list (priority rule). Maione and Naso [49] viewed negotiation tasks are inherently multi-objective in micro-economic environments and expert contractors always tend to find a rewarding compromise between conflicting aspects. We adopt Multi-Objective Optimization on the basis of Ratio Analysis (MOORA) under Fuzzy Multi Criteria Decision Making (FMCDM) environment for product ranking as the method provides certain advantages: (i) the method is simple, easy to implement, and reliable, (ii) the foundation of the method is based on simple ratio analysis and involves least mathematical computation and therefore amenable for application in MAS environment, and (iii) the analysis of MOORA method is quite robust since its calculation procedure is not affected by the introduction of any extra parameter.

Table 1
Degree of performance rating.

Linguistic variable	Abbreviation	Triangular fuzzy number
Very poor	VP	(0, 1, 3)
Poor	P	(1, 3, 5)
Medium	M	(3, 5, 7)
Good	G	(5, 7, 9)
Very good	VG	(7, 9, 10)

The MOORA method employs a ratio system in which each performance rating of an alternative on a criterion is compared to a denominator which is a representative of all alternatives concerning that criterion [50,51]. For this denominator, the best choice is the square root of the sum of squares of each alternative per objective. The units of performance ratings as well as their range of magnitude being different are normalized to convert the criteria into dimensionless attributes in the range 0–1. The MOORA method relies on the reference point approach that implies that the chosen alternative should have the highest composite score which represent the difference between sum of benefit (SOB) and sum of non-benefit (SONB) scores.

Five degrees of linguistic variables and their corresponding triangular fuzzy numbers (TFN) for both responses (performance rating) of alternatives and criteria weights are shown in Tables 1 and 2, respectively.

The stepwise procedures of MOORA technique is presented below.

Table 2
Degree of weights of criteria.

Linguistic variable	Abbreviation	Triangular fuzzy number
Very low	VL	(0, 0.1, 0.3)
Low	L	(0.1, 0.3, 0.5)
Moderate	M	(0.3, 0.5, 0.7)
High	H	(0.5, 0.7, 0.9)
Very high	VH	(0.7, 0.9, 1.0)

Step 1: Formation of decision matrix with performance scores. Performance score or performance rating is the value of alternative on each criterion provided by the decision maker.

$$\begin{array}{cc} & \begin{array}{ccccc} C_1 & \dots & C_j & \dots & C_n \end{array} \\ D = \begin{array}{c} A_1 \\ \dots \\ A_i \\ \dots \\ A_m \end{array} & \begin{bmatrix} x_{11} & \dots & x_{1j} & \dots & x_{1n} \\ \dots & \dots & \dots & \dots & \dots \\ x_{i1} & \dots & x_{ij} & \dots & x_{in} \\ \dots & \dots & \dots & \dots & \dots \\ x_{m1} & \dots & x_{mj} & \dots & x_{mn} \end{bmatrix} \end{array} \quad (1)$$

where x_{ij} is the performance rating of alternative A_i with respect to criterion C_j m is the number of alternatives and n is the number of criteria. Here the performance rating x_{ij} is either crisp or fuzzy expressed as $x_{ij} = (a_{ij}, b_{ij}, c_{ij})$.

Step 2: Formation of weight matrix.

$$W = [\,\tilde{w}_1 \quad \dots \quad \tilde{w}_j \quad \dots \quad \tilde{w}_n\,] \quad (2)$$

where $\tilde{w}_j = (\alpha_j, \beta_j, \gamma_j)$ is weight of criterion j.

Step 3: Defuzzification of performance score by the equation

$$\bar{x}_{ij} = \frac{1}{3}(a_{ij} + b_{ij} + c_{ij}) \quad (3)$$

Step 4: Defuzzification of weight

$$\bar{w}_j = \frac{1}{3}(\alpha_j + \beta_j + \gamma_j) \quad (4)$$

Step 5: Construct final defuzzified decision matrix

$$D' = [x'_{ij}]_{m \times n} \quad (5)$$

where $x'_{ij} = x_{ij}$ for objective criteria and $x'_{ij} = \bar{x}_{ij}$ for subjective criteria.

Step 6: Normalization of score of alternative is obtained using following equation

$$r_{ij} = \frac{x'_{ij}}{\sqrt{\sum_{i=1}^{m} \left(x'^{2}_{ij}\right)}} \quad (6)$$

Step 7: Normalization of importance weights of criteria is carried out using following equation

$$w_j^N = \frac{\bar{w}_j}{\sum_{j=1}^{n} \bar{w}_j}, \quad \text{where} \quad \sum_{j=1}^{n} w_j^N = 1 \quad (7)$$

Step 8: Computations of sum of benefit criteria measures (SoB) and sum of non-benefit criteria measures (SoNB) by following equations

$$\mathrm{SoB} = \sum_{j \in B} w_j^N r_{ij} \quad (8)$$

$$\mathrm{SoNB} = \sum_{j \in NB} w_j^N r_{ij} \quad (9)$$

where w_j^N is the normalized weight of jth criterion.

Step 9: Computation of composite score. In most of the real life problems different weights are given to the attributes of the alternatives as per their relative importance. When the weights of attributes are taken into consideration then Eq. (3) can be expressed as

$$y_i^* = \sum_{j \in B} w_j^N r_{ij} - \sum_{j \in NB} w_j^N r_{ij} \quad (10)$$

where y_i^* is composite score of alternative i. The value of y_i^* may be positive, negative or zero. The best alternative is one which is associated with the highest y_i^* value and the worst alternative is one which is associated with the lowest y_i^* value.

Step 10: Ranking the products in descending order of composite score y_i^*.

4.2. *Task contracting and scheduling*

The products get the opportunity for negotiation with the resources according to the merit list, settle the agreement by negotiation and prepare the local schedules.

4.2.1. *Negotiation process*

During negotiation, products as well as resources communicate with the process planning holon to get the required operational plan. Negotiation between a product and resources is a two-tier process. In the first level, the product announces the task(s) along with the eligibility criteria and other attributes (e.g. quality of raw material, quantity, etc.) and the resources respond as "Yes/No"

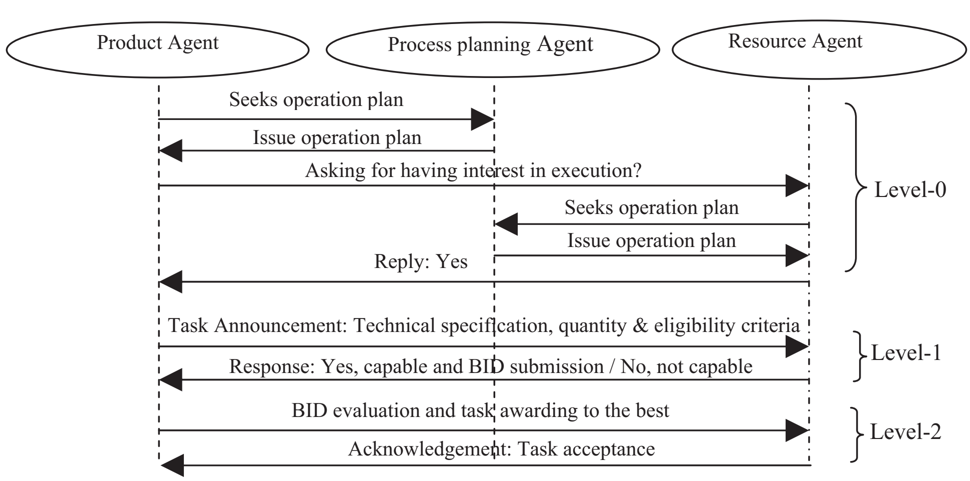

Fig. 3. Sequence diagram showing message transfer among agents.

depending on the ability to carry out the task. If the response is affirmative, the reply is accompanied by a BID. In the second phase, the product evaluates the BID, selects the best one and awards the task. The selected resource finally reciprocates with an acceptance message to complete the negotiation. However, if the best one is not available due to preoccupancy in other work, the product selects the next one following the outcome of the BID evaluation. A generalized message transfer scheme among three different categories of agents is shown by sequence diagram in Fig. 3.

4.2.2. Communication

Communication among different agents is envisaged to be accomplished through J2EE technology using XML language following a specific ontology. J2EE is a platform which provides transactional, secularized and other facilities. It is a technology used to develop enterprise application like 2, 3 or *n*-tier architecture using a client/server model over distributed enterprise environment and also for developing web applications through TCP/IP protocol as shown in Fig. 4. The schema of document type definitions (DTD's) for messages related to task announcement by the products to any resource **(Announcement_1.dtd)** and the responses **(Response_1.dtd)** are presented below.

