A New Construction of Job-Shop Scheduling System Integrating ILOG and MAS

LiZhi Qin[1], QingSong Li[1]
[1]School of Transport and Automotive Engineering, Xhu University,
610039 Chengdu Sichuan
[1]liqs73@163.com

***Abstract*—While manufacturing industry facing tough market competition, quickly responsive and flexible Job-Shop Scheduling System, JSSS, seems more attractive in nowadays. Research and applications on JSSS were summarized in this paper. A new framework of JSSS was designed in our research. A synthetic method was used here in order for efficiency and practicality. We took advantage of Multi-Agent System (MAS) and ILOG, which is a powerful calculation platform in the field of optimization. Interactive models between MAS and ILOG, among agents in MAS were also devised. Prototype of this JSSS represents more convenient and flexible scheduling plan in test by engineer of an automobile factory.**

***Index Terms*—JSSP, framework, MAS, ILOG**

1 Introduction

Job-Shop Scheduling Problem, JSSP, is mainly concerned with the resource distribution in order to meet requirements of task configuration and order constraints. Job-shop scheduling has become a little complicated, because JSSP in the actual operation involves raw materials, equipments, manpower, funds and so on, and it is complex, random, and multi-targeted. And all these show that a better Job-Shop Scheduling System is of overriding importance.

A. Current Status of the JSSS Study

Job-Shop Scheduling System is mainly used to balance the logistics and information in the shop. It is an interface between the management and processing. Existing research on JSSP mainly consists of two aspects, which are system on theory of various algorithms and system on multi-agent.

A highly efficient hybrid genetic algorithm has been applied to the JSSP by Ju Quanyong, from which you can get a better answer in a more complex environment [1]. Bacterial foraging algorithm advocated by Wang Wenyao optimizes JSSS and it can greatly improve the efficiency of plant [2]. In addition, JSSS includes putting ant colony algorithm into job-shop scheduling, applying theory of TOC into job-shop scheduling and so on. All these systems were verified better in solving scheduling problems. A Multi-Agent System is more popular for its high degree of flexibility, quick responsiveness, intelligence and compatibility to data and knowledge uncertain. Further study on MAS has been made so as to make Multi-Agent System closer to the actual JSSP and better to solve practical scheduling problems. For example, Ma le optimized the contract net protocol (CNP) which can solve conflicts caused by the pursuit of maximizing their own interests among agents [3]. Gather the advantages of genetic algorithm and ant colony algorithm and put them into the Multi-Agent Systems, the system we got can be more closed to the actual production scheduling problems and of course a better scheme for decision.

This paper aims to combine advantages of ILOG and Multi-Agent Systems, and design a Job-Shop Scheduling System which can get a better scheduling project in the actual job-shop scheduling.

B. Application Status of JSSS

The complexity of JSSP shows the necessity of diversification and customization of JSSS. JSSS applications based on a variety of scheduling rules and algorithms have made good results at present. For example, APO (advanced planner and optimizer), developed by SAP company in Germany, is applied to the industrial environment; and Trade Matrix JSSS is widely used in the industrial sector, and so on [4]. .

MAS have a great number of advantages for solving the problem of interaction between the units. Research institutes have developed the corresponding multi-agent production management systems, and it shows that multi-agent research has walk up to application stage and has made a series of results. And this shows that discrete JSSP is the main problem. Job shop scheduling in discrete manufacturing enterprises is especially complex and the production environment is full of uncertainty. Multi-agent is characterized by high flexibility, quick responsiveness, intelligence and better compatibility to undeterministic data, which has good results for solving complex shop scheduling problems.

Scheduler, one of the sub-products of ILOG CP, specifically deal with the timing of activities and resources-based scheduling, distribution and configuration, and its powerful algorithms can effectively solve large-scale complex problems. Developers use the decision variables, constraints and objective to model for business problems, then ILOG CP narrows the search space by using domain method, until you find the right solution. Then it will optimize the solution, and finally

doi:10.4304/jsw.7.2.269-276

find the best scheduling program. For example, ILOG Fab PowerOps (ILOG FPO), developed by ILOG, is popular in the semiconductor industry. ILOG FPO can create the best looking production scheduling for all the tools in each wafer processing areas. It can greatly improve processing of the production cycle, improve tool utilization, and allows more efficient manufacturing processes to respond to interrupts, process verification and emergency formula [5].

