COALITION FORMATION WITH UNCERTAIN TASK EXECUTION

Hosam Hanna
GREYC - University of Caen
Bd Maréchal Juin
14032 Caen - France

Keywords: Coalition formation, Uncertainty, Group decision, Markov decision process.

Abstract: We address the problem of coalition formation in environments where tasks' executions are uncertain. Although previous works provide good solutions for coalition formation problem, the uncertain task execution problem is not taken into account. In environments where task execution is uncertain, an agent can't be sure whether he will be able to execute all the subtasks that are allocated to him or he will ignore some of them. That is why forming coalition to maximize the real reward is an unrealizable operation. In this paper, we propose a theoretical approach to form coalition with uncertain task execution. We view the formation of a coalition to execute a task as (1) a decision to make and (2) as an uncertain source of gain. We associate then the allocation of a task to a coalition with an expected reward that represents what agents expect to gain by forming this coalition to execute this task. Also, the agents' aim is to form coalition to maximize the expected reward instead of the real reward. To reach this objective, we formalize the coalition formation problem by a Markov Decision Process (MDP). We consider the situation where decisions are taken by one agent that develops and solves the corresponding MDP. An optimal coalition formation which maximizes the agents' expected reward is then obtained.

1 INTRODUCTION

Coalition formation is an important cooperation methode for applications where an agent can't efficiently execute a task by himself. The coalition formation problem has widely been studied and many approaches have been proposed. In game theory, we find some works that treated this problem without taking into account the limited time calculation (Aumann, 1959), (Bernheim et al., 1987), and (Kahan and Rapoport, 1984). In cooperative environments, many algorithms were suggested to answer the question of group formation (Shehory and Kraus, 1998). In multiagent systems, there are several coalition formation mechanisms that include a protocol as well as strategies to be implemented by agents given the protocol (Klusch and Shehory, 1996), (Shehory and Kraus, 1998), (Zlotkin and Rosenschein, 1994), (Learman and Shehory, 2000). All these works have common assumptions: *resources consumption is perfectly controlled by agents and the formation of a coalition to execute a task is a certain source of reward*. In other words, an agent can exactly determine the quantity of resources he will consume to execute any subtask, and the formation of a coalition to execute a task is sufficient to obtain the corresponding reward. In this study, we relax this assumption in order to adapt coalition formation to more real cases, and we investigate the problem of formation coalition in environments where agents have uncertain behaviors.

Several works have investigated the coalition formation problem where coalition value is uncertain or known only to a limited degree of certainty. In (Ketchpel, 1994), author considered the case where agents do not have access to coalition value function, and he proposed a two-agents auction mechanism that allows to determine coalitions of agents that will work together, and to decide how to reward the agents. In (Blankenburg et al., 2003), authors studied situations where coalition value is known only to a limited degree of certainty. They proposed to use fuzzy quantities instead of real numbers in order to express the coalitions value. A fuzzy Kernel concept has been introduced in order to yield stable solutions. Although the complexity of the fuzzy kernel is exponential, it has been shown that this complexity can be reduced to polynomial complexity by placing a cap on the size of coalitions. The uncertainty on coalition value can be due to the unknown execution cost. In fact, when agents reason in term of utility, the net benefits of a

Hanna H. (2006).
COALITION FORMATION WITH UNCERTAIN TASK EXECUTION.
In *Proceedings of the Eighth International Conference on Enterprise Information Systems - AIDSS*, pages 164-169
DOI: 10.5220/0002459601640169

coalition is defined as the coalition value minus the execution cost of all the coalition's members. When an agent of the coalition does not know with certainty the execution costs of the other members, it is uncertain regarding both the coalition's net benefits and its net benefits. A protocol allowing agents to negotiate and form coalition in such a case has been proposed in (Kraus et al., 2003) and (Kraus et al., 2004). Another source of uncertainty on coalition value can be the imperfect or deceiving information. A study for this case has been proposed in (Blankenburg and Klusch, 2004). In (Chalkiadakis and Boutilier, 2004), authors proposed a reinforcement learning model to allow agents to refine their beliefs about others' capabilities.

