# A Cooperative Multi-Agent System for Plan Generation in Batch Manufacturing

Chunhua Ju
Dept. of Information Engineering
Hangzhou University of Commerce
Hangzhou, China

Yongjian Fu
Dept. of Computer Science
University of Missouri-Rolla
Rolla, MO, USA

**Abstract**

*This paper proposes an approach to generate blending plans using a cooperative multi-agent system. First, a model for the blending problem in batch manufacturing is developed based on operations research principles, which takes into consideration of factors involved in blending plan generation. Second, the architecture of the system is introduced. We describe the structures of control agent and execution agents that make up the cooperative multi-agent system. Third, the cooperative problem solving process for blending plan generation using the multi-agent system is described. Finally, an application of our system in tea blending is presented. The system has been in operation successfully for a tea manufacturer. The proposed approach is efficient and effective in finding blending plans.*

**Keywords**: agent, multi-agent system, cooperative problem solving, batch manufacturing.

## 1 Introduction

Blending has been widely used in many areas of batch manufacturing, for example, in the manufacturing of tea, feed, beer, and chemical materials. One of the major problems in blending is to find the best blending plan, i.e., the proportion of raw materials used in making the product. Traditionally blending plans are decided by human experts who estimate the proportion of materials based on their experiences. However, the optimal blending plan for a particular task usually requires the knowledge of raw materials, the consideration of cost and quality, the proper usage of stale materials, as well as the maintenance of reasonable inventory. It takes much time for experts to learn about the huge amount of information about raw materials, such as name, type, weight, quality, cost and much more. Moreover, in such a manual operation, it is almost impossible to accurately compare the costs of different plans. Furthermore, it is hard to maintain a consistent and uniform quality standard among different experts. Hence, it is very desirable for batch manufacturers to obtain automated solutions that are more robust, flexible and comprehensive.

In this paper, we first introduce a model for the blending problem, which incorporates materials information, target quality, and other constraints in blending. We then develop a multi-agent system (MAS) to find an optimal blending plan based on the blending model and blending knowledge.

The MAS consists of a control agent (CA) and a group of execution agents (EAs). The CA analyses user's requirements, generates and allocates subtasks for each EA, coordinates the interactions among EAs, and records and monitors the process. An EA mainly performs the planning and computation in order to reach a blending decision. The actions of an EA includes analyzing and understanding of subtasks, requesting data from CA, computing based on the complicated blending model, generating the final decision, and submitting the result to CA.

The MAS has been successfully applied in tea blending. In most cases, the system takes only a few minutes to generate the final blending plan for over 1000 materials. The system has been in operation successfully for a tea manufacturer, with hundreds of blending plans generated. It is clear that the proposed approach is efficient and effective in finding blending plans.

This paper is organized as follows. Section 2 briefly surveys related work in agent-based manufacturing. Section 3 defines the problem statement and the blending model, as well as some optimization technique in using the model. The architecture of the MAS and the cooperative problem solving process are described in Section 4. Section 5 presents an application of the MAS in tea blending. Section 6 concludes the study and discusses possible future work.

## 2 Related Work

There has been a lot of research on intelligent agents and multi-agent systems (MASs) [1,3,4,5,6]. Moreover, agent technology has been successfully applied in many areas from air traffic control to information filtering to manufacturing.

Parnuak [8] developed an agent-based system, YAMS (Yet Another Manufacturing System), for manufacturing process control. Fletcher and Deen [7] presented a model for task rescheduling in multi-agent manufacturing that intends to maximize system efficiency and reliability in an environment with predictable failure patterns. Kouiss et al. [9] proposed a multi-agent architecture in FMS for dynamic scheduling for dynamic job shop scheduling. Jennings et al. [2] developed a method of using multi-agent system for business process management, which helps managers make informed decisions based on a combination of judgement and information from each distributed department which has its own IT system.

As discussed above, most applications of agent and MAS in manufacturing are in process management and scheduling. The application of agent technology for blending in batch manufacturing has not been widely studied.

In this paper, we explore the application of agent technology for blending in batch manufacturing. Because of the complexity of the problem, it is natural to use a multi-agent system. We build a cooperative multi-agent system for finding optimal blending plans in batch manufacturing. The system generates optimal blending plans based on a blending model and blending knowledge. The agents in the system cooperatively solve the problem through inference, calculation, and interaction.