So this paper designed a JSSS which connects the advantages of MAS and ILOG, in order to suit the complex environment of job-shop scheduling.

II FRAME DESIGN OF JOB-SHOP SCHEDULING SYSTEM BASED ON ILOG AND MAS

A. Description of JSSP

The type of JSSP in this study is n / m / G / Cmax. Here n means quantity of parts, m means machines, and Cmax represents that minimizing the maximum completion time. The mathematical model is used in this paper as follow[6].

$$\min T = \max_{1 \le k \le m} \left\{ \max_{1 \le i \le n} (c_{ik}) \right\}$$

$$s.t.\ \ \{c_{ik} - p_{ik} + M(1 - a_{ihk}) \ge c_{ih}, i \in \{1,2,\ldots n\},\ h,k \in \{1,2,\ldots m\}$$

$$\{c_{jk} - c_{ik} + M(1 - x_{ijk}) \ge p_{jk}, i, j \in \{1,2,\ldots n\},\ k \in \{1,2,\ldots m\}$$

$$c_{ik} \ge 0, i \in \{1,2,\ldots n\},\ k \in \{1,2,\ldots m\}$$

$$x_{ijk} = 0 or 1, i, j \in \{1,2,\ldots n\},\ k \in \{1,2,\ldots m\}$$

c_{ik} : Completion time for part i on machine j;

p_{ik} : Operation time for part i on machine j;

M : A positive number large enough;

a_{ihk} : Indicates coefficient, the value is 1 if machine h is priority to machine k to operate part i, otherwise the value is 0;

x_{ijk} : Indicator variable, the value is 1 if part I is priority to part j operated on machine k, otherwise the value is 0;

B. Frame Design of JSSS Based on ILOG and MAS

In order to finish a task, the Job-shop Scheduling System should have features as follows: (1) Decision making. The feature of this part is receiving assigned tasks and feeding back arrangements to the decision makers. (2) Shop scheduling. In this part what should you do is task allocation and then assigning it to each workshop. (3) Task arranging. Do reasonable arrangements to the given tasks from shop scheduling, insuring that one task matches with one machine. (4) Resources utilizing (including machinery and equipment). This section should include the utilization of machinery. When got a new task, each machine should make reasonable arrangements for the new tasks without delay the other tasks.

Therefore the Job-Shop Scheduling System based on ILOG and MAS in this paper consists of four agents: Decision Agent, Shop Scheduling Agent, Task Agent and Resources Agent. All agents perform their duties, and cooperate with each other to complete the task shop scheduling. So the JSSS, based on MAS and ILOG, should consist of decision-making agents, shop scheduling agent, task agent and resource agent. These four agents are independent, and can determine their respective tasks in accordance with predetermined rules. In the process they achieve the goals of the system; these agents will coordinate their behavior and share resources, information and functions[7].The figure.1 shows the framework of Job-Shop Scheduling System based on MAS and ILOG. This kind of structure allows each module to complete their tasks independently without the influence of other modules, and can guarantee the system to run efficiently. When getting the tasks from Decision Agent, Shop Scheduling Agent will decompose the tasks by using ILOG. Each sub-task will be generated corresponds to a Task Agent. Each Task Agent completes its tasks independently, and it takes information exchange with Resource Agent through the communication interface to achieve distributions from sub-tasks to various machines and equipment.

C. Architecture of Agents

Architecture of agent is generally divided into three categories: Deliberative Architecture, Reactive Architecture and Hybrid Architecture. In this paper, the Hybrid Architecture is used. An agent has two parts, communication interface and reasoning decision-making. Communication interface includes communication modules and communication protocol to ensure that information take-over and sending between each agent. Reasoning decision-making concludes reasoning model, action model, knowledge base, database and rule base. Functional structure of each agent is in figure 2.

III. INTERACTIVE MODE OF JOB-SHOP SCHEDULING SYSTEM BASED ON ILOG AND MAS

A. Interactive Mode of Job-Shop Scheduling System

The proposed Job-Shop Scheduling System based on ILOG and MAS is as shown in Figure 3.

