A Multi-Agent Based Framework for Real-Time Monitoring and Scheduling to Handle Dynamic Changes

Kwan Hee Han, Yongsun Choi and Sung Moon Bae

***Abstract*—**Manufacturing industries are under great pressure caused by the rising costs of energy, materials, labor, capital, and intensifying worldwide competition. In particular, frequent change of customer requirements is a tough challenge to manufacturing company. To deal with this problem adequately, real-time monitoring and scheduling is an essential task to be competitive. Conventional static scheduling system cannot deals with this problem effectively. The new requirements to overcome the limitations of conventional scheduling method are as follows: First, a new approach can monitor and reflect the real situation of shop floor in real-time. Second, it can model the various parameters and performance objectives of shop floor in a scheduling system with accuracy.

Agent technology is particularly appealing to model and solve the monitoring and scheduling problems in manufacturing industries. Proposed in this paper is real-time monitoring and scheduling framework based on autonomous multi-agents. And to show the usefulness of proposed framework, a prototype system is developed and implemented. The main advantage of proposed framework is to generate a realistic and easy-to-understand schedule by imitating real-world decision-making process.

***Keywords*—** Monitoring**,** Scheduling, Real-time**,** Production System, Multi-Agent, Dynamic Change

I. INTRODUCTION

As external environment of enterprise becomes more competitive, production planning and scheduling (PP&S) plays a pivotal role in meeting due date on time and allocating resources optimally. PP&S deals with short-term decision making in the production process of whole supply chain. The task of PP&S is to seek a balance between customer orders and limited resources. Production planning usually fulfils its function by determining the orders to be executed and by determining the required capacities and materials for these orders in quantity and time. The function of production scheduling on the other hand is to provide the release and execution of orders according the conditions of production planning in a certain situation. In other words, production scheduling is the process of selecting and assigning manufacturing resources for specific time periods to the set of manufacturing processes in the plan [1]. Recently, emerging requirement of production scheduling is real-time scheduling capability to cope with continuous disturbances in today’s JIT (Just-In-Time) production environment.

Production scheduling is a difficult problem, particularly when it takes place in an open, dynamic environment. In a manufacturing system, rarely do things go as expected. The set of things to do is generally dynamic. The system may be asked to do additional tasks that were not anticipated, and is sometimes allowed to omit certain tasks. The resources available to perform tasks are subject to change. Certain resources can become unavailable, and additional resources introduced. The beginning time and the processing time of a task are also subject to variation. A task can take more time than anticipated or less time than anticipated, and tasks can arrive early or late [2]. In these situations, production scheduling tasks must be coupled with real-time monitoring of hop-floor.

During two or more decades, centralized computer-based information system such as MRP (Material Requirement Planning) and ERP (Enterprise Resource Planning) which has features of top-down and sequential nature are mostly applied as a tool for PP&S in manufacturing industries. It is claimed that these batch-mode PP&S systems revealed critical shortcomings because they assume that the capacity is infinite [3]. Besides this limitation, they could not deal with production disturbances properly. In real shop floor, schedule changes occur frequently to reduce the negative impact of disturbances such as machine failure, delay of material supply, and employee absence. However, schedule change is a difficult task in the conventional scheduling system to adapt to dynamic real situations because it lacks real-time capability of shop floor status monitoring.

In summary, conventional centralized batch-mode scheduling system have limitations as bellows: 1) Static scheduling systems do not reflect dynamic changes of shop floor in real-time. 2) Conventional scheduling system does not achieve global optimization through coordinating various conflicting performance criteria. 3) It has not capability of sensitivity analysis in case of parameter changes of production environment.

Kwan Hee Han is with the Department of Industrial & Systems Engineering, Gyeongsang National University, Korea (phone: +82-55-772-1702; fax: +82-55-772-1699; e-mail: hankh@ gnu.ac.kr).

Yongsun Choi (Corresponding Author) is with Department of System Management & Engineering, Inje University, Korea (phone: +82-55-320-3117; fax: +82-55-322-3632; e-mail: yschoi@ inje.ac.kr).

Sung Moon Bae is with the Department of Industrial & Systems Engineering, Gyeongsang National University, Korea (phone: +82-55-772-1705; fax: +82-55-772-1699; e-mail: bsm@ gnu.ac.kr).

