Decentralized decision making approaches for Dynamic Coalition of Resource-Bounded Agents

Abdel-Illah Mouaddib, Laurent Jeanpierre

Decentralized decision-making technique for dynamic coalition of resource-bounded autonomous agents

Abdel-Illah Mouaddib and Laurent Jeanpierre
GREYC-CNRS, Computer Science Department,
University of Caen Basse-Normandie, Caen, France

Abstract

Purpose – The purpose of this paper is to extend the existing approaches of coalition formation to how to adapt dynamically the size of the coalition according to the complexity of the task to be accomplished.

Design/methodology/approach – A considerable amount of attention has been paid to the coalition formation problem to deal efficiently with tasks needing more than one agent (i.e. robot). However, little attention has been paid to the problem of monitoring a coalition during the execution by modifying it according to the progress of the accomplishment of the task. In this paper, the authors consider a coalition of resource-bounded autonomous agents with anytime behavior solving a common complex task. There is no central control component. Agents can observe the effect of the other agents' actions. They can decide whether they should continue to contribute in solving the common task or to stop their contribution and to leave the coalition. This decision is made in a distributed way. The objective is to avoid the waste of resources and time by using the same coalition along the task accomplishment while some agents become unnecessary to pursue the accomplishment of the task. The authors formalize this decentralized decision-making problem as a decentralized Markov decision process (DEC-MDP).

Findings – The paper results in a framework leading to Coal-DEC-MDP, which allows each agent to decide whether to stay in the coalition or leave it by estimating the progress on the task accomplishment.

Research limitations/implications – The approach could be extended to deal with more than one coalition.

Practical implications – Decentralized control of a fleet of robots accomplishing a mission.

Originality/value – The paper deals with a new problem of adapting dynamically the coalition to the target task and the use of DEC-MDPs.

Keywords Decision making, Programming and algorithm theory, Adaptive system theory, Intelligent agents

Paper type Research paper

1. Introduction

Situations in which more than one agent is required to undertake a problem is common in many domains such as robots moving a heavy object, fire fighter robots or robots sweeping a specific area in a short time. Such problems have been addressed from the point of view of forming coalitions. This problem has received a considerable attention in recent years (Shehory and Kraus, 1995; Abdallah and Lesser, 2004; Chalkiadakis and Boutilier, 2004). Once the coalition is formed, a little attention has been paid to monitor the coalition and to modify it during the accomplishment of the problem. Indeed, the performance of a coalition depends on how the achievement of the task progresses and sometimes the presence of all the agents is not needed. Such situations exist in many

real-world applications where initial coalitions are formed to achieve tasks and during the achievement of tasks, coalitions need to evolve by reducing or increasing their size to act better. For example, in robocup rescue, to sweep better the area by extinguishing the fires and rescuing the injured persons, coalitions are initially formed, but during the execution, fires in different locations can change differently needing more or less agents in the corresponding coalitions. Consequently, a reformation of coalitions is needed. Some previous works were concerned with this problem but they assume a central control (Shehory and Kraus, 1995). In this paper, we are interested in this problem where agents can decide to stay or to leave the coalition with no central control module (Mouaddib and Jeanpierre, 2007). This problem can be seen as a sequential decision process where, given the state of the problem, the agents can decide to stay or to leave the coalition. In addition to that, agents have limited resources and have to decide how to use these resources best (Mouaddib and Zilberstein, 1998). If they decide to stay they use their resources to help accomplish the goal of the coalition, but if they leave they preserve their resources.

In the example of robocup rescue, the longer the agent spends in extinguishing fire, the better it is (damages are reduced). The communication between agents is considered indirect because direct communication is not permanently possible and it is costly as it is in the robocup rescue context. Finally, this problem becomes more difficult when a central controller does not exist. All these characteristics motivate the work we present in this paper. To overcome this problem, we present an approach of decentralized monitoring of dynamic coalition where agents decide to stay in the coalition or to leave it.

In other words, we consider a coalition of resource-bounded autonomous agents with anytime behavior which solve a common complex task. There is no central control component. Agents can observe the effect of the actions of the others. Particularly, they observe the state of the task accomplishment, the agents in the coalition and the state of its own local resource. Agents should construct a joint policy allowing them to decide whether they should continue to contribute in solving the common task or to stop their contribution and to leave the coalition. Agents decide to stay in the coalition when the problem is still complex and the coalition is small while they decide to leave it in the opposite situations. This decision is made in a distributed way with no direct communication.

This problem can be reformulated in a more general one where there are agents and tasks, coalitions are formed to achieve tasks and during the achievement of tasks, the coalitions can become inappropriate and some modifications of their composition become necessary. We focus our study just on the monitoring of one coalition by deciding which agents should leave first. A second step of this study concerns, once agents leave a coalition how they decide to integrate a new one. This second problem is out of scope of this paper.

We formalize this decentralized control with a decentralized Markov decision process (DEC-MDP). The decision process is not considered as partially observable Markov decision process (POMDP) because each local observation of an agent is complete allowing to observe the state of the coalition (agents in the coalition), the state of the task (how many resources are needed to accomplish it) and its local resources. Such situations can be met in our robocup example.

This approach could be dedicated to many applications such as control of multi-robots and spatial coordination, coalition of retrieval information engine, military mission, etc.

