# Distributed Coordination of Project Schedule Changes: An Agent-Based Compensatory Negotiation Approach

## 건설공사 공정변경의 분산조정 : 에이전트기반의 보상협의 방식

김 기 수*
Kim, Kee-Soo

### Abstract

In the construction industry, projects are becoming increasingly large and complex, involving multiple subcontractors. Traditional centralized coordination techniques used by the general contractors become less effective as subcontractors perform most work and provide their own resources. When subcontractors cannot provide enough resources, they hinder their own performance as well as that of other subcontractors and ultimately the entire project. Thus, construction projects need a new distributed coordination approach wherein all of the concerned subcontractors can reschedule a project dynamically. To enable the distributed coordination framework of project schedule changes, the author developed an agent-based compensatory negotiation methodology, which allows intelligent software agents to simulate negotiations on behalf of their human subcontractors. In addition to this theoretical work, I designed and implemented a prototype to demonstrate the effectiveness of the framework. Thus, this research formalizes the necessary steps that would help construction project participants to increase the efficiency of their resource use, which in turn will enhance successful completions of whole projects.

Keywords : Scheduling, Project management, Software agent, Distributed coordination

## 1. Introduction

### 1.1 Research Backgrounds

Despite the ubiquity of change in large, complex construction projects, current approaches to change coordination are mostly reactive, and therefore lead to less than optimal solutions. If, however, changes in a given schedule were coordinated prior to execution, then better solutions could be found. Previous researchers have explored the various causes of schedule changes in construction projects. Discrepancies between the needed resources for activities and the resources available to subcontractors are one major cause of change. The resource discrepancies occur when the timing of the activities is not well matched with the available resources, i.e., when subcontractors have different perspectives of scheduling.

Soon after the general contractor awards subcontracts according to the master project schedule, subcontractors often want to change the master schedule because resource discrepancies cause additional costs either through over-utilizing currently available resources or importing new resources (O 'Brien and Fischer 2000). Therefore, the subcontractors may try to change the project schedule in order to accommodate their wishes. Changes are likely to cause schedule conflicts among subcontractors because any move affects the activities of other subcontractors in tightly coupled construction project schedules.

In most cases, these schedule conflicts cannot easily be resolved simply by delaying the succeeding activities since such delays would affect the resource profiles of succeeding subcontractors, which would cause additional costs for them. Delays could also extend the project completion beyond the deadline. Therefore, there is a need for a methodology to handle subcontractors resource-driven schedule changes.

---

* 일반회원, (주)대우건설 기술연구소 연구기획팀 차장, 공학박사

### 1.2 Literature Review

Numerous research papers have recognized a major problem in the Critical Path Method (CPM) network approach, which assumes unlimited resource supplies, and have provided frameworks to address various limited-resource issues in construction planning and scheduling, however, few current frameworks address the difficulties of gathering information in the coordination of subcontractors' resource-driven schedule changes. The existing centralized frameworks are insufficient because the information needed for centralized resource-based scheduling, such as the resource constraints, is usually kept private by subcontractors (Choo and Tommelein 2000). Therefore, subcontractors need a distributed coordination framework for project schedule changes (DCPSC) that includes a monetary conflict-resolution methodology, while maintaining schedule logic and keeping their information private.

Current distributed frameworks in construction and in the broader project management and AI research literature have inadequately addressed challenges for distributed coordination of project schedule changes. They have not provided a monetary conflict-resolution mechanism, which is their main shortcoming, even though some of them have provided various conflict-resolution mechanisms for interactions between participants (Koo 1987; Khedro et al. 1993; Jin and Levitt 1993). ProcessLink (Petrie et al. 1998) identifies dependencies among activities and participants but does not specify a conflict-resolution mechanism.