Although these previous works deal with an important uncertainty issue (uncertain coalition value), they have several restrictive assumptions regarding another possible sources of uncertainty as the uncertain resources consumption (uncertain task execution) that can be due to the uncertain agent's behavior and to the environment's dynamism. In addition, they do not take into account the effects of forming a coalition on the future possible formations, a long-term coalition formation planning can not then be provided. In applications as planetary rovers, for example, an agent is confronted with an ambiguous environment where he can not control his resources consumption when executing tasks as good as he does in laboratory. A coalition formation planning is important so that agents adapt coalition formation to their uncertain behaviors.

The problem is more complex when resources consumption is uncertain for all the agents. Unfortunately, in such a system, an agent can't be sure whether he (or another agent) will be able to execute all the subtasks that are allocated to him or he will ignore some of them. So, forming coalitions to maximize the agents' real reward is a complex (even unrealizable) operation. In fact, a task is considered as non executed if at least one of its subtasks is not executed. That is why, forming a coalition to execute a task is a necessary but not sufficient constraint to obtain a reward, and the agents' reward must be subjected to the task execution and not only to the coalition formation and task allocation. In this paper, we take into account these issues and we present a probabilistic model, based on Markov Decision Process (MDP), that provides a coalition formation planning for environments where resources consumption is uncertain. We will show that according to each possible resources consumption, agents can decide by an optimal way which coalition they must form.

We begin in Section 2 with a presentation of our framework. In section 3, we sketch our solution approach. We explain how to form coalition via MDP in Section 4.

2 FRAMEWORK

We consider a situation where a set of m fully-cooperative agents, $A = \{a_1, \ldots, a_m\}$ have to cooperate to execute a finite set of tasks $\mathbb{T} = \{T_1, \ldots, T_n\}$ in an uncertain environment. The tasks will be allocated in a commonly known order: without loss of generality, we assume that this ordering is $T_1, T_2, \cdots, T_n$. Each agent a_k has a bounded quantity of resources $\mathcal{R}^k$ that he uses to execute tasks. Each task consists of subtasks: for simplicity, we assume that every task $T_i \in \mathbb{T}$ is composed by q subtasks such as $T_i = \{t_i^1, \ldots, t_i^q\}$. Agent a_k is able to perform only a subset $E_i^k \subset T_i$ of the subtasks of a given task T_i. We assume that each task $T_i \in \mathbb{T}$ satisfies the condition $T_i \subseteq \bigcup_{a_k \in A} E_i^k$, otherwise it is an unrealizable task. For each subtask $t_i^l, T_i, \in \mathbb{T}, l = 1, \ldots, q$, we can define the set of agents, $AE(t_i^l)$, that are able to perform t_i^l as follows: $AE(t_i^l) = \{a_k \in A | t_i^l \in E_i^k\}$. Since an agent can't execute a task T_i by himself, a coalition of agents must be formed in order to execute this task. Such a coalition can be defined as a $q-$tuple: $\langle a^1, \ldots, a^q \rangle$ where agent $a^l \in A$ executes subtask $t_i^l \in E_i^{a^l}$. We let $\mathbb{C}(T_i)$ denote the set of all possible coalitions that can perform task T_i, it can be defined as follows: $\mathbb{C}(T_i) = \{\langle a^1, \ldots, a^q \rangle | a^l \in A, t_i^l \in T_i, t_i^l \in E_i^{a^l}, l = 1, \ldots, q\}$. A task is considered as realized if and only if all its subtasks have been performed. For each realized task T_i, agents obtain a reward. We consider a general situation where the tasks can be executed with different qualities. For example, two agents can take photos for the same object, but the resolution can be different. The reward corresponding to the execution of a task depends then on the coalition that executes the task. We assume that agents have a function $w(T_i, c)$ that expresses the reward that can be obtained if the coalition c *executes* task T_i.