Therefore, in order to increase the predictability of shop floor progress and to ensure the reliability of production schedule, a new approach to production monitoring and scheduling is needed to overcome the limitations of existing batch-mode static scheduling. The new requirements are as follows: First, a new approach should monitor and reflect the real situation of shop floor in real-time. Second, it should model the various parameters and performance objectives of shop floor in a scheduling system with accuracy.

Because of its highly combinatorial aspects, its dynamic nature and its practical interest for manufacturing systems, the scheduling problem has been widely studied in the literature by various methods as depicted in Fig. 1 [4]: 1) Mathematical programming such as LP (linear programming) and mixed integer programming, 2) Heuristic method, 3) Simulation method. Widely used meta-heuristic techniques are as follows [5]: Genetics approaches, ant colony optimization, bees algorithm, electromagnetic like algorithm, simulating annealing, Tabu search and neural networks.

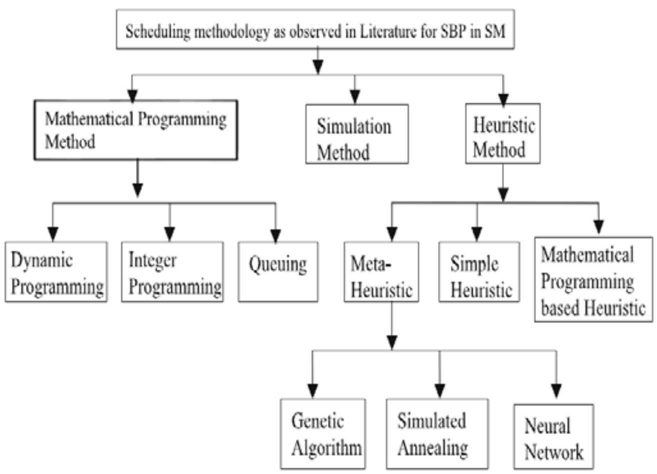

Fig. 1 Classification of solution approaches for production scheduling

Agent technology has recently been used in attempts to resolve this problem as a promising way to provide optimization [2, 4]. Agents help to capture individual interests, local decision making using incomplete information, autonomy, responsiveness, robustness and modular, distributed, reconfigurable organizational structures [7].

The most important common properties of computational agents are as follows [8]: 1) Agents act on behalf of their designer or the user they represent in order to meet a particular purpose. 2) Agents are autonomous in the sense that they control both their internal state and behavior in the environment. 3) Agents exhibit some kind of intelligence, from applying fixed rules to reasoning, planning and learning capabilities. 4) Agents interact with their environment, and in a community, with other agents. 5) Agents are ideally adaptive, i.e., capable of tailoring their behavior to the changes of the environment without the intervention of their designer.

The requirements for today's and future computer-supported manufacturing systems suggest autonomy, distribution, decentralization and flexibility, while stressing the need for control and coordination among production units based on real-time monitoring. It is expected that rigid, static and centralized manufacturing systems will give way to systems that are more adaptable to continuous rapid change.

Baker reported on a non-hierarchical multi-agent scheduling system, named market-driven contract net that performed forwards and backwards cost-driven scheduling, on a first-come-first-served basis. The system used horizontal loading in a bucket-less environment [9]. The Multi-agent paradigm represents one of the most promising approaches to building complex, flexible, and cost-effective scheduling systems because of its distributed and dynamic nature. A multi-agent based model for support dynamic scheduling in manufacturing environments was proposed [10]. They considered that a good global solution for a scheduling problem may emerge from a community of machine agents solving locally their schedules and cooperating with other machine agents that shares some relations between the operations/jobs. They addressed that the self-parameterization of the meta-heuristics will allow a better adaptation to the situation being considered (related with the problem dimension and neighborhood/population size).