### 1.3 Research Objectives

To overcome past research limitations for DCPSC, the overall purpose of the research was threefold: (1) To formalize and generalize a DCPSC framework, (2) to formalize and generalize an agent-based compensatory negotiation (ABCN) methodology to enable the DCPSC framework, and (3) to implement a distributed subcontractor agent system (DSAS) to demonstrate the DCPSC framework

### 1.4 Research Questions

The author poses six research questions intended to fulfill the research objectives for DCPSC: (1) What formalism and approach can enable every subcontractor to consider its own activities, but will also enhance global outcomes? (2) What formalism can enable agents to compensate other agents for disadvantageous agreements? (3) What protocols can enable agents to identify and resolve schedule conflicts in a tightly coupled network of related activities? (4) What mechanism can enable agents to maintain work logic and ensure convergence of distributed coordination? (5) How can a multi-agent system be developed to implement the distributed agent-based coordination methodology? (6) What are the impacts on DCPSC of distributed agent-based coordination methodology?

### 1.5 Research Methodology

To answer the reserch questions, the research methodology had four major components: (1) Theory building, (2) system building, (3) verification through testing, and (4) evaluation.

The author built a theory about distributed coordination of subcontractors' resource-driven schedule changes through the case example and a survey of background literature on AI planning, Cooperative Distributed Problem Solving (CDPS), and Multi-Agent Systems (MAS) research. Using the developed theory, I built an agent-based distributed decision-making system, in which the author tested and analyzed several case examples, with changing resource profiles to prove the concept in the theory and verify the system. Finally, the author evaluated the results of my research according to its contributions.

The author identified six specific tasks corresponding to the research methodology as follows: (1) Build up research background, (2) devlop a DCPSC framework, (3) develop an ABCN methodology, (4) develop and implement the DSAS, (5) verify DCPSC, ABCN methodology, and DSAS, and (6) evaluate the research results.

The next three sections describe the DCPSC, ABCN methodology, and DSAS.

## 2. DCPSC

### 2.1 Formalization of DCPSC

In order for subcontractors to consider their own activities, and also enhance global outcomes, the author developed the distributed coordination framework for project schedule changes (DCPSC), based on the social welfare function (Feldman 1980).

To increase individual utility and social welfare together, therefore, the author set the objective of distributed coordination of project schedule changes so as to lower E, i.e., the sum of subcontractors costs associated with their resource constraints, subject to the precedence relationship among project activities:

$$\text{lower} \quad E=\sum_{i=1}^{n}\sum_{j=1}^{m} Cost_{(i,j)}$$

subject to:

$$\forall x, Finish_x \le \min_{y\in S_x}\left\{Start_y - 1\right\}$$

where E and Cost(i, j) = the sum of the subcontractors extra costs for all m activities of n subcontractors and the jth activity, which belongs to the ith subcontractor, respectively; Finishx = finish date of activity x; Starty = start date of activity y; and Sx = set of activities which must succeed activity x.

### 2.2 Two Current Centralized Coordination Methodologies

DCPSC is different from centralized coordination methodologies used in current practice. Under the centralized coordination methodologies below, the general contractor (GC) is obligated to coordinate the subcontractors. It is reasonable to assume that the information needed for coordination, such as the preferred schedules of subcontractors and resource information, is not available to the GC but is kept within subcontractors (Choo et al. 2000). The author examines two methodologies of centralized coordination tight and loose and set practical challenges for DCPSC to set up the criteria against which the author will measure goodness of a new methodology for DCPSC.

Under Tight "Iron-Fist" Centralized Coordination (TCC), the objective is to finish the project on time and the subcontractors are instructed to finish their activities before the latest finish date of each activity respectively. Under TCC, the GC can coordinate the subcontractors to finish the project on time. However, some subcontractors might experience cost overruns when their available resources differ from their resource requirements. For the GC, the cost overruns could be regarded as faults of subcontractors for poor management of their resources.

Under Loose "Laissez-Faire"Centralized Coordination (LCC), the objective is to match the resources available to produce a workable schedule. Under LCC, activities are finished when enough resources are provided. Without knowing subcontractors'resource availability, the GC instructs subcontractors to start their activities when the preceding activities have been finished and when enough resources are available; i.e., the job is ready for it and its work can proceed unimpeded (Clough and Sears 1991). LCC usually delays the project and some subcontractors might experience cost overruns due to delays of preceding activities as well as their resource deviations. The GC also incurs liquidated damages due to the project delay.