Announcement_1.dtd

```
<!ENTITY %AnnouncementName "CDATA">
<!ENTITY %ProductId "CDATA">
<!ENTITY %Type "(Product|Resource)">
<!ENTITY %Specification "(Turning|Facing|Grooving|Profiling)">
<!ELEMENT Announcement (ProductAnnouncement)>
<!ELEMENT ProductAnnouncement (Type,To,From,MaterialQuality,
LotSize,TaskEligibility,TaskSpecifications)>
<!ATTLIST ProductAnnouncement name %AnnouncementName; #REQUIRED>
<!ATTLIST ProductAnnouncement product_id %ProductId; #REQUIRED>
<!ATTLIST ProductAnnouncement id ID #REQUIRED>
<!ELEMENT Type (%Type;)>
<!ELEMENT To (#PCDATA)>
<!ELEMENT From (#PCDATA)>
<!ELEMENT MaterialQuality(#PCDATA)>
<!ELEMENT LotSize(#PCDATA)>
<!ELEMENT TaskSpecifications (TaskSpecification*)>
<!ELEMENT TaskSpecification(%Specification;)>
<!ELEMENT TaskEligibility(#PCDATA)>
```

Response_1.dtd

```
<!ENTITY %ResponseName "CDATA">
<!ENTITY %Type "(Product|Resource)">
<!ENTITY % ResponseType "(Bid|Confirmation)">
<!ENTITY % Boolean "(true|false|yes|no|confirm|reject)">


<!ELEMENT Response (ResponseData)>
<!ELEMENT ResponseData (Type,
ResponseType,AnnouncementId,ResponseId,To,From,Decision,Message,Time)>
<!ATTLIST   ResponseData name %ResponseName; #REQUIRED>
<!ATTLIST   ResponseData id ID #REQUIRED>
<!ELEMENT ResponseType (%ResponseType; )>
<!ELEMENT Type (%Type; )>
<!ELEMENT ResponseId (#PCDATA)>
<!ELEMENT AnnouncementId (#PCDATA)>
<!ELEMENT To (#PCDATA)>
<!ELEMENT From (#PCDATA)>
<!ELEMENT Decision (%Boolean;)>
<!ELEMENT Message(#PCDATA)>
<!ELEMENT Time(#PCDATA)>
```

4.2.3. Scheduling strategy

The proposed scheduling considers following assumptions.

(i) There is no set up time and no pre-emption.
(ii) Products are to be manufactured in batch mode unless there is any resource breakdown.
(iii) No resource would remain idle.

The products are allowed to negotiate with the resources sequentially according to the rank and consider *the lowest operation time* as criterion for resource selection. Due to resource constraints (the number of resources is less than the number of products); all the products cannot be allocated to the resources at the 1st instance which results in formation of product queue. The resource that becomes free at the earliest deals with the next product. This procedure is repeated for other products that are waiting. The resources thus prepare their local schedules from which global schedule is prepared.

The initial negotiation process for all the products is completed before commencement of actual processing. This is necessary to see the prima-facie complete global picture of the schedule under normal condition to facilitate the system supervisor to compute the earliest possible completion time and to affix the deadline. It is noteworthy that outcome of this negotiation is liable to changes in the event of changing circumstances. System supervisor establishes a deadline following forward scheduling approach considering the longest time taken by any resource(s) and an allowance of 10% to take care of any deviation. This deadline is intimated to the resources so that the products can be completed within this time frame.

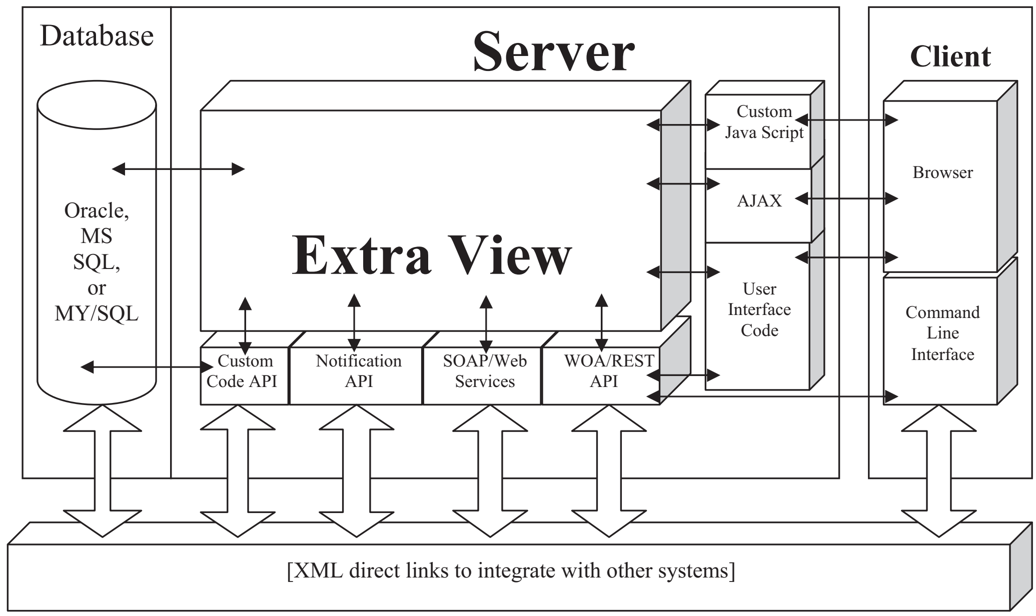

Fig. 4. J2EE communication diagram using XML language.

4.3. Schedule implementation

This subsection presents implementation of the scheduling technique in the proposed holonic manufacturing system.

4.3.1. Product ranking

In order to streamline the negotiation process the products are ranked according to their merit as evaluated by MOORA considering several attributes of different importance. The criteria and their weights are chosen in such a manner that the interests of the products (hence the customer), the resources, and the organization are satisfied.

4.3.1.1. Evaluating the criteria. The various criteria considered here are *Revenue earned* (C_1), *Urgency* (C_2), *Complexity of work* (C_3) and *Precision* (C_4).

- *Revenue earned* (C1): Revenue is perhaps the most attractive criterion for an organization. Whenever a product is processed successfully, the organization earns revenue. The revenue that can be earned from a product is computed by the following relationship.

Revenue = Material cost + Conversion (machining) cost + Profit, where profit is considered as 20% of the total cost.

Since revenue is a function of the machining cost, resources would be able to recover the machining cost incurred and can enjoy a fraction of profit after successful completion of a job. This attribute is therefore considered to satisfy the requirements of the resources as well as the system. A product is considered lucrative if high revenue is earned and hence revenue is viewed as benefit criteria. It is expressed in Indian Rupee.

- *Urgency* (C2): The products are generated at a time and it is expected that these are to be completed within a single deadline. Therefore, the concept of exact due date seems incompatible in this situation. Further, unlike traditional systems, the exact processing time is not known at this stage since it would be provided by the resources during later phase of bidding. Hence rules like SPT, EDD, and CR etc. can't be used to evaluate the relative merit of the products regarding the time frame of execution. We therefore, introduce a generic and vague criterion urgency to capture the relative importance of the products to establish preference for execution. Products requiring significant time for post machining activities like cleaning, deburring, dimension checking, etc. are assigned with high urgency level and expressed in terms of linguistic variables as per Table 1. Thus higher urgency of a product implies that the product would be given higher priority and therefore urgency is envisaged as benefit criteria.
- *Complexity* (C3): Complexity refers to the intricate geometry of the products. Since the agents are intelligent, a capable resource tries to augment its reputation and credibility by undertaking products involving intricate shapes. A resource therefore, considers this attribute as attractive and hence it is considered as benefit criteria. Complexity is a qualitative attribute and therefore it is expressed in linguistic term (according to Table 1) and its values are assigned by investigating product geometry.
- *Precision* (C4): Precision refers to a very stringent or close dimensional tolerance that has to be maintained as specified in the part drawing. Thus a product requiring high precision is attractive to the capable and sophisticated resources. Accordingly, lower numeric value of tolerance is desirable and viewed as non beneficial criteria and it is expressed in mm.