The paper is organized into 6 sections. Section 2 will introduce the problem and its characteristics. Section 3 presents our approach. Section 4 gives an analysis of the approach. Section 5 presents some first experimental results and comparison. Sections 6 and 7 concludes the paper with some perspectives and a discussion on the related work.

2. Stating the problem

We have a set of agents to accomplish a complex task such as extinguishing fire by many fire fighter robots. Agents can have a continuous behavior. The longer an agent spent in contributing to accomplish a task, the better the quality of the accomplishment is. However, at the end of the accomplishment of the task, all the agents are not necessarily needed. In other words, we have a coalition of autonomous progressive processing agents accomplishing a task T. Each agent i has a limited amount of resources r_i. Initially, task T needs an amount of resources R to be accomplished. At each stage t of the accomplishment of T, each agent consumes an amount of resources Δr. Agents do not communicate directly but they observe the effect of the other agents'actions. Agents can decide to leave the coalition or to stay. The state of the coalition changes, then, over time. The problem is how to derive a policy for each agent to decide to stay in the coalition or to leave it considering the current state of the task, the coalition and its local resources. The state of the decision process should contain the information on the task, the coalition and the internal state of the agent.

We formalize this problem as a DEC-MDP. The process is not a DEC-POMDP because each agent fully observes the state of the whole system. Indeed, the state of the system is represented in our case by three parameters: the state of the task (estimated needed resources), the state of the coalition (number of agents in the coalition) and the state of the agent's resources.

The resulting decision process is observation independent and transition dependent. Indeed, the observation of an agent cannot affect the observation of another one while the action of an agent can have an effect on the decision of the other agents. Each agent can observe the state of the accomplishment of the task and the state of the coalition. We assume in the current version of the problem that agents have a complete observation about the task and the coalition. However, agents have no information on the policy followed by each agent that it is affected by the state of the task, the state of the coalition and its internal state corresponding to its available resource. We assume that the state of the task only changes with actions of agents and there is no external factor modifying this state. We consider this state fully observable. The goal of the coalition is to accomplish the task in a short time with fewer resources without the need of any central controller:

- Let $\mathbf{A}$ be a coalition of resource-bounded agents. We assume agents have different initial resources. The initial resources of each agent are supposed to be known to each other. For example, a troupe of military has a mission to arrest terrorists where each soldier knows the available equipments (guns) of each others.
- Let $\mathbf{T}$ be a complex task which is a mission assigned to agents (the arrest terrorists mission).
- Each agent $i \in \boldsymbol{A}$ has a limited resource r_i (the water reservoir for the fire fighter robots). All agents are considered with different initial resources (different reservoirs).

- Each agent i has a set of available actions: { **Leave**, **Stay** } where **Leave** is the action to leave the coalition and to preserve resources and **Stay** is the action to stay in the coalition and to consume resources.

3. The approach

Each agent observes the progress of the task achievement (for fire extinguishment task, we can observe the surface on fire representing the state of the task achievement or for military mission it is the number of neutralized terrorists), its available resources and the coalition represented by the agents present in the coalition. Consequently, the local state of an agent i is represented by task, available resource, coalition. The agents commonly observe the state of the task and the coalition but agents ignore the state of each other's available resources. The observability is collective and complete. Indeed, when all local states are gathered we assume that they identify the current world state. However, agents to make a decision need to know if the current coalition is able to solve the task. For that they need to estimate both the resources needed to solve the task and the available resources among agents. Since, we assume that an agent observes the coalition it can estimate the resources that could be consumed next. More precisely, we assume that the agents can know the types of agents in the coalition.

The approach we propose to solve this problem is in the spirit of the wide range of work on DEC-MDP similar to multi-agent team decision problem (Pynadath and Tambe, 2002), partially observable identical payoff stochastic game (Peshkin *et al.*, 2000) and opportunity cost-DEC-MDP (Beynier and Mouaddib, 2005) all of which formalize cooperative multi-agent teams as our coalition under collective common partial observability. This approach is also in the spirit of collective intelligence which is about inducing a collection of agents with no exhibition of desired global behavior.

3.1 The framework

At each t, an agent i should decide to **Leave** or to **Stay** given the state of the accomplishment of the mission **T** and the state of the coalition **Coal**. The agent tends to stay in the coalition when it is small and the task is still complex while it tends to leave it in the opposite states. Each local decision has a global effect in the coalition. Indeed, deciding to leave the coalition can tend the other agents to make a decision to stay in the coalition while deciding to stay in the coalition can push some agents to make a decision to leave. Consequently, when an agent makes a decision it should take into consideration its effect on the decision of the other agents. This decision model could be seen as a decentralized sequential decision process where agents in the coalition make local decisions by considering the state of the coalition and the tasks achievement. We formalize this decision model by Coal-DEC-MDP which consists of a tuple $< Coal, S, A, Pr, B, C >$:

- *Coal* is the coalition of progressive processing agents.
- S is a set of local states of agents.
- A is a set of joint possible actions.
- Pr is the transition function, $Pr : S \times A \times S \rightarrow [0, 1]$.
- $\mathcal{B}$ is the reward function, $\mathcal{B}$:$S \rightarrow \{0,B\}$, returning a reward B when the task is achieved and 0 elsewhere.
- C is the cost function, $\mathcal{C}$:$A \rightarrow \Re$, assigning to each action a a cost.