From the shortcomings of two current coordination methodologies, the author set practical challenges for DCPSC to set up the criteria against which the author will measure goodness of a new methodology for DCPSC. The first criterion is whether a new distributed coordination methodology can enable subcontractors to compensate the affected subcontractors for disadvantageous agreements. The second criterion is whether a new methodology can allow subcontractors to identify schedule conflicts, consider alternatives, and resolve schedule conflicts in a tightly coupled network of related activities. The third criterion is whether a new methodology maintains work logic and ensures convergence of distributed coordination.

Thus, DCPSC reveals three important issues: distributed coordination by competitive subcontractors, socially rational decision-making, and maintaining the logical sequence of the work.

### 2.3 Agent-Based Negotiation Approach

The author introduces an agent-based negotiation approach to overcome the difficulties stemming from these issues. In the context of this paper, the author defines agent-based negotiation as the process of resolving conflicts among affected agents by increasing knowledge about others intentions through the structured exchange of relevant information (Glossary 1999).

By adopting agent-based negotiation approach, the author can model the subcontractors as software agents performing a task on behalf of human subcontractors, while modeling the interactions among subcontractors as agent negotiation protocols based on agent communication language. Software agents can communicate rapidly with each other over the Internet, which allows subcontractors to coordinate project schedule changes with the agent-based compensatory negotiation methodology to be summarized in the next section.

## 3. Agent-Based Compensatory Negotiation Methodology

In this section, the author formalizes three main aspects of the agent-based compensatory negotiation (ABCN) methodology: (1) The compensatory negotiation strategy based on utility to the agents;

(2) the multi-linked negotiation protocols by which agents interact with other agents; and (3) message-handling mechanisms for agents to evaluate alternatives and simulate the decision-making.

### 3.1 Compensatory Negotiation Based on Utility

Each agent calculates the utility for each activity k using the following function:

$$Utility_k = AC_k - \sum_{x \in all\_suceeding\_activities_k} DC_x$$

where ACk is the extra acceleration cost for accelerating the kth activity; DCx is the extra delay cost for delaying the succeeding activity x. Note that the agent, which has activity k, knows ACk, but does not know the summation of DCx until getting DCs from the succeeding activities.

By calculating utility of timing, agents can evaluate the impacts of its schedule changes and compensate other agents for disadvantageous agreements through a utility transfer scheme. After getting DC through negotiation, if AC is more than DC, i.e., there is positive utility, the agent decides to make an extension, and transfers the DC portion of the utility to other agents for compensation of disadvantageous agreements.

### 3.2 Multi-Linked Negotiation Protocols

Negotiation protocols governthe interaction among agents by constraining the way the agents interact. In this research, agents need negotiation protocols to get DCs from succeeding agents and transfer utility to other agents for compensation of disadvantages agreements. This is simple when agents can reschedule their activities without affecting others or the counterpart agent is one, which is the case of pair-wise negotiation (Smith 1980). In a more complicated case, an agent needs to negotiate with another agent, which in turn need to negotiate with a third, and so on, until the last agent. the author calls it multi-linked negotiations.

Multi-linked negotiation protocols are needed because of the tightly coupled nature of construction project schedules. The multi-linked negotiation differs from multilateral negotiation (auction) protocols (Sandholm 1993) because multi-linked negotiation allows agents to negotiate with other agents within precedence relationships rather than restricting them to negotiate solely with an auctioneer. The negotiation protocols provide the performatives, which are shared primitive message types for agents to use in negotiation, and conversation sequence, which decides who to talk with and how to initiate and maintain the communication.

The multi-linked negotiation performatives fall into three classes: human interaction, negotiation control, and negotiation. The human interaction performatives allow a human subcontractor to provide input data to its agent and an agent to inform its subcontractor of the current status of negotiation (see table 1).

Table 1. Human interaction performatives

| Performative | Description |
|---|---|
| input | the action of providing its agent with agent information and activity informationcalling for a proposal to perform a given action |
| ready | the action of informing its agent that input is finished on the specific activity |
| final | the action of informing the subcontractor of the final result of negotiation |

The negotiation control performatives manage the states of negotiation processes (see Table 2).

