

# A Free Market Architecture for Distributed Control of a Multirobot System

M. Bernardine Dias and Anthony Stentz  
Robotics Institute, Carnegie Mellon University  
5000 Forbes Ave., Pittsburgh, Pennsylvania 15213  
mbdias@ri.cmu.edu, tony+@cmu.edu

**Abstract**—The coordination of a large group of robots to solve a specified task is a difficult problem. Centralized approaches can be computationally intractable, brittle, and unresponsive to change. Distributed approaches are not as prone to these problems, but they can be highly sub-optimal. This work introduces a novel economic approach for coordinating robots based on the free market system. The free market approach defines revenue and cost functions across the possible plans for executing a specified task. The task is accomplished by dividing it into sub-tasks and allowing the robots to bid and negotiate to carry out these sub-tasks. Cooperation and competition emerge as the robots execute the task while trying to maximize their personal profits. Initial simulation results indicate the approach is successful at producing effective global plans for a team of several robots performing an interior sensing task.

## I. INTRODUCTION

For many applications, a team of robots can be effectively used. A robot team can accomplish a given task more quickly than a single agent can by dividing the task into sub-tasks and executing them concurrently. A team can also make effective use of specialists designed for a single purpose (e.g., scouting an area, picking up objects, hauling payload), rather than requiring that a single robot be a generalist, capable of performing all tasks but expert at no tasks.

The difficulty arises in coordinating all of these robots to perform a single, global task. One approach is to consider the robot team to be a single robot “system” with many degrees of freedom. A central computer coordinates the group optimally to perform the specified task. The problem is that optimal coordination is computationally difficult—the best known algorithms are exponential in complexity. Thus, the approach is intractable for teams larger than a few robots. Additionally, the approach assumes that all information about the robots and their environment can be transmitted to a single location for processing and that this information does not change during the time that an optimal plan is constructed. These assumptions are unrealistic for problems in which the environment is unknown and/or changing, communication is limited, and robots behave in unpredictable ways. Another weakness with this approach is that it produces a highly vulnerable system. That is, if the leader (the central planning unit) malfunctions, a new leader must be available or the entire team is disabled.

Local and distributed approaches address the problems that arise with centralized, globally coordinated approaches. Each robot operates largely independently, acting on information that is locally available through its sensors. A robot may coordinate with other robots in its vicinity, perhaps to divide a problem into multiple sub-problems or to work together on a sub-task that cannot be accomplished by a single robot. Typically, little computation is required, since each robot need only plan and execute its own activities. Also, little communication is required, since the robots only communicate with others in their vicinity. The robots are better able to respond to unknown or changing environments, since they sense and respond to the environment locally. Moreover, the system is more robust since the entire team’s performance no longer depends on the

guidance of a single leader. The approach works best for problems that can be decomposed into largely unrelated sub-problems, or problems for which a desired group behavior results from the aggregate of individual behaviors and interactions.

Consider an economic system for coordinating robots. An economy is nothing more than a population of agents (i.e., citizens) producing a global output. The agents coordinate with each other to produce an aggregate set of goods. Centralized economies, such as socialist/communist systems, suffer from an inability to gather all salient information, uncertainty in how to optimize with it, and unresponsiveness to changing conditions. Additionally, since economic output is divided equally amongst the entire population, individuals have little incentive to work harder or more efficiently than what is required to minimally comply with the economic plan. Individual input is de-coupled from individual output. The net effect is a sluggish, brittle, inefficient economy.

Free market economies are unencumbered by centralized planning; instead, individuals are free to exchange goods and services and enter into contracts as they see fit. *Despite the fact that individuals in the economy act only to advance their own self-interests, the aggregate effect is a highly productive society.* Individuals are in the best position to understand their needs and the means to satisfy them. Thus, individuals reap the direct benefits of their own good decisions and suffer the direct consequences of their bad ones. At times they cooperate with other members of the society to achieve an outcome greater than that possible by each member alone. At times they compete with other members to provide goods or services at the lowest possible cost, thus eliminating waste and inefficiency. But at every turn, the individual members act solely to reap the greatest profit for themselves. In this report, we describe a method for applying these powerful mechanisms to the task of coordinating a team of robots.