In the following, we develop two models to implement Coal-DEC-MDP and the assumptions under which they are valid.

3.2 The centralized decision model: multi-agent Markov decision process (MMDP)
An optimal approach is to use a MMDP, where each agent knows the global state of all the agents including their current local states (available resources). More formally, MMDP in this context consists of $< Coal, \times_{i \in Coal} S_i, \times_{i \in Coal} \{\mathbf{Leave}, \mathbf{Stay}\}, Pr, B, C >$ and an individual action consists of **Leave** or **Stay**. An element of a joint action space represents the concurrent execution of the individual actions **Leave** or **Stay** by each agent. A convention consists in prohibiting the joint action $< \mathbf{Leave}, \mathbf{Leave}, \ldots, \mathbf{Leave} >$ in some states where the task requires that the coalition has not to be distracted. The resulting model could be solved by "value iteration with state expansion" algorithm suggested by Boutilier (1999).

3.2.1 States. The state space $\boldsymbol{S} = \{s^t = (R^t, < R_i^t >_{i \in Coal(t)}, Coal^t) | t \in 1, \ldots, T\}$ is the set of states where R^t is the estimated needed resource to accomplish the task, R_i^t is the available resource of each agent i and $Coal^t$ is the state of the coalition at time t (agents composing it) and T is the horizon. The initial state is defined by $s^0 = (R_{init}^0, < R_i^0 >_{i \in \boldsymbol{A}}, Coal^0 = \boldsymbol{A}, 0)$ where R_{init}^0 is the initial resource needed to accomplish the task, R_i^0 is the initial resource available in each agent i. The coalition reaches a final state when the task is accomplished, a all resources of agents have been fully elapsed or the coalition is distracted (all agents leave it). These states s_{final} have the form of $(0, *, *)$ (for the accomplished mission), $(*, \vec{0}, *)$ the agents have no more resources and $(*, *, \varnothing)$ the coalition is distracted.

3.2.2 Actions. The action space is given by $\times_{i \in Coal} \{\mathbf{Leave}, \mathbf{Stay}\}$ representing the joint actions $\vec{a}$ of leaving or staying in the coalition.

3.2.3 Transitions.

$$Pr(s^{t+1} | s^t, \vec{\boldsymbol{a}}) = Pr((R^{t+1}, < R_i^{t+1} >_{i \in Coal^{t+1}}, Coal^{t+1}) + Pr(FailureCoalt^{t+1})$$

$$Pr((R^{t+1}, < R_i^{t+1} >_{i \in Coal^{t+1}}, Coal^{t+1}) = \Pi_{j \in Coal^{t+1}} Pr(\delta R_j)$$

$$Pr((R^t, < R_i^t >_{i \in Coal^t}, FailureCoal^{t+1}) | (R^t, R_i^t, Coal^t), \mathbf{L}) \quad =$$

$$Pr(FailureCoalt^{t+1}) \quad = \sum_{R^t > \sum_{j \in FailureCoalt^{t+1}} \Delta r_j} \Pi_j Pr(\Delta r_j) \cdot \sum_{\Delta r_j, j \in Coalt^t} Pr\left(\sum_{\Delta r_j} \Delta r_j \geq R^t \right)$$

with $Coalt^{t+1} = Coalt^t - \{j \in A | \vec{\boldsymbol{a}} \cdot j = \mathbf{L}\}$ where $\vec{\boldsymbol{a}} \cdot j$ is the composant j of vector $\vec{\boldsymbol{a}}$

$FailureCoalt^{t+1} = Coalt^t - \{j \in A | \vec{\boldsymbol{a}} \cdot j = \mathbf{L}\}$ where $\vec{\boldsymbol{a}} \cdot j$ is the composant j of vector $\vec{\boldsymbol{a}}$

3.2.4 Value function. The Bellman equation adapted to this context is given by:

$$V(s^t) = U(s^t) + \max_{\vec{\boldsymbol{a}} \in \times_{i \in Coal^t} \{\mathbf{LeaveStay}\}_{s^{t+1}}} \sum Pr(s^{t+1} | s^t, \vec{\boldsymbol{a}}) \cdot V(s^{t+1})$$

This equation can be solved by any standard algorithm based on value iteration algorithm or policy iteration algorithm. Since an agent leaving the coalition cannot re-integrate this coalition and the resource can only be consumed, the obtained MMDP is with no cycle and the joint policy is optimal.

The main drawback of this approach is its strong assumptions about the full global observability of the global state of the system by all agents and their free communication. This assumption is not valid in many real-life applications concerned with the control of dynamic coalition technology. Another approach relaxing this assumption is described below.

3.3 The decentralized decision model: DEC-MDP

3.3.1 States. The state space $\boldsymbol{S} = \{s^t = (R^t, R_i^t, Coal^t) | t \in 1, \ldots, T\}$ is the set of states where R^t is the estimated needed resource to accomplish the task, R_i^t is the available resource of agent i and $Coal^t$ is the state of the coalition at time t (agents composing it) and T is the horizon. The initial state $s^0 = (R_{init}^0, R_i, Coal^0 = \boldsymbol{A}, 0)$ is when the accomplishment of the task has not started yet, the agent does not start and the coalition contains all the agents. The agent reaches a final state when the task has been fully accomplished, the agent has no resource or the coalition is empty. These states s_{final} have the form of $(0, *, *)$ (for the accomplished mission), $(*, 0, *)$ theagent has no more resources and $(*, *, \varnothing)$ the coalition is distracted.