4.3.1.2. Assigning weights to the attributes. We assign the importance of different attributes in terms of fuzzy linguistic variables as per Table 2. The weights are chosen in a manner to satisfy the intended need of different communities such as the product, the resource, and the organization as a whole following the business rule. Products are given the paramount importance because they belong to the customer. Since products are interested about early processing, 'very high (VH)' weight is assigned to urgency. Higher revenue earning is the prime objective of the organization and the resources and therefore, 'high (H)' weight is provided to the revenue. The other two attributes namely complexity of work and precision are primarily relevant to the resource goal (augmentation of credibility and reputation) and therefore both are given 'moderate (M)' weight.

The initial decision matrix combined with weight matrix is shown in Table 3. Normalized decision matrix along with defuzzified and normalized weight matrix is presented in Table 4. The MOORA algorithm is executed in MATLAB 7.5 to yield results. The

Table 3
Initial decision matrix combined with weight matrix (in linguistic variables).

Product	Criteria →	C_1 (+)	C_2 (+)	C_3 (+)	C_4 (−)
	Weight →	H	VH	M	M
P_1		1973	VG	G	0.06
P_2		1350	G	G	0.02
P_3		1450	M	G	0.20
P_4		685	M	P	0.80
P_5		1305	G	M	0.20
P_6		1100	G	G	0.10
P_7		1207	M	P	0.12
P_8		756	M	M	0.04
P_9		840	P	G	0.08
P_{10}		320	G	M	0.06

NB: Benefit criteria are marked (+) and non benefit criteria are marked (−).

Table 4
Normalized decision matrix along with defuzzified and normalized weight matrix.

Product	Criteria →	C_1 (+)	C_2 (+)	C_3 (+)	C_4 (−)
	Weight →	0.2727	0.3376	0.1948	0.1948
P_1		0.5267	0.4447	0.3807	0.0688
P_2		0.3604	0.3590	0.3807	0.0229
P_3		0.3871	0.2564	0.3807	0.2294
P_4		0.1829	0.2564	0.1632	0.9177
P_5		0.3481	0.3590	0.2720	0.2294
P_6		0.2936	0.3590	0.3807	0.1147
P_7		0.3222	0.2564	0.1632	0.1376
P_8		0.2018	0.2564	0.2720	0.0459
P_9		0.2242	0.1539	0.3807	0.0918
P_{10}		0.0854	0.3590	0.2720	0.0688

weighted normalized decision matrix, SoB, SoNB, y_i^* and ranking of products are shown in Table 5.

4.3.2. Negotiation based schedule execution

For the proposed system, it is considered that each product would be manufactured in two numbers in a batch mode. P_1 having the highest y_i^*, gets the first opportunity for negotiation. P_1 is a long shaft (quality: mild steel) that requires operations like turning, facing, grooving and profiling. To carry out these operations, a resource must be equipped with steady rest.

A sample task announcement message from product P_1 to resource R_1 based on 'announcement_1 DTD' is provided in Appendix A. A similar message is also issued to R_2, R_3 and R_4.

R_1 verifies its capability in terms of task specifications and responses with a decision "yes" and associated message "capable" along with the execution time as *BID*. The machining time (minutes) of various product-resource combinations are shown in Table 6. The message issued from R_1 following 'Response_1 DTD' is presented in Appendix B.1.

R_2 and R_4 also respond in affirmative with only difference in machining time which is 32 min and 40 min respectively. However, R_3 cannot participate in bidding since the work is beyond its capacity (R_3 is a smaller machine of shorter bed length) and hence regrets. The regret message from R_3 is presented in Appendix B.2.

P_1 evaluates the bids of R_1, R_2, and R_4 and selects R_1. The message from P_1 for awarding task to R_1 is presented in Appendix B.3. Finally, R_1 acknowledges the agreement by sending message "accepted" to complete the information exchange. Similar procedure is followed for other products.

Following the proposed scheduling strategy, the entire scheduling is done. Once the negotiation and task allocations are completed, the system supervisor computes and establishes deadline as 108 min (98 min + 10% allowance, considering R_2 as the critical one) which is conveyed to the resources along with the instruction to initiate execution. The complete local schedules and global schedule is shown in Fig. 5.

Table 6
Machining time (minutes) of various product-resource combination.

Product	Resource			
	R_1	R_2	R_3	R_4
P_1	26	32	–[a]	40
P_2	15.5	17	19	23
P_3	8	9	12	14
P_4	4	5	6.5	8
P_5	9	10	14	15
P_6	18	20	23	25
P_7	14	16	20	22
P_8	9	10	13	16
P_9	20	22	25	27
P_{10}	15	17	20	22

[a] An infinitely large positive value is assigned against $(\tau_{op})_{P_{(1)/R_3}}$ in the algorithm.

5. Cooperation based schedule modifications under disturbances

Disturbances refer to random occurrence of events such as rush order, cancelation/modification of ordered quantity, machine malfunctioning associated with longer down time, change in delivery pattern and priority, addition/alteration of resources etc. that hinders pursue the goal and causes perturbations to the system. Presence of disturbances calls for modifications in the earlier schedule to combat the situation. Agents being *smart* and *intelligent*, embark on the challenges posed by the disturbances. The schedule modifications are usually done at local level independently, and if it is not possible, resources cooperate with each other to find any solution.

In the present work, two different types of disturbances are dealt with – (a) alteration of initial ordered quantity and (b) machine malfunctioning.

The challenge is to get solutions in the face of changing circumstances without exceeding the deadline.

Table 5
Weighted normalized decision matrix, SoB, SoNB, y_i^* and ranking.

Product	Criteria →	C_1 (+)	C_2 (+)	C_3 (+)	C_4 (−)	SoB	SoNB	y_i^*	Rank
P_1		0.1436	0.1502	0.0742	0.0134	0.3680	0.0134	0.3545	1
P_2		0.0983	0.1212	0.0742	0.0045	0.2937	0.0045	0.2892	2
P_3		0.1056	0.0866	0.0742	0.0447	0.2663	0.0447	0.2216	5
P_4		0.0499	0.0866	0.0318	0.1788	0.1683	0.1788	−0.011	10
P_5		0.0949	0.1212	0.0530	0.0447	0.2691	0.0447	0.2245	4
P_6		0.0801	0.1212	0.0742	0.0223	0.2755	0.0223	0.2531	3
P_7		0.0879	0.0866	0.0318	0.0268	0.2062	0.0268	0.1794	8
P_8		0.0550	0.0866	0.0530	0.0089	0.1946	0.0089	0.1857	6
P_9		0.0611	0.0520	0.0742	0.0179	0.1873	0.0179	0.1694	9
P_{10}		0.0233	0.1212	0.0530	0.0134	0.1975	0.0134	0.1841	7

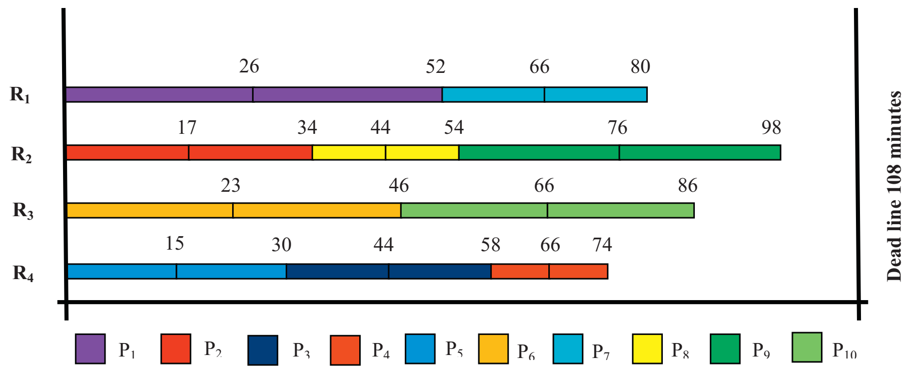

Fig. 5. Schedule under normal condition (N).

5.1. Modified scheduling due to alteration of initial ordered quantity

Alteration of initial ordered quantity combinations [for few products, quantity may increase and for others quantity may decrease] leads to disturbances in the system. Schedule modifications under these circumstances are based on the following premises.

(a) If a particular product is already under process, it is to be completed first.
(b) A resource that deals with a particular product is responsible for attempting extra quantity of the same. If it is not possible to do it completely, then it seeks help from others for balance quantity.
(c) A resource cannot off-load its agreed quantity (as finalized through negotiation) of products to others unless it malfunctions.
(d) Resources accept the proposal for extra quantity of any product provided the execution of agreed quantity for all other products (including those waiting in the queue, if any) are also possible within the same stipulated time.
(e) Quantity changes are intimated only when all the resources are engaged with their first category of product.