## II. RELATED WORK

The past decade has witnessed a growing focus on multi-agent systems. Matarić [9] presents a comprehensive summary of some of the principal efforts in this area of research. Jensen and Veloso [6], Švestka and Overmars [17], and Brumitt and Stentz [4] are examples of the centralized approach to control a multi-robot system organized hierarchically. A number of researchers have developed biologically inspired, locally reactive, behavior-based systems to carry out simple tasks [1, 2, 3, 9]. These distributed systems have found applications in many different domains.

Other novel approaches have been adopted to control multi-robot teams. Tambe [18] introduces a method of enabling flexible teamwork by providing the agents with general models of teamwork. Pagello et al. [10] examine multi-agent cooperation in the soccer domain through implicit communication. Veloso et al. [19] investigate methods of anticipation in order to improve cooperation of multi-robot teams in the soccer domain. Schneider-Fontán and Matarić [12, 13] present an approach of territorial division of tasks for a multi-robot team. Taking a similar approach, Parker [11] introduces a temporal division of tasks to allow fault-tolerant multi-robot cooperation.

Work most similar to our approach has been carried out mainly in the software-agent domain. Smith [16] introduces the contract net for distributing a task across agents that negotiate for subtask assignments. Golfarelli et al. [5] use a task swapping approach, based on the barter system, to globally optimize the solution to a problem. Wellman and Wurman [20] advocate many of free market's mechanisms for agents interacting in domains such as the Internet. Johnson et al. [7] examine the volatility and agent adaptability in a "bar-attendance" market model, while Lux and Marchesi [8] use a multi-agent model of financial markets to investigate scaling and criticality in the market. Zeng and Sycara [21] probe the benefits of learning in negotiation. Schwartz and Kraus [14] explore the use of negotiation for data allocation in the multi agent domain while Shehory

and Kraus [15] introduce a means of building coalitions using negotiations in order to allocate tasks in a multi agent domain. To the best of our knowledge, we are the first to use a free market architecture, with self-interested agents exchanging tasks for money, for controlling a multi-robot team to solve problems in a distributed manner.

### III. THE FREE MARKET SYSTEM

#### A. *Determining Revenues and Costs*

Consider a team of robots assembled to perform a particular task. The goal of the team is to perform the task well while minimizing costs. A function, **trev**, is needed that maps possible task outcomes onto revenue values. Another function, **tcost**, is needed that maps possible schemes for performing the task onto cost values. As a team, the goal is to execute some plan **P** such that profit, **trev(P) – tcost(P)**, is maximized.

But it is not enough to define just the revenue and cost functions for the team. These functions must provide a means for distributing the revenue and assessing costs to individual robots. Preferably, these individual revenues and costs are assigned based on factors over which the individuals have direct control. For example, if the task is to find and retrieve a set of objects, the team’s revenue, **trev**, could be the number of objects retrieved (converted to a “cash” value), and the team’s cost, **tcost**, could be the amount of energy consumed by the entire team to find the objects. The individual robot revenues and costs, **rrev** and **rcost**, could be the cash value of the number of objects turned in and the energy expended, respectively, by *that* individual.

Therefore, the sum of the individual revenues and costs equals the team’s revenues and costs. However, the distribution is not even: individuals are compensated in accordance with their contribution to the overall task, based on factors that are within the control of the individual. An individual that maximizes its own personal production and minimizes its own personal cost receives a larger share of the overall profit. Therefore, by acting strictly in their own self-interests, individuals maximize not only their own profit but also the overall profit of the team.