Table 2. Negotiation control performatives

| Performative | Description |
|---|---|
| ready | the action of informing the preceding activity that the activity is ready for the negotiation |
| hand-over | the action of passing control to the succeeding activity, informing it of starting a new negotiation |
| done | the action of informing the preceding activity that the activity has finished all the negotiation |

The negotiation performatives facilitate the actual compensatory negotiation processes (see Table 3).

Conversation sequence decides who to talk with and how to initiate and maintain the communication. It is important to note that the project activity precedence relationships govern the agent message exchange. Since the author developed the multi-linked negotiation protocols for the distributed coordination of project schedule changes, agents should exchange messages according to the project schedule.

The author can guarantee that our protocols will converge,e from the succeeding activity reject-allthe action of rejecting the cost response from the succeeding activity confirm-all the action of confirming the “accept” from the preceding activity renege-all the action of reneging the “accept” or “reject” from the preceding activity which use static activity precedence relationships in the project schedule for message passing among agents. The author describes the conversation sequence for the multi-linked negotiation and compare it to the conversation sequence of pair-wise negotiation.

Table 3. Negotiation performatives

| Performative | Description |
|---|---|
| ask-cost | the action of askingthe succeeding activity to find out any ost, which is incurred by the delay with the proposed start date |
| reply-cost | the action of replying to the preceding activity with the cost, which is incurred by the delay |
| accept | the action of accepting the cost response from the succeeding activity |
| reject | the action of rejecting the cost response from the succeeding activity |
| confirm | the action of confirming the "accept" from the preceding activity |
| renege | the action of reneging the "accept" or "reject" from the preceding activity |
| accept-all | the action of accepting the cost response from the succeeding activity |
| reject-all | the action of rejecting the cost response from the succeeding activity |
| confirm-all | the action of confirming the "accept" from the preceding activity |
| renege-all | the action of reneging the "accept" or "reject" from the preceding activity |

Figure 1 shows a conversation sequence diagram that includes the multi-linked negotiation performatives represents his negotiation process. For his multi-linked negotiations, loops represent recursive sequences of conversation.

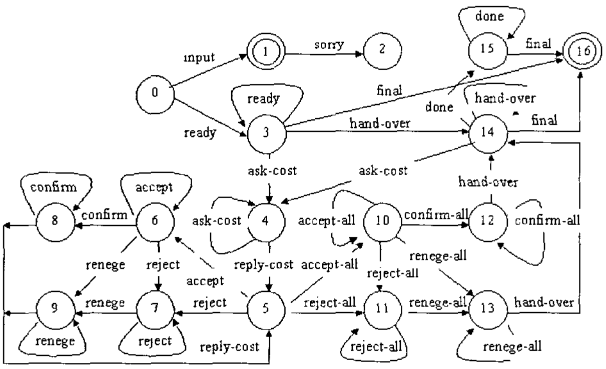

Figure 1. Diagram of multi-linked negotiation protocols

### 3.3 Message-Handling Mechanisms

The negotiating agent reacts according to what message it gets. Therefore, the negotiating agent should have the functionality of handling messages for each type of multi-linked message protocol. When the agent handles a message, it should also make a decision accordingly.

Message-handling mechanisms use the Critical Path Method (CPM) for coordination of message passing among agents so that agents exploit the sequence logic in the project schedule for coordinating message passing and to ensure successful completion of distributed computation. the author assumes that project schedules have fixed work logic and precedence relations among activities. Therefore, the message-handling mechanisms

do not allow agent to change the work logic in project schedule, but only to find a better schedule within the fixed work logic.

### 3.4 Summary of ABCN Methodology

Therefore, agents can accomplish four crucial procedures: (1) calculating utility with a utility function from the predefined schedule-change options; (2) exploring feasible alternatives by collaboration with other agents using multi-linked negotiation protocols;

(3) evaluating the impact of their alternatives; and (4) making appropriate decisions based on the evaluation. When the decisions affect other agents' activities, agents transfer utility to compensate other agents that are forced to make disadvantageous agreements, and, as a result, agents cooperatively enhance the overall project schedule in a distributed manner.