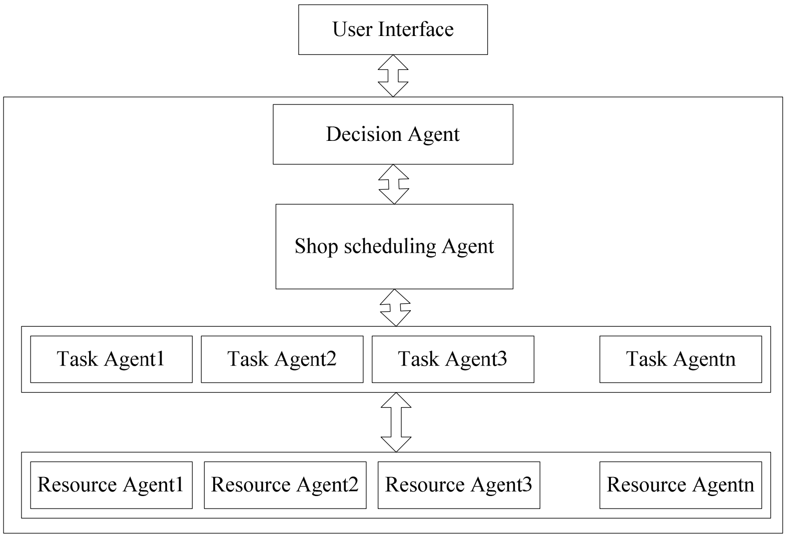

Fig. 1. Frame of Job-Shop Scheduling System based on ILOG and MAS

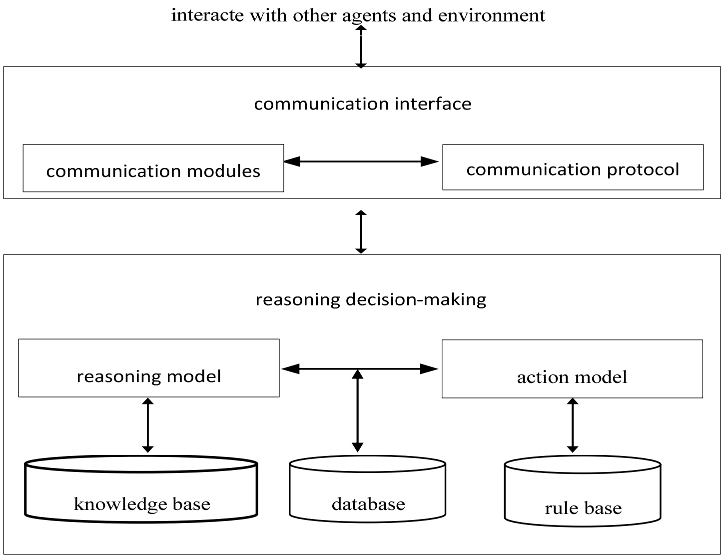

Fig.2. Functional structure of each agent

The system contains five modules as Decision Agent, Shop Scheduling Agent, Task Agent, Resource Agent and workshop capacity library.

In Fig.3, each agent has the knowledge base, human-machine interfaces and communication interfaces, even a management decision-making in decision agent. Each agent's function is marked in Fig. 3.

The agents in the system mainly are as follows:

(1) Decision Agent. Decision Agent is responsible for receiving tasks, and testing the capacity of the workshop through workshop capacity library, and then assigning tasks to the Shop Scheduling Agent. Decision Agent is also responsible for receiving feedback from the Shop Scheduling Agent, and presenting generated production tasks to decision makers for reference.

(2) Shop Scheduling Agent. When getting a new task, Shop Scheduling Agent will decompose it by using ILOG, and generate some sub-task automatically. Each sub-task will be generated corresponds to a Task Agent. In this Task Agent, there will be process list, product delivery, performance indicators, and the number of requirements and so on.

