Web Intelligence and Agent Systems: An international journal 3 (2005) 1–30 1
IOS Press

Hybrid negotiation for resource coordination in multiagent systems

Xin Li* and Leen-Kiat Soh
Department of Computer Science and Engineering, University of Nebraska-Lincoln, NE, USA

Abstract. In this paper, we present a coordination approach to resource allocation problem in multiagent systems. Agents adaptively coordinate resources among themselves to handle resource shortage crises resulted from events they encounter in dynamic, uncertain, real-time, and noisy environments. The coordination approach is implemented with a hybrid negotiation mechanism. The hybrid negotiation mechanism combines competitive and cooperative negotiations. In competitive negotiations, agents are self-interested and negotiate to maximize their individual performance; while in cooperative negotiations, agents are altruistic and negotiate to find a solution to help others. We define a hybrid negotiation model based on the Belief-Desire-Intention (BDI) architecture of agents, and implement the model with a specific negotiation protocol and strategy. To help agents negotiate better, we equip agents with profiling and learning capabilities. Agents profile others and learn the negotiation experience to make decisions on with whom to negotiate, and how to negotiate. We have implemented a multiagent system for coordinating the CPU resource allocation among agents based on the hybrid negotiation mechanism and conducted a series of experiments. The experimental results show that our coordination approach to resource allocation is able to reduce resource shortage crises, make the multiagent system adaptive to the variation of load, and provide efficient resource coordination among autonomous agents. The experimental results also show that the hybrid negotiation mechanism is stable for resource coordination in complex environments.

Keywords: Negotiation, coordination, multiagent systems, learning, resource allocation

1. Introduction

In recent years, Multiagent Systems (MASs) have become a highly active research area for complex applications such as distributed information processing, distributed problem solving, resource sharing, and so forth. *Coordination*, the process by which an agent reasons about its local actions and the actions of others to try to ensure the community acts in a coherent manner [13,23,27], is an important issue in multiagent systems. There are three main reasons why it is necessary for agents to coordinate [13]:

(1) There are dependencies between agents' tasks or goals.
(2) There is a need to meet global constraints such as cost and time limits.
(3) No individual agent has sufficient competence, resources, or information to solve the entire problem.

Achieving effective coordination in a multiagent system is non-trivial as no agent possesses the global view of the problem space. If all the agents in the system could have complete knowledge of the goals, actions and interaction among themselves, and could also have infinite processing capability, it would be possible to avoid conflicting and redundant efforts and the system could be perfectly coordinated [27]. In complex real-world environments, however, an agent only has incomplete information about the dynamically changing world and the occurrence of events may require the agents to react in a real-time manner. The possible noise in the environment may prevent the agents from sensing events and perceiving information accu-

*Corresponding auhtor: Xin Li, 256 Avery Hall, University of Nebraska-Lincoln, Lincoln, NE 68588-0115, USA. Tel.: +1 402 472 4257; Fax: +1 402 472 7767; E-mail: xinli@cse.unl.edu.

rately. The bandwidth is limited for agents to be constantly informed of all variations. Also, the computational and communication costs associated with a complete analysis on activities of each agent far outweigh the improvement in problem solving performance [3, 12]. Given this situation, we propose a coordination approach using hybrid negotiation mechanism to solve the multiagent resource coordination problem in complex real-world environments.

Coordination among agents can be classified as *competition* among competitive or self-interested agents, and *cooperation* among cooperative or altruistic agents [10]. These two contrasting patterns of coordination apply to different application domains. In the applications of competition, multiple agents work against each other due to their conflicting goals; in the applications of cooperation, multiple agents need to work together and utilize the collection of their capabilities and knowledge to achieve their common goals. Competitive agents try to maximize their own benefits at the expense of others, and so the success of one implies the failure of others [39]. In such coordination, the multiagent system benefits from assigning the tasks for solving a problem to the agent that can perform the tasks with the least cost. Cooperative agents try to form a team to accomplish what the individuals cannot accomplish and so fail or succeed together. In such coordination, the multiagent system benefits from agents working together.

To coordinate, agents may negotiate. *Negotiation* is a process by which two or more agents make a joint decision to coordinate their activities, each trying to reach an individual goal or objective [7,10,17,28]. Negotiation allows a bottom-up generation of an any-time solution. It means the negotiation partners can leave negotiations at any time with the solution generated, which is suitable to the real-time requirement in the complex environments. Similar to the different coordination patterns, negotiations can be competitive or cooperative. The self-interested agents negotiate competitively to achieve their mutually exclusive goals while the altruistic agents negotiate cooperatively to achieve their non-mutually exclusive goals.

Our coordination approach uses *hybrid negotiation* mechanism, combining competitive negotiation and cooperative negotiation for resource coordination in multiagent systems. The agents coordinate the resource allocation among themselves via hybrid negotiations. We adopt this mechanism taking into account the agents' characteristics. In our research domain, the agents are motivated to be both self-interested and altruistic: (1) agents work in an overall system and the overall system's resources are limited and shared by agents; (2) each agent has its local resource requirement and it knows it may encounter and suffer from resource shortage problems so it is very natural for an agent to try to maximize its own resource utilization; and (3) when agents have additional resources they are not using, they are willing to give up resources to help others. Such agents are similar to people who are generally both self-interested and altruistic. They care about their own benefits and they also would like to help others when their needs are satisfied. We define a hybrid negotiation model based on the Belief-Desire-Intention (BDI) architecture of agents [29] and implement the model with a specific negotiation protocol and strategy. In such a model, each agent holds a set of beliefs about its knowledge of the environment and other agents. These beliefs motivate them to be both self-interested and altruistic. Starting from their beliefs, the agents have different desires and goals in a negotiation as different negotiation counterparts. Since some of their goals are mutually exclusive and some are non-mutually exclusive, competition and cooperation coexist in the hybrid negotiation.

To increase the likelihood of coordination success, specifically to help agents to be better at being self-interested or altruistic in negotiations, we equip agents with profiling and learning capabilities so that they can profile other agents' behavior in negotiations and learn from their behaviors as well as their interaction patterns to adapt to the complex environment [37]. In complex environments, an agent cannot specify another agent's behavior optimally in advance due to the uncertainty of the agents' actions, and the agent also cannot know how to negotiate optimally. But to achieve the necessary degree of flexibility in coordination, an agent is required to dynamically make decisions on with whom to coordinate (or negotiate), and how to coordinate (or negotiate). Equipped with profiling and learning capabilities, an agent will be able to know which agents are more helpful to it, which agents should receive more help from it, and which negotiation strategies are better. That makes negotiations more effective and more efficient.

The objective of this paper is to build a generic, adaptive framework for coordinating resource allocation in a multiagent system. There are three inherent characteristics in our approach. First, our coordination approach is to address the requirement of complex environments. Second, the hybrid negotiation mechanism integrates the complex characteristics of autonomous agents in a multiagent system. Third, the profiling and learning capabilities of agents provide the chance of improving coordination activities.

The proposed coordination approach is aimed for a variety of applications in which agents that are both self-interested and altruistic show both competitive and cooperative behavior when coordinating sharable resources among themselves. We further applied the approach to CPU resource allocation. In a multiagent system, the CPU resource is needed by all agents and its total amount is limited in the overall system, which makes each agent to be self-interested and compete for their own performance. But an agent might be altruistic to give up part of CPU resource that it is not using to others. This results in both competitive and cooperative behavior during coordination.

The remainder of this paper is organized as follows. Section 2 introduces the coordination approach in a MAS framework. Section 3 presents the hybrid negotiation model. Section 4 describes the implementation of the hybrid negotiation mechanism in the coordination approach. Section 5 addresses the application domain of the coordination approach in the adaptive multiagent resource allocation. Section 6 presents our experiments and results in the specific application domain. In Section 7, we present some related work. Finally, Section 8 concludes and touches upon some future work.

2. Agents and multiagent system framework

In this section, we first discuss the model of our agents and the multiagent system framework. Then we describe the environment in which the agents are situated.

2.1. Agents

We assume that in our multiagent system there is a set of n autonomous agents, $Ag = \{a_i| \quad i \in 1...n\}$. Each autonomous agent a_i has a vector of resources (or capabilities) specified as $\vec{R}_{a_i} =< r_1^{a_i}, r_2^{a_i} \ldots, r^{a_i}_{|\vec{R}_{a_i}|} >$, where $|\vec{R}_{a_i}|$ refers to the size of the vector. Each resource is a property of an agent performing a specific task and each agent's resource vector size may be different since the agents may have different types of resources. That is, some resources are properties of all agents while some other resources are properties of only a subset of agents. Although it is not necessary to suppose each agent has the same resource, here, we assume such a sharable resource that is the property of some agents and the total amount of this resource is finite in the overall system (e.g., CPU resource). This assumption makes it possible and necessary for coordination to take place.

The agents in the system are assumed to have *social ability*. That means the agents are capable of interacting with other agents in order to satisfy their design objectives [40]. Specifically, each agent is assumed to have a *neighborhood* consisting of other agents it knows about and it can interact with. The neighborhood of agent a_i can be specified as $N_{a_i} = \{a_k|Know(a_i, a_k)\}$ where $Know(a_i, a_k)$ means that agent a_i knows about the existence of another agent a_k and can interact with it. Obviously, $N_{a_i} \subseteq Ag$ and each agent in N_{a_i} is a neighbor of a_i. We also have $Know(a_i, a_k) \rightarrow Know(a_k, a_i)$ which indicates agent a_i and agent a_k are neighbors to each other. The neighborhood assumption can simplify the agent's decision-making procedure and reduce the communication and computational cost in resource coordination as an agent only needs to be "familiar" and communicate with the agents in its neighborhood. An agent can be a member of multiple agents' neighborhoods, that is, agents' neighborhoods may overlap. This design supports the integrity of the system as all agents in the system can be reached. In addition, each agent's social ability is assumed to be limited in that the available communication channels are limited.

We assume that the agents in the system are both *reactive* and *rational*. Being reactive means that the agents are able to perceive their environment, and respond in a timely fashion to changes that occur in it in order to satisfy their design objectives; while being rational means that the agents are able to exhibit goal-directed behavior by taking the initiative in order to satisfy their design objectives [40]. Taking into consideration of the dynamic and uncertain environment and the real-time requirement, our agents must be reactive. On the other hand, to attain more "successful" goals, our agents need to rationalize to make the right decision according to their perception and knowledge.

We also assume that the agents in the system are both *self-interested* and *altruistic*. Being self-interested means that the agents try to maximize their own benefits and pursue their own objectives; while being altruistic means that the agents are willing to help others. Here, we also assume that our agents are *honest* and do not cheat or intend to hurt other agents. So an agent will not conceal or mislead other agents. When each agent has its own requirement of a specific resource but the resource is finite in the overall system, it is very natural for the agents to be self-interested to satisfy their own needs. On the other hand, when an agent does not have the immediate resource need, it also likes to help others by giving up the resource. This dual characteristic results in agents' both competitive and cooperative behavior in resource coordination.

2.2. Multiagent system framework

In our multiagent framework, we adopt a two-stage resource allocation mechanism: (1) reactive fixes by self, and (2) rational coordination among agents. The delineation of the two stages is based on whether the agent coordinates with other agents to solve its resource shortage problem. The two-stage mechanism also corresponds to the agent's dual characteristics of being both reactive and rational. Figure 1 shows the framework from the perspective of one agent.

Initially, each agent is allocated an initial resource amount. At run-time, each agent's resource usage changes dynamically. When the initial allocation no longer supports the agent's usage, the agent realizes that a resource shortage has occurred. If the agent can afford a coordination effort, it will initiate one. If it cannot, then it continues to monitor its resource usage, endures the impact of the resource shortage on task execution, and tries to come up with quick, reactive fixes to get over the current shortage: discarding some ongoing tasks consuming the resource. During a resource shortage, the agent evaluates its tasks to decide which ones to keep: the high-priority, time-constrained tasks are kept while low-utility tasks are discarded.

Once the current resource shortage is resolved, the agent makes a decision on *whether to coordinate*. To do it, the agent computes its possible resource requirement in the immediate future. If it thinks more resource is needed, then it initiates a series of rational coordination activities. Firstly, the agent needs to make a decision on *with whom to coordinate*. To do it, the agent will choose a list of potential coordination partners from the set of agents. It scores and ranks the helpfulness degree of each neighbor based on the profiling result of neighbors (Section 4.3). Due to the limitation of computational and communication cost, the agent only approaches the most helpful neighbors as potential coordination partners. Secondly, the agent makes a decision on *how to coordinate*. It proposes the requested resource amount, proportionately assigns the amount to each chosen coordination partner based on their degree of helpfulness, uses case-based reasoning (CBR) to select negotiation strategies (Section 4.3), and negotiates with each partner concurrently on resource reallocation. During negotiation, the agent will look to the dynamic change of its resource need. Once the need has been met, remaining ongoing negotiations will be aborted since the agents are altruistic and not greedy. Also, if the resource need changes, the agent will modify the requests. The coordination partners also need to decide how to coordinate. They use case-based reasoning to select negotiation strategies, and decide whether to grant the agents' requests fully, partially, or just reject them.

At the first stage of the resource allocation mechanism, an agent is reactive and responds in a timely manner to the resource shortage event that it encounters. It tries to get over the current crisis on its own through adaptively reducing its resource usage. At the second stage, the agent is rational and exhibits goal-directed behavior. It tries to get more resource from others in case of its future resource need. The reactive behavior allows an agent to satisfy the real-time problem-solving requirement while the rational behavior reflects the agent's long-term goal. In this paper, we focus on the second stage of the two-stage mechanism, that is, the resource coordination among agents.

2.3. Environment description

An important element in any multiagent system is the environment where the agents inhabit and share [14]. We have touched upon a dynamic, uncertain, real-time, and noisy environment in Section 1. To address the concern of our coordination approach to the complex environment, we describe it more specifically as following.

(1) The dynamism of the environment refers that, the initial states that prompt the agents' decision making process in the first place may change while the decision making process is still going on. During resource coordination, the initial states that prompt the coordination process may change. In our approach, the agents' dynamic checking and modification to the resource need during negotiation makes the flexible coordination possible. Also, the dynamic profiling on each neighbor is aimed to help the agent learn the dynamic and uncertain behavior of other agents for the future decision-making.

(2) The uncertainty of the environment refers that actions performed are not guaranteed to result in expected outcomes. Since other agents' be-

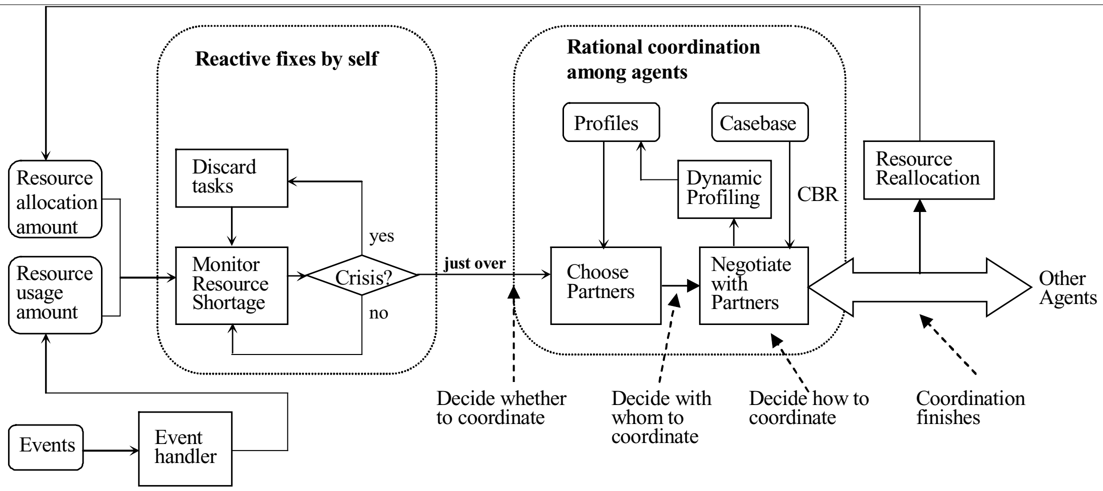

Fig. 1. Multiagent system framework.

havior may influence the outcome of an agent's action, the coordination outcome might be different from the expected by the agent that initiates the coordination process. In our approach, the agents may negotiate with multiple neighbors instead of only one, and proportionately assign requested resource amount among them. This strategy can reduce the influence of any specific partner's uncertain actions, and increase the chance of obtaining the expected amount.