The author concludes that the proposed ABCN methodology facilitates the distributed coordination of project schedule changes by meeting practical challenges as follows: (1) by using schedule-change options based on utility of timing, an agent can compensate other agents for disadvantageous agreements; (2) by employing multi-linked negotiation protocols, agents can identify schedule conflicts, consider alternatives, and resolve schedule conflicts in a tightly coupled network of related activities; and (3) by directing message-passing based on the CPM, agents can maintain work logic and ensure convergence of distributed coordination.

## 4. Distributed Subcontractor Agent System

To test the effectiveness of the DCPSC-based ABCN methodology, the author needs to build a multi-agent system, wherein all of the concerned subcontractors can

reschedule a project dynamically through negotiations with help of software agents.

### 4.1 Requirements for DSAS

The requirements for developing the multi-agent system called distributed subcontractor agent system (DSAS) are as follows: (1)

Subcontractor agents should have the functionalities of ABCN methodology, (2) Human subcontractors can interact with their agents to provide them with the needed information for negotiations and to get the negotiation results that needed to reschedule the project, which the objective of the DCPSC.

### 4.2 DSAS Architecture

In order to test the agent-based compensatory negotiation methodology for the distributed coordination of project schedule implemented a multi-agent system called the Distributed Subcontractor Agent System (DSAS). DSAS consists of multiple subcontractor agents, multiple Graphic User Interfaces (GUIs), and an Agent Message Router (AMR), as shown in Figure 2.

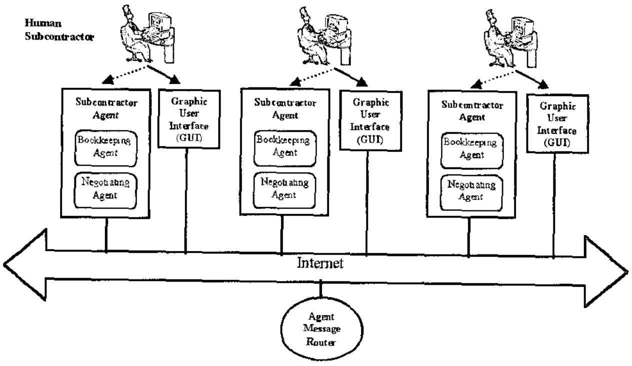

Figure 2. DSAS architecture

In DSAS, subcontractor agents negotiate with other subcontractor agents for distributed coordination of project schedule changes exchanging messages over the Internet using the Knowledge and Query Manipulation Language (KQML) (Finin et al. 1994). Through GUIs, human subcontractors can interact with their subcontractor agents to provide options and get negotiation results. The AMR provides a robust message-passing infrastructure. We implemented the subcontractor agents in the Java-based JATLite agent development tool (Jeon et al. 2000). Figure 3 shows a screen shot of DSAS.

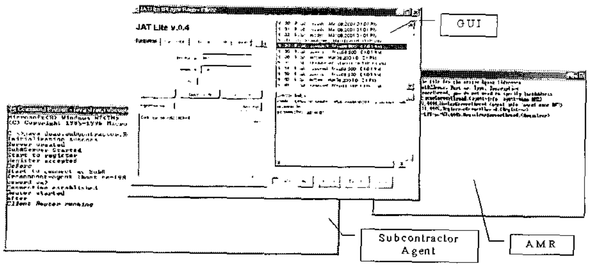

Figure 3. DSAS screen shot

## 5. Evaluations

This section demonstrates the effectiveness of the ABCN methodology for DCPSC through evaluation tests. It compares two centralized coordination methodlogies used in current practice to DCPSC-based ABCN methodology in terms of extra costs and project duration. the author conducted charrette tests of the DSAS to test the effectiveness compared to manual centralized processes. The author also conducted a series of experimental tests with different schedules to measure the system performance of DSAS.

### 5.1 Comparison Tests

The author examined two methodologies of centralized coordination -- TCC and LCC -- to DCPSC-based ABCN methodology in terms of extra costs and project duration.

Table 4 summarizes the comparison among three different coordination methodologies on a case example, which has seven activities assigned to three subcontractors.