(3) Task Agent. After receiving the tender, Task Agent will calculate their existing processing time, processing costs on machines, and send message to Resources Agents to ask if they will accept each task. After receiving answers from resources agents, Task Agents will take into account processing time, cost and other factors to choose satisfied Resource Agents. At last Task Agents will send the results of feedback to the Shop Scheduling Agent and Resource Agent.

(4) Resource Agent. Resource Agent will do some calculation referring processing time, equipment utilization to decide if it is capable to bid. Satisfied resources should get the Task Agent known of its bender.

If each agent at last gets the conclusion that there is no bidding, it means that the shop cannot finish the task on the premise of meeting the constraints. Decision makers should cancel or postpone the production task. Workshop capacity library in figure 2 is used to determine whether the receiving task is in the production range. If they meet, the task was assigned; if not met, exit and display on the window that the task is not in production range.

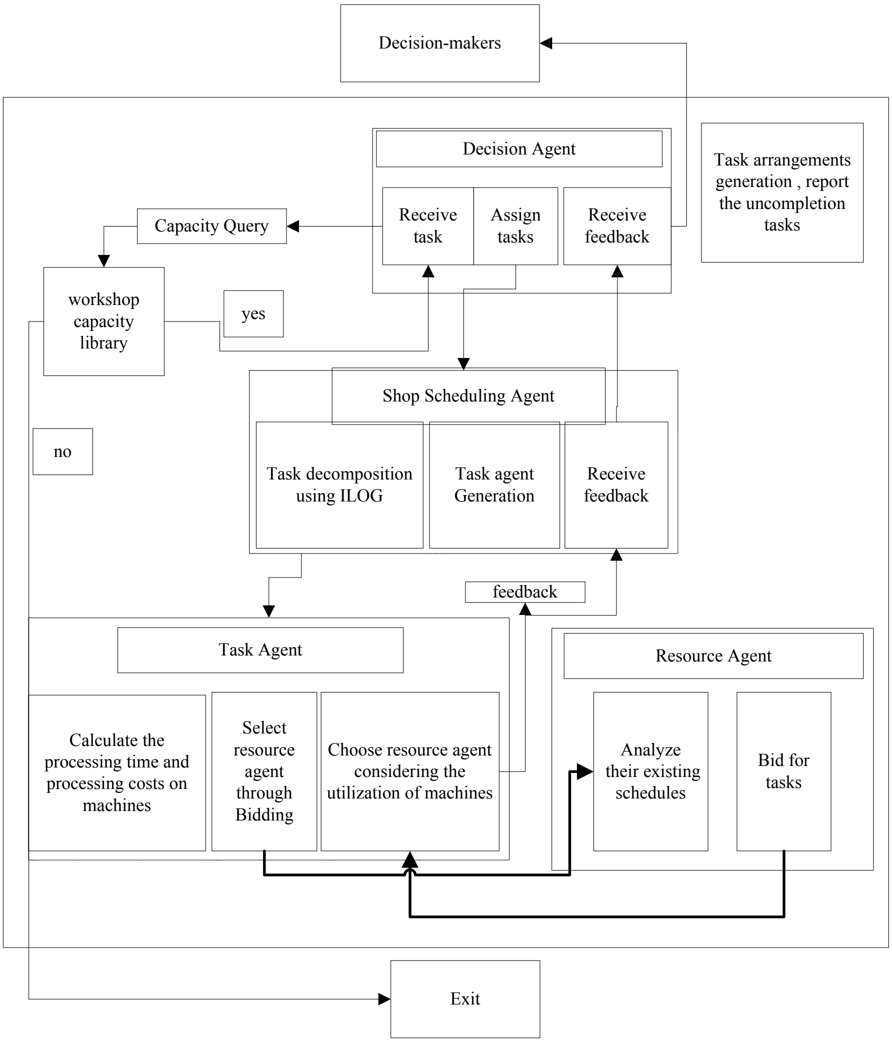

Fig. 3. Job-Shop Scheduling System Based on ILOG and MAS

B. Consultative Mechanism

Each agent has its own independence, and takes its action according to their purpose, knowledge and ability, so there will be conflicts between their activities. In order to achieve the goal, a consultative mechanism must be established.