(3) In the real-time environment, each agent must perform tasks in a timely manner. The coordination activities should be done within the specified time limit in order not to miss the deadlines of tasks. In our approach, we address the real-time factor in specific negotiation processes and a negotiation can finish at any time before the time limit with full offer, counter-offer, or no offer. Then the resource coordination can satisfy the real-time requirement.

(4) In a noisy environment, the sensed events by the agent may not be accurately described and the communication channels between agents may be jammed and unreliable. Thus the needed resource amount may be estimated inaccurately and the negotiations may be interrupted or proceed with incomplete information exchange. In our approach, due to its self-interested characteristic, an agent always requests more resource than the needed from other agents. Through that it also can reduce the impact of estimating less resource than the actually needed. During negotiations, negotiation-initiating agents decide which information pieces are more important to send to the responding agents first to persuade them soon. Through that they also can minimize the number and length of messages sent, since with fewer messages the agents can reduce the impact of message loss.

Due to the complexity of the environment, our coordination approach is not to obtain an optimal solution, but a "good enough, soon enough" one. In such a coordination environment, agents cannot precisely predict the outcome of coordination and it is not possible that an agent can always get the requested resource amount. Under time constraints, agents cannot afford to perform complicated coordination. Negotiation, which can finish at any time, is a good coordination approach.

3. Hybrid negotiation mechanism

In our multiagent system framework we employ a hybrid negotiation mechanism that allows agents to interact with each other and coordinate resources through negotiation. It is the heart of the proposed coordination approach. A generic negotiation may be competitive that occurs among self-interested agents [18,20], or cooperative that occurs among altruistic agents [26, 36]. In our research domain, the agents are both self-interested and altruistic, which results in the negotiations in which competition and cooperation coexist. We refer such negotiations as hybrid negotiations. In this section, we formally describe a hybrid negotiation model.

3.1. Basic definitions

To distinguish hybrid negotiations from generic negotiations, which are either competitive or cooperative (e.g. [18,20,26,36]), we present some definitions in generic negotiations. In a generic negotiation, i.e., a competitive or cooperative negotiation, autonomous agents interact with each other and seek to reach an agreement on a set of issues at the end of the negotiation process. We give the formal definitions of negotiation issues, negotiation processes, competitive negotiations, and cooperative negotiations.

Definition 1. Negotiation issue: A *negotiation issue* is the negotiated content. From the perspective of applications (e.g., e-market), specifically, a negotiation issue is any good or service that one agent can provide to another. From the perspective of problem solving in multiagent systems, specifically, a negotiation issue is a (scarce) resource or capability.

The concept of a negotiation issue is a basic concept in negotiation research [19,30,38]. It is also called a negotiation item. In this paper, we refer a negotiation issue as a resource. A negotiation may address multiple issues (resources) or only one issue (resource). Our hybrid negotiation mechanism is applicable in negotiations over multiple resources. In our domain, we assume that the agents can negotiate over different sets of resources concurrently.

Definition 2. Negotiation process: A *negotiation process* is a series of continuous negotiation activities over specific negotiation issues (resources) between a negotiation-initiating agent (initiator) that requires resources and a negotiation-responding agent (responder) that responds to the negotiation request when approached by the initiator.

Similar to the concept of a negotiation issue, a negotiation process is also a basic concept in negotiation research [19,30,38]. We define a negotiation process to specify the 1-to-1 negotiation mode (between one initiator and one responder), which is the basis of other subsequent definitions in this section. A negotiation process can be briefly represented as in Fig. 2. In general, an initiator is more conceding and agreeable as the side needing some resources and a responder is more demanding and unyielding as the side holding some resources.

Definition 3. Competitive negotiation: A *competitive negotiation* is a decision-making process of resolving a conflict between two negotiation sides (initiators and responders) over a single mutually exclusive goal.

We define a competitive negotiation based on the description in [9,24]. More specifically, in competitive negotiations, both negotiation sides are self-interested agents. The single goal of the self-interested agents is to maximize their own benefits but their individual benefits are conflicted. When a competitive negotiation process ends, only one negotiation side's goal can be achieved. One side is better off but the other is worse off as the benefit gained from negotiation shifts in a single direction.

Definition 4. Cooperative negotiation: A *cooperative negotiation* is a decision-making process of resolving a conflict between two negotiation sides (initiators and responders) over multiple interdependent, but non-mutually exclusive goals.

We define a cooperative negotiation based on the description in [9,24]. More specifically, in cooperative negotiations, both negotiation sides are altruistic agents. Either side has a set of goals but the goals of both sides are not mutually exclusive. The altruistic agents would like to help others to achieve the goals. When a cooperative negotiation process ends, it is possible that both of the sets of goals are achieved and both sides are better off as the benefits gained from negotiation shift in double directions.

3.2. Hybrid negotiation model

In essence, competitive negotiation is a win-lose type of negotiation between self-interested negotiation sides while cooperative negotiation is a win-win type of negotiation between altruistic negotiation sides. In our research domain, however, the negotiations between agents are hybrid rather than merely competitive or cooperative since our agents are both self-interested and altruistic, and their decision-making processes in negotiations will be also motivated by the dual characteristics. Similar to the definitions of competitive negotiations and cooperative negotiations, we define hybrid negotiations as follows.

Definition 5. Hybrid negotiation: A *hybrid negotiation* is a decision-making process of resolving a conflict between two negotiation sides (initiators and responders) over multiple interdependent goals, some of which are mutually exclusive and some of which are non-mutually exclusive.

Fig. 2. A negotiation process, where the arrow indicates that the initiator "initiates" the negotiation to the responder.

The definition of a hybrid negotiation integrates the definitions of a competitive negotiation and a cooperative negotiation. To describe the hybrid negotiation and distinguish the goals in hybrid negotiations from generic negotiations, we present a hybrid negotiation model from the perspective of the Belief-Desire-Intention (BDI) architecture of agents [29]. In the BDI architecture, *beliefs* represent the knowledge of the agent about the state of the environment and the agents, *desires* represent the motivations of the agent to act, and *intentions* represent the goals that the agent wants to achieve. In our research domain, we assume that when each agent, a_i, is created, it is assigned two initial beliefs on the resources $\vec{R}_{a_i}$ it holds:

$$Bel1: \ Bel_{a_i}(\neg UseWisely(a_i, \vec{R}_{a_i}) \rightarrow Suffer(a_i))$$

$$Bel2: \ Bel_{a_i}(\neg Using(a_i, \vec{R}_{a_i}) \wedge Need(Ag - \{a_i\}, \vec{R}_{a_i}) \rightarrow GiveUp(a_i, \vec{R}_{a_i}))$$

The first belief, $Bel1$, means that the agent a_i believes that it must use its resources wisely; otherwise, it will suffer from the resulted resource shortage problems. The second belief, $Bel2$, means that the agent a_i believes that when it has additional resources it is not using, it can give them up to others who are in need of the resources (Ag represents the set of autonomous agents specified in Section 2.1).

When each agent a_i is created, it is also assigned two initial beliefs about any other agent a_k it knows about:

$$Bel3: \ Bel_{a_i}(Bel_{a_k}(\neg UseWisely(a_k, \vec{R}_{a_k}) \rightarrow Suffer(a_k)))$$

$$Bel4: \ Bel_{a_i}(Bel_{a_k}(\neg Using(a_k, \vec{R}_{a_k}) \wedge Need(Ag - \{a_k\}, \vec{R}_{a_k}) \rightarrow GiveUp(a_k, \vec{R}_{a_k})))$$

The beliefs $Bel3$ and $Bel4$ indicate that each agent believes other agents that it knows about also hold the same beliefs on their resources.

Summarizing the above beliefs that agents hold, (1) $Bel1$ is the basis of motivating an agent to be self-interested in that it tries to maximize the benefits of resource utilization and it tries to avoid possible resource shortage problems; (2) $Bel2$ is the basis of motivating an agent to be altruistic in that it is willing to help others and respond to others' resource requests; and (3) $Bel3$ and $Bel4$ constitute the basis of motivating an agent to initiate negotiations with others for resource help, since the agent believes that it is possible that other agents will help.

The hybrid negotiation is defined to be applicable to multiple negotiation modes: *1-to-1*, *1-to-many*, *many-to-1*, and *many-to-many*. The *1-to-1* mode means there is one single initiator and one single responder in the negotiation over some resources. The *1-to-many* mode means there are only one single initiator but more than one responder in the negotiation over the same resources. The *many-to-1* mode means there are more than one responder but only one single initiator in the negotiation over the same resources. The *many-to-many* mode means there are more than one initiator and more than one responder in the negotiation over the same resources. Based on the above beliefs assigned to and held by agents that are both self-interested and altruistic, we infer the goals of agents in negotiations of these modes respectively.

3.2.1. The 1-to-1 mode

In the 1-to-1 negotiation mode, there are only one single initiator and one single responder, which corresponds to the negotiation process defined in Section 3.1. Based on its assigned beliefs, the initiator, a_{ini}, which is in need of resources (e.g., r), has a set of desires in the negotiation:

$$Des^1_{ini}: \ Des_{a_{ini}}(\neg Suffer(a_{ini}))$$

$$Des^2_{ini}: \ Des_{a_{ini}}(NegotiateWith(a_{res}) \wedge mostHelpful(a_{res}, a_{ini}, Ag))$$

$$Des^3_{ini}: \ Des_{a_{ini}}(Obtain(a_{ini}, r, \rho_r) \wedge \neg lessThan(amountGivenUp(a_{res}, a_{ini}, r), \rho_r))$$

$$Des^4_{ini}: \ Des_{a_{ini}}(Maximize(amountGivenUp(a_{res}, a_{ini}, r)))$$

$Des^5_{ini}:\ Des_{a_{ini}}(notEnough(a_{ini}, r) \rightarrow$

$Persuade(a_{ini}, a_{res}, r))$

$Des^6_{ini}:\ Des_{a_{ini}}(enough(a_{ini}, r) \rightarrow$

$StopNegotiation(a_{ini}, a_{res}, r))$

Here, Des^1_{ini} means the initiator a_{ini} desires not to suffer from the resource shortage problem. Des^2_{ini} means a_{ini} desires to negotiate with a responder a_{res} which is the most helpful agent to a_{ini} among all the agents in the set Ag. Des^3_{ini} means a_{ini} desires to obtain an amount of resource r, ρ_r, from a_{res}. ρ_r is the minimally needed amount by a_{ini} and is desired to be given up by the responder. Des^4_{ini} means a_{ini} desires to maximize the amount of resource r given up by a_{res} as holding additional resource will not incur any cost and getting more resource is beneficial to it. Des^5_{ini} means when the resource r is not enough to satisfy its need, a_{ini} desires to persuade a_{res} to give up resource. Des^6_{ini} means a_{ini} desires to stop the negotiation with a_{res} on resource r once resource is enough to satisfy its need. Among all the six desires, Des^1_{ini}, Des^2_{ini}, Des^3_{ini}, Des^4_{ini} and Des^5_{ini} are generated from the initiator's belief $Bel1$ while Des^6_{ini} is generated from the initiator's belief $Bel2$. So, in a way, when an agent needs resources, it is self-interested. But when it has enough resources, it is altruistic enough to stop the negotiation.

Similarly, based on its beliefs, the responder, a_{res}, which responds to the initiator's request on resources (e.g., r), has a set of desires:

$Des^1_{res}:\ Des_{a_{res}}(\neg Suffer(a_{res}))$

$Des^2_{res}:\ Des_{a_{res}}(\neg GiveUp(a_{res}, r))$

$Des^3_{res}:\ Des_{a_{res}}(GiveUp(a_{res}, r) \wedge$

$Rationalize(amountGivenUp(a_{res}, a_{ini}, r)))$

Here, Des^1_{res} means the responder desires not to suffer from the resource shortage problem. Des^2_{res} means the responder desires not to give up resource at all to avoid the resource shortage problem. Des^3_{res} means the responder desires to give up additional resource rationally to satisfy the initiator's need. Among all the desires, Des^1_{res} and Des^2_{res} are generated from the responder's belief $Bel1$ while Des^3_{res} is from both $Bel1$ and $Bel2$.

Based on these desires, the initiator and the responder will have a series of intentions (or goals) to achieve respectively in a negotiation process. *Goals* are a mutually consistent set of desires [33]. The initiator has two goal sets, $\{Des^1_{ini}, Des^2_{ini}, Des^3_{ini}, Des^4_{ini}, Des^5_{ini}\}$ and $\{Des^1_{ini}, Des^2_{ini}, Des^3_{ini}, Des^5_{ini}, Des^6_{ini}\}$, since Des^4_{ini} and Des^6_{ini} are inconsistent with each other. The responder also has two goal sets, $\{Des^1_{res}, Des^2_{res}\}$ and $\{Des^1_{res}, Des^3_{res}\}$, since Des^2_{res} and Des^3_{res} are inconsistent with each other.

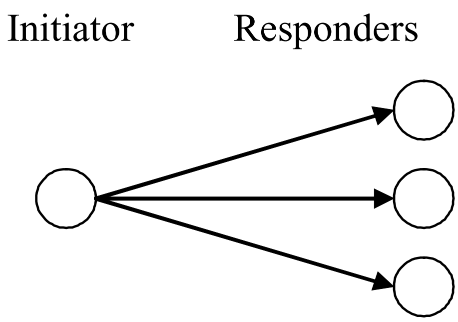

Fig. 3. The 1 – to – many negotiation mode.

Comparing the goals of the initiator and the responder, we can find that in a negotiation process, the initiator and the responder have multiple interdependent goals. Some of these goals are mutually exclusive (e.g., $\{Des^1_{ini}, Des^2_{ini}, Des^3_{ini}, Des^4_{ini}, Des^5_{ini}\}$ and $\{Des^1_{res}, Des^2_{res}\}$) while some are non-mutually exclusive (e.g., $\{Des^1_{ini}, Des^2_{ini}, Des^3_{ini}, Des^5_{ini}, Des^6_{ini}\}$ and $\{Des^1_{res}, Des^3_{res}\}$). To achieve these goals, the negotiation between the initiator and the responder integrates the characteristics of competitive negotiation and cooperative negotiation, which forms the hybrid negotiation.

3.2.2. *The 1-to-many mode*

In the 1-to-many negotiation mode, there is only one single initiator but there is more than one responder. To request resources, an agent initiates a set of concurrent negotiation processes with multiple responders at the same time, which can be briefly represented as in Fig. 3.

Obviously, the 1-to-1 negotiation mode is the special case of the 1-to-many negotiation mode. We can generalize our discussion on agents' goals in the 1-to-1 negotiation mode to the 1-to-many negotiation mode.