3.3.2 Actions. $A_i = \{\textbf{Leave}, \textbf{Stay}\}$. For a simplicity of notation, we note **Leave** by **L** and **Stay** by **S** to continue in achieving the task.

3.3.3 Transitions.

- *Leave transition.* This action has an effect on the coalition state but it has no effect on the state of the task nor on the resource of the agent. The decision process moves from $(R^t, R_i^t, Coal^t)$ to $(R^t, R_i^t, Coal^{t+1})$ when the coalition $Coal^{t+1}$ is still able to solve the task. In the other case, the decision process fails and the agent should pay a penalty provoking this failure. We name this state $(R^t, R_i^t, FailureCoal^{t+1})$ where $FailureCoalition^{t+1}$ is a coalition unable to solve the task. Consequently, the probability is as follows:

$$Pr(s^{t+1} | s^t, \mathbf{L}) = Pr(Coal^{t+1}) + Pr(FailureCoalt^{t+1})$$

where the transition with failure is given by:

$$Pr((R^t, R_i^t, FailureCoal^{t+1}) | (R^t, R_i^t, Coal^t), \mathbf{L}) \quad = \quad Pr(FailureCoalt^{t+1})$$

$$= \sum_{R^t > \sum_{j \in FailureCoalt^{t+1}} \Delta r_j} \Pi_j Pr(\Delta r_j) \cdot \sum_{\Delta r_j, j \in FailureCoalt^{j+1} \cup \{i\}} Pr\left(\sum_{\Delta r_j} \Delta r_j \geq R^t \right)$$

This equation means the probability that an agent provokes a failure from his leaving the coalition. We do not take into account the cases where the failure could occur even if the agent stayed in the coalition. Indeed, an agent computes the probability of the opportunity failure of its decision of leaving the coalition.

- *Probability of a coalition* ($Pr(Coal^t)$). The probability of the state of the coalition depends on the probability that an agent selects an action. The probability distribution on the set of actions of an agent are considered to be uniform. We mean by uniform the fact that all actions have the same probability to be selected. This probability p is given by:

$$p = \frac{1}{|A^t|}$$

Then, there is a probability p that an agent decides to leave the coalition and $1 - p$ to stay. In addition to that, the number of agents in the coalition goes along a binomial law:

$$Pr(Coal^{t+1}) = \sum_{l=0}^{l=|coal^t|} p^l.(1-p)^{|coal^t|-l}$$

Future work will concern a representation of this probability as a Boltzman distribution and its effect on the performance of the approach.

- *Stay transition.* This transition has an affect on the state of the task and the remaining resources of the agent, while the state of the coalition is observed. The decision process moves from $(R^t, R_i^t, Coal^t)$ to $(R^{t+\Delta t}, R_i^{t+\Delta t}, Coal^{t+\Delta t})$ where $R^{t+\Delta t} = R^t - \sum_{j \in Coal^t} \Delta r_j$, $R_i^{t+\Delta t} = R_i^t - \Delta r_i$ where Δr_i is consumed by level $l+1$ while $Coal^{t+\Delta t}$ is one of possible subsets of $Coal^t$. Consequently, the probability of this transition is as follows:

$$Pr(s^{t+1}|s^t, \mathbf{S}) = \Pi_{j \in Coal^{t+\Delta t}}.Pr(\Delta R_j).Pr(Coal^{t+\Delta t})$$

$$Pr(Coal^{t+\Delta t}) = \sum_{l=0}^{l=|coal^{t+\Delta t}|} p^l.(1-p)^{|coal^t|-l}$$

However, this transition can lead to the state where the agent consumes all its available resources. The new state becomes $(R^{t+1}, 0, *)$. The probability to reach this state is the probability that the agent consumes an amount of resources greater than available. This probability is:

$$Pr((R^{t+1}, 0, *)|(R^t, R_i^t, Coal^t), \mathbf{S}) = Pr(R^{t+1}).\sum_{\Delta r_k > R_i} Pr(\Delta r_k)$$

3.3.4 Reward and value functions. The Bellman equation adapted to our former:

$$\begin{aligned} V((R^t, R_i^t, Coal^t)) = {} & U((R^t, R_i^t, Coal^t)) \\ & + max(V((R^t, R_i^t, Coal^t, l), \mathbf{L}), V((R^t, R_i^t, Coal^t), \mathbf{S})) \end{aligned} \tag{1}$$

where the value of action **L** is given by:

$$\begin{aligned} V((R^t, R_i^t, Coal^t), \mathbf{L}) = {} & Pr(Coal^{t+1}) \cdot V((R^t, R_i^t, Coal^{t+1}) \\ & + Pr(FailureCoal^{t+1}) \cdot V((R^t, R_i^t, FailureCoal^{t+1}) \end{aligned}$$

and the value of action **S** is given by:

$$\begin{aligned} V((R^t, R_i^t, Coal^t), \mathbf{S}) = {} & \Pi_{j \in Coal^{t+\Delta t}}.Pr(\Delta R_j) \cdot \\ & Pr(Coal^{t+\Delta t}) \cdot V((R^{t+\Delta t}, R_i^{t+\Delta t}, Coal^{t+\Delta t})) \end{aligned}$$