#### B. *The Role of Price and the Bidding Process*

Robots receive revenue and incur costs for accomplishing a specific team task, but the team’s revenue function is not the only source of income. A robot can also receive revenue from another robot in exchange for goods or services. For example, a robot may not be equipped to find objects for which the team function provides revenue, but it can transport the objects to the goal once they have been found. Therefore, this haulage robot provides a service to the robots that find the objects, and it receives payment for performing such a service.

In general, two robots have incentive to deal with each other if they can produce more aggregate profit together than apart—such outcomes are win-win rather than zero-sum. The *price* dictates the payment amount for the good or service. How is the price determined? Assume that robot **A** would like to purchase a service from robot **B**. Robot **B** incurs a cost **Y** for performing the service. Robot **A** can make an additional revenue of **X** if **B** performs the service for it. Therefore, if  $X > Y$ , then both parties have an incentive to execute the deal. But how should the composite profit,  $X - Y$ , be divided amongst the two parties? It may sound fair to split the winnings  $(X - Y) / 2$  by setting the price at  $(X + Y) / 2$ . But robots **A** and **B** may have other opportunities—they may be considering other deals that contend for the same money and resources. Since these factors may be hidden or complex, a common approach is to *bid* for a good or service until a mutually acceptable price is found. For example, robot **A** could start by bidding a price of **Y** (i.e., robot **A** receives the entire profit). Robot **B** could decline and counter

with a bid of  $\mathbf{X}$  (i.e., robot  $\mathbf{B}$  receives the entire profit). The idea is to start by bidding a price that is personally most favorable, and then successively retreat from this position until a price is mutually agreed upon.

Note that a given robot can negotiate several potential deals at the same time. It begins by bidding the most favorable price for itself for all of the deals, successively retreats from this position with counter bids, and closes the first deal that is mutually acceptable. Note also that a deal can be multi-party, requiring that all parties agree before any part of the deal is binding. The negotiated price will tend toward the intersection of the supply and demand curves for a given service. If a service is in high demand or short supply, the price will be high. This information will prompt other suppliers to enter the fray, driving the price down. Likewise, if demand is low or supply high, the low price will drive suppliers into another line of business. Thus, price serves to optimize the matching of supply to demand.

Finally, it is important to note that price and bidding are low bandwidth mechanisms for communicating aggregate information about costs. When consumers decide between purchasing apple juice or orange juice for breakfast, they do not analyze land acreage dedicated to both crops, the costs of producing each, the demand for each, and the impact of weather and pest infestations. Instead, they merely look at the price of each and weigh them against their own personal preferences. Yet the price *encodes* all of these factors in a concise fashion that enables them to make a locally optimal decision based on low-bandwidth information available at the point of sale.

### C. Cooperation vs. Competition

As described in the previous section, robots interact with each other to exchange goods and services. Two robots are *cooperative* if they have complementary roles, that is, if both robots can make more profit by working together than by working individually. Generally, robot teams foster cooperation between members of different types (heterogeneous). For instance, a robot able to grasp and lift objects and a robot able to transport objects could team together to provide a pick-and-place service that neither one could offer independently.

Conversely, two robots are competitive if they have the same role; that is, if the amount of profit that one can make is negatively affected by the presence of the other robot. Generally, robot teams foster competition amongst members of the same type (homogeneous). For instance, two robots that are able to transport objects compete for the services of a given grasping robot, thus driving the price down. Either one could charge more money if the other were not present.

These delineations are not strict however. Subgroups of heterogeneous robots could form that provide a given service. These subgroups would compete with each other, thus providing an example where robots of different types compete rather than cooperate with each other. Heterogeneous robots could also compete if the same task can be accomplished in different ways. Conversely, two robots of the same type may cooperate by agreeing to segment the market. Homogeneous robots can also cooperate if accomplishing a specific task requires more than one robot. For example, several robots with grasping capability may need to cooperate in order to move a heavy object. The flexibility of the market-model allows the robots to cooperate and compete as necessary to accomplish a task, regardless of the homogeneity or heterogeneity of the team.