Based on its assigned beliefs, the initiator, a_{ini}, which is in need of resources (e.g., r), has a set of desires in negotiations with the responders. Note that an initiator can initiate multiple negotiations to request resources from multiple responders at the same time, so we use $\{a_{res}\}$ to represent all responders in the following desires:

$Des^1_{ini}:\ Des_{a_{ini}}(\neg Suffer(a_{ini}))$

$Des^2_{ini}:\ Des_{a_{ini}}(Maximize(numberOf$

$Negotiations(a_{ini}, r)))$

$Des^3_{ini}:\ Des_{a_{ini}}(NegotiateWith(\{a_{res}\}) \wedge mostHelpful(\{a_{res}\}, a_{ini}, Ag))$

$Des^4_{ini}:\ Des_{a_{ini}}(Obtain(a_{ini}, r, \rho_r) \wedge \neg lessThan(amountGivenUp(\{a_{res}\}, a_{ini}, r), \rho_r))$ $Des^5_{ini}:\ Des_{a_{ini}}(Maximize(amountGivenUp(\{a_{res}\}, a_{ini}, r)))$

$Des^6_{ini}:\ Des_{a_{ini}}(notEnough(a_{ini}, r) \rightarrow Persuade(a_{ini}, \{a_{res}\}, r))$

$Des^7_{ini}:\ Des_{a_{ini}}(enough(a_{ini}, r) \rightarrow StopNegotiation(a_{ini}, \{a_{res}\}, r))$

The desires in 1-to-many mode are similar to the ones in 1-to-1 mode but they are specific to the set of responders rather than to one single responder. Specifically, Des^2_{ini} means the initiator a_{ini} desires to maximize the number of negotiations over the resource r so that it may maximize the chance of getting resource. The difference is also in how the initiator behaves. It has more flexibility and room to change its request. That is, it can make use of the different negotiation outcomes achieved so far to refine its ongoing negotiations to be self-interested – to salvage from failed negotiations, and to be altruistic – to relax its requests due to successful negotiations.

In the 1-to-many negotiation mode, each negotiation process is separate and each responder makes decisions independently. So a responder is not aware of other responders and other negotiation processes. The desires of a responder in the 1-to-many mode are same as the ones in the 1-to-1 mode:

$Des^1_{res}:\ Des_{a_{res}}(\neg Suffer(a_{res}))$

$Des^2_{res}:\ Des_{a_{res}}(\neg GiveUp(a_{res}, r))$

$Des^3_{res}:\ Des_{a_{res}}(GiveUp(a_{res}, r) \wedge Rationalize(amountGivenUp(a_{res}, a_{ini}, r)))$

Based on these desires, the initiator and the responders will have a series of goals to achieve respectively in the overall negotiation. Their goals are interdependent. Some of these goals are mutually exclusive (e.g., $\{Des^1_{ini}, Des^2_{ini}, Des^3_{ini}, Des^4_{ini}, Des^5_{ini}, Des^6_{ini}\}$ and $\{Des^1_{res}, Des^2_{res}\}$) while some are non-mutually exclusive (e.g., $\{Des^1_{ini}, Des^2_{ini}, Des^3_{ini}, Des^4_{ini}, Des^6_{ini}, Des^7_{ini}\}$ and $\{Des^1_{res}, Des^3_{res}\}$). To achieve these goals, the 1-to-many negotiation between the initiator and the responders integrates the characteristics of competitive negotiation and cooperative negotiation, which forms the hybrid negotiation.

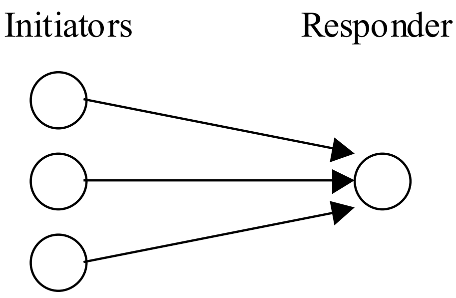

Fig. 4. The many – to – 1 negotiation mode.

3.2.3. The many-to-1 mode

In the many-to-1 negotiation mode, there is only one single responder but there is more than one initiator. Through different negotiation processes, multiple initiators request for same resources from the responder at the same time, which can be briefly represented as in Fig. 4.

Obviously, the 1-to-1 negotiation mode is the special case of the many-to-1 negotiation mode. We can generalize our discussion on agents' goals in the 1-to-1 negotiation mode to the many-to-1 negotiation mode.

Based on its assigned beliefs, each initiator a_{ini} in the set of initiators ($\{a_{ini}\}$), which is in need of resources (e.g., r), has a set of desires in the individual negotiation process with the same responder a_{res}:

$Des^1_{ini}:\ Des_{a_{ini}}(\neg Suffer(a_{ini}))$

$Des^2_{ini}:\ Des_{a_{ini}}(NegotiateWith(a_{res}) \wedge mostHelpful(a_{res}, a_{ini}, Ag))$

$Des^3_{ini}:\ Des_{a_{ini}}(Obtain(a_{ini}, r, \rho_r) \wedge \neg lessThan(amountGivenUp(a_{res}, a_{ini}, r), \rho_r))$

$Des^4_{ini}:\ Des_{a_{ini}}(Maximize(amountGivenUp(a_{res}, a_{ini}, r)))$

$Des^5_{ini}:\ Des_{a_{ini}}(notEnough(a_{ini}, r) \rightarrow Persuade(a_{ini}, a_{res}, r))$

$Des^6_{ini}:\ Des_{a_{ini}}(enough(a_{ini}, r) \wedge Need(\{\neg a_{ini}\}, r) \rightarrow GiveUp(a_{ini}, r))$

$Des^7_{ini}:\ Des_{a_{ini}}(enough(a_{ini}, r) \rightarrow StopNegotiation(a_{ini}, a_{res}, r))$

The desires of an initiator in the many-to-1 mode are similar to the ones in the 1-to-1 mode except for the addition of Des^6_{ini}. The desire Des^6_{ini} means when the acquired resource is enough to satisfy its own need, the initiator a_{ini} desires to give up resource r to other initiators $\{\neg a_{ini}\}$ who are in need of r.

Based on its beliefs, the responder, a_{res}, which responds to multiple initiators' requests on resources (e.g., r) concurrently, has a set of desires:

$Des^1_{res}:\ Des_{a_{res}}(\neg Suffer(a_{res}))$

$Des^2_{res}:\ Des_{a_{res}}(\neg GiveUp(a_{res}, r))$

$Des^3_{res}:\ Des_{a_{res}}(GiveUp(a_{res}, r)\wedge$
$Rationalize(amountGivenUp(a_{res}, \{a_{ini}\}, r)))$

$Des^4_{res}:\ Des_{a_{res}}(GiveUp(a_{res}, r)\wedge$
$IdentifyNeed(a_{res}, \{a_{ini}\}, r))$

$Des^5_{res}:\ Des_{a_{res}}(GiveUp(a_{res}, r)\wedge$
$(moreJustifiableThan(\{a'_{ini}\}, \{a_{ini}\}$
$-\{a'_{ini}\}, r) \rightarrow HelpMore(\{a'_{ini}\}, \{a_{ini}\}$
$-\{a'_{ini}\}, r)))$

The desires of the responder in the many-to-1 mode are similar to the ones in the 1-to-1 mode except that (1) in Des^3_{res} the responder a_{res} desires to give up additional resource and rationally assign the resource among initiators to satisfy the initiators' needs, (2) Des^4_{res} is added which means the responder a_{res} desires to give up additional resource but it will identify the resource need of each initiator so as to decide which ones are more justifiable to receive help on resource r, and (3) Des^5_{res} is added which means the responder a_{res} desires to give up additional resource to be more helpful to some initiators ($\{a'_{ini}\}$), which it thinks are more justifiable to receive resource r, than others ($\{a_{ini}\} - \{a'_{ini}\}$). The desires Des^4_{res} and Des^5_{res} indicate that the responder is more responsible in its giving away resources in the many-to-1 mode than in the 1-to-1 mode, and it desires to choose among the initiators the most justifiable ones to help. These two desires are generated from both of its initial beliefs on resources: (1) $Bel1$ – it will suffer from the resource shortage problem unless it uses its resources wisely, and (2) $Bel2$ – it can give them up to others who are in need of the resources when it has additional resources it is not using. Based on these two beliefs, for the sake of the wise use of additional resources in helping others, the responder will be more self-interested to those initiators that it believes less justifiable to receive help, and more altruistic to those ones more justifiable to receive help.

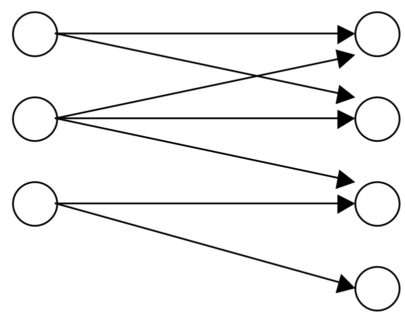

Fig. 5. The many-to-many negotiation mode.

Based on these desires, the initiators and the responder will have a series of goals to achieve respectively in the overall negotiation. Their goals are interdependent. Some of these goals are mutually exclusive (e.g., $\{Des^1_{ini}$, Des^2_{ini}, Des^3_{ini}, Des^4_{ini}, $Des^5_{ini}\}$ and $\{Des^1_{res}$, $Des^2_{res}\}$) while some are non-mutually exclusive (e.g., $\{Des^1_{ini}$, Des^2_{ini}, Des^3_{ini}, Des^5_{ini}, Des^6_{ini}, $Des^7_{ini}\}$ and $\{Des^1_{res}$, Des^3_{res}, Des^4_{res}, $Des^5_{res}\}$). To achieve these goals, the many-to-1 negotiation between the initiators and the responder integrates the characteristics of competitive negotiation and cooperative negotiation, which forms the hybrid negotiation.

3.2.4. The many-to-many mode

In the many-to-many negotiation mode, there are more than one initiator and more than one responder. One agent negotiates with multiple responders over same resources; at the same time, the responders are also negotiating with other agents over the same resources, which can be represented as in Fig. 5.

Obviously, the many-to-many negotiation mode is the mixture of the 1-to-many mode and the many-to-1 mode. We also can generalize our discussion on agents' goals in the 1-to-1 negotiation mode to the many-to-many negotiation mode.

Based on its assigned beliefs, each initiator, a_{ini}, which is in need of resources (e.g., r), has a set of desires in its negotiation processes:

$Des^1_{ini}:\ Des_{a_{ini}}(\neg Suffer(a_{ini}))$

$Des^2_{ini}:\ Des_{a_{ini}}(Maximize(numberOf$
$Negotiations(a_{ini}, r)))$

$Des^3_{ini}:\ Des_{a_{ini}}(NegotiateWith(\{a_{res}\})\wedge$

$mostHelpful(\{a_{res}\}, a_{ini}, Ag))$

$Des_{ini}^{4} : Des_{a_{ini}}(Obtain(a_{ini}, r, \rho_r) \wedge \neg$
$lessThan(amountGivenUp(\{a_{res}\}, a_{ini}, r), \rho_r))$

$Des_{ini}^{5} : Des_{a_{ini}}(Maximize(amountGiven$
$Up(\{a_{res}\}, a_{ini}, r)))$

$Des_{ini}^{6} : Des_{a_{ini}}(notEnough(a_{ini}, r) \rightarrow$
$Persuade(a_{ini}, \{a_{res}\}, r))$

$Des_{ini}^{7} : Des_{a_{ini}}(enough(a_{ini}, r) \wedge$
$Need(\{\neg a_{ini}\}, r) \rightarrow GiveUp(a_{ini}, r))$

$Des_{ini}^{8} : Des_{a_{ini}}(enough(a_{ini}, r) \rightarrow$
$StopNegotiation(a_{ini}, \{a_{res}\}, r))$

The desires of the initiator in the many-to-many mode are similar to the ones in the 1-to-many mode except for the addition of Des_{ini}^{7}. The desire Des_{ini}^{7} indicates that the initiator is also involved in the many-to-1 negotiation mode.

Based on its beliefs, each responder, a_{res}, which is in need of resources (e.g., r), has a set of desires in its negotiation processes with the set of initiators ($\{a_{ini}\}$):

$Des_{res}^{1} : Des_{a_{res}}(\neg Suffer(a_{res}))$

$Des_{res}^{2} : Des_{a_{res}}(\neg GiveUp(a_{res}, r))$

$Des_{res}^{3} : Des_{a_{res}}(GiveUp(a_{res}, r) \wedge$
$Rationalize(amountGivenUp(a_{res}, \{a_{ini}\}, r)))$

$Des_{res}^{4} : Des_{a_{res}}(GiveUp(a_{res}, r) \wedge$
$IdentifyNeed(a_{res}, \{a_{ini}\}, r))$

$Des_{res}^{5} : Des_{a_{res}}(GiveUp(a_{res}, r) \wedge$
$(moreJustifiableThan(\{a'_{ini}\}, \{a_{ini}\}$
$-\{a'_{ini}\}, r) \rightarrow HelpMore(\{a'_{ini}\}, \{a_{ini}\}$
$-\{a'_{ini}\}, r)))$

The desires of a responder in the many-to-many mode are just same as the ones in the many-to-1 mode. Although the responder is also involved into the 1-to-many negotiation mode, it is not aware of other responders that are negotiating with their common initiator.

Based on these desires, the initiators and the responders will have a series of goals to achieve respectively in the overall negotiation. Their goals are interdependent. Some of these goals are mutually exclusive (e.g., $\{Des_{ini}^{1}$, Des_{ini}^{2}, Des_{ini}^{3}, Des_{ini}^{4}, Des_{ini}^{5}, $Des_{ini}^{6}\}$ and $\{Des_{res}^{1}$, $Des_{res}^{2}\}$) while some are non-mutually exclusive (e.g., $\{Des_{ini}^{1}$, Des_{ini}^{2}, Des_{ini}^{3}, Des_{ini}^{4}, Des_{ini}^{6}, Des_{ini}^{7}, $Des_{ini}^{8}\}$ and $\{Des_{res}^{1}$, Des_{res}^{3}, Des_{res}^{4}, $Des_{res}^{5}\}$). To achieve these goals, the many-to-many negotiation between the initiators and the responders integrates the characteristics of competitive negotiation and cooperative negotiation, which forms the hybrid negotiation.

Summarizing the above BDI-based inference on different negotiation modes: 1-to-1, 1-to-many, many-to-1, and many-to-many, it is obvious that the negotiations in our research domain are hybrid negotiations. The hybrid negotiation occurs between agents that are both self-interested and altruistic, and it integrates the characteristics of competitive negotiation and cooperative negotiation.

4. Implementation of hybrid negotiation for resource coordination

In our multiagent system framework, the agents coordinate resource allocation among themselves via hybrid negotiations. Motivated by their desires, the agents compete and cooperate with each other in negotiation to achieve their goals. We also implement hybrid negotiation in the many-to-many mode. The many-to-many negotiation mode is able to represent the most complex negotiation case in resource coordination: multiple initiators negotiate with some common responders over same resources at the same time. Further, it is the most general one among all the four negotiation modes and its implementation mechanism can be specialized to other modes.