In the manufacturing system, consultation is defined as a process that some agents discuss a task until one agent or a group of agents are chosen to perform the task[8]. The consultation process of JSSS agents in this paper is as follows[9]:

(1) Accepting tasks, and generating tasks. When Shop Scheduling Agent receives the task, it will divide it into several sub-tasks by using ILOG technology, one for each sub-task agent tender. The tender includes: description, expected completion date, the quality, the closing date and so on. The tender is issued to related Resources Agent through the net.

(2) Biding for Resource Agents. Resource Agents receive tender information according to their ability, and determine whether bids or not. Resource Agent can also make alliance proposal to related Resource Agents. Each proposal received alliance can choose to accept or refuse the proposal or exit the consultation process. Finally Resource Agents which decide to tend is asked to give the value.

(3) Assigning tasks to Resource Agents according to the feedback. As deadline expires, Task Agent will select the best standard value based on feedback to release tasks. Then it will allocate tasks to each related Resource Agents. If there is not a satisfied standard value, it will return to the first stage, and begins to assign tasks again.

C. Function of Models in Each Agent

Each agent has its own attribute, knowledge base, rule base, inference engine, behavior module and communication modules, and each perform different functions. Attribute records the information of each agent. Knowledge base stores information of environment and other related agents. Rule base prescribes the action of the agent and agreement with other agents. Inference engine processes the obtained information and make the appropriate decisions. Behaviour module drives the agent's own action, or the interaction with other agents. Communication module is responsible for implementing the interface between agents and environment, and between agents and agents. All agents perform their respective duties, cooperate together to complete the task shop scheduling [6].

1 Function of Models in Decision Agent

Module structure of Decision Agent was constructed with communication interface,man-machine interface,knowledge base and so on. Communication interface can ensure the communication between Decision Agent and Shop Scheduling Agent. Man-machine interface is used to interact with decision-maker, receiving tasks and submit scheduling project to decision-maker. There are resources information, process information in knowledge base, and these information can help to assess the order then decide whether it is in capable or not. Detailed information is as follows:

<order information>: : =<order number><arrived time><releasing time>< delivery ><weight for delay>
< Process Flow>: : =<product name><process information><material information>

2 The concrete introduction of Shop Scheduling Agent

The role of Shop Scheduling Agent is to use the advanced ILOG technology to decompose the received tasks, and sub-tasks automatically generate more than one task agent. Shop Scheduling Agent has the following five main functions: business analysis, interpreter, modeling, solving model and receiving feedback. Detailed structure and function of the agent as shown in Fig.4.

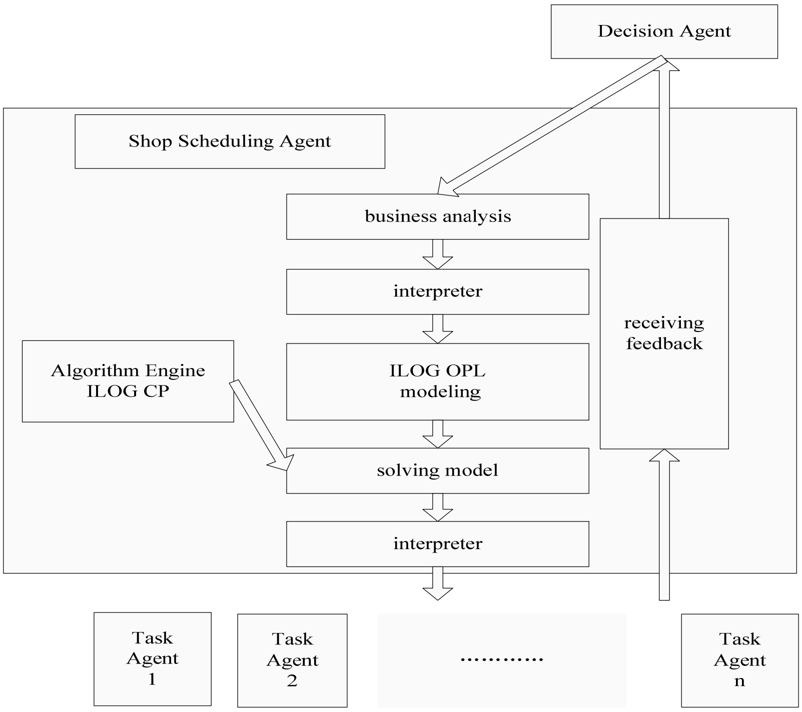

Fig.4. Detailed structure and function of Shop Scheduling Agent

(1) Business analysis. After receiving the task tested by workshop capacity library, Decision Agent will assign task to Shop Scheduling Agent. Shop Scheduling Agent analyzes the task at first, then recording the content as $\alpha \mid \beta \mid \gamma$. α is the machine environment of processing task, β describes constraints of task, γ is the task targets.