## 3 Problem Statement

The objective of a blending system is to find the optimal blending plan that satisfies the quality, cost, blending rules, and other requirements. A blending plan specifies the proportion of raw materials. The input of the system includes data about the raw materials, TQS, blending rules, and current inventory of materials. The raw materials will be blended according to the proportion in the blending plan to manufacture the product.

### 3.1 Data and Rules

The fundamental data and rules and their structures are explained in this section, including raw material data, TQS, and blending rules.

#### 3.1.1 Raw Material Data

The quality, inventory, and price information about raw materials should be available from the information system. For each raw material, its code, type, check date, mark, unit price, weight, storage, and quality factors are stored.

The values of quality-factors are to be determined when materials are being checked. The value of a quality factor reflects the material's measurement in a quality dimension, such as moisture. All raw materials have the same set of quality factors. For example, the description of a raw tealeaf will be:

(1,HSC,2000/06/21,Song41605,2300,11.32,Ming4-4,(0.9,0.8,0.8,0.9,0.8,0.8,0.9,0.8,0.8,0.9,0.8,0.8,0.9,0.8))

There are 14 quality factors representing various quality measures.

#### 3.1.2 Target Quality System (TQS)

The target quality system (TQS) specifies the quality of the final blended product in terms of quality factors. The final blended product's quality should be at least as good as that specified in TSQ. More specifically, the *i*-th quality factor of the final product should be no less than the *i*-th quality factor in TQS.

The number and order of quality factors in the TQS should be in accordance with that of the materials so that the quality factors can be compared. For example, a TQS for tea blending from the previous tealeaves may be:

(9371,Senegal,(0.8,0.8,0.8,0.9,0.8,0.8,0.9,0.8,0.8,0.9,0.8,0.8,0.8,0.8))

where 9371 and Senegal are product code and customer name, respectively.

#### 3.1.3 Blending Rules

A blending rule empirically specifies the proportion of raw material types for a particular product. They are expressed in *IF-THEN* form. The *IF* part is a set of pairs each specifying a raw material type and its proportion. Note that a raw material type is given, rather than a specific raw material. The *THEN* part records the frequency of the rule's use, reflecting the degree of the rule's acceptance. For example, one rule for tea blending is:

R101: **IF** ([HSC, 25%][HC, 40%][WC, 20%][SC, 15%])

**THEN** (9371, use (12))

The rule says that for product 9371, it should consist of 25% HSC, 40% HC, 20% WC, and 15% SC. This rule has been successfully used 12 times.

### 3.2 A Model for Blending Problem

According to the practical blending process, several requirements for the blending problem are listed below.

(1) The cost of the final product should be as low as possible.

(2) The quality of the final produce should meet the requirements in TQS.

(3) The use of stale and fresh materials should be combined appropriately.

(4) The remaining of materials should be reasonable so the storage places are best utilized.

Based on these requirements, a model for blending problems is proposed which is based on linear programming. Assuming that there are *n* raw materials, the symbols used in our model are summarized in Table 1.

Although it is theoretically possible to reach an ideal solution where the blended material's quality factors match exactly these of TQS's, it is practically impossible to find such a solution. Hence, two parameters, $\alpha$ and $\beta$, are introduced to enhance the model. The values of $\alpha$ and $\beta$ ($\alpha \in [0,1]$, $\beta \in [1,2]$) could be acquired from statistics of actual data from past or from the expert's experiences.

Table 1. Symbols in the blending model.

| Symbol | Description |
|---|---|
| $C_i$ | unit price of the *i*-th material |
| $a_{1,i}, a_{2,i}, .., a_{k,i}$ | quality factors of the *i*-th material |
| $W_i$ | weight of the *i*-th material in inventory |
| $X_i$ | blending percentage of the *i*-th material |
| C | cost of a blending plan |
| WT | product weight |
| $b_1, b_2, \ldots, b_k$ | TQS for the product |
| $\alpha, \beta$ | Relaxation parameters of TQS |

The first two requirements in the model for blending problem are represented as follows:

$$\text{Min } C=(C_1X_1+C_2X_2+...+C_iX_i+...+C_nX_n)*WT$$

$$\begin{cases} a_{1,1}X_1+a_{1,2}X_2+...+a_{1,i}X_i+...+a_{1,n}X_n \geqslant \alpha b_1 \\ a_{1,1}X_1+a_{1,2}X_2+...+a_{1,i}X_i+...+a_{1,n}X_n \leqslant \beta b_1 \\ \quad\vdots \qquad \vdots \\ a_{k,1}X_1+a_{k,2}X_2+...+a_{k,i}X_i+...+a_{k,n}X_n \geqslant \alpha b_k \\ a_{k,1}X_1+a_{k,2}X_2+...+a_{k,i}X_i+...+a_{k,n}X_n \leqslant \beta b_k \\ X_1+X_2+...+X_i+...+X_n=1 \end{cases} \qquad \text{(St1)}$$

St1 is a basic linear programming problem with multiple constraints. The goal function, Min(C), and St1, represent the requirements (1) and (2) listed in the beginning of this section respectively. However, there are two additional requirements in practice that should be incorporated into the model.