We implement hybrid negotiation in our multiagent system framework with a specific negotiation protocol and a specific negotiation strategy. To help initiators to be better at being self-interested, and help responders to be better at being self-interested and altruistic in negotiation, we also apply profiling and learning techniques in the implementation. In our implementation, since we apply the hybrid negotiation mechanism in resource coordination in complex environments, we also take into account the environmental factors addressed in Section 2.3: dynamic, uncertain, real-time, and noisy. Although we implement hybrid negotiation in a specific way, the negotiation protocol and negotiation strategy can be defined in other forms. As long as the agents are built following the presented model in Section 3, hybrid negotiation can be achieved.

4.1. Negotiation protocol

A *negotiation protocol* is a set of rules by which agents will come to agreements or disagreements [44]. It specifies the kinds of deals they can make, as well as the sequence of offers and counter-offers that are supported. We implement the hybrid negotiation mechanism with the argumentation-based negotiation protocol [32,36]. In argumentation-based negotiation protocols, agents exchange proposals and counter-proposals backed by *arguments* that summarize the reasons why the proposal should be accepted. Agents generate and exchange arguments to support their negotiation stance. The argumentation is persuasive because the exchanges may alter the stance of the agents involved.

We describe the argumentation-based negotiation protocol with finite state machines (FSMs) as shown in Figs 6 and 7, respectively corresponding to an initiator and a responder. In each negotiation process, the initiator and the responder interact and negotiate over resource with this protocol (rules). The specification of rules is based on the work of [36]. Either FSM is composed of a set of states, and pairs of input and output. The latter forms a series of state transitions. An *input* is a received message including a negotiation primitive, or a current decision result, or both, or nothing. An *output* is a message being sent, or an action being activated by the input information, or both, or nothing.

In our multiagent system framework, we assume that there is no centralized information shared by agents. The agents exchange information directly during negotiation. Only what is considered relevant and useful information is communicated to reduce the communication cost. The negotiation protocol provides the following primitives that indicate the type and purpose of each message:

request: the initiator asks the responder for a certain amount of resource.

moreInfo: the responder requests more information from the initiator.

info: the initiator provides more information to support its resource request.

infoNull: the initiator cannot provide information to support its resource request any more.

accept: the initiator accepts the amount of resource from the responder, or the responder agrees the initiator with the requested amount.

decline: the initiator decides not to accept resource provided by the responder since it has obtained enough resource (from negotiations with other agents).

counter: the responder proposes a less resource amount to the initiator than the requested.

reject: the responder rejects the resource request of the initiator and stops negotiation. *abort*: the initiator or the responder stops negotiation because the time is out of bound.

Of all the primitives, *request*, *info*, *infoNull*, *accept*, *decline*, and *abort* are used by an initiator as shown in Fig. 6, and *moreInfo*, *accept*, *counter*, *reject*, and *abort* are used by a responder as shown in Fig. 7.

The competitive and cooperative characteristics of hybrid negotiation can be shown in the negotiation protocol, as the agents are both self-interested and altruistic. As an initiator, an agent competes for the resource through sending messages with support information (arguments) to the responder (see the input/output pair *moreInfo/SndInfoMsg* in Fig. 6). Once the resource need of the initiator is satisfied, the ongoing negotiation is aborted and the agent shows cooperative with other agents (see the input/output pairs *RcvAcceptMsg&enoughResourceObtained/SndDeclineMsg* and *enoughResourceObtained/SndDeclineMsg* in Fig. 6). As a responder, on one hand, the agent is competitive in that it rejects the initiator's request when it is using the resource (see the input/output pair *Reject/SndRejectMsg* in Fig. 7), and that it needs more support information when that is possible to give up its additional resource (see the input/output pair *possible/SndMoreInfoMsg* in Fig. 7); on the other hand, the agent is cooperative in that it tries to give the initiator an opportunity to get the resource (see the input/output pairs *possible/SndMoreInfoMsg* and *RcvInfoNullMsg&possible/SndCounterMsg* in Fig. 7).

4.2. Negotiation strategy

A negotiation protocol specifies the rules of the interaction between agents, while a *negotiation strategy* is a specification of what to do in every alternative during negotiation. Given a set of interaction alternatives, what actions should the initiator and the responder take or which decision to make while participating in the negotiation process? In this section, we describe the negotiation strategies of agents as initiators and responders separately. We also apply case-based reasoning into agents' negotiation strategy making. The application of case-based reasoning is aimed to provide agents with basic strategies and avoid constructing a strategy from scratch.

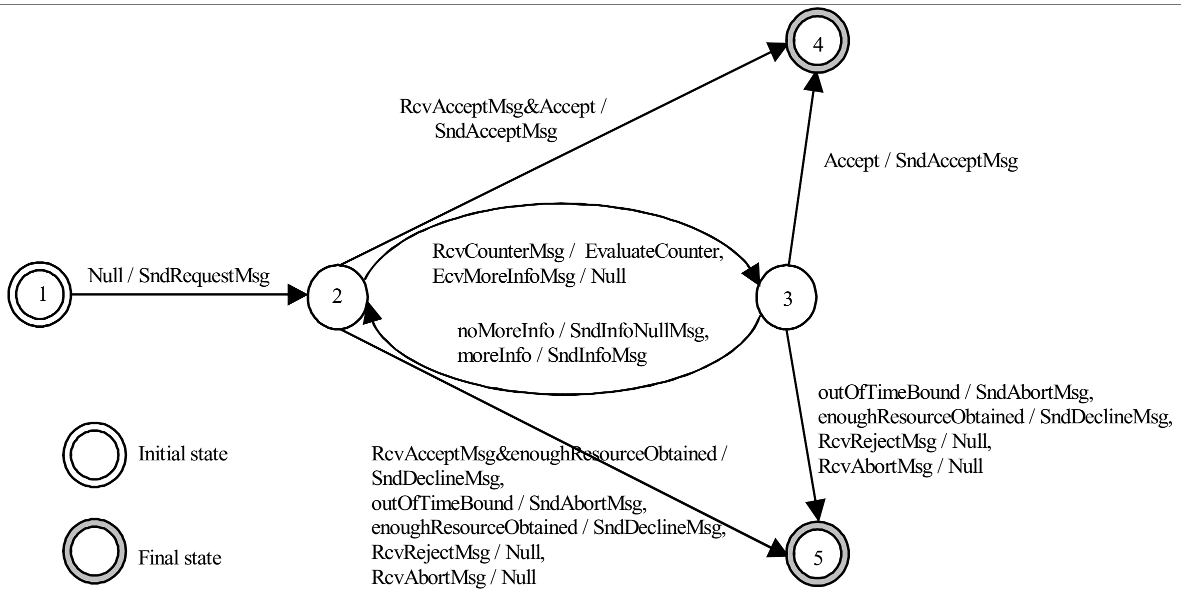

Fig. 6. An initiator's FSM in negotiation protocol.

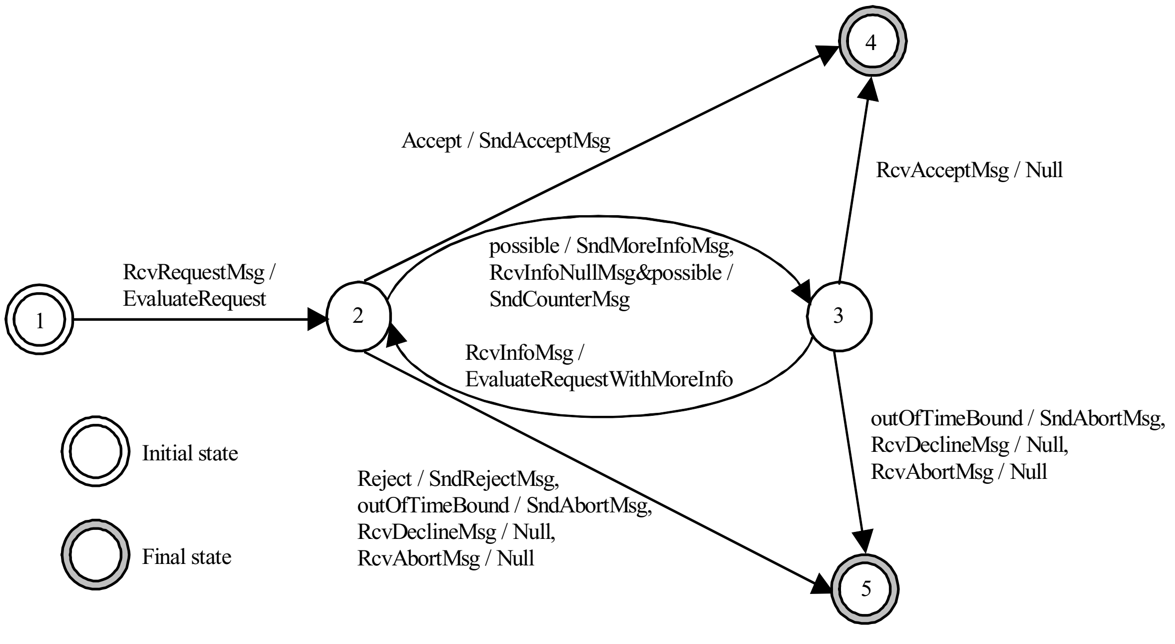

Fig. 7. The responder's FSM in negotiation protocol.

4.2.1. Initiators' negotiation strategy

For an initiator, it needs to make decisions in a series of situations: (1) with whom to negotiate and how to assign the requested resource amount to the responder(s), (2) how much resource to request, (3) how to provide support information, (4) what to do when its resource need changes during negotiation, (5) what to do when its resource need has been satisfied, (6) what to do when the negotiation time runs beyond the allowed time, and (7) what to do when there is exception that interrupts an ongoing negotiation. The above decision-making process forms the initiator's negotiation strategy. The initiator is both self-interested and altruistic. With its varieties of desires, the initiator's negotiation strategy will show the competitive and cooperative characteristics of hybrid negotiation. Assume that we have the following notations:

a_{ini} :	the initiator,
r :	the resource requested,
$\{a_{res}\}$:	the set of responders with whom a_{ini} negotiates,
$\{nego_{a_{ini}}^{a_{res}}\}$:	the set of negotiation processes between a_{ini} and $\{a_{res}\}$,
ρ_r :	the resource amount a_{ini} minimally needs,
$\rho_r^{\{nego_{a_{ini}}^{a_{res}}\}}$:	the total resource amount requested by a_{ini} from $\{a_{res}\}$,
$\rho_r^{nego_{a_{ini}}^{a_{res}}}$:	the resource amount requested by a_{ini} from a_{res},
$\hat{T}_{\{nego_{a_{ini}}^{a_{res}}\}}$:	the allowed time of the negotiations,
$T_{nego_{a_{ini}}^{a_{res}}}^{a_{ini}}$:	the time a_{ini} has spent on the negotiation with a_{res} so far,
$\hat{\tau}$:	the maximal round-trip time of message passing between agents, and
$\tau_{nego_{a_{ini}}^{a_{res}}}^{a_{ini}}$:	the time of a_{ini} waiting for the next message sent from a_{res} during negotiation.

Motivated by its desires in the many-to-many negotiation mode (see Section 3.2.4), the negotiation strategy of an initiator a_{ini} on resource r is shown in Fig. 8. First, at the decision points 1 and 3 where the initiator needs to decide with whom to negotiate and how to assign the requested resource amount, it selects a number of the most helpful negotiation partners based on its beliefs, and assigns the requested resource amount proportionally to each partner. The initiator makes these decisions motivated by its desires: Des_{ini}^1, Des_{ini}^2, Des_{ini}^3, and Des_{ini}^5. It desires not to suffer from the resource shortage problem. It also desires to maximize the number of helpful responders and maximize the resource amount obtained. On the other hand, in an uncertain environment, the negotiation outcome may be different from what is expected by the initiator. These decisions can maximize the chance of the initiator getting resource from others, and minimize the impact of uncertain behavior of other agents. The principle of selection and assignment is based on the result of neighbor profiling (see Section 4.3). Second, at the decision point 2 where the initiator needs to decide how much resource to request, motivated by its desires Des_{ini}^1 and Des_{ini}^4, the initiator requests more resource than the needed from others. For simplicity, we assume that an initiator always requests the resource amount twice the minimal need that it has estimated. Third, at the decision points 4 and 5 where the initiator needs to decide how to provide support information, motivated by its desires: Des_{ini}^1, Des_{ini}^4, Des_{ini}^5 and Des_{ini}^6 (especially Des_{ini}^6), the initiator tries to convince each responder by supplying arguments to support its request. Specifically, the initiator has to decide which arguments (information pieces) are more important to be sent to each responder first. Through that it can minimize the number and length of messages sent. With fewer messages the agents can avoid suffering from message loss in the noisy environment, and the message transfer can be faster. Fourth, in a dynamic environment, the resource needs of agents may change as the time progresses. At the decision points 6 and 7 where the initiator's resource need changes during negotiation, it proportionally increases or decreases the requested resource amount. The former is motivated by its desires Des_{ini}^1, Des_{ini}^4, and Des_{ini}^5 while the latter is by Des_{ini}^1 and Des_{ini}^4. Note that when the initiator's resource need decreases, it requests less resource amount than its initial request. Through that it can obtain resource easier. Fifth, at the decision point 8 where the initiator's resource need has been satisfied, motivated by its desires Des_{ini}^1, Des_{ini}^7, and Des_{ini}^8, it declines the resource given up by the responder and stops all remaining negotiations. Sixth, at the decision point 9 where the negotiation time runs beyond the allowed time, motivated by its desire Des_{ini}^5, the initiator accepts the resource given up by the responders. And motivated by its desire Des_{ini}^1, it stops all the remaining negotiations to avoid suffering from missing other tasks. This decision addresses the real-time requirement of the negotiation environment. Finally, in the noisy environment, the communication channels between agents are not reliable and may be jammed. At the decision point 10 where the initiator perceives that its waiting time for a response message exceeds the limit, it stops the current negotiation to avoid suffering from endless waiting.

The competitive and cooperative characteristics of hybrid negotiation can be seen in the initiators' negotiation strategy. An initiator is cooperative at the decision point 8 while competitive at other decision points. It is because our agents are both self-interested and altruistic.

4.2.2. Responders' negotiation strategy

For a responder, it needs to make decisions in a series of situations: (1) whether to give up resource to an initiator, (2) how much resource it can give up to an initiator, (3) which initiators should be committed first when there is more than one initiator requesting the same resource at the same time, (4) what to do when the negotiation time runs beyond the allowed time, and (5) what to do when there is exception that interrupts an ongoing negotiation. Similar to the initiators' negotiation strategy, with its varieties of desires, the responder's negotiation strategy also shows the competitive and cooperative characteristics of hybrid negotiation. Assume that we have the following notations:

a_{res} :	the responder,
r :	the resource requested,
$\{a_{ini}\}$:	the set of initiators with whom a_{res} negotiates at the same time,
$\rho_r^{nego_{a_{ini}}^{a_{res}}}$:	the resource amount requested by a_{ini} from a_{res},
$\rho_r^{possiblyGivenUp(a_{res})}$:	the resource amount possibly given up by a_{res} that is not being used,
$\rho_r^{counteredGivenUp(a_{res})}$:	the countered resource amount proposed by a_{res},
$util_{arguments_{a_{ini}}}$:	the utility of all the support information provided by a_{ini} that is evaluated by a_{res},
δ_r :	the persuasion threshold a_{res} holds about the resource r,
$\hat{T}^{a_{res}}_{nego^{a_{res}}_{a_{ini}}}$:	the allowed time of a_{res} negotiating with a_{ini},
$T^{a_{res}}_{nego^{a_{res}}_{a_{ini}}}$:	the time a_{res} has spent on the negotiation with a_{ini} so far,
$\hat{\tau}$:	the maximal round-trip time of message passing between agents, and
$\tau^{a_{res}}_{nego^{a_{res}}_{a_{ini}}}$:	the time of a_{res} waiting for the next message sent from a_{ini} during negotiation.