(2) Interpreter. Translate information classified in (1) into ILOG languages.

(3) Modeling in ILOG OPL. According to the information in (2), build the model file (.Mod) and the data file (.Dat) in the interface of ILOG OPL. You can also build the file (.ops) to control the solving process. Set configuration in the interface of OPL for the scheduled task.

(4) Solving model. After modeling, to get a scheme of scheduling you should call the ILOG CP algorithm. ILOG CP narrows the search space by using domain method, until you find the right solution, and then optimizes the solution, finally find the best scheduling program.

(5) Interpreter. Translate scheduling scheme of ILOG language into common language that can be recognized by each agent. Each part of scheduling scheme will be automatically generated to a task agent.

Shop Scheduling Agent, Task Agent and Decision Agent receive and feedback information by communication modules to ensure the flow of information throughout the system.

3 Function of Models in Task Agent

Task Agent is sub-task information got by the decomposition of task order, and every specific processing task corresponds to a Task Agent, which including process list, product delivery, performance indicators and quantity order. The main structure modules in Task Agent are database module, learning module and reasoning decision-making module. Database module was mainly used to storage the ability, state, location, unfinished tasks, and some processing associated parameters of Resource Agent. Learning modules adjusts task allocation strategies and reasoning strategy based on task completion to obtain satisfied sub-task allocation. Reasoning decision-making module is mainly for the calculation and assessing tasks to capable Resource Agents by the way of bidding. The main information of Task Agent contains is below:

<Task Agent>: : =<task number><task type><task sources>< delivery>< standard hours >< Processing accuracy requirement ><processing state >

< Feedback information >: : =<processing state >

4 Function of Models in Resource Agent

Resources Agent refers to the information of the workshop, such as processing capacity, processing type, processing state and others for the specific equipment resources. Resources Agent has the ability to make an independent decision analysis and scheduling performance goals and it pursues maximization of individual character abilities on the basis of meeting constraint conditions. Structure modules of Resource agent mainly consisted of date base, learning modules, reasoning decision-making module, scheduling executive module, the user interface, etc.

1) Date base: including processing time information of every task in capable for Resources Agent and providing reference for parameter settings.

2) Learning modules: According to completion information, modify processing time information, and at the same time apply for processing task circumstance to modify related parameters of scheduling strategy.

3) Decision-making module: Do some calculation related to application of processing tasks. On receiving task, put it into the queue, calculate the ability needed to complete the task according to its current state, application strategy and related parameters, and finally return information to Task Agent.

4) Scheduling executive module: Schedule the tasks in the queue.

5) User interface: Normally, Resources Agent can complete the application and processing independently, but if there are some abnormalities, personnel is needed to complete scheduling task.

Main information in Resource Agent is below:

<Resource Agent>: : =<machine information><information state>

<machine information>: : =<machine number>< products range ><processing state>< depreciation situation>

< information state>: : =< normal/fault>< queue stay processing ><competing tasks >

< Task execution >: : =< executive order >< condition monitoring >

D. Communication Mechanism Between Each Agent

Agents in the scheduling system based on ILOG and MAS communicate with each other through a finite language to complete tasks. In the present paper, KQML （Knowledge Query and Manipulation Language）is used for agents' communication. Here we give a communication way between Task Agent and Resource Agent. Msg_announce_id is the bidding contents structure, and announce and bid mean bidding.

```
(announce
        : content msg_announce_id
        : language KQML
        : ontology kqml_ontology
        : reply-with TASK-annl
        : in-reply-to RESOURCE-adv
        : sender A
        : receiver B）
(bid
        : content msg_announce_id
        : language KQML
        : ontology kqml_ontology
        : reply-with RESOURCE-adv
        : in-reply-to TASK-annl
        : sender A
        : receiver B）
Struct msg_announce_id
{
    int tasknum;
    int resourcenum;
    int jobnum;
    int bid;
    int start time;
    int end time;
  }
```