(a) The use of stale materials

Stale materials should be used with fresh materials in practical blending production process. Usually, the percentages of materials of different ages can be determined from experience. If the inventory of a particular year's materials is not adequate, its deficient percentage will be assigned to the next year, i.e., we will try to use stale materials as much as possible. Assuming that the materials have been sorted according to the number of years that they have been in stock, the requirement can be expressed as the following equations.

$$\begin{cases} X_i+...+X_j=PW[0] \\ X_l+...+X_m=PW[1] \\ \quad\dots \\ X_t+...+ X_s=PW[T] \\ PW[0]+PW[1]+\dots+PW[T] = 1 \end{cases} \qquad \text{(St2)}$$

$X_i$, …, $X_j$ are the percentages of materials of this year. $X_l$, …, $X_m$ are the percentages of materials one year old. $X_t$, …$X_s$ are the percentages of materials $T$ years old. Usually the materials are blended within three years, i.e., T is usually 2. PW[$t$] represents the total percentage of materials that are $t$ years old.

(b) The reasonable remaining of materials

For each material, we have to make sure there is enough inventory. Besides, the amount of remaining material should be reasonable because otherwise the storage space is not efficiently used. The relationship between a material's inventory weight $W_i$ and its blending weight $P_i = X_i*WT$ can be expressed as follows.

① $P_i$ must be less than or equal to $W_i$.

② If $W_i$ if greater than $P_i$, then $W_i$-$P_i$ (remaining of material) should be reasonable. Otherwise, if $W_i$-$P_i$ is a quite small amount, a whole storage unit is occupied by a small amount of material.

When $P_i$ approximates $W_i$, that is, when $W_i$-$P_i$ < MRW (Minimum Remaining Weight), it is a good idea to use all of the *i*-th material, that is, to make $P_i = W_i$. In that way, one storage unit will be emptied and available. When $W_i$-$P_i$ > MRW, the calculated percentage will be used without change. The constraint can be represented as the following formulas.

$$\begin{cases} W_1\text{-}WT*X_1=0 \text{ or } W_1\text{-}WT*X_1 \geqslant MRW \\ W_2\text{-}WT*X_2=0 \text{ or } W_2\text{-}WT*X_2 \geqslant MRW \\ \qquad\qquad \vdots \\ W_n\text{-}WT*X_n=0 \text{ or } W_n\text{-}WT*X_n \geqslant MRW \end{cases} \qquad \text{(St3)}$$

The model for the blending problem consists of St1, St2, St3, and the goal function Min(C).

### 3.3 Optimization Techniques

Since St3 contains disjunctions (OR conditions), the model is different from that of standard linear programming. Since St3 has $n$ conjuncts, each of which has two disjuncts, there are a total of $2^n$ possible combinations of these disjuncts. Thus St1, St2, and St3, will result in $2^n$ possible sets of linear equations that need to be solved. It will take too much time to solve these $2^n$ sets of linear equations.

By analyzing the conditions in St3, we realize that some combinations of disjuncts will prohibit a solution. These combinations can be pruned immediately to reduce total solving time.

We first transform St3 into:

$$\begin{cases} X_1=W_1/WT \text{ OR } X_1 \leqslant (W_1\text{-}MRW)/WT \\ X_2=W_2/WT \text{ OR } X_2 \leqslant (W_2\text{-}MRW)/WT \\ \qquad\qquad \vdots \\ X_n=W_n/WT \text{ OR } X_n \leqslant (W_n\text{-}MRW)/WT \end{cases} \qquad \text{(St3')}$$

Assume there are $p$ equalities in St3'. Then there are $n$ - $p$ inequalities in St3'. The following findings can help us to reduce the number of combinations.