The utility function U takes into account both the gain rewarded when the task is achieved and the cost of resources. It also takes into account the penalty to pay for wasted resources ($R < 0$). The reward function is given by $\mathscr{B}$. For all final states, the utility function U is thus:

- for states $(0, *, *)$ the task is achieved. The gain rewarded is:

$$U((0, *, *)) = B$$

- for states $(R < 0, *, *)$, the task is achieved but some resources of agents have been wasted. Indeed, agents consume more that necessary to achieve the task. Consequently, the gain is:

$$U(R < 0, *, *) = B - Cost(R)$$

- for states $(R, R_i, \varnothing, l)$, the coalition is empty, all the agents left the coalition and the task has not been achieved. For this situation, the agent pays a cost as follows:

$$U(R, R_i, \varnothing, l) = Cost(R - R_i) - B$$

- for states $(R, 0, Coal^t)$, the resources of the agent have been fully consumed and the task is not achieved, for that, the agent pays a cost of the remaining resources to achieve the task:

$$U((R, 0, Coal^t)) = Cost\left(R - \sum_{j \in Coal^t - \{i\}} r_j\right) - B$$

- finally, for states $(*, *, FailCoal^{t+1})$,

$$V((*, *, FailCoal^{t+1})) = U((*, *, FailCoal^{t+1}))$$

- where the agent leaves the coalition, which becomes unable to achieve the task. For that the agent has to pay a penalty that is the cost of resources needed to achieve the task, the cost of the wasted resources and the cost of the wasted benefit B:

$$U((R, *, FailCoal^{t+1})) = Cost\left(R + \sum_{j \in FailCoal^{t+1}} r_j\right) - B$$

4. Discussion and properties

4.1 Which class of DEC-MDP?

This approach is a independent observation DEC-MDP but dependent action DEC-MDP. Goldmann and Zilberstein (2004) show that this class of DEC-MDP is non-deterministic polynomial-complete. The interdependence between MDPs of agents comes from the fact that leaving action can make the coalition in a failure state while staying makes the coalition strong but may be with low utility. The balance is taken into account by agents by considering the opportunity cost of failure of the coalition. However, agents, in our approach, do not consider the positive effect of staying which can make the leaving

action of other agents possible. This gain is not considered. This means that the decision process under-estimated the expected value of the action stay. This is the main reason of the loss of optimality of our approach. However, an exact computation of the expected value of action **S** makes the complexity of the approach higher and the computation of the joint policy difficult. In fact, our approach could be seen as a set of MDPs where each MDP is solved locally at each agent, while the computation of the joint policies is known to be NEXP (Bernstein and Zilberstein, 2000). The future work will develop a better approximation of the opportunity gain of the staying action.

Goldman *et al.* show that introducing communication can in some settings improve the performance of the system. Our concern consists in introducing a resource-bounded communication in the model and its effects on the performance of the system. The local MDPs are separately solved where their interactions are considered among the cost of failed coalition which represents a cost an agent should pay when it leaves the coalition.

4.2 On the complexity of Coal-DEC-MDP and MMDP

In the previous section, we described the properties of Coal-DEC-MDP and to which class of DEC-MDP belongs. We also described the reasons leading to a loss of optimality but we present experimentally that this loss is reduced and the expected value of our approach is not far from the expected value of optimal policy that could be derived from an MMDP. However, from the complexity point of value, our approach reduces significantly the complexity since Coal-DEC-MDP is solved as a set of separate MDPs each of which is solved independently from the others as explained previously. The complexity of solving and MDP is $o(|S|^2 \cdot |A|)$ (Goldmann and Zilberstein, 2004). Considering that each agent i has a set of states $|S_i|$ and a set of actions $|A_i|$, the complexity of Coal-DEC-MDP in the worst case is:

$$o(\max_i(|S_i|^2 \cdot |A_i|))$$

Note that in our case and from the definitions of Coal-DEC-MDP in Section 3.3, $|A_i| = 2$ (two actions: stay or leave) and $|S_i| = 2^n \cdot R_i^0 \cdot R_{init}^0$. Factored state space considering the size of the task $R_i nit$, the initial resource of the agent R_i^0 and all the possible coalitions (power set of n agents). In the opposite to Coal-DEC-MDP, MMDP is a factored MDP consdering the joint actions of agents and their joint states. Consequently, the complexity of the MMDP is:

$$o\left(\prod_{i \in \{1,2,\ldots,n\}} |S_i|^2 \cdot |A|\right)$$

Experimental results will confirm these complexity, particularly Figure 3.

4.3 How to avoid a joint action $< \mathbf{L}, \ldots, \mathbf{L} >$

One of the weakness of this approach is the possibility that the joint action is $< \mathbf{L}, \ldots, \mathbf{L} >$ while the task has not been achieved. In a more general point of view, this issue is similar to ALFAROL BAR problem and Braess'paradox (Wolpert and Tumer, 2000). This situation is possible when:

$$V((R^t, R_i^t, Coal^t, l), \mathbf{L}) > V((R^t, R_i^t, Coal^t, l), \mathbf{S})$$

for all agents in the coalition. One way to avoid that agents use jointly action **L**, is to have the utility of a failed coalition with a very high penalty as $-\infty$. But with such penalty our agents tend to use only action **S** which leads to an emergent global behavior similar to a static coalition where all agents stay in the coalition along the task accomplishment. Our approach is to make the decision of leaving at the right time by adjusting the cost of failed coalition. In our current version, the cost of failed coalition is determined during an off-line learning step.