Aguilar *et al.* proposed the reference model of the agent for planning and management of the production factors in the second abstraction level of the SADIA model, in order to offer a solution for the problem concerning to the enterprise resource planning and management [11]. This agent is modeled and specified by using the SCDIA framework and the MASINA methodology [12]. The SCDIA is a multi-agent platform specifically designed for control systems of distributed processes which proposes the use of a set of agents, representing the elements in a processes control loop. MASINA methodology defines 5 models as follows:

- Agent model: shows the characteristics of all the agents involved in the problem resolution.
- Task model: allows the description of the agent activities, through which the agent provides the services and accomplishes its objectives.
- Intelligence model: describes the reasoning, learning and knowledge representing mechanisms, used by the agents to accomplish their tasks.
- Coordination model: this model describes the coordination schemes between the agents, the direct and indirect communication mechanisms, the meta-languages, and the communication ontologies, among others. The coordination model is focused on services, where an agent can offer the realization of determined tasks to other agents, called services.
- Communication model: this model describes the interactions (speech acts) between the agents.

In already mentioned above, production monitoring and scheduling plays an important role in shop floor planning. A schedule shows the planned time when the processing of a specific job will start on each machine that the job requires. In real shop floor, field workers monitor shop floor status continuously and make decisions with knowledge and experiences about shop floor to cope with unexpected events. If they cannot solve these problems by themselves, they cooperate with workers of pre-/post-process or consult a supervisor on that problem. In other words, to build or adjust a production schedule requires sophisticated interactions between participants and coordination.

Software agent, which is a rule-based software object, is suitable for modeling field worker's behavior with complex interactions. Therefore, agent technology is a powerful tool to solve the dynamic problem of production monitoring and scheduling.

Besides production scheduling applications of agent technology in manufacturing organizations, Paolucci *et al.* proposed a multi-agent based system that would enable small and medium-size manufacturing organizations to dynamically build cost-effective aggregate sales and operations plans in supply chain contexts. The simulation of the agent's interactions supports planners, in supply chain operations planning, to provide multiple scenarios with respect to the balance between supply and demand [13]. This paper presented the main features of the proposed system and it finally discusses the benefits and limitations highlighted by its application in real industrial contexts. In an area of product design, Ospina and Fougéres presented the context for the use of the knowledge produced by the introduction of a mediator actor in a group of actors that work together at distance with the help of a cooperative system to give support to participants in mechanical system designer activities. They defined the types of knowledge for proposed mediation system. Then they illustrated the use of memorized knowledge by the mediator during an activity of technical functional analysis [14].

The aim of this paper is to propose a multi-agent based framework for real- time monitoring and scheduling to handle dynamic changes adequately in manufacturing industries. To fulfill this objective, a multi-agent based framework is proposed, and a prototype system based on this proposed framework is implemented to show the usefulness of proposed approach.

The rest of the paper is organized as follows. Section 2 addresses proposed multi-agent based real-time monitoring and scheduling framework. Section 3 describes an implemented prototype system for automotive parts industry as a case study. Finally, the conclusion and suggestions for further research are found in section 4.

II. Multi-Agent Based Real-Time Monitoring and Scheduling Framework

Production scheduling is a decision-making process that is used on a regular basis in many manufacturing and service industries. It deals with the allocation of resources to tasks over given time periods and its goal is to optimize one or more objectives. The objectives can also take many different forms. One objective may be the minimization of the completion time of the last task and another may be the minimization of the number of tasks completed after their respective due dates. Scheduling, as a decision-making process, plays an important role in most manufacturing systems [7].

Real world scheduling requirements are related with complex systems operating in dynamic environments. This means that they are frequently subject to several kinds of random occurrences and perturbations, such as new job arrivals, machine breakdowns, employee's sickness and job cancellation causing prepared schedules becoming easily outdated and unsuitable [16].

As an illustrative example of solution approach to deal with these dynamic disturbances, to pursuit the scheduling goal, the general strategy of coordinated rule-based scheduling approach is as follows [17]:

1) Identify the critical resources in the factory. Critical resources are those that increase the overall throughput of the facility when their constraints are relaxed. Analyze production loading to determine the resources with the greatest load.
2) Keep critical resources busy. Keep manageable levels of work in the resources input queue, choosing tasks that minimize setups, and avoiding unnecessary setups.
3) Identify server resources. Server resources have low to moderate loads and can be used to aid critical resources. Server resources can insure that critical resources have work that minimizes setup time. Server resources should look downstream and feed the most productive work to critical resources.
4) Run the model and analyze the results. Analyze the schedule performance measures established in your scheduling strategy. Pay particular attention to the utilization of critical resources. If you see a trend that could possibly be improved by making a rule change, enhance the rule and run the model again.
5) Repeat the previous steps until you are satisfied with the schedule performance.
6) Employ the selected rules with live data to schedule the facility.