IV. Scheduling Problem in a Heavy Machinery Company

We applied the method to a heavy machinery company in Sichuan's scheduling problem, and the resolute turns to be able to satisfy the goal of minimizing the maximum completion time. These parts in the company are ordered to meet two constraints, machine constraint and processing time constraints. Machine constraints and processing time constraints are:

$$\begin{bmatrix} 1 & 4 & 1 & 5 & 6 & 1 & 7 & 5 & 6 & 1 & 7 & 0 & 0 & 0 \\ 1 & 4 & 1 & 5 & 1 & 7 & 5 & 1 & 7 & 0 & 0 & 0 & 0 & 0 \\ 1 & 4 & 1 & 8 & 9 & 2 & 7 & 8 & 9 & 3 & 7 & 0 & 0 & 0 \\ 1 & 7 & 8 & 8 & 1 & 7 & 1 & 7 & 8 & 1 & 7 & 8 & 7 & 0 \\ 1 & 7 & 8 & 8 & 1 & 7 & 1 & 7 & 8 & 1 & 7 & 8 & 7 & 0 \\ 3 & 10 & 11 & 11 & 10 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 7 & 12 & 12 & 4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 7 & 12 & 12 & 4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 4 & 5 & 5 & 4 & 6 & 1 & 10 & 10 & 5 & 5 & 9 & 1 & 10 \\ 2 & 10 & 11 & 11 & 2 & 10 & 11 & 11 & 0 & 0 & 0 & 0 & 0 & 0 \\ 2 & 10 & 11 & 11 & 2 & 10 & 11 & 11 & 0 & 0 & 0 & 0 & 0 & 0 \\ 2 & 10 & 13 & 13 & 2 & 10 & 13 & 13 & 14 & 13 & 13 & 15 & 2 & 10 \end{bmatrix}$$

6	100	2	120	100	6	24	200	120	5	38	0	0	0
8	100	2	350	6	24	400	6	16	0	0	0	0	0
5	60	5	86	68	5	16	90	78	5	16	0	0	0
4	14	120	140	2	42	4	10	80	2	6	130	10	0
4	14	120	140	2	42	4	10	80	2	6	120	10	0
4	36	50	68	10	0	0	0	0	0	0	0	0	0
5	36	140	150	38	0	0	0	0	0	0	0	0	0
5	32	110	130	30	0	0	0	0	0	0	0	0	0
6	40	200	210	30	260	5	50	5	90	140	100	4	66
2	3	26	30	2	3	15	14	0	0	0	0	0	0
2	6	28	32	2	3	16	16	0	0	0	0	0	0
2	7	22	33	1	6	19	20	36	3	36	14	1	12

To solve the scheduling problem, we made some assumptions: consider the processing times of jobs only, regardless of transport time and some other time associated with processing. Finally, the results are shown in Figure 5. The completion time with 1778 working-hours is better than the current 2000 working-hours.