### D. Self Organization

Conspicuously absent from the free market system is a rigid, top-down hierarchy. Instead, the robots organize themselves in a way that is mutually beneficial. Since the

aggregate profit amassed by the individuals is directly tied to the success of the task, this self-organization yields the best results.

Consider a group of ten robots. An eleventh robot, **A**, offers its services as their leader. It does not become their leader by coercion or decree, but by convincing the group that they will make more money by following its advice than by acting individually or in subgroups. **A** does this by investigating “plans” for utilizing all ten robots. If **A** comes up with a truly good plan, it will maximize profit across the whole group. The prospective leader can use this large profit to bid for the services of the group members, and of course, retain a portion of the profit for itself. The leader may be bidding not only against the individuals’ plans, but also against group plans produced by other prospective leaders. Note that the leader acts both as a benevolent and a self-interested agent—it receives personal compensation for efforts benefiting the entire group.

But there is a limit to this organization. As the group becomes larger, the combinatorics become intractable and the process of gathering all of the relevant information to produce a good plan becomes increasingly difficult. A leader will realize this when it can no longer convince its subjects (via bidding for their services) to follow its plans.

### *E. Learning and Adaptation*

The robot economy is able to learn new behaviors and strategies as it executes its task. This learning applies to both individual behaviors and negotiations as well as to the entire team. Individual robots may learn that certain strategies are not profitable, or that certain robots are apt to break a contract by failing to deliver the goods or proper payment. Individuals may also learn successful bidding strategies or which deals to offer when. The robot team may learn that certain types of robots are in over-supply, indicated by widespread bankruptcy or an inability to make much money. Conversely, the robot team may learn that certain types of robots are in under-supply, evidenced by excessive profits captured by members of the type. Thus, the population can learn to exit members of one type and enter members of another. Moreover, in this approach, successful agents are able to accumulate wealth and perpetuate their winning strategies because of their ability to offer higher payments to other agents.

One of the greatest strengths of the market economy is its ability to deal successfully with changing conditions. Since the economy does not rely on a hierarchical structure for coordination and task assignment, the system is highly robust to changes in the environment, including malfunctioning robots. Disabling any single robot should not jeopardize the system’s performance. By adding escape clauses for “broken deals”, any tasks undertaken by a robot that malfunctions can be re-bid to other robots, and the entire task can be accomplished. Thus, the market model allows the robots to deal with dynamic environments in an opportunistic and adaptive manner.

## IV. INITIAL IMPLEMENTATION AND RESULTS

We developed an initial version of the free market architecture and tested it in a distributed sensing problem in a simulated interior environment. A group of robots, located at different starting positions in a known simulated world, were assigned the task of visiting a set of pre-selected observation points. This problem is equivalent to the distributed traveling salesman problem, where the observation points are the cities to visit. Each robot was equipped with a map of the world, which enabled it to calculate the cost associated with visiting each of these cities. The costs were the lengths of the shortest paths between cities in an eight-connected grid, interpreted as money. Let  $c_{ij}$  be the cost for the  $j^{\text{th}}$  robot to visit the  $i^{\text{th}}$  city from the  $(i-1)^{\text{th}}$  city in its tour (where the  $0^{\text{th}}$  city is the

starting location). The robot cost function for the  $j^{\text{th}}$  robot was computed as follows:

$$\mathbf{rcost}(\mathbf{j}) = \sum_{i=1}^{n_j} \mathbf{c}_{ij}$$

where  $n_j$  is the number of cities in the tour for robot  $j$ . The team cost function was:

$$\mathbf{tcost} = \sum_{j=1}^m \mathbf{rcost}(\mathbf{j})$$

where  $m$  is the number of robots. The team revenue and robot revenue functions,  $\mathbf{trrev}$  and  $\mathbf{rrev}$ , were determined by the negotiated prices. The maximum available team revenue was chosen to exceed team costs for reasonable solutions to the problem.