Motivated by its desires in the many-to-many negotiation mode (see Section 3.2.4), the negotiation strategy of a responder a_{res} on resource r is shown in Fig. 9. First, at the decision point 1 where the responder does not have additional unused resource, it just rejects any initiator's resource request. The responder makes this decision motivated by its desires Des^1_{res} and Des^2_{res} in that it desires to wisely use its resources. Second, when it has additional resources, the responder is altruistic and willing to help others. But even that, it still desires to wisely use its resources. Motivated by its desires Des^3_{res}, Des^4_{res}, and Des^5_{res}, the responder tries to identify the resource need of each initiator so as to decide which ones are more necessary to receive help. It evaluates the utility of each initiator's support information based on specific rules, and then compares the utility value $util_{arguments_{a_{ini}}}$ with a dynamic persuasion threshold δ_r it holds. A *persuasion threshold* is a value associated with a resource that indicates the degree to which an agent needs to be convinced to offer an amount of resource. At the decision point 2 where the utility value is lower than the threshold value, the responder is not convinced and it requests more support information from the initiator. At the decision point 3 where the initiator cannot provide more information, the responder counter-offers with less resource than the requested. This countered resource amount, $\rho_r^{counteredGivenUp(a_{res})}$, is the weighted sum of $\rho_r^{nego_{a_{ini}}^{a_{res}}}$, $\rho_r^{possiblyGivenUp(a_{res})}$, and $util_{arguments_{a_{ini}}}$. Third, if the utility value $util_{arguments_{a_{ini}}}$ is not lower than the threshold value δ_r, the responder is convinced. At the decision point 4 where the requested resource amount is not greater than the possibly given up, the responder accepts the request and gives up the resource of the requested amount $\rho_r^{nego_{a_{ini}}^{a_{res}}}$. Similarly, at the decision point 5 where the requested amount is greater than the possibly given up, it proposes a counter-offer with the resource amount $\rho_r^{possiblyGivenUp(a_{res})}$. The responder makes these decisions motivated by its desires Des^3_{res} and Des^5_{res}. These desires motivate the responder to give up resource first to the initiators that can convince it first. When the responder negotiates with more than one initiator over the same resource and at the same time, it desires to give more help to the initiators that it thinks more justifiable to receive help. The initiators that can convince it first are assumed as those ones. Then the responder continues negotiating with other initiators over the remained resource amount. Fourth, at the decision point 6 where the negotiation time runs beyond the allowed time, motivated by its desire Des^1_{res}, the responder immediately stops the negotiation to avoid suffering from missing other tasks. Finally, similar to the initiators' negotiation strategy, at the decision point 7, the responder stops the negotiation to avoid suffering from endless waiting.

Similar to the initiators' negotiation strategy, the competitive and cooperative characteristics of hybrid negotiation can also be seen in the responders' negotiation strategy. A responder is competitive at the decision points 1, 6, and 7 while cooperative at other decision points in negotiation, as our agents are both self-interested and altruistic.

4.2.3. *Deriving negotiation strategy via case-based reasoning*

In the negotiation strategies of initiators and responders, there are a series of issues that need to be determined dynamically. For an initiator, it needs to dynamically determine the number of agents that can be requested in a negotiation, the allowed negotiation time, the way of providing support information, etc. For a responder, it needs to dynamically determine the resource amount that can be given up, the allowed negotiation time, the number of negotiation steps allowed, the persuasion threshold, how to compute the utility of arguments provided by the initiator, etc. We use case-based reasoning (CBR) to help an agent derive a negotiation strategy that determines these issues.

Case-based reasoning is to solve a new problem by remembering a previous similar situation and by reusing information and knowledge of that situation [15]. It utilizes the specific knowledge of previously experienced, concrete problem situations (cases). A new problem can be solved by finding a similar past case and reusing it in the new problem situation, which can avoid the solution construction from scratch. In such a way, it is possible for an agent to derive the current negotiation strategy based on past ones, which is helpful to meet the real-time requirement in negotiations.

We equip each agent with a set of cases to provide a set of possible negotiation strategies. All of the strategies are valid in the current situation, but only one of them is best to produce a "good enough, soon enough" solution. We embed the negotiation strategies in cases so that agents can derive a strategy for the current situation in a real-time manner, rather than constructing a strategy from scratch every time. Specifically, we use case-based reasoning to get the values of the previously addressed issues. Those values are set in cases according to the past negotiation experience.

Each agent maintains two casebases?an initiating casebase for the cases when the agent was the initiator of negotiations, and a responding casebase when the agent was the responder of negotiations. The different casebase setting is necessary since an agent's negotiation strategies are different as an initiator and as a responder. So the initiating cases in the initiating casebase have different compositions (attributes) from the responding cases in the responding casebase. But any case consists of three parts: *input*, *output*, and *outcome*. Respectively, these three parts describe the situation information relevant to negotiation, the negotiation strategy, and the negotiation outcome.

The input part of an initiating case has three attributes: the type of resource being requested, the description of the agent's observation of the world, and the agent's status. Its output part has four attributes: the way of providing support information, the allowed negotiation time, the number of agents that can be requested, and the number of negotiation steps allowed. Its outcome part has only one attribute: the negotiation outcome. One example of the initiating case is as follows.

(*initiatingCase* (*_request CPU_RESOURCE*) (*_target Speed 1.0*) (*_numberOfTasks 7*) (*_negotiating false*) (*_infoClass* (*info0 3*) (*info1 2*) (*info2 1*)) (*_timeAllowed 7000*) (*_numberOfPartners 3*) (*_numberOfSteps Allowed 6*) (*_outcome N_SUCCESS*))

As shown in this example, (1) *_request* corresponds to the type of resource being requested by the initiator in the past negotiation; (2) *_targetSpeed* is domain-specific and it corresponds to the agent's observation of the world; (3) *_numberOfTasks* and *_negotiating* describe the agent's status; (4) *_infoClass* describes the priority order in which the initiator provided different classes of support information to the responder, where *info0*, *info1*, and *info2* respectively represent the different priorities; (5) *_timeAllowed* corresponds to the allowed negotiation time; (6) *_numberOfPartners* corresponds to the number of agents that were requested; (7) *_numberOfStepsAllowed* corresponds to the number of negotiation steps allowed, which indicates the total number of support information messages the initiator could send within the time limit; and (8) *_outcome* records the outcome of the past negotiation (in this example, the negotiation was successful). If a negotiation fails, there may be three reasons: (i) the responder rejected the resource request (*N_REJECTED*); (ii) the negotiation time ran beyond the allowed time (*N_OUT_OF_TIME*); and (iii) the communication channel between the initiator and the responder was jammed during negotiation (*N_CHANNEL_JAMMED*).

The input part of a responding case has three attributes: the type of resource requested, the description of the agent's observation of the world, and the agent's status. Its output part has four attributes: the allowed negotiation time, the number of negotiation steps allowed, the way of calculating the utility of arguments

Decision Points	Decisions	Underlying Desires
1.	Selects a number of the most helpful agents $\{a_{res}\}$ to negotiate with.	Des^1_{ini}, Des^2_{ini}, Des^3_{ini}, Des^5_{ini}
2.	Proposes the requests with the total resource amount $\rho_r^{\{nego^{a_{res}}_{a_{ini}}\}}$ more than the minimal need ρ_r.	Des^1_{ini}, Des^4_{ini}
3.	Proportionately assigns $\rho_r^{\{nego^{a_{res}}_{a_{ini}}\}}$ to $\{nego^{a_{res}}_{a_{ini}}\}$.	Des^1_{ini}, Des^2_{ini}, Des^3_{ini}, Des^5_{ini}
	for each negotiation process $nego^{a_{res}}_{a_{ini}}$, **do**	
4.	Tries to convince a_{res} to help by supplying arguments to support its request of $\rho_r^{nego^{a_{res}}_{a_{ini}}}$.	Des^1_{ini}, Des^4_{ini}, Des^5_{ini}, Des^6_{ini}

5.	**if** a_{res} requests more information from it, **then**
	go to 4.
	endif
		Des^1_{ini}, Des^4_{ini}, Des^5_{ini}, Des^6_{ini}

6.	**if** the resource amount currently needed increases, **then**
	Increases $\rho_r^{nego^{a_{res}}_{a_{ini}}}$.
	go to 4.
	endif
		Des^1_{ini}, Des^4_{ini}, Des^5_{ini}

7.	**if** the resource amount currently needed decreases, **then**
	Decreases $\rho_r^{nego^{a_{res}}_{a_{ini}}}$.
	go to 4.
	endif
		Des^1_{ini}, Des^4_{ini}

8.	**if** the resource need has been satisfied, **then**
	Declines the resource given up by a_{res}.
	Stops all the remaining negotiations.
	endif
		Des^1_{ini}, Des^7_{ini}, Des^8_{ini}

9.	**if** $T^{a_{ini}}_{nego^{a_{res}}_{a_{ini}}} \geq \hat{T}_{\{nego^{a_{res}}_{a_{ini}}\}}$, **then**
	Accepts the resource given up by a_{res}.
	Stops all the remaining negotiations.
	endif
		Des^1_{ini}, Des^5_{ini}

10.	**if** $\tau^{a_{ini}}_{nego^{a_{res}}_{a_{ini}}} \geq \hat{\tau}$, **then**
	Stops $nego^{a_{res}}_{a_{ini}}$ due to the communication channel jammed.
	endif
		Des^1_{ini}

	endfor	

Fig. 8. The initiator's negotiation strategy where the numbers 1∼ 10 represent the series of decision points.

provided by the initiator, and the persuasion threshold. Its outcome part has only one attribute: the negotiation outcome. One example of the responding case is as follows.

(*respondingCase* (*_request CPU_RESOURCE*) (*_dataQ low*) (*_power 0.00*) (*_numberOfTasks 1*) (*_negotiating false*) (*_resourceUsage 0.1*) (*_maxResourceGiveUp 0.1*) (*_timeAllowed 6000*) (*_numberOfStepsAllowed 8*) (*_function LINEAR*) (*_kappa 2.0*) (*_beta 4.0*) (*_persuasionThreshold 1.0*) (*_outcome N_SUCCESS*))

In this example, (1) *_request* corresponds to the type of resource being requested by the initiator; (2) *_dataQ and _power* are domain-specific and they consist of the agent's observation of the world; (3) *_numberOfTask*, *_negotiating*, *_resourceUsage*,

Decision Points	Decisions	Underlying Desires
	for each negotiation process with the initiator a_{ini}, **do**	
1.	**if** $\rho_r^{possiblyGivenUp(a_{res})} \leq 0$, **then** Rejects the resource request of a_{ini}. Stops the negotiation with a_{ini}.	Des_{res}^1, Des_{res}^2
2.	**else if** $util_{arguments_{a_{ini}}} < \delta_r$, **then** Requests more support information from a_{ini}.	Des_{res}^3, Des_{res}^4, Des_{res}^5

3.	**if** a_{ini} cannot provide support information any more, **then**
	Proposes a counter-offer of $\rho_r^{counteredGivenUp(a_{res})}$.
	endif
	go to 1.
		Des_{res}^3

4.	**else if** $\rho_r^{nego_{a_{ini}}^{a_{res}}} \leq \rho_r^{possiblyGivenUp(a_{res})}$, **then** Accepts the request and gives up the resource of the requested amount $\rho_r^{nego_{a_{ini}}^{a_{res}}}$. Stops the negotiation with a_{ini}.	Des_{res}^3, Des_{res}^5
5.	**else if** $\rho_r^{nego_{a_{ini}}^{a_{res}}} > \rho_r^{possiblyGivenUp(a_{res})}$, **then** Proposes a counter-offer of $\rho_r^{possiblyGivenUp(a_{res})}$. **go** to 1. **endif**	Des_{res}^3, Des_{res}^5

6.	**if** $T_{nego_{a_{ini}}^{a_{res}}}^{a_{res}} \geq \hat{T}_{nego_{a_{ini}}^{a_{res}}}^{a_{res}}$, **then**
	Stops the negotiation with a_{ini}.
	endif
		Des_{res}^1

7.	**if** $\tau_{nego_{a_{ini}}^{a_{res}}}^{a_{res}} \geq \hat{\tau}$, **then**
	Stops the negotiation with a_{ini} due to the communication channel jammed.
	endif
	endfor
		Des_{res}^1

Fig. 9. The responder's negotiation strategy where the numbers 1$\sim$ 7 represent the series of decision points.

and *_maxResourceGiveUp* describe the agent's status, where *_resourceUsage* represents the resource amount being used and *_maxResourceGiveUp* represents the resource amount that could be given up; (4)*_timeAllowed* corresponds to the allowed negotiation time; (5) *_numberOfStepsAllowed* corresponds to the number of negotiation steps allowed, which indicates the total number of support information messages the responder would wait for the initiator to provide within the time limit; (6) *_function*, *_kappa*, and *_beta* specify the way of calculating the utility of arguments provided by the initiator, where *_function* indicates the computation complexity of the utility function, *_kappa*, *_beta* correspond to the parameters of the function; (7) *_persuasionThreshold* specifies the value of the persuasion threshold held by the responder on the resource; and (8) *_outcome* records the outcome of the past negotiation.

We use case-based reasoning to retrieve and adapt a negotiation strategy. To retrieve one from all the strategies available, the agent compares the similarity of each case with the current situation. If there are multiple cases that are equally similar to the current situation, the negotiation outcomes in these cases will be further compared to select out one with the best outcome. The negotiation strategy in the best case might not be appropriate in the current situation, so the agent needs to do corresponding adaptation to the strategy. The application of case-based reasoning is based on the work of [35,36]. In case-based reasoning, the available cases in each agent's casebases do not vary. In Section 4.3, we will describe how to improve cases in casebases.

4.3. *Profiling and learning*

In the hybrid negotiation, the initiator and the responder are motivated by their individual beliefs and desires. The initiator is motivated to obtain resource through negotiation while the responder is motivated to help others with resource it is not using. To help initiators to be better at being self-interested – to obtain resource, and help responders to be better at being self-interested and altruistic – to wisely give up resource, we apply profiling and learning techniques in the implementation of hybrid negotiation.