In a real world situation, as depicted in Fig. 2, production is started by issuing production order to shop floor. Shop floor consists of multiple workers and machines. Eventually, these resources make finished products. During a production, when unexpected events occur, production schedule must be adjusted by a supervisor using accumulated domain knowledge and experiences.

This real world situation can be mapped to a production model using autonomous multi-agents as follows, which is also depicted in Fig. 3: A production order is modeled by task agent; A product is modeled by product agent; A human or machine is modeled by workstation agent; A supervisor is divided and modeled by 3 manager agent type as follows: A task manager agent interacts with task agent instances. A product manager agent interacts with product agents. A workstation manager agent interacts with workstation agent instances. Three types of manager agent also interact with each other independently.

Harmonious interactions through the cooperation among agents enable dynamic real-time monitoring and scheduling, which is difficult to attain in the conventional centralized rule-based batch-type approach [18, 19].

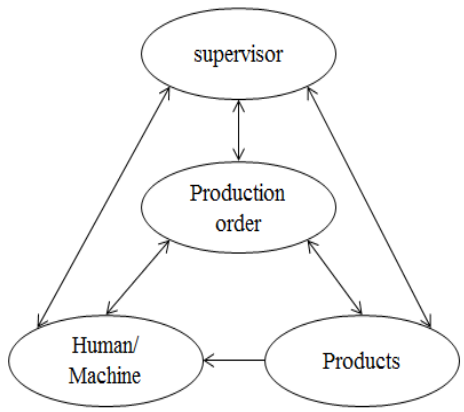

Fig. 2 Elements of real production system

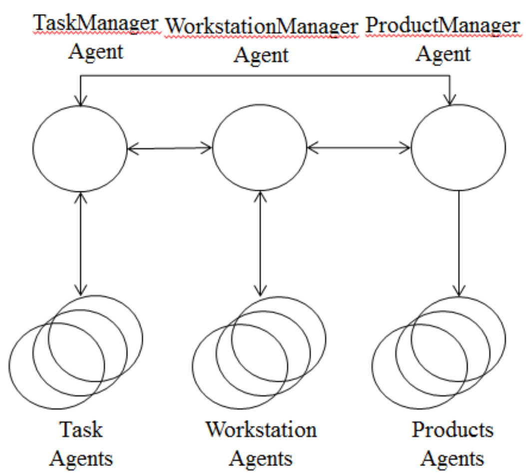

Fig. 3 Elements of multi-agent system for real-time monitoring and scheduling

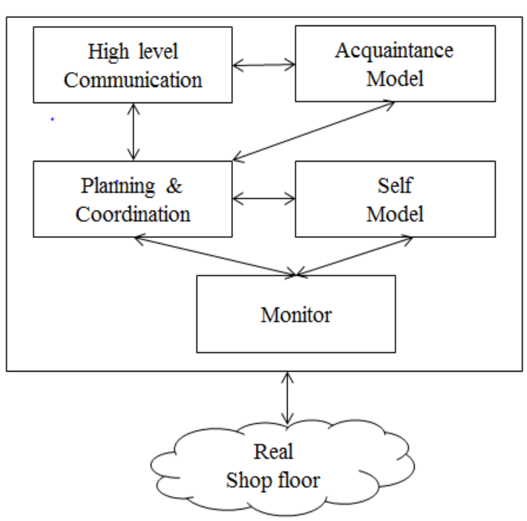

Fig. 4 Functional structure of an autonomous agent

For dynamic adaptation to continuously changing production environment, adequate allocation of agent functionality for real-time production monitoring and scheduling, like human workers, is necessary as follows: 1) decision making part, 2) communication part, 3) planning and coordination part, 4) monitoring part.

To attain the goal of production, decision making part of an agent is further classified into two sub-types: self-model maintaining domain knowledge deals with local decisions, whereas acquaintance model deals with global decision, which manages the information about other agent. These two elements are essential parts of agent for real-time scheduling [20]. Functional structure of autonomous agent is shown in Fig. 4.