A cooperative problem solving algorithm is presented to look into the feasibility of accommodating changes considering m number of resources and n different types of products such that $m \le n$.

The algorithm is essentially three-tier – (i) the algorithm [1(a)] identifies how many extra quantity of the initially assigned (ongoing) products are not possible by the corresponding resources, (ii) second phase of the algorithm [1(b)] tries to get a solution by others in view of the situation (i), and (iii) the last part of the algorithm [1(c)] verifies the possibility of processing those products [only when $m < n$] which are waiting, within the same deadline.

The executions of the algorithm in two different occasions are shown by the flow charts presented in Fig. 6(a) and (b).

Once the modifications are announced; the quantity of each product due for execution are determined by taking the algebraic sum of quantity initially ordered, quantity altered, and the quantity already executed (including those that are undergoing processing) at the time of computation. The resources now look into the possibility of executing different products that are due within the deadline. This is accomplished by computing *Left out Time* (LOT) recursively for all the resources separately and by verifying the number of different products that can be done. LOT of any resource is defined as the time remaining to reach the deadline after completion of the present job or at the instant envisaged. Finally, decision is taken unanimously regarding the acceptance of the new proposal.

Mathematically, $[q_{P_j}]_{due} = [q_{P_j}]_{initial} + [q_{P_j}]_{alt} - [q_{P_j}]_{exec}$, such that $[q_{P_j}]_{due} > 0$.

Following symbols are used in algorithm 1(a):

$R_{(i)}$ ordered rank of resources where $i = 1, ..m$
$P_{(j)}$ ordered rank of products where $j = 1, ..n$
$LOT_{R(i)}$ LOT of ordered rank of resources $R_{(i)}$
$(\tau_{op})_{P_{(j)}/R_{(i)}}$ operation time of $P_{(j)}$ by resource $R_{(i)}$
$[q_{P_{(j)}}]_{exec}$ quantity of $P_{(j)}$ executed by $R_{(i)}$ at the instant considered
$[q_{P_{(j)}}]_{poss/R_{(i)}}$ quantity of $P_{(j)}$ possible by $R_{(i)}$
$[q_{P_{(j)}}]_{Np/R_{(i)}}$ quantity of $P_{(j)}$ not possible by $R_{(i)}$
$[q_{P_{(j)}}]^B$ balance quantity of $P_{(j)}$ that would be carry forwarded to algorithm 1(b)

Algorithm 1(a)

Step 0: $R_{(i)} \leftarrow P_{(j)}$ [This implies ordered rank of product $P_{(j)}$ is assigned to ordered rank of resource $R_{(i)}$]
Step 1: set $i = 1$
Step 2: set $j = 1$
Step 3: while ($i \le m$ and $j \le n$) do Step 4 to Step 12
Step 4: $LOT_{R_{(i)}} = [Deadline] - [q_{P_{(j)}}]_{exec}(\tau_{op})_{P_{(j)}/R_{(i)}}$
Step 5: $[q_{P_{(j)}}]_{poss/R_{(i)}} = \text{int}\left[\frac{LOT_{R_{(i)}}}{(\tau_{op})_{P_{(j)}/R_{(i)}}}\right]$
Step 6: If $[q_{P_{(j)}}]_{poss/R_{(i)}} < [q_{P_{(j)}}]_{due}$ then Go to Step 10
Step 7: $LOT_{R_{(i)}} = LOT_{R_{(i)}} - [q_{P_{(j)}}]_{due}(\tau_{op})_{P_{(j)}/R_{(i)}}$
Step 8: set $[q_{P_{(j)}}]^B = 0$
Step 9: Go to Step 13
Step 10: $LOT_{R_{(i)}} = LOT_{R_{(i)}} - [q_{P_{(j)}}]_{poss}(\tau_{op})_{P_{(j)}/R_{(i)}}$
Step11: $[q_{P_{(j)}}]_{Np/R_{(i)}} = [q_{P_{(j)}}]_{due} - [q_{P_{(j)}}]_{poss/R_{(i)}}$
Step 12: set $[q_{P_{(j)}}]^B = [q_{P_{(j)}}]_{Np/R_{(i)}}$
Step 13: $i = i + 1$
Step 14: $j = j + 1$
[end of Step3 loop]
Step 15: If ($[q_{P_{(j)}}]^B > 0$ and $m \le n$) for any $R_{(i)} \leftarrow P_{(j)}$, then refer to algorithm 1(b)
Step 16: If ($[q_{P_{(j)}}]^B = 0$ and $m < n$) for all $R_{(i)} \leftarrow P_{(j)}$, then refer to algorithm 1(c)
Step 17: Decision ***"possible"*** and "***accept the proposal"*** [when $[q_{P_{(j)}}]^B = 0$ and $m = n$]
Step 18: exit

Following symbols are used in algorithm 1(b):

$R_{(i)}$ the ordered rank of resource where $i = 1, ..m$ and $P_{(j)}$ is the ordered rank of products where $j = 1, ..m$
$\underset{i \neq j}{LOT_{R_{(i)}}}$ left out time of resource $R_{(i)}$ where $i \neq j$
$(\tau_{op})_{P_{(j)}/R_{(i)}}$ operation time of $P_{(j)}$ by $R_{(i)}$ such that $i \neq j$
$\underset{i \neq j}{[q_{P_{(j)}}]}_{poss/R_{(i)}}$ quantity of $P_{(j)}$ possible by $R_{(i)}$ such that $i \neq j$
$[q_{P_{(j)}}]^B$ balance quantity of $P_{(j)}$ carry forwarded from algorithm 1(b)

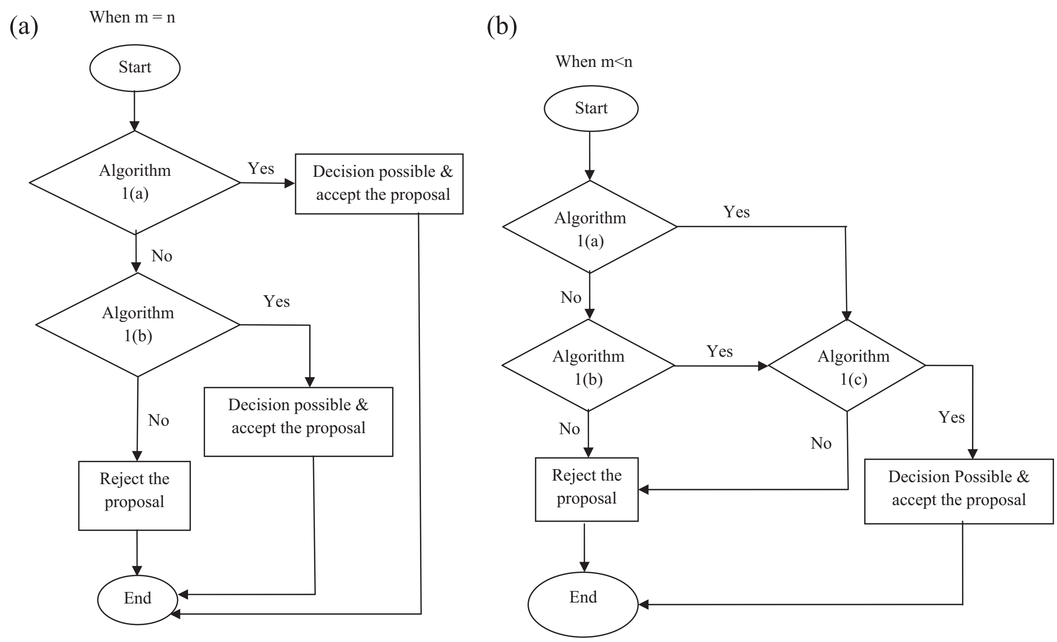

Fig. 6. (a) Flowchart showing execution of algorithms when $m = n$. (b) Flowchart showing execution of algorithms when $m < n$.