5. Experimental results

The criteria used to assess the performance of Coal-DEC-MDP consists of:

- The resource consumed in comparison with a static coalition according to the complexity of task to accomplish what we represent with the required resource for the accomplishment.
- The expected value of the policy of Coal-DEC-MDP in comparison with the expected value of MMDP policy in order to assess the quality of our approach.
- How Coal-DEC-MDP scales up in comparison with MMDP according to the size of the coalition and the task. The task considered for experiments is the fire fighting scenario inspired from the robocup rescue (simulation computation) (Rescue, 1995).

We considered different size of the fire (the space burning) and how the coalition evolves according to the evolution of the fire. The experiments are computer-simulated using the robocup rescue simulation. The experiments on real robots is left to the future work.

5.1 Comparison with an optimal static coalition

Figure 1 shows that using a static coalition consumes much more resource than needed even if this coalition is optimal. We have developed experiments by varying the

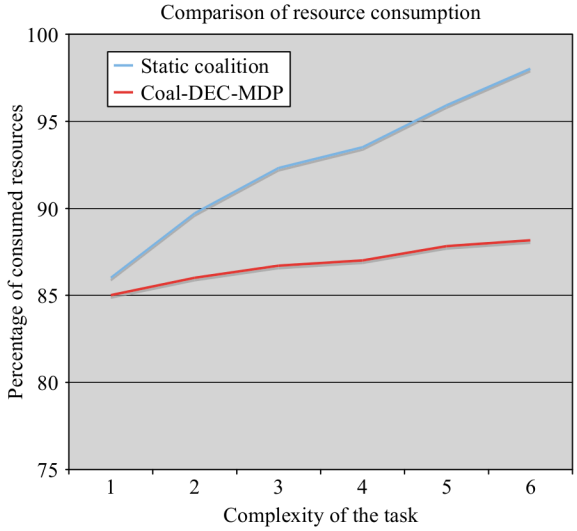

Figure 1.
Comparison of resource consumption

resource needed to accomplish the task (increasing the complexity of the task) and we can see that the percentage of consumed resources by the static coalition is higher than with our approach. This result confirms the main motivation of this work that using the same coalition from the beginning to the end of the task is resource consuming while our approach is more economical.

5.2 Comparison of expected value

Figure 2 shows the distance between the expected value of our approach and the optimal expected value. The interest of that is to assess the performance of the approach. To do so, we use a task with needed resource about 100 and we modify the size of the coalition from 5 to 35 agents. Two important results: the first one shows that the expected value of our approach has an error under 3 per cent and the second one is that the MMDP approach cannot deal with situations with tasks requiring 100 resources and a coalition with more than 20 agents, while with our approach we developed experiments up to 35 agents. This result confirms the second motivation of our approach that can better scale up than a centralized technique. The next experiments will confirm our claim.

5.3 Scalability according to the size of coalitions

Figure 3 shows how both approaches MMDP and Coal-DEC-MDP scale up. What we can see from Figure 3 is that problems with task requiring 100 and more than 20 agents cannot be considered by MMDP while our approach can.

5.4 Scalability according to the complexity of task

Figure 4 shows the same difficulty of MMDP to scale up. Indeed, in this experiment we have a coalition with 15 agents and we can see that MMDP cannot consider situations where task to accomplished requires more than 125 units of resources. In the same time Coal-DEC-MDP can scale up better and considering up to 135 and more.