Table 4. Summary of comparison test results

| Coordination methodology | | Extra cost($) | | | | | Duration (days) |
|---|---|---|---|---|---|---|---|
| | | Sub-α | Sub-β | Sub-δ | GC | Total | |
| Distributed | ABCN | +480 | +1,024 | 0 | 0 | +1,504 | 12 |
| Centralized | TCC | +480 | +1,920 | 0 | 0 | +2,400 | 12 |
| | LCC | +512 | +960 | +768 | +4,000 | +6,240 | 14 |

After comparing the results, the author found that DCPSC-based ABCN can find a solution that is better than or equal to any of the results from the centralized coordination methodologies. Kim (2001) shows the generalization of these evaluation results with mathematical proofs.

### 5.2 DSAS Charrette Tests

In order to test the effectiveness of DSAS, the author used the charrette test method (Clayton et al. 1998), whichFigure 4. Test results in terms of no. of messages the Center for Integrated Facility Engineering (CIFE) at Stanford University has used to test the effectiveness of software systems. The author conducted the charrette tests to compare two processes: one was a manual centralized coordination process and another was a computer-aided ABCN process on DSAS. The propositions to be tested are whether a computerized DSAS coordination produces the lower cost solution faster than a manual centralized coordination. The task of the participants was to find a better project schedule from schedule options, which were given to participants, in terms of costs and time taken. The reason why the author used the charrette test method was that it could test the effectiveness of the prototype system from the human perspective.

After two sessions of charrette tests, computerized DSAS coordination produced a lower cost solution faster than any of the manual centralized coordination efforts by two groups.

The reason for finding a soution faster is that computerized DSAS coordination used software agents that could communicate rapidly, and reasoning mechanisms inside software agents made decisions automatically. If the number of subcontractors and activities in schedules grows, the power of DSAS to produce a solution quickly will be more evident.

The reason for finding a lower-cost solution is that computerized DSAS coordination considered more schedule-change options than manual centralized coordination because humans bounded rationality limited them. However, the author did not conclude that computer agents always perform better than humans in any case because the human rationality is required in negotiation over soft issues that cannot be reduced/translated to mathematical models and discrete numerical values.

### 5.3 System Performance of DSAS

The author also measured DSAS system performance on the five CPM schedules from various sources, ranging three to twenty-seven activities; with zero to maximum changes. A change means that an activity does not have enough resources for the initial schedule.

Figure 4 shows the test results in terms of the number of messages in graphical form. The test results showed that the number of messages exchanged among subcontractor agents does not grow exponentially with the number of activities or with the number of changes. The author estimated that the worst-case computational complexity of DSAS is O(n3), where O is the approximate running time of DSAS, measured as a function of the number of activities, n, in a schedule. However, under three problems, which is common at a time in real cases, the common computational complexity is O(n2).

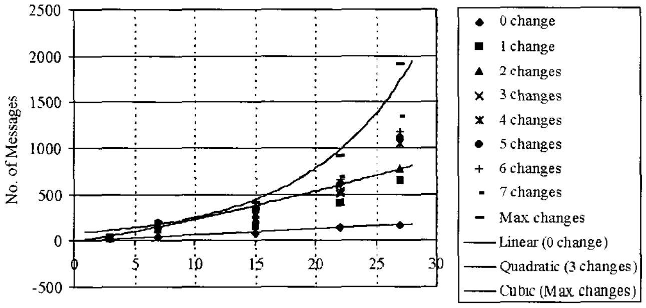

Figure 4. Test results in terms of no. of messages

## 6. Conclusions

The paper shows a distributed approach to coordination of project schedule changes and demonstrates agent-based compensatory negotiation as a vehicle for enabling this approach. The author conclude that the agent-based compensatory negotiation methodology facilitates the distributed coordination of project schedule changes by enabling subcontractors to compensate the affected subcontractors for disadvantageous agreements; by allowing subcontractors to identify and resolve schedule conflicts in a tightly coupled network of related activities; and by enabling subcontractors to maintain work logic and ensure convergence of distributed coordination. This in itself is a significant departure from prior and recent research, particularly in the area of construction project planning and scheduling, which has traditionally attempted to centralize the coordination process.