All robots (bidders) adopted the same simplistic strategy of bidding a fixed percentage of the maximum profit they could obtain. According to this strategy, if a task were on offer for a maximum price of  $\mathbf{r}$ , and the cost to carry out the task were  $\mathbf{c}$ , a robot computed its bid  $\mathbf{b}$  as follows:

$$\mathbf{b} = 0.9 * (\mathbf{r} - \mathbf{c}) + \mathbf{c}$$

Thus, the robots bid for each city based on their estimated costs to visit that city.

The interface between the human operator and the team of robots was a software agent, the *operator executive* (*exec*). The *exec* conveyed the operator's commands to the members of the team, managed the team revenue, monitored the team cost, and carried out the initial city assignments. Being a self-interested agent, the *exec* aimed to assign cities quickly while minimizing revenue flow to the team. In our initial implementation, the *exec* adopted the following greedy algorithm for assigning tasks:

- Announce all cities to all robots and wait for all incoming bids
- Insert each incoming bid in a priority queue with the lowest bid claiming the highest priority
- Assign  $m$  cities (one to each robot) starting with the highest priority bid. (Note, once a city is assigned from the priority queue, all other bids for that city and all other bids submitted by that robot are removed from the queue before making the next assignment)
- Delete all bids, and call for re-bids for all remaining cities
- Repeat procedure until all  $n$  cities are assigned

Once the *exec* had completed the initial city assignments, the robots negotiated amongst themselves to subcontract city assignments. Unlike a bartering system where robots can only make "city-for-city" deals, our architecture allows robots to make "city-for-revenue" deals, thereby enabling transactions that reduce team costs but increase a robot's individual costs. Each of the robots, in turn (the initial implementation was fully synchronous), offered all the cities on its tour (individually) to all the other robots for a maximum price equal to the offerer's cost reduction by removing that city from its tour. Each bidder then submitted a bid for that city greater than the cost for adding the city to its tour. In order to estimate the additional cost of inserting a city into its tour, the bidder evaluated the cost of inserting that city at each point of the tour and picked the point of insertion that resulted in the lowest cost increase. In this initial implementation, only single-city deals were considered, and the robots continued to negotiate amongst themselves until no new, mutually profitable deals were possible. Thus, negotiations ceased once the system settled into a local minimum of the global cost.

Some preliminary results are illustrated in figures 1 and 2 below:

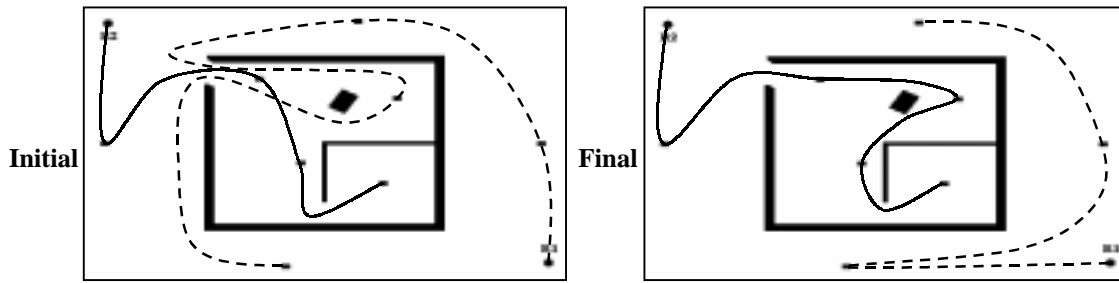


Figure 1: Initial assignments and final tours for 2 robots and 8 cities (14.7% decrease in team cost)