Each agent profiles its neighbors through recording the various negotiation relationships between them. The agent records the various negotiation relationships with a neighbor to identify whether and how well they have helped each other in the past. The neighbor profiling plays two roles in negotiation. First, when an agent as an initiator selects neighbors to request resources, it can check its profiles of all neighbors to find the ones with which it has had good relationships. And then it will select those neighbors as negotiation partners and assign requested resource amount proportionately to the relationship with each of them. Second, when an agent as a responder determines whether to give up resource to an initiator, it can check its profile of the initiator to find whether they have helped each other very well. And then it prefers to give up resource to the initiator with which it has had good relationship. The first role factors in the initiator's negotiation strategy when the initiator selects negotiation partners. The second role factors in the responder's negotiation strategy when the responder evaluates the utility of the initiator's support information. The profiling plays these roles based on the following beliefs assigned to agents:

$Bel5: \ Bel_{a_i}(high(\beta_{HelpWell(a_i,a_k,r)}) \rightarrow high(\gamma_{UseWisely(a_k,r)})),$

$Bel6: \ Bel_{a_i}(high(\beta_{HelpWell(a_i,a_k,r)}) \rightarrow helpful(a_k,r)),$ and

$Bel7: \ Bel_{a_i}(GiveUp(a_i,a_k,r) \wedge high(\gamma_{UseWisely(a_k,r)}) \rightarrow UseWisely(a_i,r)).$

The beliefs $Bel5$, $Bel6$, and $Bel7$ of agents are about what to do to identify helpful neighbors and what to do to wisely give up resources in negotiations. They are the continuation of the four beliefs – $Bel1$, $Bel2$, $Bel3$, and $Bel4$ – described in Section 3.2. Here, β and γ respectively represent the help degree and the wisdom degree of the neighbor a_k over resource r, in the belief of the agent a_i. The ***help degree*** of a neighbor refers to the overall negotiation relationship between the agent and the neighbor. It indicates how the neighbor is helpful to the agent and how the agent is helpful to the neighbor. So it represents the mutual help relationship between them. The agent profiles it from its negotiation experience. The *wisdom degree* of a neighbor indicates how wisely the neighbor uses its resources. An agent measures a neighbor's wisdom degree based on its help degree. The reason is as follows. In the agent's belief, both of them have to wisely use their resources to avoid suffering from resource shortage problems. Each time when the agent asked resource help from the neighbor, it did need help since it is rational. Then if the neighbor could realize this point and helped it very well, that indicates the neighbor could wisely use its resources. Further, if the agent helped the neighbor very well as requested, then that indicates the neighbor was wisely using its resource and could convince the agent very well. So the agent can derive the wisdom degree of a neighbor from its help degree.

Respectively, the above three beliefs mean each agent believes that: (1) a neighbor of high help degree is using its resource wisely; (2) a neighbor of high help degree is helpful to the agent's resource request; and (3) if it gives up resource to a neighbor of high wisdom degree, then it is wisely using its resource. The agent's belief $Bel5$ is generated from $Bel1$, $Bel2$, $Bel3$, and $Bel4$, and it is the basis of $Bel6$ and $Bel7$. The belief $Bel6$ helps an initiator select neighbors to request resources. The initiator prefers to select the neighbors of higher help degrees to obtain resource from them. The belief $Bel7$ helps a responder wisely give up its additional resource. The responder prefers to give up its resource to the neighbors of higher wisdom degrees.

Based on its negotiation experience, the agent a_i profiles its neighbor a_k, specifically the help degree (also the wisdom degree) of a_k, from their following negotiation relationships:??

$$\text{the helpfulness of } a_k \text{ to } a_i: \frac{\sum_{negotiate}^{success}(a_i \to a_k)}{\sum_{negotiate}(a_i \to a_k)},$$

$$\text{the importance of } a_k \text{ to } a_i: \frac{\sum_{negotiate}(a_i \to a_k)}{\sum_{negotiate}(a_i \to N_{a_i})},$$

$$\text{the reliance of } a_i \text{ on } a_k: \frac{\sum_{negotiate}^{success}(a_i \to a_k)}{\sum_{negotiate}^{success}(a_i \to N_{a_i})},$$

and

$$\text{the helpfulness of } a_i \text{ to } a_k: \frac{\sum_{negotiate}^{success}(a_k \to a_i)}{\sum_{negotiate}(a_k \to a_i)}.$$

Specifically, $\sum_{negotiate}(a_i \to a_k)$ indicates the total number of negotiations initiated from a_i to a_k, $\sum_{negotiate}^{success}(a_i \to a_k)$ indicates the total number of successful negotiations initiated from a_i to a_k, $\sum_{negotiate}(a_i \to N_{a_i})$ indicates the total number of negotiations initiated from a_i to its any neighbor, and $\sum_{negotiate}^{success}(a_i \to N_{a_i})$ indicates the total number of successful negotiations initiated from a_i to its any neighbor. Among all the four relationships, the first three relationships record whether the neighbor a_k has helped a_i very well in the past while the last one records whether a_i has helped a_k very well in the past. The help degree of the neighbor a_k can be profiled by a_i from these negotiation relationships. It is represented as the weighted sum of all of them.

In Section 4.2.3, we described how agents make negotiation strategy via case-based reasoning. To help agents to be better at negotiation with more and better strategies, we apply the case-based learning technique [15] into the implementation of hybrid negotiation. The case-based learning is used as the supplement of the case-based reasoning (see Section 4.2.3). Based on case-based reasoning, an agent can get an adapted negotiation strategy to conduct its negotiation. If the negotiation outcome is good and the new case is significantly different from the existing cases in the casebase, the agent will store the new case into its initiating casebase or responding casebase.

With the addition of new cases, the size of each casebase increases, which will lead to less efficient case retrieval since there are more cases to be evaluated. To keep a reasonable size of each casebase, an agent periodically updates its casebases. The agent records the time when each case is used and once the casebase is out of size, the case that has not been used for a long time will be removed from the casebase. This case removal strategy is helpful to keep the available cases (and negotiation strategies in cases) up to date and adaptive to the dynamic environment. The application of case-based learning is based on the work in [25].

In case-based reasoning, from its initiating casebase, an agent retrieves negotiation strategies about how to obtain resources, which helps it to be self-interested. Similarly, from its responding casebase, the agent retrieves negotiation strategies about whether and how to give up additional resources to others, which helps it to be both self-interested and altruistic. Through case-based learning, the agent improves its casebases and thus improves its negotiation strategies. That helps it to be better at being self-interested as an initiator, and to be better at being self-interested and altruistic as a responder.

5. Application domain

Our hybrid negotiation mechanism is objected to be applicable in generic coordination problems. The coordination approach based on it is aimed for varieties of applications. In such applications agents have to coordinate resources among themselves. Their coordination is both competitive and cooperative as the agents are both self-interested and altruistic. To validate the approach, specifically, we apply it in adaptive CPU resource allocation. The environmental events are multi-sensor target-tracking tasks that cause real-time CPU shortages in resource-constrained agents.

We focus on the application of CPU resource allocation for three reasons. First, the CPU resource is necessary in multiagent systems to support agents' behaviors that vary widely depending upon the availability of CPU. So, how to reallocate CPU resource to agents adaptively to their needs at run-time is a non-trivial problem, especially in the dynamic and real-time environment. Second, in a multiagent system where there is no hierarchical organization among the agents or centralized management, each agent is of high autonomy and the agents have to dynamically coordinate CPU allocation by themselves. The coordination approach is crucial for agents to reallocate CPU resource among themselves. Third, agents can use the CPU resource in both competitive and cooperative manner. The agents should compete for CPU since each of them needs it to perform tasks and its total amount is limited in the

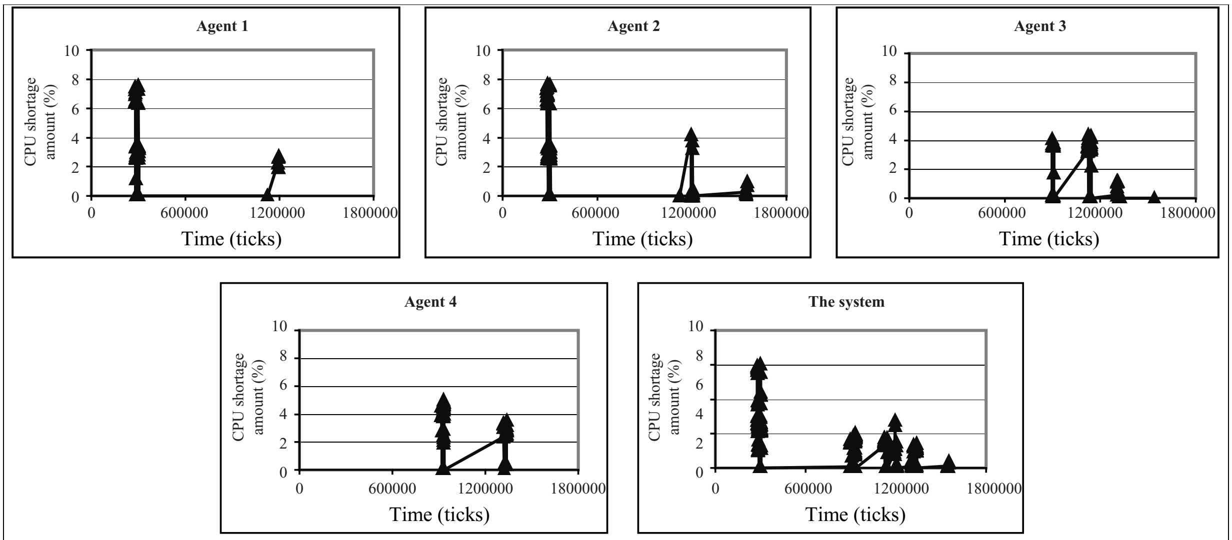

Fig. 10. Alleviation of CPU shortage crises.

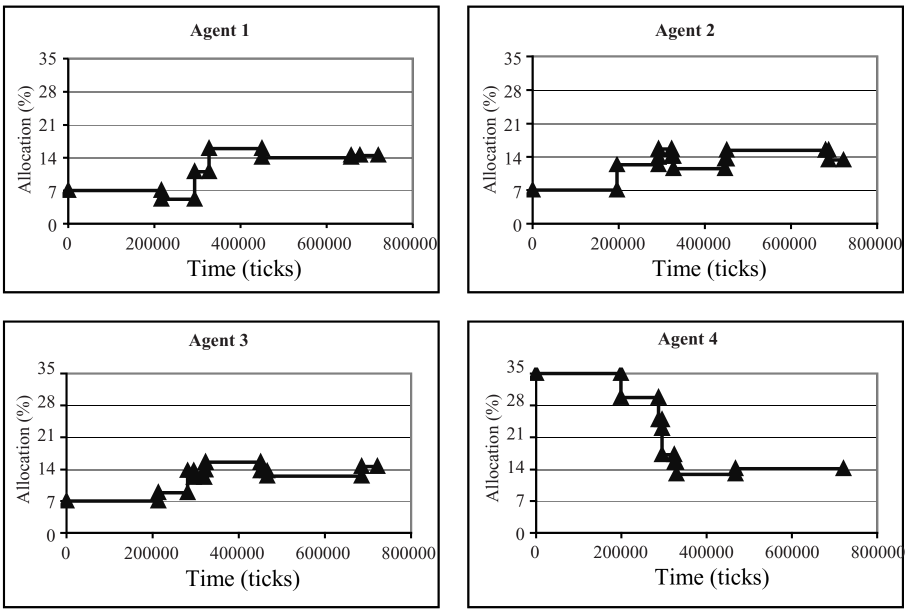

Fig. 11. Variation in agents' CPU allocations.

overall system. So it is very natural for an agent not to cooperate when requested by other agents – even the negotiation process will consume the agent's CPU resource no matter whether it will give up resource – or having chance to get more. However, it is also possible to reallocate CPU resource among agents at run-time in an *adaptive* manner to satisfy the agents' dynamic needs and thus improve the global performance. So agents might use CPU resource cooperatively shown as (1) one agent gives up part of its CPU to another as requested when it has additional CPU resource not being used, and (2) an agent will exit CPU requests immediately when it thinks its need can be satisfied even if it has chance to get more.

To test our coordination approach, we implemented a multiagent system in a dynamic, uncertain, real-time, and noisy environment. In the system, there are a group of agents that perform multi-sensor target tracking and adaptive CPU reallocation. Each agent controls a sensor and can activate the sensor to search-and-detect the environment. Each sensor has a set of consumable resources, such as beam-seconds (the amount of time a sensor is active), battery power, communication channels, and CPU resource that each sensor desires to utilize efficiently. Each sensor is at a fixed physical location and, as a target passes through its coverage area, it has to collaborate with neighboring sensors to triangulate their measurements to obtain an accurate estimation of the position and velocity of the target. And this is when a CPU shortage may arise: the activity may consume more CPU resource. When an agent detects a CPU shortage (usage greater than allocated), it needs to coordinate with other agents and will ask for help from its neighbors. In such an environment, the target-tracking tasks dynamically take place and also the CPU shortage events. The agents' behavior is uncertain and cannot be exactly expected by others due to their high autonomy. The CPU reallocation must be done in a real-time manner to satisfy the requirement of task fulfillment. The noisy environment is simulated with a JAVA-based program called RADSIM [22].

6. Experiments and results

Our multiagent system was implemented in C++. We employed the proposed coordination approach as well as the hybrid negotiation mechanism in the system. In the current design, each agent has $3+N$ threads. The *core* thread is responsible for the agent's making decisions, managing tasks, and overseeing negotiations. A *communication* thread is used to interact with the message passing system of the sensor. An *execution* thread actuates the physical sensor: calibration, search-and-detect for a target, and so on. Each agent also has N negotiation threads to concurrently negotiate with other agents. During a CPU shortage crisis, an agent suffers with its threads slowing down. This is simulated by activating sleeping functions in the threads.

Our multiagent system was designed to enable a random number of agents to operate in it. To make the experimental results distinct, specifically we implemented four agents in the system, and we assume all four agents are neighbors each other. On one hand, such an experimental setting will not impact the generality of applying our coordination approach in multiagent systems; on the other hand, such a setting makes it easy to observe each agent's resource coordination status at run-time.

Our experiments concentrated on: (1) verifying the sufficiency and necessity of the coordination approach implemented with the hybrid negotiation mechanism, more specifically, whether the coordination among agents can reduce the CPU shortage crises of each agent as well as the overall system, and whether the hybrid negotiation is necessary in resource coordination; (2) checking the agents' capabilities of dynamic resource coordination, more specifically, whether the agents can reallocate resource among themselves adaptively and fairly; (3) evaluating the efficiency of the coordination approach and the negotiation mechanism, more specifically, what about the success situations of coordination and individual hybrid negotiations, how the agents' characteristics of being both self-interested and altruistic influence the efficiency of coordination and hybrid negotiation, and whether the profiling and learning techniques can help agents to negotiate better; and (4) verifying the stability of the hybrid negotiation mechanism for resource coordination in complex environments.

We set the experiments to make the agents in the system have chances of coordinating CPU resource among them. By varying the initial CPU amount assigned to each agent, we created a series of scenarios for the resource-constrained agents: overly-constrained, mildly-constrained, symmetrically-constrained, and asymmetrically-constrained. Based on these scenarios, our hypotheses were: (1) that the coordination approach can alleviate CPU shortage crises of each agent as well as the overall system; (2) that the hybrid negotiation is necessary to alleviate CPU shortage crises of each agent as well as the overall system; (3) that the CPU resource will be reallocated more evenly among agents gradually; (4) that the coordination and negotiation are more successful in the mildly-constrained scenarios than in the overly-constrained scenarios, and more successful in the symmetrically-constrained scenarios than in the asymmetrically-constrained scenarios; (5) that the good balance of agents' being self-interested and being altruistic can improve the success rates of coordination and negotiation; (6) the profiling and learning techniques can help agents to negotiate better; and (7) that the agents' negotiation outcomes are stable in complex environments.

In our experiments, each agent has the same maximally needed CPU usage, 15.75%, for optimal processing (when all threads are concurrently running and actuating the physical sensors). Thus the overall system would need 63% of CPU for optimal processing. We conducted four experiments shown in Table 1.