(a)If the sum of $X_i$'s in the *p* equalities is more than 1, then there is no solution to problem, (contradicting $X_1+...+X_n=1$ in St1);

(b)If the sum of $X_i$'s in the *p* equalities is less than L, and M is the maximum of the sum of $X_i$'s in the *n* - *p* inequalities, then L+M must be greater or equal to 1. Otherwise, it contradicts $X_1+X_2+...+X_i+...+X_n=1$ in St1, and is unsolvable as well.

## 4 Multi-Agent System

The MAS consists of a Control Agent(CA) and a group of Execution Agents(EAs), connected through a network. The architecture of the MAS is shown in Figure 1.

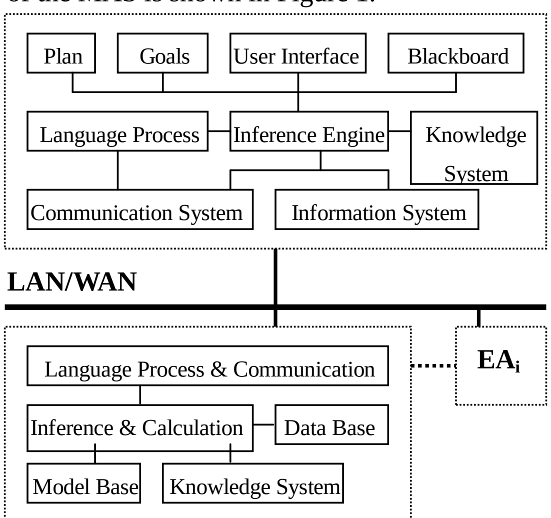


Figure 1. The architecture of the MAS.

The detailed workflow in a cooperative problem solving process is as following.

1. A user inputs the blending task through human interface **UI**. CA gets the set of blending requirements ***DS*** (product code, customer, product weight, and other control parameters);
2. CA searches to get target quality set ***TQS*** for ***DS*** from its information system **IS** based on product code, and infers to get blending rule set ***RS*** from its knowledge system **KS**, according to ***DS***, product code, and customer;
3. CA analyses and interprets the rule set ***RS***, then generates subtask set ***TS*** ($\mathbf{T_0,T_1,\ldots,T_n}$) from ***RS***, ***DS***, ***QS***;
4. CA allocates subtasks ($\mathbf{T_0,T_1,\ldots,T_n}$) to corresponding EA ($EA_0,..,EA_n$) through its communication system **CS**;
5. After $EA_0,..,EA_n$ receive the subtasks $\mathbf{T_0,T_1,\ldots,T_n}$, each EA does processing and calculating independently;
6. $EA_i$ analyses the $\mathbf{T_i}$ to get the ingredient list which is in the blending rule's **IF** clause. $EA_i$ then knows the percentages of each material type ***PS*** ($\mathbf{P_0,..P_c}$), and generates the material requirements ***MS*** ($\mathbf{M_0,..M_c}$) for these types;
7. $EA_i$ interacts with CA, requesting the information of each type of materials from CA;
8. Based on $\mathbf{M_0}$, ..,$\mathbf{M_c}$, CA extracts information about each type of material from the information system **IS**, and generates the material data set ***FS*** ($\mathbf{F_0,..F_c}$), then sends ***FS*** to $EA_i$. $EA_i$ stores ***FS*** in its database **DB**;
9. With the information, $\mathbf{T_i}$ (***DS***, ***QS***, ***RS***), ***PS*** ($\mathbf{P_0,..,P_k}$), and ***FS*** ($\mathbf{F_0,..,F_c}$), $EA_i$ runs C+1 times (C is the number of material types in the rule). During each run, $EA_i$ employs the optimization techniques. After each run, $EA_i$ obtains the percentage of each material using the blending model. These percentages form a blending plan. All the interactions and the result will be written in CA's blackboard **B**;
10. CA monitors the state of the blackboard, and keeps track of the execution and requests of EAs and all the interactions;
11. When CA notices that the execution of $EA_0,..EA_n$ has stopped, it knows that blending plans have been generated;
12. Through the human interface **UI**, the user chooses the best available plan and modifies related data in the system. CA automatically stores all processes in blackboard **B**, and modifies the rule use function.

## 5 Application in Tea Blending

In this section, we give an example application of the MAS in tea blending. Due to the space limit, an instance of blending for the product coded 9371 is given here.