Algorithm 1(b)

Step 1: $LOT_{R_{(i)}}$ of algorithm 1(a) is now brought forwarded to algorithm 1(b)

Step 2: set $j = 1$

Step 3: while $j \leq m$ do Step 4 to Step 13

Step 4: Repeat Step 5 to Step 11 for $i = 1, ..m$ where $i \neq j$

Step 5: $\underset{i \neq j}{[q_{P_{(j)}}]}_{poss/R_{(i)}} = \text{int}\left[\frac{LOT_{R_{(i)}}}{(\tau_{op})_{P_{(j)}/R_{(i)}}}\right]$

Step 6: if $\underset{i \neq j}{[q_{P_{(j)}}]}_{poss/R_{(i)}} < [q_{P_{(j)}}]^B$ then Go to Step 10

Step 7: $\underset{i \neq j}{LOT_{R_{(i)}}} = \underset{i \neq j}{LOT_{R_{(i)}}} - (\tau_{op})_{P_{(j)}/R_{(i)}}[q_{P_{(j)}}]^B$

Step 8: set $[q_{P_{(j)}}]^B = 0$

Step 9: Go to Step 13

Step 10: $\underset{i \neq j}{LOT_{R_{(i)}}} = \underset{i \neq j}{LOT_{R_{(i)}}} - (\tau_{op})_{P_{(j)}/R_{(i)}}[q_{P_{(j)}}]_{poss/R_{(i)}}$

Step 11: $[q_{P_{(j)}}]^B = [q_{P_{(j)}}]^B - [q_{P_{(j)}}]_{poss/R_{(i)}}$

[end of Step 4 loop]

Step 12: if $[q_{P_{(j)}}]^B > 0$ for any j where $j = 1, ..m$, then Go to Step 15

Step 13: $j = j + 1$

[end of Step 3 loop]

Step 14: If $m < n$ then Decision "***may be possible***" and then refer to ***Algorithm 1(c)***.
else [when $m = n$]
Decision "***possible***" and exit
[end if]

Step 15: Decision "***not possible***" and exit

Algorithm 1(c)

Step 1: $R_{(i)}$ is defined as ordered rank of resources where $i = 1, ..m$ and $P_{(j)}$ is the ordered rank of products where $j = m + 1, m + 2, ..., n$.

Step 2: $LOT_{R_{(i)}}$ of algorithm 1(b) is now brought forwarded to algorithm 1(c).

[If algorithm 1(b) becomes redundant then $LOT_{R_{(i)}}$ of algorithm 1(a) would be used.]

Step 3: set $j = m + 1$

Step 4: while $j \leq n$ do Step 5 to Step 14

Step 5: Repeat Step 6 to Step 12 for $i = 1, ..m$

Step 6: $[q_{P_{(j)}}]_{poss/R_{(i)}} = \text{int}\left[\frac{LOT_{R_{(i)}}}{(\tau_{op})_{P_{(j)}/R_{(i)}}}\right]$

Step 7: if $[q_{P_{(j)}}]_{poss/R_{(i)}} < [q_{P_{(j)}}]_{due}$ then Go to Step 11

Step 8: $LOT_{R_{(i)}} = LOT_{R_{(i)}} - (\tau_{op})_{P_{(j)}/R_{(i)}}[q_{P_{(j)}}]_{due}$

Step 9: set $[q_{P_{(j)}}]_{due} = 0$

Step 10: Go to Step 14

Step 11: $LOT_{R_{(i)}} = LOT_{R_{(i)}} - (\tau_{op})_{P_{(j)}/R_{(i)}}[q_{P_{(j)}}]_{poss/R_{(i)}}$

Step 12: $[q_{P_{(j)}}]_{due} = [q_{P_{(j)}}]_{due} - [q_{P_{(j)}}]_{poss/R_{(i)}}$

[end of Step 5 loop]

Step 13: if $[q_{P_{(j)}}]_{due} > 0$ for any j where $j = m + 1, m + 2, ..., n$, then Go to Step 16

Step 14: $j = j + 1$

[end of Step 4 loop]

Step 15: Decision "***accept the proposal***" and exit.

Step 16: Decision "not possible" and exit.

The proposed algorithm is efficient enough to verify the feasibility of accommodating changes in the volume-mix without extending the deadline by virtue of cooperation. The algorithm is advantageous in many respects as compared to the other available algorithms reported in [19,31,36,42]. (i) The complete algorithm satisfies multiple objective of different communities with optimum results, (ii) the algorithm ensures better resource utilization and load balancing without necessitating any additional procedure, when work load is increased, (iii) while attempting for extra quantity of the current products, the commitment of in time execution of other products waiting in the queue is not sacrificed, and (iv) the deadline, as set by the supervisor, is never violated since the algorithm takes it into account from the very beginning. Therefore, the solutions are automatically in good agreement with the global objective.

5.1.1. Message protocol for seeking help

When a particular resource seeks help from others in view of attempting extra quantities of a product, a request message is issued following a separate DTD schema **(RequestDelegation_1.dtd)** as presented below to delegate the task to others.

RequestDelegation_1.dtd

```
<!ENTITY %RequestName "CDATA">
<!ENTITY %ResourceId "CDATA">
<!ENTITY %Type "(Product|Resource)">
<!ELEMENT Request (ResourceRequest)>
<!ELEMENT ResourceRequest (Type,To,From,Products)>
<!ATTLIST ResourceRequest name %RequestName; #REQUIRED>
<!ATTLIST ResourceRequest id ID #REQUIRED>
<!ATTLIST ResourceRequest resource_id ID #REQUIRED>
<!ELEMENT Type (%Type; )>
<!ELEMENT To (Resources)>
<!ELEMENT From (#PCDATA)>
<!ELEMENT Resources (Resource*)>
<!ELEMENT Resource (#PCDATA)>
<!ELEMENT Products (Product*)>
<!ELEMENT Product (AdditionalProductName,AdditionalProductQty)>
<!ELEMENT AdditionalProductName (#PCDATA)>
<!ELEMENT AdditionalProductQty(#PCDATA)>
```

5.2. Modified scheduling when a resource confronts any malfunctioning

When a particular resource malfunctions and breaks down completely, it tries to recover at the earliest. If it is diagnosed that recovery is not possible in a short while, the faulty resource communicates with others to take the onus of incomplete work. However, success or failure to get rid of such situation depends on how much progress has already been made at the time of failure. If the lag or delay is found to be trivial and can be mitigated, resources try to avail modified process plan, if any, to expedite the execution time. Failure to provide solution calls for extension of the deadline.

It may so happen that some accessories/components (e.g. steady rest, tail stock, indexing mechanism, etc.) of a resource do not work properly and put constraints in view of specific products and specific operations. Under this situation it can partially progress the work to its maximum possible extent and then seeks help from others for the remaining work.

5.3. Implementation

In order to validate the proposed strategies for handling disturbances, few case studies have been conducted. Four different scenarios' (D1 to D4) are considered for investigating the outcome of the proposed cooperative problem solving algorithm in regard to product volume-mix changes. We consider another situation (D5) that deals with partial malfunctioning of a machine without any change in the initial ordered quantity. The product-volume combinations under disturbances and under normal situation are presented in Table 7.

Table 7
The product volume-mix in different situations.

Scenario	P_1	P_2	P_3	P_4	P_5	P_6	P_7	P_8	P_9	P_{10}
N	2	2	2	2	2	2	2	2	2	2
D1	5	2	2	2	1	2	2	2	2	2
D2	1	7	2	4	3	2	2	2	2	2
D3	2	2	3	4	2	5	2	1	2	3
D4	3	2	2	2	2	2	3	2	3	2
D5	2	2	2	2	2	2	2	2	2	2

5.3.1. Product volume-mix changes (D1 to D4)

The modifications of the quantity changes for the ongoing products are intimated to the corresponding resources. If help is required while attempting for extra quantities, the resources communicate with others. A sample message based on 'RequestDelegation_1.dtd' is developed (when R_1 is processing P_1 in scenario D1) and presented in Appendix C.

The proposed cooperative problem solving algorithm is implemented using HTML code in front-end with JSP program through Apache Tomcat 6.02 server. The JSP output helps visualize different product-resource combinations along with the time left for resources that helps in schedule preparation. Additionally, if the proposed volume-mix is not possible to execute entirely as in case D4, the JSP output exhibits to what extent it is possible to carry out. The outcome of the algorithm as reflected through JSP is shown in Table 8 for the situations D1 to D4. The schedules for D1, D2 and D3 are shown by Gantt charts in Fig. 7.

5.3.2. Steady rest malfunctioning of R_1 (D5)

R_1 confronts malfunctioning of steady rest due to some hydraulic/mechanical problem while processing P_1. However, tailstock support is working properly. It is noteworthy that P_1 requires steady support while machining spherical radius in one end. So except profiling, other operations can be done with the help of tailstock.