High level communication part, which is implemented in CORBA (Common Object Request Broker Architecture) object**,** determines communication mode between agents. There are two messaging modes as follows: 1) Broadcasting is a messaging where a piece of information is sent from one point to all other points. In this case, there is just one sender, but the information is sent to all connected receivers. 2) Multicasting is a messaging where a piece of information is sent from one or more points to a set of other points. In this case, there is may be one or more senders, and the information is distributed to a set of receivers (there may be no receivers or any other number of receivers).

Planning and coordination part of an agent defines knowledge according to the selected decision-making strategy for cooperation. In this paper, bidding mechanism was adopted as a decision making strategy.

Monitor part of an agent traces shop floor situation in real-time and acquires input from humans or machines.

To sum up, each agent is comprised of 5 parts, and each part of an agent consists of attributes and its operations. For example, Table 1 shows the internal structure of task agent except monitor part, which is mapped to a production order in a real world. Attributes of task agent's self-model are such as production order number, processing time, order quantity and so on. Operations of self-model, which are responsibilities of task agent, are 'get_child_task()', 'set_earliest_start_time()' and so on.

Table 1. Internal structure of task agent

<< agent name>>:	Task Agent	
part name	attributes	operation
	-product number	-get/set_child_task()
	-productin line	-get/set_earliest_start/end_time()
	-production order number	
self model	-processing time	
	-order quantity	
	-production priority	
	-production ststus(wait, in-process,done)	
	-task type (auto, manual)	
	-bidding criteria	
	-parent/child task agents	
acquaintance model	-available workstation agents	
planning/coordination		-handle_event()
		-handle_event_start_task()
		-handle_event_end_task()
		-handle_evwent_delete_task()
		-do_bidding()
		-calculate_bid()
		-campare_bid()
High_level Communication		CORBA IDL

During the pursuit of their goal, autonomous agents act independently with intelligence in ordinary times. If necessary, they make decisions through the cooperation, competition and negotiation with other agents like humans. Therefore, they belong to an agent community, which has features as follows: 1) Join and withdrawal to community should be allowed to each agent when each agent is created or deleted. 2) Within a community, each agent ought to notify its capability represented as self-model to other agents.

In this paper, agent community is composed of several sub-group communities and their corresponding supervisor agent rather than one whole community as depicted in Fig. 5.

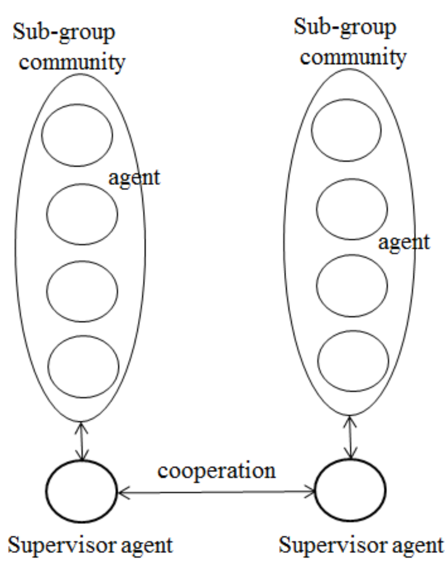

Fig. 5 Structure of agent community

This type of agent community structure has advantages as follows: 1) communication overhead for the cooperation among many agents could be reduced. 2) Addition and deletion of agents to an agent community is flexible without having impact on whole agent community.

Each autonomous agent has domain knowledge within self-model individually, which is different form conventional central rule base. Therefore, cooperation among agents is indispensable to decision making because necessary knowledge for good decision-making is distributed to related agents.

There are two approaches to agent decision making: First is a sequential pairwise comparison of all alternatives. It is suitable for small scale agent community, because it causes considerable computational and communicational overhead in the large scale community. Second is a bidding method, which is effective in the large scale agent community [21]. In this method, there exist one bidding controller and many bidders. Bidding controller requests a bid, and each bidder proposes a bid. Bidding controller selects the bidder with best bid value.

Communication among agents is usually accomplished by message transmission and receipt. To provide inter-operability between heterogeneous agents, a commonly understood agent communication language (ACL) is used: examples include KQML (Knowledge Query and Manipulation Language), Arcol, and FIPA's ACL [12]. The first ACL was the KQML that included many performatives, assertives and directives which agents use for telling facts, asking queries, subscribing to services and/or finding other agents. In this paper, CORBA (Common Object Request Broker Architecture) is adopted as a message calling among agents. In real production shop floor, message passing is occurred frequently within one communication node rather than communications between nodes, and has characteristics of small classes and many instances. In this situation, CORBA is an effective method to facilitate the communication of agents which is resulted in reducing messaging overhead substantially.