In addition to this theoretical work, the author designed and implemented a new Java-based DSAS to demonstrate the effectiveness of the DCPSC framework through a series of comparison tests, charrette tests, and measurements. This research formalizes, implements, and tests the necessary steps to help subcontractors coordinate schedule changes in order to increase the efficiency of their resource use, which in turn enhances successful completion of whole projects.

## References

1. Choo, H. J., and Tommelein, I. D. (2000). "Interactive coordination of distributed work plan." Proc., 6th Constr. Congress, Orlando, FL, ASCE, 11-20.
2. Choo, H. J., Tommelein, I. D., Ballard, G., and Zabelle, T. D. (1999). WorkPlan: Constraint-based database for work package scheduling. Journal of Construction Engineering and Management, ASCE, 125(3), 151-160.
3. Clayton, M. J., Kunz, J. C., and Fischer, M. A. (1998). "The charrette test method." Technical Rep., No. 120, CIFE, Stanford Univ., Stanford, Calif.

4. Clough, R. H., and Sears, G. A. (1991). Construction Project Management, 3rd ed., John Wiley & Sons, Inc., New York.
5. Feldman, A. M. (1980). Welfare Economics and Social Choice Theory, Martinus Nijhoff Publishing, Boston.
6. Finin, T., R. Fritzson, D. McKay, and R. McEntire (1994), "KQML as an agent communication language." Proc. 3rd International Conference On Information and Knowledge Management, ACM Press, New York, 456-463.
7. Glossary. (1999) Multiagent Systems: A Modern Approach to Dustributed Artificial Intelligence, G. Weiss ed., MIT Press, Cambridge.
8. Jeon, H., C.J. Petrie, and M.R. Cutkosky (2000), "JATLite: A Java agent infrastructure with message routing." IEEE Internet Computing, IEEE, 4(2), 87-96.
9. Jin, Y., and Levitt, R. E. (1993). "i-AGENTS: Modeling organizational problem solving in multi-agent teams." Intelligent Systems in Accounting, Finance and Management, John Wiley & Sons, Ltd., New York, 2, 247-270.

## 요 약

최근 건설 프로젝트가 대형화 및 복잡화됨에 따라 다수의 하도급업체들이 참여하는 형태로 진행되고 있다. 특히 하도급업체들이 자체 자원을 운용하여 공사를 수행함에 따라, 기존의 원 도급업체가 주도하는 중앙집중식 조정방식은 실효성이 저하되고 있다. 하도급업체의 자원수급이 건설공사의 공정과 일치하지 않을 경우 해당업체의 공사가 지연될 뿐만 아니라, 다른 하도급업체의 공사들과 더 나아가 건설공사의 지연을 초래한다. 이에 따라, 관련 하도급업체들이 해당공사에 공정변경이 생길 경우에 이에 맞추어 건설공사 공정을 조정하는 새로운 건설공사 공정변경의 분산조정에 관한 연구가 필요하다. 연구자는 건설공사 공정변경의 분산조정방식 및 이를 위한 소프트웨어 에이전트를 이용한 보상협의방법을 정의하였다. 본 연구는, (1) 타이밍 유틸리티의 새로운 정의, (2) 공정계획에 근거한 소프트웨어 에이전트간 중첩 협의를 위한 새로운 프로토콜, 그리고 (3) CPM (Critical Path Method) 에 근거한 소프트웨어 에이전트간 새로운 메시지 처리 알고리즘을 제시한다. 위와 같은 연구결과를 검증하기 위해, 연구자는 컴퓨터 프로그래밍 랭귀지인 자바(™)를 이용하여 다중 소프트웨어 에이전트 시스템의 프로토타입을 개발하여, 중앙집중식 조정방식과 비교하고, 실제 사용자를 대상으로 테스트를 수행하고, 시스템 성능검사를 마침으로서 공정변경 분산조정방식을 검증하였다. 이로서 본 연구는 현재의 건설공사 주체인 하도급업체의 자원수급의 효율성을 향상시키는 데 필요한 방법을 정의하고, 구현하고, 검증함으로서 하도급업체의 이윤추구 및 건설공사의 성공적인 수행을 함께 달성할 수 있도록 한다.

**키워드** : 공정관리, 프로젝트 매니지먼트, 소프트웨어 에이전트, 분산조정