In each experiment, we equipped each agent with the same initial casebases. But the initial CPU re-

Table 1
Configuration of experiments

Experiment	Agents' initial CPU allocation (%)				
	Agent 1	Agent 2	Agent 3	Agent 4	Total
#1	7	7	7	7	28
#2	7	7	7	35	56
#3	7	7	21	21	56
#4	7	14	14	21	56

source distribution among agents in each experiment is different. In experiment #1, the CPU resource was *overly-constrained* (28% allocated to the agents in total). This serves as a baseline scenario. In other experiments, each system had the total CPU allocation of 56% and thus was only *mildly-constrained*. In experiment #1, all four agents had poor initial CPU allocations (7%); while other experiments had three, two, and one poor agent, respectively. In the mildly-constrained scenarios (experiments #2, #3, and #4), we further created the *symmetrically-constrained* scenario (experiment #3) and *asymmetrically-constrained* scenarios (experiments #2 and #4). Taking into account the maximally needed CPU usage of each agent (15.75%), it is obvious that the resource competition degree of agents in each of these four experiments was incremental from the first one to the last one. So the degree of agents' being self-interested and altruistic will vary in these experiments. The different scenario setting allowed us to examine that when each agent had same negotiation strategies available, how coordination among agents behaved in different resource environments.

6.1. Alleviation of CPU shortage crises

To verify the sufficiency and necessity of the coordination approach implemented with the hybrid negotiation mechanism, we observed all the four experiments, and examined whether the coordination among agents could alleviate CPU shortage crises of each agent as well as the overall system and what would happen if there were no hybrid negotiation between agents. We evaluate a CPU shortage crisis with the CPU shortage amount in the crisis. In all experiments, the reduction in CPU shortage amount of each agent as well as the overall system was obvious. Figure 10 shows the variation in their CPU shortage amounts in experiment #3.

From Fig. 10, we have the following observations: (1) at the beginning, agents 1 and 2 encountered more CPU shortage crises while agents 3 and 4 did not encounter CPU shortage crises; (2) gradually, agents 1 and 2 encountered less CPU shortage crises while agents 3 and 4 began to encounter more CPU shortage crises; and (3) as the time progressed, the CPU shortage amount of each agent as well as the overall system decreased. These observations can be explained from the agents' initial CPU resource distribution and their characteristics of being both self-interested and altruistic. The first observation is due to the insufficient initial CPU allocations to agents 1 and 2 yet sufficient allocations to agents 3 and 4. The second observation is due to the hybrid negotiation mainly driven by self-interested initiators and altruistic responders. When agents 3 and 4 had additional resource they were not using, they gave up it to agents 1 and 2, which resulted in their own CPU shortage crises later. The third observation indicates that the agents were wisely and effectively coordinating their resources. This is due to our hybrid negotiation mechanism in which each agent is both self-interested and altruistic. On one hand, if the agents were not self-interested, no agent would desire to get resources from others at all. Then no coordination would happen, and each agent's CPU shortage degrees would be always similar rather than decrease. Also no responder would give up resources sensibly. For example, agents 3 and 4 would have given up more and more resources to others to a point that they would suffer from more serious CPU shortage crises than what are shown in Figure 10. On the other hand, if the agents were not altruistic, no responder would give up its resource at all. Then agents 3 and 4 would have just looked on other two agents suffering from serious CPU shortage problems with their own additional resources idle. Also each initiator would keep asking for resources more than what it needs. Then the CPU shortage crises of agents 1 and 2 could not be both alleviated. It is because only one agent that could provide the strongest support information could win the competition. In either case, without hybrid negotiation the agents would not positively coordinate their resources.

The similar experimental results can be observed in other three experiments. It indicates that through hybrid negotiations the agents can implement the reallocation of CPU resource among them and alleviate their CPU shortage crises. Based on the analysis to the third observation, we also can conclude that the hybrid negotiation is necessary for CPU coordination. Although our results only showed the variation tendency of the agents' CPU shortage problems within the time period of 1800000 time ticks, the CPU shortage crises happened after that did not resume to a worse degree.

6.2. Adaptive and fair resource reallocation

The main objective of coordination among agents is to enable the agents to adapt to variations in resource load. To check the agents' capabilities of dynamic resource coordination, we observed all the four experiments, and examined whether the CPU resource could be reallocated among agents adaptively and fairly. Figure 11 shows the variation in each agent's CPU allocation in experiment #2, which is an extremely asymmetrically-constrained scenario.

As can be seen in Fig. 11, after a series of dynamic CPU reallocation activities among agents, each agent's CPU resource allocation converged to an *average* level, 14% (recall that the total CPU resource amount of all four agents is 56% in experiment #2). Such convergence tendency can be observed in all experiments although in each experiment the agents' initial CPU resource allocations are different. The reason of convergence is that the agents are both self-interested and altruistic. The initiators try to get more resources from others but stop asking for resources once they have enough (see all the three poor agents in Fig. 11 could get resources from agent 4 at about the 280000th time tick). The responders are willing to help others through giving up their additional CPU resources to others but reluctant to hurt themselves (see agent 4's resource amount was never below the average level too much). From this convergence tendency, we can conclude that the coordination approach based on the hybrid negotiation mechanism is able to coordinate the agents' behavior and reallocate their CPU resource adaptively to variations in load.

6.3. Efficiency evaluation of coordination and negotiation

To evaluate the efficiency of the coordination approach and the hybrid negotiation mechanism in different scenarios, we compared the four experiments in terms of coordination success rate and negotiation success rate respectively. The *coordination success rate* of an agent refers to the percentage of the successful resource coordination activities among all the ones initiated by the agent. A resource coordination activity is successful if through that the agent successfully obtains CPU from at least one negotiation responder, regardless of the amount of CPU obtained. The *negotiation success rate* of an agent refers to the percentage of the successful negotiation processes among all the ones initiated by the agent. A negotiation process is successful if through that the agent can obtain the amount of CPU as requested. Figures 12 and 13 document the comparison results.

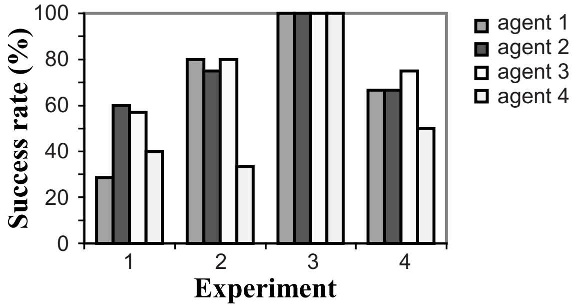

Fig. 12. Comparison among coordinations.

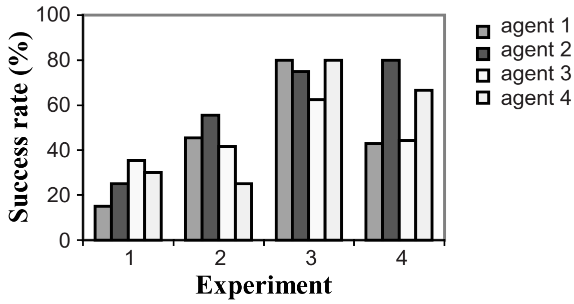

Fig. 13. Comparison among negotiations.

From Fig. 12, we can observe that (1) the average coordination success rate of agents in experiment #1 was the lowest among all the four scenarios; (2) the average coordination success rate of agents in experiment #3 was significantly higher than in other scenarios; and (3) in experiment #2, agent 4, the significantly richest agent of 35% CPU, had the lowest success rate. The similar patterns can be observed from Fig. 13. The first experimental result observed is due to that when the resource is overly constrained the chances of agents having additional resource are fewer than when the resource is mildly constrained. As a result, there is more competition among agents. Each agent is more self-interested and less altruistic.

Intuitively, the average coordination (negotiation) success rate of agents in experiment #4 should be better than the success rate in experiment #3, as the competition degree of agents in the former is the lowest among all experiments. The rich agents should be more willing to give up resources to the poor agents. However, the second observation shows that the average coordination (negotiation) success rate of agents in experiment #3 was significantly higher than in other experiments.

This is due to the symmetric initial CPU distribution among agents. In this case, there were equal numbers of *poor* agents and *rich* agents initially, which leads to a better balance of agents between being self-interested and being altruistic. In experiment #4, although agents 2 and 3 were relatively richer than agent 1, their resources possibly given up were both less than agent 3's in experiment #3. So compared to the latter, they were more self-interested as responders to wisely use their resources. Although there were more poor agents in experiment #3, these agents were more altruistic as initiators because both of them could get enough resources from just one rich agent. On the other hand, if a responder were only altruistic, or if an initiator were only self-interested, the average coordination (negotiation) success rate of agents in experiment #3 would have been lower than that observed in experiment #4, since there were more rich agents in the latter, and only one of poor agents (rather than both) could win the competition in experiment #3. However, we observed the opposite in the actual results due to the hybrid negotiation mechanism. Further, that means when the self-interested characteristics and the altruistic characteristics of agents are in a good balance, the hybrid negotiation mechanism could play a more significant role in coordination (negotiation).

The third observation shows that the significantly richest agent had the lowest coordination (negotiation) success rate. In our application domain, the events that possibly result in agents' CPU shortage crises occurred evenly to each agent. Thus after agents' CPU resource amounts converged to the average level, the chance that each agent initiated negotiations was not significantly different. However, at the beginning, the poor agents (agents 1, 2, and 3) initiated negotiations more often than the rich agent (agent 4). Through profiling neighbors and learning negotiation strategies, the initially poor agents were becoming better at deciding with whom to coordinate and how to coordinate, as they had more opportunities in such activities to obtain better evaluation abilities. This experimental result indicates that our neighbor profiling and case-based learning techniques played a significant role in agents' coordination (negotiation). They helped initially poor agents to be better at being self-interested as negotiation initiators. They also helped initially rich agents to be better at being self-interested and altruistic as negotiation responders. So the initially poor agents were more successful in obtaining resources than the initially rich agents.

Comparing Figs 12 and 13, we can find that the average coordination success rate of agents in each experiment was higher than their average negotiation success rate. Note that each coordination activity may consist of multiple negotiation processes. There are three reasons for not successful negotiations: (1) in the noisy environment, agents' communication was not efficient, which resulted in that negotiations were likely to stop earlier than the expected time; (2) in the resource-constrained environment, the agents' threads slowed down during CPU shortage, which resulted in that time-bounded negotiations were more likely to fail; and (3) there were competitions among multiple agents that requested resource from the same responders. This comparison result shows the benefit of an agent negotiating with multiple agents at the same time rather than with only one agent. Although the outcome of each individual negotiation is uncertain, the agent can request resource from multiple negotiation partners, to reduce the impact of both the environment and the competition among initiators. Then the overall coordination outcome can be better. The difference in the comparison result indicates that via case-based reasoning the agent could decide the appropriate number of negotiation partners rather than just negotiate with only one agent.

6.4. *Stability of the hybrid negotiation mechanism*

Table 2 further validates the previous experimental results in detail. From Table 2, we can find three stable experimental results: (1) initially poorer agents always obtained more total CPU than initially richer agents; (2) for the agents of significantly large (extremely rich) initial CPU amount (e.g. agent 4 in experiment #2, agents 3 and 4 in experiment #3, and agents 4 in experiment #4), the obtained was always far less than the given up; and (3) the agents of similar initial CPU allocations obtained similar amounts of CPU from negotiations, especially in the mildly-constrained scenarios (experiments #2, #3, and #4). These results show the stable similarity of agents' negotiation outcomes – the agents that had similar initial CPU allocations had similar degrees of being self-interested and altruistic in hybrid negotiations. In our multiagent system, each agent was assigned the same set of beliefs and held the same initial casebases, so the agents' negotiation strategies were similar as initiators and were also similar as responders. Although the similarity of agents' negotiation outcomes can be understood from the similarity of their negotiation strategies, it is not occasional for agents to get negotiation outcomes of the stable similarity. That is because in the complex environments the negotiation outcomes were uncertain. The above results imply that the hybrid negotiation mechanism is stable for resource coordination in complex environments. That means our coordination approach based on the hybrid negotiation mechanism is reliable in complex environments.

7. Related work

There has been a variety of research work that deals with coordination in multiagent systems. Much earlier work assumes it is a design-time problem (e.g.'[4, 30,31,34]). The agents are designed to follow certain rules (social laws or conventions) of behavior that constrain them to act in coordinated ways. However, in dynamic and uncertain environments, it is necessary for the agents to act flexibly to the change of the environment. The rules might result in too restricted agents. In our research domain, the agents are both self-interested and altruistic about the coordinated resources, and their behavior is motivated by their complex characteristics. Since the agents' motivations may vary at run-time with the change of the resource coordination status, it is difficult to set the appropriate rules at design-time to constrain the agents' complex run-time behavior. So our work focuses on the run-time coordination in multiagent systems.

So far, more and more research stresses the run-time coordination among agents (e.g. [1,2,5,6,8,11,41, 43]. In [1,5], a coordination framework is presented that enables agents to dynamically select the particular coordination protocol at run-time from a set of available protocols in order to coordinate their inter-related activities. Although each agent tries to maximize its own benefit through taking optimal actions, the agents' goals are not mutually exclusive, so there is only cooperation and no competition. In their work, agents do not negotiate with each other at all. In [41,43], a multi-step negotiation mechanism is discussed. But they also only concern cooperation among agents, and the negotiation is cooperative negotiation. Moreover, the run-time coordination work in [1,5,41,43] is built in grid worlds and focuses on the cooperation between two agents on task allocation. Their grid-world environment provides a finite, discrete, certain, and noiseless problem space for agents. However, we focus on the resource coordination problem among multiple agents. The problem space is infinite and continuous due to the dynamic variation of resource requirement and the continuity of resource allocation. Our agents work in a dynamic, uncertain, real-time, and noisy environment, which increases the complexity of the problem. In [2,6, 8,11], resource coordination among multiple agents is concerned. Specific approaches are proposed to handle resource reallocation triggered by run-time variation of application needs in multiagent systems. There are two general strategies. First, the reallocation is solved from the viewpoints of resource users (agents) [2,6]. Resources are passive goods to be managed and the decision makers are the users. Second, the reallocation is solved in which resources are represented by their own agents [8,11]. These resource agents work with the user agents to decide how best to assign the resources. We also focus on the approach to resource reallocation triggered by run-time variation of application needs in multiagent systems and our strategy is akin to the first one. Compared to the agents in all these run-time coordination work, our agents are both altruistic and self-interested about limited resources. The dual characteristics of agents complicate their decision making process and make their resource coordination behavior not only cooperative but also competitive. Thus we introduce the hybrid negotiation mechanism to run-time resource coordination in such a complex environment.

A generic negotiation may be competitive that occurs among self-interested agents (e.g. [18,20]), or cooperative that occurs among altruistic agents (e.g. [26, 36]). In our research domain, the agents are both self-interested and altruistic, which results in the hybrid negotiation in which competition and cooperation coexist. We build a hybrid negotiation model and implement the hybrid negotiation mechanism based on the argumentation-based negotiation protocol, case-based reasoning, neighbor profiling, and case-based learning. This negotiation mechanism was designed to adapt to the agents' distinct characteristics and the complex environment.