III. Application of Multi-Agent Based Real-Time Production Monitoring and Scheduling System

Building a detailed schedule that can efficiently utilize resource capacity requires careful consideration of many interacting constraints. Conventional scheduling is often so time-consuming that only one possible schedule can be produced and no other scenarios can be tried. Worse still, when order changes come along or a machine goes down, the whole thing may have to be re-worked.

In particular, in the assemble-to-order (ATO) environment, different options may be used to assemble a product that a customer wants, for example, building a computer system on customer specifications, or adding options to a basic product based on customer preference, such as in an automobile. Here, although the number of basic products is limited, the number of final products can be large. Conventional scheduling systems also revealed limitations to handle this problem.

In this paper, to improve the limitation of conventional scheduling method, and to generate a realistic production schedule, agent-based framework for real-time monitoring and scheduling system is proposed.

As a case study for implementing a proposed framework, a prototype system was applied to the press line of automotive parts company. A press line, which produces stamped panels, is comprised of blanking sub-line and stamping sub-line in series. Finished product is moved to post-process which is called a panel assembly line. In a press line, production schedule and production order is made for the stamping sub-line, and the schedule of blanking sub-line is made by backward scheduling.

Generally, the goal of production scheduling is to generate an optimal schedule meeting post-process's requirements. The need of real-time monitoring and scheduling is for adapting to disturbances due to the dynamic fluctuations of production environment.

A scheduling system developed in this paper generates a real-time reschedule if the following conditions are met as shop floor status is continuously monitored: 1) inventory level is changed (production report of stamped panel (+inventory), delivery of stamped panel to post-process (-inventory), adjustment of inventory (±inventory). 2) machine failure and repair are occurred, 3) emergency order is issued.

The above disturbing conditions cause the adjustment of current schedule. So, in a real shop floor, line supervisor solves this situation by monitoring shop floor situations and rescheduling existing schedule in cooperation with other workers. Agents of proposed multi-agent based framework imitate these human behaviors realistically.

Major reason of rescheduling is due to the low stock level of stamped panel. So, in case of above 1), safety stock level is calculated as follows: SSL=PCPH * PLT where SSL=safety stock level, PCPH=panel consumption in the post-process per hour, PLT=production lead time of press line. If (SSL > current stock level), new production order is generated, i.e., reasonable production reschedule and priority of new production order is determined with consideration to existing production orders. Production schedule of new order has time stamp such as earliest start time, latest start time, earliest finish time and latest finish time. After the schedule of stamping sub-line is determined, the schedule of blanking sub-line is calculated automatically by backward scheduling.

As illustrated in Fig. 6, in case of low stock level due to the delivery of stamped panel to post-process, detailed schedule generation procedure for new order in a press line is as follows:

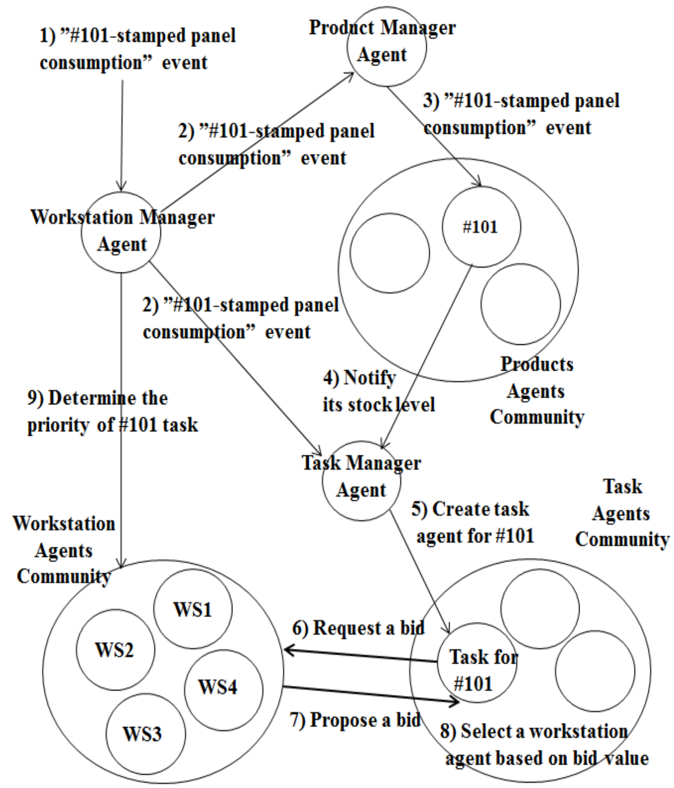

Fig. 6 schedule generation procedure for new order

1) The event of 'stamped panel (part number: #101) consumption' is notified to workstation manager agent.

2) If (safety stock level of #101> current stock level of #101), workstation manager agent notifies this event to product manager agent and task manager agent.

3) Product manager agent sends this event to product agent for #101 stamped panel within product agent community.

4) Product agent for #101 sends its stock level to task manager agent

5) Task manager agent creates the task agent for # 101 stamped panel, and registers it to agent community managed by task manager agent.

6) Task agent for #101 requests for bid to workstation agent community to determine the production line. The selection criterion of bid is an EST (Earliest Start Time) of each production line to produce this stamping panel.

7) Workstation agents that can process this task participate in bidding, and propose a bid value.

8) Task agent for #101 selects the workstation agent which proposes best value of EST.

9) After workstation agent is determined, workstation manager agent determines the processing priority of task #101 in the selected workstation by considering the relationship with existing production orders. For example, if there is an existing production order (called A) sharing same stamping die with new order (called B), the priority of B is adjusted to let it be next to A.

IV. Conclusions and Further Research

Manufacturing industries are under great pressure caused by the rising costs of energy, materials, labor, capital, and intensifying worldwide competition. In other words, external environment of enterprises are rapidly changing brought about majorly by global competition, cost and profitability pressures, and emerging new technology.

In particular, frequent change of customer requirements is a tough challenge to manufacturing company. To cope with these challenges, a scheduling function might be established to provide accountability for both customer service and operational efficiency. Scheduling encompasses allocating workloads to specific work centers and determining the sequence in which operations are to be performed. Generally, the objectives of scheduling are to achieve trade-offs among conflicting goals, which include efficient utilization of staff, equipment, facilities and minimization of customer waiting time, inventories and cycle times to cope with dynamic disturbances to shop floor.

Conventional static centralized batch-mode scheduling system could not deal with this problem effectively. Agent technology is particularly appealing to model and solve the production planning and control problems realistically in manufacturing industries.

Proposed in this paper is a real-time production monitoring and scheduling framework based on autonomous multi-agents, and prototype system was implemented in the press line of automotive parts company. As production line supervisor monitors the continuously changing shop floor situations which disturb initial schedule, agent behaves like human in making decisions and generating real-time production schedule effectively. The main advantage of proposed framework is to generate a realistic and easy-to-understand schedule by imitating real-world decision-making process.

Further research is the integration of a real-time production scheduling (RPS) system and a simulation system: Current prototype system cannot ensure whether a generated schedule is optimal or not. In the integrated structure, a RPS system generates schedule alternatives, and a simulation system evaluates each alternative in terms of pre-defined performance indicators, and the best one is selected as a new schedule.

Acknowledgment

This work was supported by Academy-oriented Research Funds of Development Fund Foundation, Gyeongsang National University, 2014.

This work was also supported by the Nuclear Power Core Technology Development Program of the Korea Institute of Energy Technology Evaluation and Planning (KETEP) granted financial resource from Ministry of Trade, Industry & Energy, Republic of Korea (No. 20131510101690).

References

[1] W. Shen, L. Wang and Q. Hao, 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 Reviewers*, Vol. 36, No. 4, pp. 563–577, 2006.

[2] W. Shen and H. N. Douglas, Agent-based systems for intelligent manufacturing: A state-of-the-art survey, *Knowledge and Information Systems*, Vol.1, No. 2, pp. 129-156, 1999.

[3] K. Udeda, Intelligent manufacturing systems-from knowledge-base to emergence-type, *Journal of the Japan Society of Precision Engineering*, Vol.5, No.11, pp. 5-10, 1993.