3 SOLUTION APPROACH

The key idea, in our approach, is to view the formation of a coalition to execute a task as a decision to make that provides an expected reward instead of a real gain. What one expects to gain by forming collation c to execute task T_i? In fact, when T_i is allocated to c, the agents expect to obtain two values. The first one is the value $w(T_i, c)$ which is subjected to the execution of task T_i. The second expected value expresses the gain that can be obtained from future formation and allocation taking into consideration resources quantity consumed to execute T_i. Indeed, when a coalition executes a task, the agents' available resources is reduced. The chances to execute another tasks can then be reduced. As the resources collection consumed to execute task T_i depends on the coalition c executing T_i, the gain the agents can obtain from future formation and allocation depends also on coalition c. Finally, the expected reward associated to the formation of a coalition to execute a task is sum of these two expected values. It is necessary to recall here that our expected reward definition is different from the *expected coalition value* defined in (Chalkiadakis and Boutilier, 2004) for a Bayesian reinforcement learning model. In fact, the expected coalition value notion is used to express what an agent, basing on his expectation regarding the capabilities of other agents, beliefs about the value of any coalition. In addition, this notion doesn't allow agents to take into account the impact of the formation of a coalition on the gain that can be obtained from the formation of another coalitions (our second expected value).

Differently from known coalition formation methods that maximize the agents' real gain[1], the goal of our agents is defined as follows: *for each task T_i, form a coalition c by such a way that it maximizes agents' long-term expected reward*. To realize this objective, we have to treat the uncertain resources consumption and to formalize the expected reward associated to coalition formation. We will use a discreet representation of resources consumption and then define an execution probability distribution. Finally, we formalize the coalition formation problem by a Markov decision process (MDP). It is well known that solving a MDP allows to determine an optimal policy maximizing the long-term expected reward (Bellman, 1957; Puterman, 1994).

3.1 Uncertain Resource Consumption

In order to deal with the uncertain resources consumption, we assume that the execution of subtask $t_i^l \in T_i$ by agent a_k can consume one quantity of resources from a finite set $R_k^{t_i^l}$ of possible quantities of resources. For simplicity, we assume that there are p resources quantities in the set $R_k^{t_i^l}$. Agent a_k doesn't know which quantity of resources will be consumed, but he can anticipate it using some probability distribution:

Definition 3.1 *With each agent $a_k \in A$ is associated an execution probability distribution PE_k where $\forall t_i^l \in T_i, \forall r \in R_k^{t_i^l}, PE_k(r, t_i^l)$ represents the probability to consume the resources quantity r at the time of the execution of subtask t_i^l by agent a_k.*

[1]Or another type of gain that doesn't include the impact of the formation of a coalition on the future formations.

If a coalition $c = \langle a^1, \ldots, a^q \rangle \in \mathbb{C}(T_i)$ executes task T_i, a resources collection such as $\langle r^1, \ldots, r^q \rangle$ can be consumed, where agent a^k consumes quantity r^k to perform subtask t_i^k. Since one of p resources quantities can be consumed by each agent a^k to execute subtask t_i^k, then the execution of T_i by c consumes one collection from p^q resources collections. We let H_i^c denote the set of all these resources collection. The probability $Pr(\langle r^1, \ldots, r^q \rangle, T_i)$ to consume collection $\langle r^1, \ldots, r^q \rangle \in H_i^c$ at the time of the execution of T_i by c is then the probability that each agent a^k consumes the quantity r^k. Using definition (3.1), this probability can be defined as follows[2]:

$$Pr(\langle r^1, \ldots, r^q \rangle, T_i) = \prod_{k=1}^{q} PE_{a^k}(r^k, t_i^k) \quad (1)$$

3.2 Coalition Expected Reward

In our context, a specific agent, "controller", is charged to form coalitions and to allocate tasks. Controller views the formation of a coalition to execute a task as a decision to make. When such a decision is made, a coalition is formed, a task is allocated to this coalition, and a resources collection will be consumed to execute the allocated task. As we have shown in Section 3, the decision to form a coalition to execute a task is associated with an expected reward. In the following, we show how controller can calculate this expected reward.