In tea blending, there are 14 quality factors ($k$ = 14) that describe the tea quality, which consists of both internal and external quality of raw tealeaves. The raw tealeaves are categorized into 4 types: HSC, SC, WC, and HC. The product code, desired product weight, and customer are 9371, 200 tons, and Senegal, respectively. The minimum remains weight (MRW) is set as 2 tons. The CA will have the following data and knowledge based on **UI**, **IS**, **KS**. Note that there are 3 rules for the product 9371 in **KS.** There are totally 1000 tealeaves in the four types, i.e., $n$ = 1000. The relaxation parameters for TQS, $\alpha$ and $\beta$ are 0.95 and 1.05, respectively.

***DS***={9371, Senegal, 200T, 2T}

***TQS***={9371,Senegal,(0.8,0.8,0.8,0.9,0.8,0.8,0.9,0.8,0.8,0.9,0.8,0.8,0.8,0.8)}

***RS***={R101, R201, R401}

R101: **IF** ([HSC, 25%][HC, 40%][WC, 15%][SC, 20%]) **THEN** (9371, use (12))

R201: **IF** ([HSC, 30%][HC, 30%][WC, 30%][SC, 10%]) **THEN** (9371, use (9))

R401: **IF** ([HSC, 30%][HC, 40%][WC, 30%]) **THEN** (9371, use (5))

The tealeaves in principal are used within 3 years. The percentages for raw tealeaves of different ages are 10% (this year), 30% (last year), and 60% (two years ago). That is, PW[0]=0.1, PW[1]= 0.3, and PW[2]=0.6. If the total weight of a particular year's tealeaves is insufficient, the deficient amount would be adjusted by CA to the next year. With ***RS***, ***DS*** and ***QS***, CA generates a task set ***TS*** ($\mathbf{T_0}$,$\mathbf{T_1}$,$\mathbf{T_2}$), and allocates $\mathbf{T_0}$,$\mathbf{T_1}$,$\mathbf{T_2}$ to $EA_0$, $EA_1$, $EA_2$ respectively.

$\mathbf{T_0}$=(9371, 200T, TQS, (HSC, 25%), (HC, 40%), (WC, 15%), (SC, 20%), (0.1, 0.3, 0.6))

$\mathbf{T_1}$=(9371, 200T, TQS, (HSC, 30%), (HC, 30%), (WC, 30%), (SC, 10%), (0.1, 0.3, 0.6))

$\mathbf{T_2}$=(9371, 200T, TQS, (HSC, 30%), (HC, 40%), (WC, 30%), (0.1, 0.3, 0.6))

Because of the parallelism of $EA_0$, $EA_1$, and $EA_2$, CA interacts with $EA_0$, $EA_1$, and $EA_2$ alternatively and continuously and keeps monitoring them. After $EA_0$, $EA_1$, and $EA_2$ complete $\mathbf{T_0}$, $\mathbf{T_1}$, and $\mathbf{T_2}$ respectively, CA can get three final full blending plans. With the CA's **UI**, the user could choose one plan for the tea manufacturer. After the final blending plan is chosen, CA will prints the plan and modifies data in **IS** and rule-use-function in **KS**.

Table 2. The blending plan for product 9371 (unit of weight: ton).

| Type | Material-age | Blending-weight | Remaining-weight | Percentage |
|---|---|---|---|---|
| HSC | 2 | 11.32 | 0.00 | 5.660 |
| HSC | 2 | 10.48 | 0.00 | 5.240 |
| HSC | 1 | 4.77 | 0.00 | 2.385 |
| HSC | 1 | 3.00 | 0.00 | 1.500 |
| HSC | 1 | 7.44 | 0.00 | 3.720 |
| HSC | 1 | 11.78 | 0.00 | 5.890 |
| HSC | 0 | 0.40 | 0.00 | 0.200 |
| HSC | 0 | 0.81 | 3.90 | 0.405 |
| HC | 2 | 6.97 | 0.00 | 3.335 |
| HC | 2 | 5.07 | 0.00 | 2.540 |
| HC | 2 | 7.83 | 0.00 | 3.915 |
| HC | 2 | 4.08 | 0.00 | 2.040 |
| HC | 1 | 5.27 | 0.00 | 2.635 |
| HC | 1 | 1.96 | 0.00 | 0.980 |
| HC | 1 | 2.20 | 0.00 | 1.100 |
| HC | 1 | 16.23 | 0.00 | 8.120 |
| HC | 0 | 9.80 | 0.00 | 4.900 |
| HC | 0 | 5.61 | 0.00 | 2.805 |
| HC | 0 | 4.68 | 0.00 | 2.340 |
| HC | 0 | 2.28 | 0.00 | 1.140 |
| HC | 0 | 1.95 | 0.00 | 0.980 |
| HC | 0 | 2.91 | 0.00 | 1.455 |
| HC | 0 | 3.16 | 5.65 | 1.580 |
| WC | 1 | 7.60 | 0.00 | 3.800 |
| WC | 1 | 7.00 | 0.00 | 3.500 |
| WC | 1 | 9.14 | 0.00 | 4.570 |
| WC | 0 | 5.80 | 0.00 | 2.900 |
| WC | 0 | 0.46 | 5.91 | 0.230 |
| SC | 2 | 14.79 | 0.00 | 7.393 |
| SC | 1 | 1.41 | 0.00 | 0.705 |
| SC | 1 | 1.60 | 0.00 | 0.800 |
| SC | 1 | 4.06 | 0.00 | 2.035 |
| SC | 1 | 7.54 | 0.00 | 3.770 |
| SC | 0 | 2.14 | 0.00 | 1.070 |
| SC | 0 | 6.94 | 0.00 | 3.470 |
| SC | 0 | 1.51 | 9.81 | 0.755 |