[4] M. Mathirajan, A.I. Sivakumar, A literature review, classification and simple meta-analysis on scheduling of batch processors in semiconductor, *International Journal of Advanced manufacturing Technology*, Vol. 29, pp. 990-1001, 2006.

[5] M. Fera, F. Fruggiero, A. Lambiase, G. Martino and M. E. Nenni, Chapter 5. Production Scheduling Approaches for Operations Management, *Operations Management*, www.intechopen.com, 2013.

[6] F. T. S. Chan, and J. Zhang, A multi-agent-based agile shop floor control system, *International Journal of Advanced Manufacturing Technology*, Vol. 19, No. 10, pp. 764–774, 2002.

[7] J. Váncza and A. MaÂrkus, An agent model for incentive-based production scheduling, *Computers in Industry*, Vol. 43, No. 2, pp. 173-187, 2000.

[8] L. Monostori, J. Váncza and S.R.T. Kumara, Agent-based systems for manufacturing, *Annals of the CIRP*, Vol. 55, No. 2, pp. 697-720, 2006.

[9] A. D. Baker, A survey of factory control algorithms that can be implemented in a multi-agent hierarchy: dispatching, scheduling and pull, *Journal of Manufacturing Systems*, Vol.17, pp. 297–320, 1998.

[10] A. Madureira and J. Santos, Proposal of multi-agent based model for dynamic scheduling in manufacturing, *Proceedings of the 6th WSEAS Int. Conf. on Evolutionary Computing*, Lisbon, Portugal, pp. 193-198, June 16-18, 2005.

[11] J. Aguilar, M. Cerrada, F. Hidrobo, J. Chacal and C. Bravo, Specification of a multi-agent system for planning and management of the production factors for automation based on the SCDIA framework and MASINA methodology, *WSEAS Transactions on Systems and Control*, Issue 2, Volume 3, pp.79-88, 2008.

[12] J. Aguilar, M. Cerrada and F. Hidrobo. A methodology to specify multi-agent systems, *Lecture Notes in Artificial Intelligence*, Springer-Verlag, Vol. 4496, pp. 92-101, 2007.

[13] M. Paolucci, R. Revetria, and F. Tonelli, An agent-based system for sales and operations planning in manufacturing supply chains, *WSEAS Transactions on Business and Economics*, Issue 3, Volume 5, pp. 103-112, 2008.

[14] V.E. Ospina and A.-J. Fougéres, Agent-based mediation system to facilitate cooperation in distributed design, *WSEAS Transactions on Computers*, Issue 6, Volume 8, pp.937- 948, 2009.

[15] M. L. Pinedo, *Scheduling-Third Edition*, Springer, New York, USA, 2008.

[16] A. Madureira, C. Ramos, and Sílvio C. Silva, Toward Dynamic Scheduling Through Evolutionary Computing, *WSEAS Transactions on Systems*, Issue 4. Volume 3, pp.1596-1604, 2004.

[17] M. Thompson, Using simulation-based Finite capacity planning and scheduling software to improve cycle time in front end operations, *1995 IEEE/SEMI Advanced Semiconductor Manufacturing Conference*, pp. 131-135, 1995.

[18] M. Wooldridge and N. R. Jennings, Agent theories, architectures, and languages: A survey, *Lecture Notes in Computer Science*, Vol. 890, pp. 1-39, 1995.

[19] Y. Zhang, G. Q. Huang, S. Sun and T. Yang, Multi-agent based real-time production scheduling method for radio frequency identification enabled ubiquitous shop floor environment, *Computers & Industrial Engineering*, Vol. 76, pp. 89–97, 2014.

[20] T. Wittig, *ARCHON: An Architecture for Multi-agent Systems*, pp. 21-22, Ellis Horwood Upper Saddle River, NJ, USA, 1992.

[21] M.K. Lim, Z. Zhang and, W.T. Goh, An iterative agent bidding mechanism for responsive manufacturing, *Engineering Applications of Artificial Intelligence*, Vol. 22, No. 7, pp. 1068–1079, 2009.

[22] J. Pitt and A. Mamdani, A Protocol-Based Semantics for an Agent Communication Language, *Proceedings of 16^{th} International Joint Conference on artificial intelligence*, pp. 486-491, 1999.