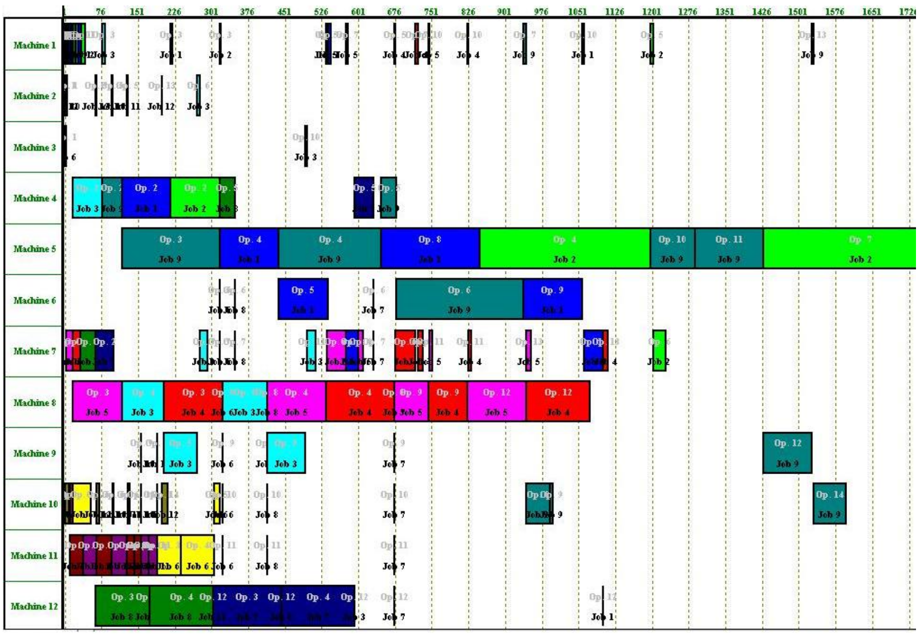

Fig.5 Scheduling result showed in Gantt

V. Conclusions

Nowadays, with the rapid development of information technology, the old mode of production cannot meet the needs of modern manufacturing systems. As the core of the entire manufacturing system, Job-Shop scheduling directly affects the production of finished products, production cycle, and core competitiveness. A good Job-shop Scheduling System is the key to improve enterprise efficiency. Prototype of this JSSS designed in this paper represents some more convenient and flexible scheduling plans in test by engineer of an automobile factory. This paper combines multi-Agent and ILOG technologies together and forms a JSSS based on MAS and ILOG. The new JSSS has a strong flexibility and global optimization, and has been proved to be a good method to solve complex shop scheduling problem.

Acknowledgments

This is one part of the study of Research on Dynamic Job Shop Scheduling Problem in Auto-mobile Industry Based on Multi-agent Method, and it is financially supported by Sichuan Youth Sci. & Tech. Foundation, Number 08ZQ026-009. It is also funded by Construction Foundation of Xihua University Transportation Planning and Management Key Subject, Number XZD0816-09-1. Thanks for their kind assistance.

Reference

[1] Lee H, Kim S.: Integration of Process Planning and Scheduling Using Simulation Based Genetic Algorithms [J]. The International Journal o f Advanced Manufacturing Technology. pp. 586-589. (2001)

[2] Mishra, S.,: A hybrid least square- fuzzy bacteria foraging strategy for harmonic estimation [J], IEEE Trans. Evol. Computer. pp. 1-3. (2005)

[3] David Zhengwen, Anthony Zhang, Ikechuk Wu.: An agent-based approach for e- manufacturing and supply chain integration [J] (in Chinese).Computer & Industrial Engineering. pp. 340- 352. (2006):

[4] (American) Ping Duo, Zhang Zhihai.: Scheduling,arithmetic and system(in Chinese). Tsinghua University Press. 2007.368-387.

[5] ILOG User's Manual, ILOG Ltd.

[6] Wang Ling: job shop schedule and genetic algorithm[M](in Chinese). Tsinghai university press2003.

[7] Li Jin,Lou Peihuang.: Application of MAS and Genetic Algorithms in Job Shop Scheduling Problem[J] (in Chinese). Machinery manufacturing and research.2004

[8] M Fox. Constrain.: directed search: A case study of job shop scheduling [Dissertation]. Carnegie-Mellon University, Pittsburgh, (1983)
[9] Zhang Jie,Shi Ke.: Research of shop floor control frame based on multi-agent system[J] (in Chinese). Huazhong University of Technology.1998.
[10] Shi Ke, Li Peigen. Agile Manufacturing Cell scheduling based on multi-Agent and CNP[J](in Chinese). Huazhong University .2001.
[11] Chu Yanhong, Cao Quanjun. Agent and CORBA-based multi-processing unit based manufacturing shop scheduling system[J] (in Chinese).2007.
[12] Yuan Kun, Zhu Jianying.: Improved genetic algorithm for the flexible job-shop scheduling with multi-object. [J] (in Chinese). Chinese Mechanical Engineering. pp. 150—158.(2007)