Once R_1 identifies the problem, it communicates with R_2 and R_4 to undertake the incomplete work. In this attempt R_3 is not considered since P_1 is beyond its capacity. From the existing workload, it is found that R_2 and R_4 become free after 34 min and 30 min respectively. Since the machining time of a particular product varies for different resources, it is estimated by interpolation that the incomplete work of R_1 can be completed by R_2 and R_4 in 56.15 min and 63.92 min respectively. R_2 being the best choice, R_1 decides to get the work done by R_2. So R_1 continues up to 34 min to progress as much as possible with the tailstock support. R_2 shoulders the responsibility of processing incomplete work of P_1 after completion of its existing work (P_2). The remaining products are allocated according to the availability of resources as per the existing rule. The modified schedule is presented in Fig. 8. The present solution also manifests that the resources can deal with the disturbance arising out of partial malfunctioning of a machine to achieve the goal by teamwork through cooperation.

6. Results and performance evaluation

The outcome of the algorithm shows that the proposed quantity variations are possible to execute within the stipulated time in scenario D1, D2 and D3, but it is not possible in D4 because one P_9

Table 8
Allocation of products to resources in different situations.

Scenario	Resource				Outcome
	R_1	R_2	R_3	R_4	
N	$2P_1+2P_7$	$2P_2+2P_8+2P_9$	$2P_6+2P_{10}$	$2P_5+2P_3+2P_4$	. . .
D1	$4P_1+1P_4$	$2P_2+1P_1+2P_3+2P_8$	$2P_6+2P_{10}+1P_7$	$1P_5+1P_7+2P_9+1P_4$	All possible
D2	$1P_1+1P_2+2P_3+2P_8+2P_{10}$	$6P_2+1P_4$	$2P_6+2P_7+3P_4$	$3P_5+2P_9$	All possible
D3	$2P_1+1P_6+3P_3+1P_8+1P_4$	$2P_2+3P_{10}+1P_7+1P_4$	$4P_6+2P_4$	$2P_5+1P_7+2P_9$	All possible
D4	$3P_1+2P_3+1P_8+1P_4$	$2P_2+1P_8+2P_{10}+1P_7+1P_4$	$2P_6+2P_7$	$2P_5+2P_9$	$1P_9$ not possible
D5	$2P_1$ (65.38%) $+2P_8+2P_7$	$2P_2+2P_1$ (34.62%) $+2P_9$	$2P_6+2P_{10}$	$2P_5+2P_3+2P_4$	All possible

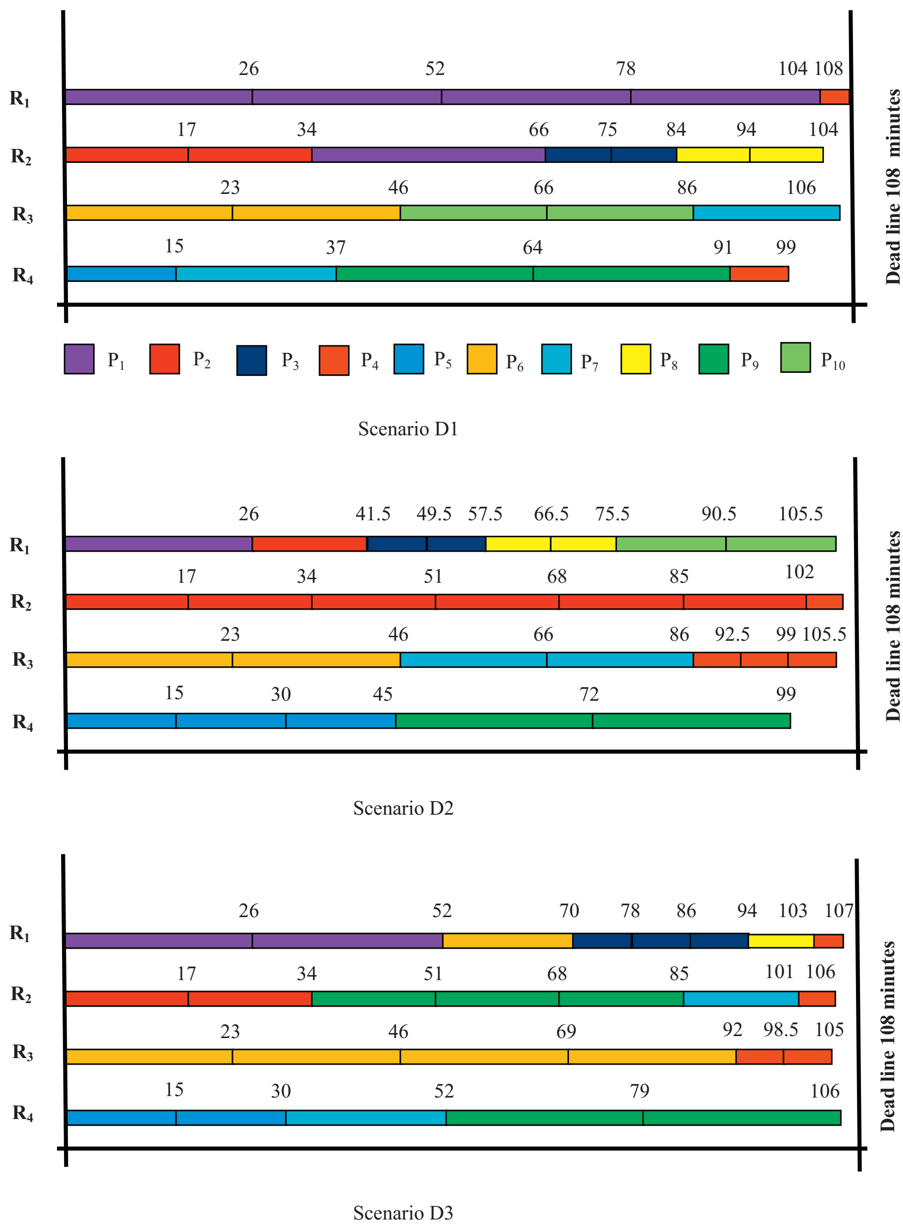

Fig. 7. Gantt charts showing schedules under the scenarios D1, D2, and D3.

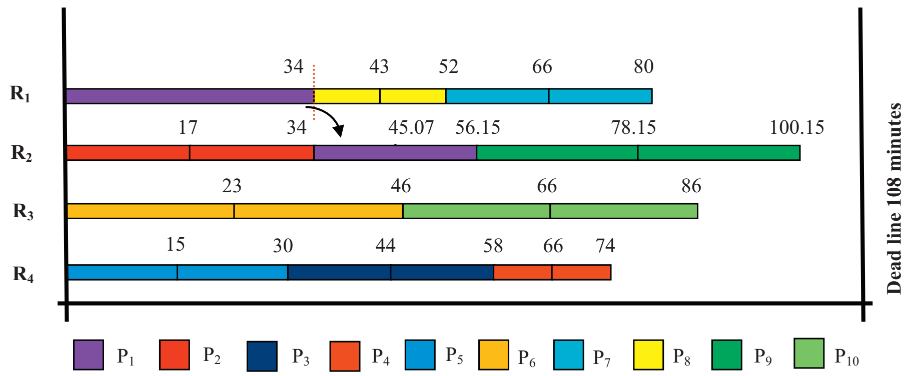

Fig. 8. Schedule when R_1 malfunctions partially (D5).

Table 9
Scheduling output.

Scenario	No. of products	Engagement time of Resources (min)				Resource utilization (%)				WIP	MFT
		R_1	R_2	R_3	R_4	R_1	R_2	R_3	R_4		
N	20	80	98	86	74	74	90.7	79.6	68.5	12.48	52.75
D1	22	108	104	106	99	100	96.3	98.2	91.7	14.15	67
D2	27	105.5	107	105.5	99	99.7	99.1	97.7	91.6	16.84	65
D3	26	107	106	105	106	99.1	98.1	97.2	98.1	16.9	68.9
D4[a]	23	...	...	...	...	...	...	...	...	...	...
D5	20	80	100.15	86	74	74	92.7	79.6	68.5	12.7	53.97

[a] This situation is not considered for performance evaluation, since proposed quantity combinations are not possible to execute.

cannot be allotted to any resource due to time constraint. When R_1 malfunctions partially while machining P_1 (scenario D5), support from P_2 helps to get rid of the crisis. Allocations of the products to the resources for task execution in normal situation and under disturbances are shown in Table 8.