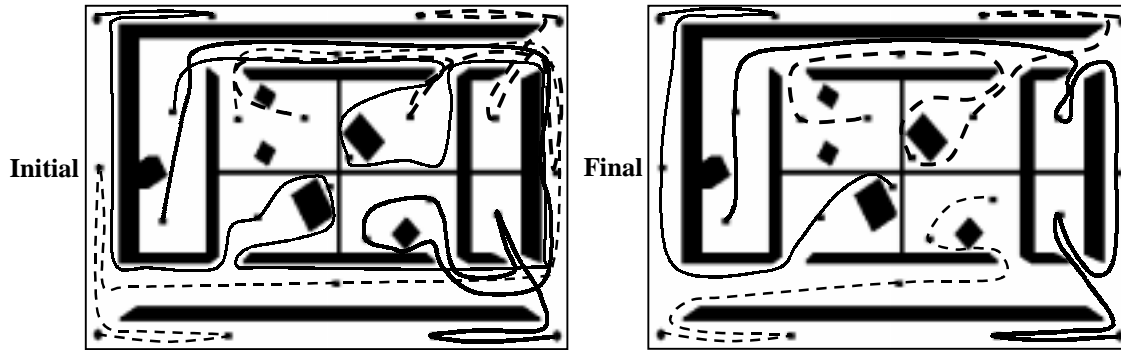


Figure 2: Initial assignments and final tours for 4 robots and 20 cities

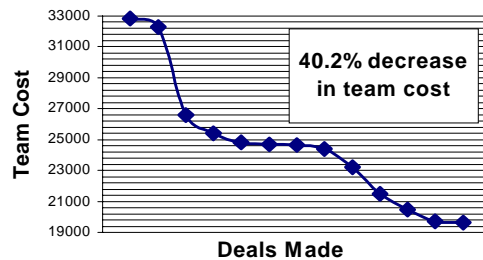


Figure 3: Team cost reduction during inter-robot negotiation

Note that the reported decrease in team costs was calculated on the operator executive's initial greedy assignment and *not on a random assignment* which would have resulted in a significantly higher initial team costs on average. Although many features of the free market architecture were not implemented in this preliminary version, initial experiments clearly show effective global plans with low team costs.

## V. CONCLUSIONS AND FUTURE WORK

This paper describes a free market architecture for distributed control of multirobot systems solving decomposable tasks. The architecture uses a revenue function to reward robots for performing subtasks. The robots negotiate amongst themselves to minimize their costs and maximize their profits. The architecture eliminates the distinction between benevolent and self-interested agents, since robots can increase their personal profits by eliminating global waste and inefficiencies. In preliminary testing on the distributed travelling salesman problem, the architecture was shown to significantly reduce the cost of the team solution through a series of city-for-revenue deals between pairs of robots. At the same time, the robots' profits increased in proportion to their contribution to the optimization process.

The results illustrate just a few aspects of the architecture. In the future, we will investigate different strategies for bidding, self-organization, and learning. We will test the architecture on a variety of problems, including those with heterogeneous robots and multiple types of subtasks. Additionally, we will benchmark our architecture against the

alternative approaches and quantify the results.

## VI. ACKNOWLEDGEMENTS

This research was sponsored in part by DARPA, under contract “Cognitive Colonies” (contract number N66001-99-1-8921, monitored by SPAWAR). The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies or endorsements of the U.S. Government. The authors thank the members of the Cognitive Colonies group for their valuable contribution: Scott Thayer, Bruce Digney, Martial Hebert, and Bart Nabbe.