Compared to the typical negotiation mechanisms in some others' work, our hybrid negotiation mechanism has the following characteristics. First, some agents negotiate using the unified negotiation protocol in worth-driven, state-driven, and task-driven domains where agents look for mutually beneficial deals to perform task distribution [30,34]. In our hybrid negotiation mechanism, the negotiations among agents are driven by their characteristics of being both self-interested and altruistic. Second, in some negotiation mechanisms, agents can negotiate in a fully prescribed manner where the negotiating parties know exactly what each other's cost and utility functions are, or when such knowledge is learned during the first step of interaction in a negotiation [16,21]. In our hybrid negotiation mechanism, the agents do not know exactly what each other's cost

Table 2
Comparison of obtained CPU amounts

Initiating agent	Total CPU amount obtained from an agent (%)															
	Experiment #1				Experiment #2				Experiment #3				Experiment #4			
	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1	–	0	0.84	0.84	–	3.01	0	8.24	–	0	7.94	8.78	–	0.82	0.85	8.05
2	2.31	–	0	0	1.98	–	3.50	7.12	0	–	8.64	8.68	0	–	0.50	0.49
3	0.18	0	–	0	1.80	3.76	–	6.71	3.63	4.08	–	0	0	0.23	–	0.60
4	0.11	2	0.12	–	0	0	1.17	–	5.85	6.08	0	–	1.54	1.62	0	–

and utility functions are, but via neighbor profiling they can learn about other agents from their negotiation experience up to now. Third, in general argumentation-based negotiation protocols, an agent sends over its inference rules to another agent to demonstrate the soundness of its arguments [32]. In our hybrid negotiation mechanism, the agents do not send over their inference rules to others. They only send over their arguments and the receivers will judge the soundness of the arguments. Finally, there are also agents that incorporate AI techniques like Bayesian learning into negotiation [42]. In our hybrid negotiation mechanism, the agents can search for appropriate negotiation strategies via case-based reasoning and improve their negotiation strategies via case-based learning. These strategy search techniques have not yet been found to be used in others' work.

In [24], a dual concern model is proposed in negotiation based on the concern degree of an agent to its own outcomes and others' outcomes. Five negotiation attitudes are proposed as following: (1) competition – negotiation partners pursue own outcomes strongly, and show no concern for the other party obtaining their desired outcomes, (2) yielding – negotiation partners show no concern in whether they attain own outcomes, but are quite interested in whether the other party attain desired outcomes, (3) inaction – negotiation partners do not concern whether any party attain desired outcomes, (4) cooperation – negotiation partners show high concern in obtaining own outcomes, as well as high concern for the other party obtaining their desired outcomes, and (5) compromising – negotiation partners show moderate concern in obtaining own outcomes, as well as moderate concern for the other party obtaining their desired outcomes. Our hybrid negotiation mechanism is consistent with this model in some degree, but we only concern the competitive and cooperative negotiation attitudes of agents. More importantly, only these attitudes can be held by our agents. As our agents are not only altruistic but also self-interested, they will concern their own outcomes and thus they will not exhibit the yielding attitude. As agents negotiate at the price of computational and communication cost, and agents are rational during coordination, the agents will not exhibit the inaction attitude. The compromising attitude can be regarded as just the combination of cooperative and competitive attitudes.

8. Conclusions and future work

In this paper, we present a coordination approach using a hybrid negotiation mechanism. We have implemented a multiagent system that employs this approach for resource allocation in a dynamic, uncertain, real-time, and noisy environment. Based on the experimental results, we conclude that: (1) our coordination approach centered on the hybrid negotiation mechanism does work for adaptive resource allocation among agents: the coordination among agents could alleviate CPU shortage crises of each agent as well as the overall system; (2) the hybrid negotiation is necessary for resource coordination: the agents would not positively coordinate resources without hybrid negotiation; (3) the agents are able to dynamically coordinate resources on their own: the agents showed adaptive and emergent behaviors in fairly reallocating their resources, and the overall system was able to converge through coordination; (4) the efficiency of the coordination approach and the negotiation mechanism is influenced by the agents' characteristics and their profiling and learning capabilities: the good balance of agents' being self-interested and being altruistic could improve the success rates of coordination and negotiation, and the profiling and learning techniques helped agents to be better at being self-interested as negotiation initiators and helped agents to be better at being self-interested and altruistic as negotiation responders; and (5) the hybrid negotiation mechanism can play a stable role in complex environments for resource coordination: the agents' negotiation outcomes were stable.

We realize that there are certain limitations in our experimental design. In the current experiments, the number of agents is not significantly large and the variety of tasks is not enough, only target-tracking tasks encountered by agents that may incur resource shortage crises of agents. The current setting of the agents' population is mostly for the convenience of comparing the impact of different resource configurations on agents' coordination behavior. Taking into account our neighborhood design, even that there are a significantly high number of agents, the coordination activities still take place among a finite number of agents that are in a neighborhood. Thus the current experimental results based on a low number of agents have been able to indicate the sufficiency, necessity, and efficiency of our coordination approach. With regard to the variety of tasks, the execution of any task may incur the possible CPU resource shortage. So one special type of tasks has been able to show the agents' resource coordination behavior. However, the increase of the population of agents and the variety of tasks will increase the complexity of the coordination problem, and as a result increase the complexity of agents' characteristics of being both self-interested and altruistic. In the future, firstly, we will increase the number of agents in the system and the variety of tasks in the environment to verify the proposed coordination approach in more complex contexts.

In the future, secondly, we will increase the flexibility of the responders' commitment strategy in negotiations. When a responder negotiates with multiple initiators simultaneously, it commits the initiators' requests with a first-convinced-first-serve strategy. This strategy can save the coordination time of agents and an initiator can get resource immediately when it makes the responder convinced. But if the responder has committed the first initiator with all its additional resource, then other initiators that are not quick enough cannot obtain resources any more. The worse thing is that maybe the first initiator requested and obtained more resource than the actually needed but the additional resource is just idle. In the future, we will introduce the utility mechanism to build a utility-relevant commitment strategy. Then the responder may reserve the resource and distribute it among all the initiators based on their different utility values. The utility values can be computed from initiators' resource shortage degrees. But this strategy may lengthen the coordination time of some agents, which conflicts with the real-time requirement of coordination. As a tradeoff, we aim to enable the responder to make a decision on which strategy to be used based on the allowed time. If the coordination time is strictly bound, the first-convinced-first-serve strategy is preferred; otherwise, the utility-relevant strategy is preferred.

In the current design of the hybrid negotiation mechanism, we use profiling and learning techniques to help initiators to be better at being self-interested, and also help responders to be better at being self-interested and being altruistic. But we do not have techniques to help initiators to be better at being altruistic. In the future, thirdly, we plan to add more learning techniques to help initiators to be better at being altruistic. For examples, the initiators will learn to anticipate resource shortage to disappear, learn whether to cut down the number of negotiation requests, learn to decrease the requested resource amount during negotiation, learn to terminate ongoing negotiations immediately when the need is satisfied, and so forth. For the learning to anticipate resource shortage to disappear, the agent may even decide not to ask for help at all, which is even more altruistic to other initiators. Further, we also plan to add more learning techniques to help responders to be better at self-interested. For example, a responder learns to predict its own resource need in the soon future more precisely, to guarantee that the resource can satisfy its own need first.

Acknowledgment

This work is partially supported by a grant from the DARPA ANTS project, subcontracted from the University of Kansas (Contract number: 26-0511-0026-001). We would like to thank Dr. Costas Tsatsoulis who helped initiate the groundwork of this research and the anonymous reviewers for their comments on the paper.

References

[1] R.A. Bourne, C.B. Excelente-Toledo and N.R. Jennings. *Run-time selection of coordination mechanisms in multi-agent systems*, in Proceedings of the Fourteenth European Conference on Artificial Intelligence (ECAI-2000), Berlin, Germany, 2000, 348–352.

[2] D.D. Corkill, *A framework for organizational self-design in distributed problem solving networks*, PhD Dissertation, University of Massachusetts, 1983.

[3] D.D. Corkill and V.R. Lesser, *The use of meta-level control for coordination in a distributed problem solving network*, in Proceedings of the Eighth International Joint Conference on Artificial Intelligence (IJCAI'83), Karlsruhe, Germany, 1983, 748–756.

[4] E.H. Durfee and V.R. Lesser, Partial global planning: A coordination framework for distributed hypothesis formation, *IEEE Transactions on Systems, Man, and Cybernetics* **21**(5) (1991), 1167–1183.

[5] C.B. Excelente-Toledo and N.R. Jennings, The dynamic selection of coordination mechanisms, *Journal of Autonomous Agents and Multi-Agent Systems* (2003).

[6] S.S. Fatima and M.J. Wooldridge, *Adaptive task and resource allocation in multiagent systems*, in Proceedings of the Fifth International Conference on Autonomous Agents (AGENTS'01), Montreal, Quebec, Canada, 2001.

[7] S.S. Fatima, M.J. Wooldridge and N.R. Jennings, *Multi-issue negotiation under time constraints*, in Proceedings of the First International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS'02), Bologna, Italy, 2002, 143–150.

[8] F. Guichard and J. Ayel, Logical reorganization of DAI systems, in: *Intelligent Agents: Theories, Architectures, and Languages*, M. Wooldridge and N. R. Jennings, eds, Springer-Verlag: Berlin, Germany, 1995, pp. 118–128.

[9] R.H. Guttman and P. Maes, *Cooperative vs. competitive multi-agent negotiations in retail electronic commerce*, in Proceedings of the Second International Workshop on Cooperative Information Agents (CIA'98), Paris, France, 1998.

[10] M.N. Huhns and L.M. Stephens, Multiagent systems and societies of agents, in: *Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence*, G. Weiss, ed., The MIT Press: Cambridge, MA, 1999, pp. 79–120.

[11] T. Ishida, L. Gasser and M. Yokoo, Organization self design of production systems, *IEEE Transactions on Knowledge and Data Engineering* **4**(2) (1992), 123–134.

[12] N.R. Jennings, Commitments and conventions: The foundation of coordination in multi-agent systems, *The Knowledge Engineering Review* **8**(3) (1993), 223–250.

[13] N.R. Jennings, Coordination techniques for distributed artificial intelligence, in: *Foundations of Distributed Artificial Intelligence*, G.M.P. O'Hare and N.R. Jennings, eds, Wiley Interscience, 1996, pp. 187–210.

[14] N.R. Jennings, An agent-based approach for building complex software systems, *Communications of the ACM* **44**(4) (2001), 35–41.

[15] N.R. Jennings, S. Parsons, P. Noriega and C. Sierra, *On argumentation-based negotiation*, in Proceedings of International Workshop on Multi-Agent Systems, Boston, MA, 1998.

[16] J. Kolodner, *Case-Based Reasoning*, Morgan Kaufmann: San Mateo, CA, 1993.

[17] S. Kraus, Beliefs, time, and incomplete information in multiple encounter negotiations among autonomous agents, *Annals of Mathematics and Artificial Intelligence* **20**(1–4) (1997), 111–159.

[18] S. Kraus, *Strategic Negotiation in Multi-Agent Environments*, The MIT Press: Cambridge, MA, 2001.

[19] S. Kraus, E. Ephrati and D. Lehmann, Negotiation in a non-cooperative environment, *Journal of Experimental and Theoretical AI* **3**(4) (1998), 255–282.

[20] S. Kraus, K. Sycara and A. Evenchik, Reaching agreements through argumentation: A logical model and implementation, *AI Journal* **104**(1–2) (1998), 1–69.

[21] S. Kraus and J. Wilkenfeld, A strategic negotiations model with applications to an international crisis, *IEEE Transactions on Systems, Man, and Cybernetics* **23**(1) (1993), 313–323.

[22] S. Kraus, J. Wilkenfeld and G. Zlotkin, Multiagent negotiation under time constraints, *Artificial Intelligence* **75** (1995), 297–345.

[23] J. Lawton, The Radsim simulator, in:*Distributed Sensor Networks: A Multiagent Perspective*, V. Lesser, C.L. Ortiz, Jr. and M. Tambe, eds, Kluwer Academic Publishers: The Netherlands, 2003, pp. 11–20.

[24] V.R. Lesser, Cooperative multiagent systems: A personal view of the state of the art, *IEEE Transactions on Knowledge and Data engineering* **11**(1) (1999), 133–142.

[25] R. Lewicki, D. Saunders, J. Minton and B. Barry, *Essentials of Negotiation*, The Third Edition, McGraw-Hill College, 2003.

[26] J. Luo, *Case-Based Learning Behavior in a Real Time Multi-Agent System*, Master Project, Department of Computer Science and Engineering, University of Nebraska-Lincoln, USA, 2003.

[27] R. Mailler, R. Vincent, V. Lessor, J. Shen and T. Middlekoop, Soft-real time, cooperative negotiation form distributed resource allocation, *AAAI Fall Symposium on Negotiation Methods for Autonomous Cooperative Systems*, North Falmouth, MA, 2001, 63–69.

[28] T.W. Malone, Modeling coordination in organizations and markets, *Management Science* **33** (1987), 1317–1332.

[29] H. Raiffa, *The Art and Science of Negotiation*, Harvard University Press: Cambridge, MA, 1982.

[30] A. Rao and M. Georgeff, *BDI agents: from theory to practice*, in Proceedings of the First International Conference on Multiagent Systems (ICMAS-95), San Francisco, CA, 1995, 312–319.

[31] J.S. Rosenschein and G. Zlotkin, *Rules of Encounter: Designing Conventions for Automated Negotiation among Computers*, The MIT Press: Cambridge, MA, 1994.

[32] Y. Shoham and M. Tennenholtz, *On the synthesis of useful social laws for artificial agent societies*, in Proceedings of the Tenth National Conference on Artificial Intelligence (AAAI-92), San Jose, California, 1992, 276–281.

[33] C. Sierra, N.R. Jennings, P. Noriega and S. Parsons, *A framework for argumentation-based negotiation*, in Proceedings of the Fourth International Workshop on Agent Theories, Architectures and Languages (ATAL'97), Providence, RI, 1997, 167–182.

[34] M.P. Singh, A.S. Rao and M.P. Georgeff, Formal methods in DAI: Logic-based representation and reasoning, in: *Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence*, G. Weiss, ed., The MIT Press: Cambridge, MA, 1999.

[35] R.G. Smith and R. Davis, Frameworks for cooperation in distributed problem solving, *IEEE Transactions on Systems, Man, and Cybernetics* **11**(1) (1981), 61–70.

[36] L.-K. Soh and X. Li, *An integrated multi-level learning approach to multiagent coalition formation*, in Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence (IJCAI'2003), Acapulco, Mexico, 2003, 619–624.

[37] L.-K. Soh and C. Tsatsoulis, *Reflective negotiating agents for real-time multisensor target tracking*, in Proceedings of the Seventeenth International Joint Conference on Artificial Intelligence (IJCAI'2001), Seattle, WA, 2001, 1121–1127.

[38] K.P. Sycara. Multiagent systems, *AI Magazine* **19**(2) (1998), 79–92.

[39] D.N. Walton and E.C.W. Krabbe, *Commitment in Dialogue: Basic Concepts of Interpersonal Reasoning*, SUNY Press: Albany, NY, 1995.

[40] G. Weiss, ed., *Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence*. The MIT Press: Cambridge, MA, 1999.

[41] M. Wooldridge, Intelligent agents, in: *Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence*, G. Weiss, ed., The MIT Press: Cambridge, MA, 1999, pp. 27–78.

[42] P. Xuan, V. Lesser and S. Zilberstein, *Communication decisions in multi-agent cooperation: model and experiments*, in Proceedings of the Fifth International Conference on Autonomous Agents (Agents-2001), Montreal, Canada, 2001, 616–623.

[43] D. Zeng and K. Sycara, Bayesian learning in negotiation, *International Journal of Human-Computer Studies* **48** (1998), 125–141.

[44] X. Zhang, R. Podorozhny and V. Lesser, *Cooperative, multi-step negotiation over a multi-dimensional utility function*, in Proceedings of the IASTED Int. Conf., Artificial Intelligence and Soft Computing (ASC 2000), ASTED/ACTA Press, Banff, Canada, 2000, 136–142.

[45] G. Zlotkin and J.S. Rosenschein, Mechanism design for automated negotiation, and its application to task oriented domains, *Artificial Intelligence* **86**(2) (1996), 195–244.