The performances are evaluated by attributes like Work in progress (WIP), Mean flow time (MFT), % utilization of the resources, and resource load balancing. The results are presented in Table 9. It is observed that WIP increases substantially when volume-mix changes arbitrarily but leading to more stringent demand pattern. The reason can be attributed to the fact that the changes in volume-mix pattern are accompanied by disproportionately large increase in processing time. Similar logic also holds true about MFT. The % utilization of resources however, is improved remarkably and it is maximum (98.12%) in D3. It is also worth finding that as the scenario becomes more complex, the loads are evenly balanced and the best result is obtained in D3. Further, when machine malfunctions (D5), the increase in WIP and MFT are merely 1.7% and 2.3% respectively in comparison to the normal situation (N). This rise is due to the swapping of products P_1 (partially) and P_8 between R_1 and R_2. This modification results in marginal increase in engagement time (2.15 min) of R_2 only coupled with 2.2% rise in machine utilization. The schedules of other resources remain unaffected in this situation and average machine utilization (78.7%) changes negligibly from N. Therefore, the problem encountered by R_1 hardly has any serious consequence on the overall system performance since the problem is minor with zero downtime. The cooperation among resources therefore helps to overcome the widely disparate nature of disturbances without exceeding the stipulated time.

7. Conclusion

The present paper deals with dynamic task allocation mechanism for machine scheduling in a job shop environment following agent based holonic control approach. The main contribution of this research is multi-objective scheduling under all conditions to satisfy the varying need of different agent communities and the organization concurrently. The well-known contract net protocol is followed for the purpose of negotiation where message based communication is accomplished by XML in compliance with FIPA-ACL. The proposed scheduling is a two stage procedure that includes framing a priority rule and developing a job allocation strategy. The MOORA technique which is adopted for product ranking considers different parameters of varying importance in congruence with the interests of different communities and therefore is compatible to the MAS environment. It is robust enough to deal with any number of products and criteria. The job allocation strategy adopted simultaneously tries to satisfy local objectives of the agents and global objectives of the organization. Furthermore, when disturbances are present, the resources initially try to solve the problem as an individual and if fails, resort to cooperation. The merit of the cooperative problem solving algorithm lies in optimum allocation of the products to the resources so that multiple objectives are satisfied even under adverse situations. The results shows that cooperation based work also help to mitigate the problem arising out of partial malfunctioning of resource. The future work is directed toward the application of the methodology and strategy with necessary modifications in more complex practical situations to augment its credential in larger perspective.

Appendix A. Task announcement message


```
<?xml version="1.0" ?>
  <Announcement>
    <ProductAnnouncement name="Task Execution" product_id="P1" id="announcement_123"
            <Type>Product</Type>
            <To>R1</To>
            <From>P1</From>
             <Task_specifications>
                   <Task_specification>Turning</Task_specification>
                   <Task_specification>Facing</Task_specification>
                   <Task_specification>Grooving</Task_specification>
                   <Task_specification>Profiling</Task_specification>
            </Task_specifications>
            <Material_quality>Mild Steel</Material_quality>
            <Lot_size>2</Lot_size>
            <Task_eligibility>Must have steady rest</Task_eligibility>
     </ProductAnnouncement>
</Announcement>
```


Appendix B. Responses

B.1. Response from R_1


```
 <?xml version="1.0" ?>
<Response>
   <ResponseData name="Reply" id="response_11">
          <Type>Resource</Type>
          <AnnouncementId>announcement_123</AnnouncementId>
          <ResponseType>Bid</ResponseType>


         <To>P1</To>
         <From>R1</From>
         <Decision>yes</Decision>
         <Message>Capable</Message>
         <Time>26</Time>
  </ResponseData>
</Response
```


B.2. Response from R_3


```
 <?xml version="1.0" ?>
<Response>
     <ResponseData name="Reply" id="response_12">
            <Type>Resource</Type>
            < AnnouncementId>announcement_123</ AnnouncementId>
            <ResponseType>Bid</ResponseType>
            <To>P1</To>
            <From>R3</From>
            <Decision>no</Decision>
            <Message>Not Capable</Message>
            <Time />
  </ResponseData>
 </Response>
```


B.3. Response from P_1 for task awarding to R_1


```
    <?xml version="1.0" ?>
 <Response>
      <ResponseData name="Task Awarding" id="response_13">
            <Type>Product</Type>
            <AnnouncementId></AnnouncementId>
            <ResponseType>Confirmation</ResponseType>
            <ResponseId>response_11</ResponseId>
            <To>R1</To>
            <From>P1</From>
            <Decision>confirm</Decision>
            <Message>Confirmed</Message>
            <Time/>
     </ResponseData>
   </Response>
```

Appendix C. Request message to delegate tasks


```
<?xml version="1.0" ?>
<Request>
    <ResourceRequest name="Delegate Tasks" resource_id="R1" id="request_124">
                <Type>Resource</Type>
                <To>
                        <Resources>
                                <Resource> R2</Resource>
                                <Resource> R3</Resource>
                                <Resource> R4</Resource>
                        </Resources>
                </To>
                <From>R1</From>

            <Products>
                    <Product>
                            <AdditionalProductName>P1</AdditionalProductName>
                            <AdditionalProductQty>3</AdditionalProductQty>
                    </Product>
            </Products>
    </ResourceRequest>
</Request>
```