The controller observes the state of the system as the couple of available resources of all the agents and the set of formed coalitions and allocated tasks. Being in a state S, the decision that consists in forming a coalition c to execute a task T_i drives the system into a new state S^h in which task T_i has been allocated to coalition c and a resources collection $h \in H_i^c$ is anticipated to be consumed when c executes T_i. In order to take into account the uncertain task execution, controller must anticipate all the possible resources collections that can be consumed when c executes T_i; each possible consumption drives the system into a different state. If agents of coalition c have enough resources to execute T_i (collection h is less than c's agents available resources), then the system receives in state S^h an immediate gain $w(T_i, c)$ (first expected value), else it receives zero. From state S^h another decision can be made and another reward can be so obtained (second expected value). We let $V[S^h]$ denote the gain in state S^h and we define it as the sum

[2]Since PE_k is a distribution probability on $R_k^{t_i^l}$, we have $\sum PE_k(r, t_i^l) = 1, \forall r \in R_k^{t_i^l}$. It is easy to verify that Pr represents a distribution probability on H_i^c: $\sum_{\forall h \in H_i^c} Pr(h, T_i) = 1$.

of both last rewards (see Section 4.3 for mathematical definition). Being in state S, the probability to gain $V[S^h]$, if coalition c is formed to execute T_i, is expressed by the probability to consume resources collection h because the system reaches state S^h if collection h has been consumed. This probability is defined by equation 1. We can say now that being in state S the decision to form coalition c to execute T_i drives to state S^h and allows to gain $V[S^h]$ with probability $Pr(h, T_i)$, where $h \in H_i^c$. The expected reward of this decision can be defined as follows:

$$E(\text{Forming } c \text{ to execute } T_i) = \sum_{h \in H_i^c} Pr(h, T_i) \times V[S^h] \quad (2)$$

We note that the expected reward associated to a decision made in the state S depends on the gain that can be obtained in each state S^h, and so on. The question is then: being in a state S and knowing that there are $|\mathbb{C}(T_i)|$ coalitions capable to execute T_i, which decision controller has to make in order to maximize his long-term expected reward? To answer this question, we formalize our coalition formation problem using a probabilistic model called Markov Decision Process (MDP). We will show that the MDP allows to determine an optimal formation coalition policy that define for each system state the coalition to form in order to maximize the system's long-term expected reward.

4 COALITION FORMATION

The coalition formation can be viewed as a sequential decision process. At each step of this process, the decision to form a coalition to execute a task has to be made. In the next step, another decision concerning the next task is made, and so on. The formation of a coalition changes the system's current state into a new one. As it has been shown in the previous section, the probability to transit between the system's current state and a new state *only depends on the system's current state and on the made decision*. So, this process is a *Markovian* one (Papoulis, 1984; Bellman, 1957).

A Markov decision process consists of a set of all system's states $\mathcal{S}$, a set of actions $\mathcal{AC}$ and a model of transition (Bellman, 1957). With each state is associated a reward function and with each action is associated an expected reward. In the following, we describe our MDP via: the states, the actions, the transition model and the expected reward.

4.1 States Representation

A state S of the set $\mathcal{S}$ represents a situation of coalition formation and resources consumption for all the agents. We let $S_i = \big(B_i, \langle R_i^1, \ldots, R_i^m \rangle\big)$ denote the system state at time i where:

- B_i is the set of couples task-coalition representing the coalition formation until time i: $B_i = \{(T_f, c_f) | f = 1, \ldots, i$, coalition c_f is formed to execute task T_f } ;
- $R_i^k, k = 1, \ldots, m$ is the available resources of the agent a_k at time i.

At time 0 the system is in the initial state $S_0 = (\emptyset, \langle \mathcal{R}^1, \ldots, \mathcal{R}^m \rangle)$, where $\mathcal{R}^k$ is the initial resources of agent a_k. At time n (number of tasks), system reaches a final state S_n where there are no more tasks to allocate or no more resources to execute tasks.