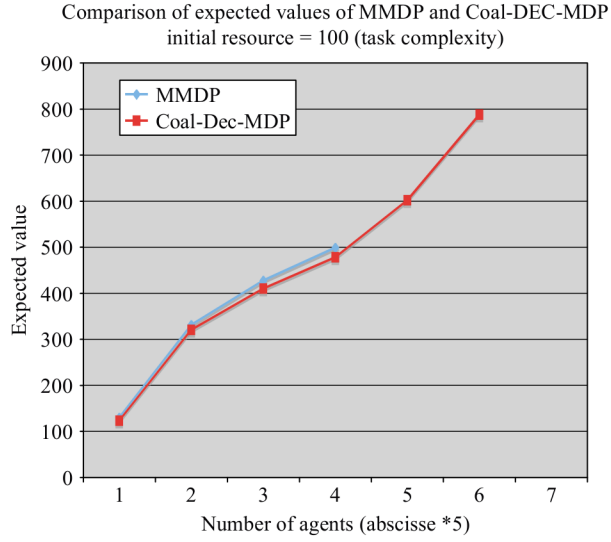

Figure 2.
Comparison of expected value

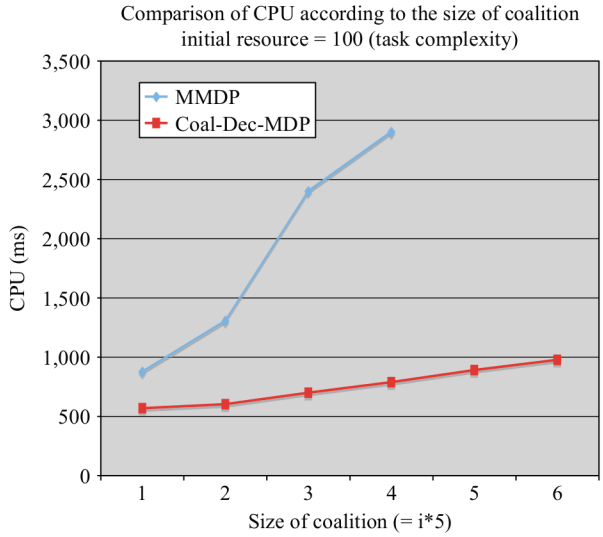

Figure 3.
CPU according to the size of coalition

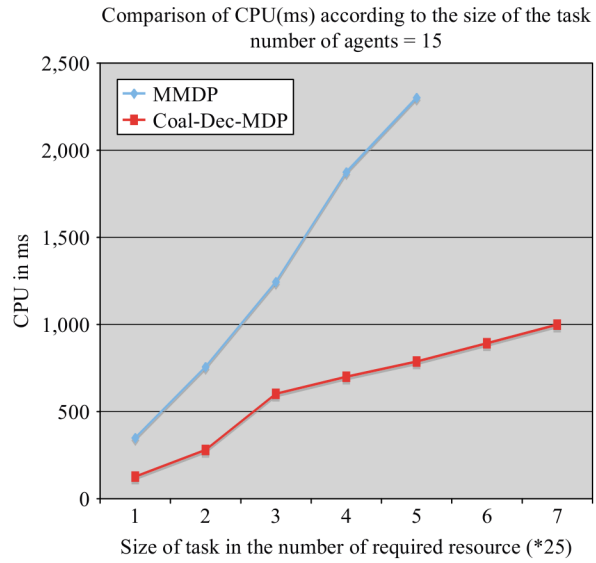

Figure 4.
CPU according to the size of the task

6. Discussion

A considerable amount of attention has been paid to the coalition formation problem to deal efficiently with tasks needing more than one agent (Shehory and Kraus, 1995; Abdallah and Lesser, 2004). However, a little attention has been paid to the problem of monitoring a coalition during the execution by modifying it according to the progress of the accomplishment of the task. In the coalition literature, we find how to form the coalition (Chalkiadakis and Boutilier, 2008; Genin and Aknine, 2008, 2010).

Some approaches consist in reforming the coalition but the reformation of the coalition could be time consuming (Shehory and Kraus, 1995; Rahwan and Jennings, 2008; Rahwan *et al.*, 2009). Coal-DEC-MDP is an important contribution from a coalition point of view since we have developed an approach to dynamically adapt the coalition according to the progress of the task. In the classical approaches, a plenty of algorithms are proposed to form coalitions or structure of coalitions and our approach contributes in completing the exiting approaches. Indeed, once the best coalition constructed for the initial objective, our approach is able to adapt this coalition to stay suitable to the progress of the initial objective. To our knowledge, this contribution is novel.

This approach uses DEC-MDP framework for which a wide range of works was dedicated last year. Most of these approaches suggest algorithms to solve DEC-MDP under different assumptions and contexts (Pynadath and Tambe, 2002; Goldmann and Zilberstein, 2004; Beynier and Mouaddib, 2005). We can use most of the obtained results but most of them do not scale up well. Our approach is based on solving MDPs separately and guaranteeing the coordination by the cost of failed coalition. We use an algorithm developed by Beynier *et al.* because they scale well. However, approaches based on common scrambling algorithm can also be used and lead to an optimal solution but they do not scale up well and only coalitions with limited size could be considered.

This approach is also related to the problem of monitoring resource-bounded reasoning. Indeed, this work is inspired from the approach developed by Hansen and Zilberstein (1996), where they present a decision process control technique allowing an anytime algorithm to stop or to continue its processing. Our approach can be seen as a generalization of this approach to multiple resource-bounded reasoning units as an anytime algorithm by deciding to stay (continue) or to leave (stop) by the difficulty that this decision should be coordinated with the other units to avoid a joint decision leave. This coordination is based on the cost of failed coalition, which is in the same case the sensitive point of this approach. In fact, this cost needs many learning steps to find the suitable function.

7. Conclusion

In this paper, we presented a formal framework of monitoring a dynamic coalition which adapts its size according to the progress of the task accomplishment. We consider the problem where the task becomes less and less difficult according to time. The resulting framework leads to Coal-DEC-MDP, which allows to each agent to decide staying in the coalition or leave it by estimating the progress on the task accomplishment. The originality of our contribution is twofold: first it is to consider a dynamic adaptation of the coalition which, to our knowledge, is novel and second the use of a robust mathematical tool to formalize this dynamic adaptation process as a distribute decision making process using DEC-MDP and solving it as a set of separate MDPs to overcome the computation complexity.

Future works will concern many directions concerning the improvement of this approach and to enrich it to deal with more complex problems such as a set of coalitions. Indeed, we are using a simple algorithm with an under-estimation of the gain based on the technique developed with Beynier *et al.* and future work will concern a better estimation of the opportunity gain and deep analysis of the cost of the failed coalition on the performance of the approach. Actually, we consider only the cost of failing a coalition but we do not consider the gain of leaving it by reducing the use of resources.