References

[1] Zhang X, Balasubramanian S, Brenan RW, Noorie DH. Design and implementation of a real-time holonic control system for manufacturing. Information Sciences 2000;127:23–44.
[2] Brussel HV, Bongaerts L, Wyns J, Valckenaers P, Ginderachter TV. A conceptual framework for holonic manufacturing: identification of manufacturing holons. Journal of Manufacturing Systems 1999;18(1):35–52.
[3] Wang L. Integrated design-to-control approach for holonic manufacturing systems. Robotics and Computer-Integrated Manufacturing 2001;17:159–67.
[4] Dominici G. Holonic production system to obtain flexibility for customer satisfaction. Journal of Service Science and Management 2008;1:251–4.
[5] Koestler A. The ghost in the machine. London: Arkana Books; 1967.
[6] Brussel HV, Wyns J, Valckenaers P, Bongaerts L, Peeters P. Reference architecture for holonic manufacturing systems: PROSA. Computers in Industry 1998;37(3):255–76 (Special Issue on Manufacturing Systems).
[7] Zhao F, Hong Y, Yu D, Yang Y, Zhang Q, Yi H. A hybrid algorithm based on particle swarm optimization and simulated annealing to holon task allocation for holonic manufacturing system. International Journal of Advanced Manufacturing Technology 2007;32:1021–32.
[8] Hsieh F-S, Chiang CY. Collaborative composition of processes in holonic manufacturing systems. Computers in Industry 2011;62:51–64.
[9] Huang B, Gou H, Liu W, Li Y, Xie M. A framework for virtual enterprise control with the holonic manufacturing paradigm. Computers in Industry 2002;49:299–310.
[10] Christo C, Cardeira C. Trends in intelligent manufacturing systems. In: IEEE international symposium on industrial electronics. 2007. p. 3209–14.
[11] Kotak D, Wu S, Fleetwood M, Tamoto H. Agent-based holonic design and operations environment for distributed manufacturing. Computers in Industry 2003;52:95–108.
[12] Fletcher M, Brennan RW, Norrie DH. Modeling and reconfiguring intelligent holonic manufacturing systems with internet based mobile agents. Journal of Intelligent Manufacturing 2003;14:7–23.
[13] Leitao P. Agent-based distributed manufacturing control: a state-of-the-art survey. Engineering Applications of Artificial Intelligence 2009;22:979–91.
[14] Wooldridge M, Jennings NR. Intelligent agents: theory and practice. The Knowledge Engineering Review 1995;10(2):115–52.
[15] Chaxel F, Charpentier P, Jacquel D. Product agent concept based on feature modeling and Java Card implementation. International Journal of Advanced Manufacturing Technology 2003;3:208–14.
[16] Babiceanu RF, Frank Chen F. Development and applications of holonic manufacturing systems: a survey. Journal of Intelligent Manufacturing 2006;17:111–31.
[17] Heragu SS, Graves RJ, Kim B, Onge A. Intelligent agent based framework for manufacturing systems control. IEEE Transactions on Systems, Man, and Cybernetics – Part A: Systems and Humans 2002;32(5):560–73.
[18] Shen W, Wang L, Hao Q. 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 2006;36(4):563–77.
[19] Wang C, Ghenniwa H, Shen W. Real time distributed shop floor scheduling using an agent-based service-oriented architecture. International Journal of Production Research 2008;46(9):2433–52.
[20] Cao Y, Yang Y, Yang L. Intelligent job shop scheduling based on MAS and integrated routing wasp algorithm and scheduling wasp algorithm. Journal of Software 2009;4(5):487–94.
[21] Sudo Y, Sakao N, Matsuda M. An agent behavior technique in an autonomous decentralized manufacturing system. Journal of Advanced Mechanical Design Systems and Manufacturing 2010;4(3):673–82.
[22] Leitao P, Restivo F. A holonic approach to dynamic manufacturing scheduling. Robotics and Computer-Integrated Manufacturing 2008;24:625–34.
[23] Ouelhadj D, Petrovic S. A survey of dynamic scheduling in manufacturing systems. Journal of Scheduling 2009;12(4):417–31.
[24] Gou L, Luh PB, Kyoya Y. Holonic manufacturing scheduling: architecture, cooperation mechanism, and implementation. Computers in Industry 1998;37:213–31.
[25] Miyashita K. CAMPS. A constraint-based architecture for multi-agent planning and scheduling. Journal of Intelligent Manufacturing 1998;9(2):147–54.
[26] Kutanoglu E, Sabuncuoglu I. An analysis of heuristics in a dynamic job shop with weighted tardiness objectives. International Journal of Production Research 1999;37(1):165–87.
[27] Wang YC, Usher JM. Application of reinforcement learning for agent-based production scheduling. Engineering Applications of Artificial Intelligence 2005;18:73–82.
[28] Priore P, Fuente D, Puente J, Parreno J. A comparison of machine-learning algorithms for dynamic scheduling of flexible manufacturing systems. Engineering Applications of Artificial Intelligence 2006;19(3):247–55.
[29] Iwamura K, Okubo N, Tanimizu Y, Sugimura N. Real-time scheduling for holonic manufacturing systems based on estimation of future status. International Journal of Production Research 2006;44(18–19):3657–75.
[30] Alvarez E, Diaz F. Framework for the dynamic scheduling of complex job shops. International Journal of Manufacturing Technology and Management 2007;11(3/4):411–25.
[31] Duffle NA, Prabhu VV. Real-time distributed scheduling of heterarchical manufacturing systems. Journal of Manufacturing Systems 1994;13(2):94–107.
[32] Shen W. Distributed manufacturing scheduling using intelligent agents. IEEE Intelligent Systems and Their Applications 2002;17(1):88–94.
[33] Rabelo RJ, Camarinha-Matos LM, Afsarmanesh H. Multi-agent based agile scheduling. Robotics and Autonomous Systems 1999;27:15–28.
[34] Cai N, Wang L, Feng H-Y. GA-based adaptive setup planning toward process planning and scheduling integration. International Journal of Production Research 2009;47(10):2745–66.
[35] Wang L, Shen W, Hao Q. An overview of distributed process planning and its integration with scheduling. International Journal of Computer Applications in Technology 2006;26(1/2):3–14.
[36] Wong TN, Leung CW, Mak KL, Fung RYK. Dynamic shop floor scheduling in multi-agent manufacturing systems. Expert Systems with Applications 2006;31:486–94.
[37] Davis R, Smith R. Negotiation as a metaphor for distributed problem solving. Artificial Intelligence 1983;20(1):63–109.
[38] Smith RG. The contract net protocol: high-level communication and control in a distributed problem solver. IEEE Transactions on Computers C 1980;29(12):1104–13.
[39] Rabelo RJ, Camarinha-Matos LM. Negotiation in multi-agent based dynamic scheduling. Robotics and Computer-Integrated Manufacturing 1994;11(4):303–9.
[40] Markus A, Kis Vbncza T, Monostori L. A market approach to holonic manufacturing. Annals of the CIRP 1996;45(1):433–6.
[41] Kanchanasevee P, Biswas G, Kawamura K, Tamura S. Contract-net based scheduling for holonic manufacturing systems. In: Proceedings of SPIE,

architectures, networks, and intelligent systems for manufacturing integration. 1997. p. 108–15.

[42] Yang C, Lin JS. The development of a hybrid hierarchical/heterarchical shop floor control system applying bidding method in job dispatching. Robotics and Computer-Integrated Manufacturing 1998;14:199–217.

[43] Sousa P, Ramos C. A distributed architecture and negotiation protocol for scheduling in manufacturing systems. Computers in Industry 1999;38: 103–13.

[44] Suesut T. Multi level contract net protocol based on holonic manufacturing system implement to industrial network. In: Proceedings of the 2004 IEEE conference on robotics, automation and mechatronics. 2004. p. 253–8.

[45] Lai L. Agent-based holonic dynamic and optimal manufacturing system for distributed manufacturing. In: International conference on environmental science and information application technology. 2009. p. 485–8.

[46] Wang XH, Wong TN, Wang G. Knowledge representation for multi-agent negotiations in virtual enterprises. International Journal of Production Research 2011;49(14):4275–97.

[47] Adam E, Berger T, Sallez Y, Trentesaux D. Role-based manufacturing control in a holonic multi-agent system. International Journal of Production Research 2011;49(5):1455–68.

[48] Renna P. Multi-agent based scheduling in manufacturing cells in a dynamic environment. International Journal of Production Research 2011;49(5/1):1285–301.

[49] Maione G, Naso D. A soft computing approach for task contracting in multi-agent manufacturing control. Computers in Industry 2003;52: 199–219.

[50] Brauers WKM, Zavadakas EK, Peldschus F, Turskis Z. Multi objective decision making for road design. Transport 2008;23(3):183–93.

[51] Brauers WKM, Zavadakas EK. Robustness of the multi objective MOORA method with a test for the facilities sector. Technological & Economic Development of Economy 2009;15(2):352–75.

T.K. Jana is currently working as an associate professor of Mechanical Engineering Department, Haldia Institute of Technology, Haldia, India. He received his B.E. degree from Bengal Engineering and Science University (BESU), Howrah, West Bengal, India in the year 1988 and M.E. degree in production engineering from the Jadavpur University, India in 1990. His active areas of interests are metal cutting, computer integrated manufacturing and holonic manufacturing. Currently he is also pursuing his Ph.D. in the Jadavpur University in the area of holonic manufacturing control.

B. Bairagi is currently working as assistant professor in Production Engineering Department, Haldia Institute of Technology, Haldia, India. The author has received his B.E. and M.E degrees from Production Engineering Department of Jadavpur University, Kolkata. He is also pursuing his Ph.D. in the Jadavpur University in the areas of production management. His area of interest is production management with special focus on multi criteria decision making.

S. Paul received the M.Stat. degree with a specialization of applied statistics and data analysis from Indian Statistical Institute (Kolkata) in the year of 1990 and received M.Tech. (IT) from Punjab University in the year of 2003. He is currently working toward Ph.D. degree in control engineering at the Department of Electronics and Telecommunication, Jadavpur University. He is currently an assistant professor in the Department of Information Technology, Haldia Institute of Technology, Haldia, West Bengal, India. His research interests include control engineering, soft computing, pattern recognition and image processing.

Dr. B. Sarkar is the professor of Production Engineering Department, Jadavpur University, Kolkata, India. He received his B.E. and M.E. degrees in production engineering from the Jadavpur University in 1983 and 1985 respectively. He received his Ph.D. degree in 1992. He is the recipient of Outstanding Paper Award at Emerald Literati Network for excellence 2006, UK. He is the member of Institution of Engineers (India), Society for Reliability Engineering (SRE). He has published more than 150 papers in the National/International Journals and Proceedings. His fields of interests include Reliability engineering, AI techniques in mechanical & production management and decision engineering.

Dr. J. Saha received his Ph.D. degree from Jadavpur University, Kolkata. He is currently professor in production engineering of Jadavpur University. He has published more than 50 papers in the area of manufacturing and related areas. His research interests are application of CAD-CAM in manufacturing, robotics, flexible manufacturing system, vibration in machine tools, non-traditional machining. He has been joint author of three books. He is member of the Institution of Engineers (India). He has also carried out few consultancy works for Industry.