## VII. REFERENCES

1. Arkin, R. C., “Cooperation without Communication: Multiagent Schema-Based Robot Navigation”, *Journal of Robotic Systems*, Vol. 9, No.3, pp. 351-364, 1992.
2. Arkin, R. C., Balch, T., “AuRA: Principles and Practice in Review”, *Journal of Experimental & Theoretical Artificial Intelligence*, Vol. 9, No. 2/3, pp.175-188, 1997.
3. Brooks, R. A., “Elephants Don’t Play Chess”, *Robotics and Autonomous Systems*, Vol. 6, pp.3-15, 1990.
4. Brumitt, B. L., Stentz, A., “Dynamic Mission Planning for Multiple Mobile Robots”, *Proceedings of the IEEE International Conference on Robotics and Automation*, No. 3, pp. 2396-2401, 1996.
5. Golfarelli, M., Maio, D., Rizzi, S., “A Task-Swap Negotiation Protocol Based on the Contract Net Paradigm”, Technical Report CSITE, No. 005-97, 1997.
6. Jensen, R. M., Veloso, M. M., “OBDD-based Universal Planning: Specifying and Solving Planning Problems for Synchronized Agents in Non-Deterministic Domains”, *Lecture Notes in Computer Science*, No. 1600, pp. 213-248, 1999.
7. Johnson, N. F., Jarvis, S., Jonson, R., Cheung, P., Kwong, Y. R., Hui, P. M., “Volatility and Agent Adaptability in a Self-Organizing Market”, *Physica A*, Vol. 258, No. 1-2, pp. 230-236, 1998.
8. Lux, T., Marchesi, M., “Scaling and Criticality in a Stochastic Multi-Agent Model of a Financial Market”, *Nature*, Vol. 397, No. 6719, pp. 498-500, 1999.
9. Mataric, M. J., “Issues and Approaches in the Design of Collective Autonomous Agents”, *Robotics and Autonomous Systems*, Vol. 16, pp. 321-331, 1995.
10. Pagello, E., D’Angelo, A., Montsello, F., Garelli, F., Ferrari, C., “Cooperative Behaviors in Multi-Robot Systems through Implicit Communication”, *Robotics and Autonomous Systems*, Vol. 29, No. 1, pp. 65-77, 1999.
11. Parker, L. E., “ALLIANCE: An Architecture for Fault Tolerant Multi-Robot Cooperation”, *IEEE Transactions on Robotics and Automation*, Vol. 14, No.2, pp. 220-240, 1998.
12. Schneider-Fontán, M., Mataric, M. J., “Territorial Multi-Robot Task Division”, *IEEE Transactions on Robotics and Automation*, Vol. 14, No. 5, 1998.
13. Schneider-Fontán, M., Mataric, M. J., “A Study of Territoriality: The Role of Critical Mass in Adaptive Task Division”, *Proceedings, From Animals to Animats 4, Fourth International Conference on Simulation of Adaptive Behavior (SAB-96)*, MIT Press/Bradford Books, pp. 553-561, 1996.
14. Schwartz, R., Kraus, S., “Negotiation On Data Allocation in Multi-Agent Environments”, *Proceedings of the AAAI National Conference on Artificial Intelligence*, pp.29-35, 1997.
15. Shehory, O., Kraus, S., “Methods for Task Allocation via Agent Coalition Formation”, *Artificial Intelligence Journal*, Vol.101, No:1-2, pp.165-200, May, 1998.
16. Smith, R., “The Contract Net Protocol: High-Level Communication and Control in a Distributed Problem Solver”, *IEEE Transactions on Computers*, Vol. C-29, No. 12, December, 1980.
17. Švestka, P., Overmars, M. H., “Coordinated Path Planning for Multiple Robots”, *Robotics and Autonomous Systems*, Vol. 23, No. 4, pp. 125-133, 1998.
18. Tambe, M., “Towards Flexible Teamwork”, *Journal of Artificial Intelligence Research*, Vol. 7, pp. 83-124, 1997.
19. Veloso, M., Stone, P., Bowling, M., “Anticipation: A Key for Collaboration in a Team of Agents”, Submitted to the 3<sup>rd</sup> International Conference on Autonomous Agents, pp. 1-16, 1998.
20. Wellman, M., Wurman, P., “Market-Aware Agents for a Multiagent World”, *Robotics and Autonomous Systems*, Vol. 24, 1998.
21. Zeng, D., Sycara, K., “Benefits of Learning in Negotiation”, *Proceedings of the AAAI National Conference on Artificial Intelligence*, pp.36-41, 1997.