4.2 Actions and Transition Model

With each state $S_{i-1} \in \mathcal{S}$ is associated a set of actions $AC(S_{i-1}) \subset \mathcal{AC}$. An action of $AC(S_{i-1})$ consists in forming coalition $c \in \mathbb{C}(T_i)$ to execute task T_i and in anticipating the resources collection which can be consumed to execute T_i. We denote such an action by $Form(c, T_i)$. So, the set $AC(S_{i-1})$ contains $|\mathbb{C}(T_i)|$ actions. Being in state $S_{i-1} = \big(B_{i-1}, \langle R_{i-1}^1, \ldots, R_{i-1}^m \rangle\big)$, the application of action $Form(c, T_i)$ drives the system into a new state S_i^h which can be any state from the following states:

$$S_i^h = \big(B_i^h, \langle R_i^1, \ldots, R_i^m \rangle\big) \quad (3)$$

where :

- $c = \langle a^1, \ldots, a^q \rangle$
- $\forall h = \langle r^1, \ldots, r^q \rangle \in H_i^c$
- $B_i^h = B_{i-1} \cup \{(c, T_i)\}$
- $\forall a_k \in A, a_k \notin c, R_i^k = R_{i-1}^k$
- $\forall a^l = a_k \in c$,
$$R_i^k = \begin{cases} R_{i-1}^k - r^l, \text{ if } R_{i-1}^k \geq r^l \\ 0, \text{ if } r^l > R_{i-1}^k \end{cases}$$

In fact, there are $|H_i^c|$ possible future states because the execution of T_i by coalition c can consume one resources collection of the set H_i^c. The case where $r^l > R_{i-1}^k$ corresponds to the situation when agent $a^l = a_k$ try to execute task t_i^l and he consumes all his resources R_{i-1}^k but t_i^l is not completely performed because it necessitates more resources (r^l). The a^l's available resources is then 0 and task T_i can't be considered as a realized task. If c's agents have enough resource to execute T_i, an immediate gain equal to $w(T_i, c)$ will be received in state S_i^h. In the other case (c's agents available resources are not sufficient to completely execute T_i), the immediate gain is equal to 0. We let $\alpha(S_i^h)$ denote the immediate gain in state S_i^h, thus:

$$\alpha(S_i^h) = \begin{cases} w(T_i, c), \text{ if } \forall a^l = a_k \in c, r^l \leq R_{i-1}^k \\ 0, \text{ otherwise: } \exists a^l = a_k \in c, r^l > R_{i-1}^k \end{cases} \tag{4}$$

Furthermore, the probability of the transition from state S_{i-1} to a state S_i^h knowing that the action $Form(c, T_i)$ is applied can be expressed by the probability to consume resources collection h by coalition c, thus $Pr(S_i^h | S_{i-1}, Form(c, T_i)) = Pr(h, T_i)$. It's important to know that state S_i^h is inevitably different from the state S_{i-1}. In fact, the task to allocate in S_{i-1} was T_i, while in any state $S_i^h, h \in H_i^c$ we form a coalition to execute task T_{i+1}. In other words, being in a state S at time i, there are no actions that can drive the system to a state S' which was the system's state at time $i' \leq i$. Consequently, the developed MDP doesn't contain loops, it is a finite horizon MDP (Sutton and Barto, 1998). This is a very important property as we will show in the following.

4.3 Expected Reward

The decision to apply an action depends on the reward that the system expects to obtain by applying this action. We denote by $E(Form(c, T_i), S_{i-1})$ the expected reward associated to the action $Form(c, T_i)$ applied in state S_{i-1}. We recall that this expected reward represents what the system, being in state S_{i-1}, expects to gain if coalition c is formed to execute task T_i. A policy π to follow is a mapping from states to actions. For state $S_{i-1} \in \mathcal{S}$, $\pi(S_{i-1})$ is an action from $AC(S_{i-1})$ to apply. The expected reward of a policy $\pi(S_{i-1}) = Form(c, T_i)$ is $E(Form(c, T_i), S_{i-1})$. An optimal policy is the policy that maximizes the expected reward at each state. In state S_{i-1} an optimal policy $\pi^*(S_{i-1})$ is then the action whose expected reward is maximal. Formally,

$$\pi^*(S_{i-1}) = \arg\left(\max_{c \in \mathbb{C}(T_i)} \{E\left(Form\left(c, T_i\right), S_{i-1}\right)\}\right) \tag{5}$$