The new direction will concern our future work is the fact that the framework will be enhanced by considering many coalitions where agents can leave a coalition to integrate another one. The main modification in the framework concerns the effect of the action leave on the expected value and the probability transition. In addition to that, the current framework concerns only the action leaving a coalition and to deal with many coalitions an action of joining a coalition should be considered. This later is also let to future work.

References

Abdallah, S. and Lesser, V. (2004), "Organization-based cooperative coalition formation", *Proceedings of the IEEE/WIC/ACM International Conference on Intelligent Agent Technology, Hong Kong, China*, pp. 162-8.

Bernstein, D. and Zilberstein, S. (2000), "The complexity of decentralized control of MDPs", *Proceedings of the 16th Conference on Uncertainty in Artificial Intelligence* (*UAI*), Morgan Kaufmann, San Francisco, CA.

Beynier, A. and Mouaddib, A. (2005), "A polynomial algorithm to solve decentralized MDP with temporal constraints", *Proceedings of the International Conference on Autonomous Agents and Multi-agent Systems* (*AAMAS*), *Utrecht, The Netherlands*.

Boutilier, G. (1999), "Sequential optimality and coordination in multiagent systems", *Proceedings of the 16th International Joint Conference on Artificial Intelligence* (*IJCAI*), *Stockholm, Sweden*.

Chalkiadakis, G. and Boutilier, C. (2004), "Bayesian reinforcement learning for coalition formation under uncertainty", *Proceedings of the International Joint Conference on Autonomous Agent and Multi-agent Systems* (*AAMAS*), *New York, NY*.

Chalkiadakis, G. and Boutilier, C. (2008), "Sequential decision making in reapeated coalition formation under uncertainty", *Proceedings of the International Joint Conference on Autonomous Agent and Multi-agent Systems* (*AAMAS*), *Estoril, Portugal*, pp. 347-54.

Genin, T. and Aknine, S. (2008), "Coalition formation strategies for self-interested agents", *Proceedings of the 18th European Conference on Artificial Intelligence* (*ECAI*), *Patras, Greece*, pp. 418-22.

Genin, T. and Aknine, S. (2010), "Coalition formation strategies for self-interested agents in hedonic games", *Proceedings of the 19th European Conference on Artificial Intelligence* (*ECAI*), *Lisbon, Portugal*, pp. 1015-16.

Goldmann, C. and Zilberstein, S. (2004), "Distributed control of cooperative systems: categorization and complexity analysis", *Journal of Artificial Intelligence Research*, Vol. 22, pp. 143-74.

Hansen, E.A. and Zilberstein, S. (1996), "Monitoring the progress of anytime problem-solving", *Proceedings of the 13th National Conference on Association for the Advancement of Artificial Intelligence* (*AAAI*), *Portland, OR*, pp. 1229-34.

Mouaddib, A.-I and Jeanpierre, L. (2007), "Dynamic coalition of resource-bounded agents", *Proceedings of the 19th IEEE International Conference of Tools of Artificial Intelligence* (*ICTAI*), *Patras, Greece*, pp. 117-23.

Mouaddib, A.-I. and Zilberstein, S. (1998), "Optimal scheduling for dynamic progressive processing", *Proceedings of the 13th European Conference on Artificial Intelligence* (*ECAI*), *Brighton, UK*, pp. 499-503.

Peshkin, L., Kim, K., Meuleu, N. and Kaelbling, L. (2000), "Learning to cooperate via policy search", *Proceedings of the 16th Conference on Uncertainty in Artificial Intelligence* (*UAI*), Morgan Kaufmann, San Francisco, CA, pp. 489-96.

Pynadath, D. and Tambe, M. (2002), "The communicative multiagent team decision problem: analyzing teamwork theories and models", *Journal of Artificial Intelligence Research*, Vol. 16, pp. 389-423.

Rahwan, T. and Jennings, N. (2008), "Coalition structure generation: dynamic programming meets anytime optimisation", *Proceedings of the International Conference on Association for the Advancement of Artificial Intelligence* (*AAAI*), *Chicago, IL*, pp. 156-61.

Rahwan, T., Michalak, T., Jennings, N. and Wooldridge, M. (2009), "Coalition structure generation in multi-agents systems with positive and negative externalities", *Proceedings of the 16th International Joint Conference on Artificial Intelligence* (*IJCAI*), *Stockholm, Sweden*, pp. 257-63.

Rescue, R. (1995), "Rescue simulation", available at: www.robocuprescue.org/

Shehory, O. and Kraus, S. (1995), "Coalition formation among autonomous agents: strategies and complexity", *Proceedings of the 16th International Joint Conference on Artificial Intelligence* (*IJCAI*), *Montréal*, pp. 57-72.

Wolpert, D. and Tumer, K. (2000), "Introduction to collective intelligence", in Bradshaw, J.M. (Ed.), *Handbook of Agent Technology*, AAAI Press, Cambridge, MA.

About the authors

Abdel-Illah Mouaddib (PhD 1993) is a Professor at the University of Basse-Normandie, Caen. His research interests concern the decentralized decision-making based on DEC-POMDP, game theory, decentralized and multi-agent planning, robotics, autonomous and intelligent systems. Abdel-Illah Mouaddib is the corresponding author and can be contacted at: mouaddib@info.unicaen.fr

Laurent Jeanpierre (PhD 2002), is an Assistant Professor at the Institute of Technology at Caen. His research interests concern autonomous systems, robotics and Markov decision process.