Table 2 is the chosen final optimal plan. It takes only 5 minutes to get the final blending plan. In Table 2, the percentages are 25%, 40%, 15%, and 20%, for the HSC, HC, WC, and SC types of tealeaves, respectively. The percentages are approximately 24%, 46%, and 30%, for tealeaves of ages 0, 1, and 2, respectively. This is because there are not enough tealeaves of age 2 and the CA assigns the remaining 30% to the next year, i.e., age 1, which makes one-year-old tealeaves to be insufficient, and CA assigns the remaining percentage to fresh tealeaves (age = 0). The remaining weight of all tealeaves is either 0 or greater than 2 tons.

In addition, we make a comparison of times for three different methods: manual, operation research (OR), and MAS, for products with 2, 3 and 4 rules in knowledge base (KS). The result is shown in Table 3. The MAS method is much faster than the other two.

Table 3. The comparison of times for different methods (in minutes).

| Method / No. Rules | Manual | OR | MAS |
|---|---|---|---|
| 2 | ⩾200 | 20 | 3.5 |
| 3 | ⩾200 | 30 | 5 |
| 4 | ⩾200 | 50 | 8 |

The MAS for tea blending has been in operation at Zhejiang tea exporting Co., China for several years. It has completed hundreds of blending plans and brought about substantial savings in both time and cost for the company.

## 6 Conclusion and Future Work

A cooperative multi-agent system for blending in batch manufacturing has been proposed. It finds an optimal blending plan based on raw material information, a blending model, and blending rules. Compared with traditional methods, our approach is much faster and often finds the best blending plan. Our system has been successfully applied to tea blending.

Our future work will focus on developing more flexible models that can incorporate more general constraints. Another direction is to extend the agent interaction in different environments.

## References

[1]N. R. Jennings. Controlling cooperative problem solving in industrial multi-agent systems using joint intentions. In Artificial Intelligent,1995,(75):195-240.

[2]N. R. Jennings, P. Faratin, M. J. Johnson. Agent-based business process management. In International Journal of Cooperative Information Systems,1996,5(2): 105-130.

[3]M. Wooldridge, N. R. Jennings. Intelligent agents: Theory and practice. The Knowledge Engineering Review, 1995,10(2):115-152.

[4]Q. M. Chen, D. Umeshwar. Multi-Agent Cooperative Transaction for E-Commerce.. CoopIS 1999:209-220.

[5]J. S. Rosenschein. Rational Interaction: Cooperation Among Agents. PhD thesis, Computer Science Department, Stanford University, Stanford,CA 94305,1985.

[6]T. Sandholm. An implementation of the contract net protocol based on marginal cost calculations. In Proceedings of the Eleventh National Conference on Artificial Intelligent, Washington, July 1993, pp.256-262.

[7]M. Fletcher, S. M. Deen. Task Rescheduling in Multi-Agent Manufacturing. In Proceedings of the 10th International Workshop on Database & Expert System Application, Sept. 1-3, 1999, Italy.

[8]V. D. Parunak. Manufacturing Experience with Contract Net. Distributed Artificial Intelligence, Huhns,M.N.ed., Pitman, pp.285-310.

[9]A. Kouiss, H. Pierreval, N. Mebarki. Using multi-agent architecture in FMS for dynamic scheduling. Journal of Intelligent Manufacturing, 8(1),41-47.