Solving equation 5 allows to determine an optimal coalition formation policy at each state S_{i-1}. To do this, the expected reward associated to action $Form(c, T_i)$ has to be defined. Defining this expected reward requires, basing on equation 2, the definition of the reward associated with each state. We define the reward $V[S_{i-1}]$ associated with a state $S_{i-1} = \left(B_{i-1}, \langle R_{i-1}^1, \ldots, R_{i-1}^m \rangle\right)$ as an immediate gain $\alpha(S_{i-1})$ accumulated by the expected reward of the followed policy (reward-to-go). We can formulate $V[S_{i-1}]$ and $E(Form(c, T_i), S_{i-1})$ using Bellman's equations (Bellman, 1957), thus:

➤ $_{i-1}$:

$$V[S_{i-1}] = \underbrace{\alpha(S_{i-1})}_{\text{immediate gain}} + \underbrace{E(\pi^*(S_{i-1}))}_{\substack{\text{reward-to-go} \\ \text{according to } \pi^*}} \tag{6}$$

$$E(\pi^*(S_{i-1})) = \max_{c \in \mathbb{C}(T_i)} \{E(Form(c, T_i), S_{i-1})\} \tag{7}$$

$$E\left(Form(c, T_i), S_{i-1}\right) = \sum_{h \in H_i^c} Pr(h, T_i) \times V\left[S_i^h\right] \tag{8}$$

where state S_i^h corresponds to the consumption of resources collection h.

➤ $_n$:

$$V\left[S_n\right] = \alpha(S_n) \tag{9}$$

Since the obtained MDP is *a finite horizon with no loops*, several known algorithms, as Value Iteration and Policy Iteration, solve BELLMAN's equations in a finite time (Puterman, 1994), and an optimal policy is obtained.

4.4 Optimal Coalition Formation

An optimal coalition formation can be obtained by solving BELLMAN's equations and then applying the optimal policy at each state starting from initial state S_0. Here, we distinguish two cases according to the execution model. The first case corresponds to the execution model where tasks must be sequentially executed in the allocation order ($T_1, T_2, \ldots, T_n$). In this case, a coalition to execute task T_{i+1} is formed at the end of T_i's execution. Let $\pi^*(S_{i-1}) = Form(c, T_i)$ be the optimal policy to apply in the state S_{i-1}. The application of this policy means that the coalition c must be formed to execute task T_i. Assuming that resources collection h has been consumed by c to execute T_i, system then reaches the state $S_i = S_i^h$ defined by equation 3. From this new state S_i, controller applies the calculated optimal policy $\pi^*(S_i)$, and so on.

The second case corresponds to the execution model where controller form all the possible coalitions before agents start the execution. In this case, after each coalition formation, controller has to anticipate the state the system will reach when executing the allocated task. Let $\pi^*(S_{i-1}) = Form(c, T_i)$ be the optimal policy to apply in the state S_{i-1}. By applying this optimal policy, coalition c is formed to execute T_i. As the execution is not immediate, controller anticipates the state S_i the system will reach when c executes T_i. This state S_i can be any state $S_i^h, h \in H_i^c$. The state that the system has big chances to reach is the state corresponding to the resources collection that can be consumed with a maximal probability. Formally, the state S_i^h

the system has a big probability to reach when c executes T_i is the state corresponding to the consumption of the resources collection h that satisfies: $Pr(h,T_i) = \max_{h \in H_i^c}\{Pr(h,T_i)\}$. From this new state $S_i = S_i^h$, controller applies the calculated optimal policy $\pi^*(S_i)$, and so on until reaching a terminal state $S_n = (B_n, \langle R_n^1, \ldots, R_n^m \rangle)$. Finally, the set B_n contains the formed coalitions and their allocated tasks.

5 CONCLUSION

Approaches that proposed solution for coalition formation problem with uncertain coalition value do not take into account the uncertain task execution and the impact of the formation of a coalition on the gain that can be obtained from the formation of another coalitions. In this paper, we addressed the problem of coalition formation in environments where the resources consumption is uncertain. We showed that in such an environment, forming a coalition to execute a task have impacts on the possibility to form another coalitions. Thus, this issue must be taken into account at each time agents decide to form a coalition. We introduced the notion of expected reward that represents what agents expect to gain by forming a coalition. The expected reward is defined as the sum of (1) what agents immediately gain if the coalition executes the task and (2) what they expects to gain by future formation. Our key idea is to view the formation of coalitions as a decision to make that provides, due to the uncertain task execution, an expected reward. Agents' aim is then to form coalition by a way that maximizes their long-term expected reward instead of real reward. The coalition formation problem has been formalized by a Markov decision process. Since the obtained MDP is a finite horizon, it can be solved in a finite time using known algorithms as value-iteration and policy iteration. After solving the MDP, the controller agent can optimally decide, for each task, which coalition must be formed. In other words, it can make optimal decisions about the coalition formation.

REFERENCES

Aumann, R. (1959). Acceptable points in general cooperative n-person games. *volume IV of Contributions to the Theory of Games*. Princeton University Press.

Bellman, R. E. (1957). A markov decision process. *journal of Mathematical Mechanics*, pages 6:679–684.

Bernheim, B., Peleg, B., and Whinson, M. (1987). Coalition-proof nash equilibria: I concepts. *Journal of Economic Theory*, 42(1):1–12.

Blankenburg, B. and Klusch, M. (2004). On safe kernel stable coalition formation among agents. In *Proceedings of International Joint conference on Autonomous Agents & Multi-Agent Systems, AAMAS04*.

Blankenburg, B., Klusch, M., and Shehory, O. (2003). Fuzzy kernel-stable coalition formation between rational agents. In *Proceedings of International Joint conference on Autonomous Agents & Multi-Agent Systems, AAMAS03*.

Chalkiadakis, G. and Boutilier, C. (2004). Bayesian reinforcement learning for coalition formation under uncertainty. In *Proceedings of International Joint conference on Autonomous Agents & Multi-Agent Systems, AAMAS04*.

Kahan, J. and Rapoport, A. (1984). *Theories of Coalition Formation*. Lawrence Erlbaum Associations Publishers.

Ketchpel, S. (1994). Forming coalition in the face of uncertain rewards. In *Proceedings of AAAI*, pages 414–419.

Klusch, M. and Shehory, O. (1996). A polynomial kernel-oriented coalition formation algorithm for rational information agents. In *Proceedings of ICMAS*, pages 157–164.

Kraus, S., Shehory, O., and Taase, G. (2003). Coalition formation with uncertain heterogeneous information. In *Proceedings of the Second International Joint conference on Autonomous Agents and Multi-Agent Systems, AAMAS03*, Australia.

Kraus, S., Shehory, O., and Taase, G. (2004). The advantages of compromising in coalition formation with incomplete information. In *Proceedings of International Joint conference on Autonomous Agents & Multi-Agent Systems, AAMAS04*.

Learman, K. and Shehory, O. (2000). Coalition formation for large-scale electronic markets. In *Proceedings of the Fourth International Conference on Multiagent Systems*.

Papoulis, A. (1984). *Signal Analysis*. International student edition, McGraw Hill Book Company.

Puterman, M. L. (1994). *Markov Decision Processes*. John Wiley & Sons, New York.

Shehory, O. and Kraus, S. (1998). Methods for task allocation via agent coalition formation. *Artificial Intelligence*, 101:165–200.

Sutton, R. S. and Barto, A. G. (1998). *Reinforcement Learning: An Introduction*. MIT Press, Cambridge MA. ISBN 0-262-19398-1.

Zlotkin, G. and Rosenschein, J. (1994). Coalition, cryptography, and stability: mechanisms for coalition formation in task orientd domains. In *Proceedings of AAAI*, pages 432–437.