Universitat Rovira i Virgili

Introduction to MultiAgent Systems

Activity 2

Analysis of the cooperation mechanisms between agents

Authors:
Hugo Bertiche Argila
Axel Brando Guillaumes
Julià Camps Sereix
María Leyva Vallina
Vicent Roig Ripoll

Supervisor:
Dr. Jordi Pujol

December 14, 2015

Contents

Introduction

Goal

In this report will study and analyse several cooperation mechanisms between agents in order to define which fits better our needs in the MAS project required on this course.

Description

We will discuss main issues related to the communication protocols and the cooperation mechanisms extracted from the theoretical concepts described in the lectures about cooperation mechanisms between agents. Finally we will define which of these mechanisms have been chosen as well as their functional description.

- Different cooperation mechanisms have been analysed based on the lectures given, including voting, auctions, coalitions formation, etc.
- Several advantages and disadvantages have been listed for each different cooperation mechanism, considering the architectures defined in the first activity.
- Once the different cooperation mechanisms have been analysed for each agent, we will justify the chosen mechanism for the coordination of the agents.

Finally we will detail the application of each chosen mechanism depending on the task to fulfil.

Cooperation mechanisms analysis

Partial Global Planning

In *distributed* multi-agent planning, each agent carries a model of every other participating agent's lans. Agents communicate with each other to update their local plans and the model until all conflicts have been resolved. **Partial global planning** (PGP) and **generalised partial global planning** (GPGP) are approaches based on multi-agent distributed planning.

The main principle of **partial global planning** [3][Durfee, 1988, 1996; Durfee and Lesser, 1987] is that cooperating agents exchange information in order to reach common conclusions about the problem-solving process. Planning is *partial* because the system does not (indeed *cannot*) generate a plan for the entire problem. It is *global* because agents form non-local plans by exchanging local plans and cooperating to achieve a non-local view of problem solving.

Partial global planning involves three iterated stages:

1. Each agent decides what its own goals are and generates short-term plans in order to achieve them.
2. Agents exchange information to determine where plans and goals interact.
3. Agents alter local plans in order to better coordinate their own activities.

In order to prevent incoherence during these processes, Durfee proposed the use of a meta-level structure, which guided the cooperation process within the system. The meta-level structure dictates which agents an agent should exchange information with, and under what conditions they should do so.

The actions and interactions of a group of agents are incorporated into a data structure known as a *partial global plan* (PGP). This data structure is generated cooperatively by agents exchanging information. It contains the following principal attributes:

- **Objective**: The objective is the larger goal that the system is working towards.

- **Activity maps**: An activity map is a representation of what agents are actually doing, and what results will be generated by their activities.
- **Solution construction graph**: A solution construction graph is a representation of how agents should interact, and what information should be exchanged, and when, in order for the system to successfully generate a result.

Note that PGP initial version was exclusively adapted to the *Distributed Vehicle Monitoring* problem.

Generalised Partial Global Planning

Keith Decker extended and refined the PGP coordination mechanisms in his TAEMS testbed [Decker, 1996]; this led to what he called generalised partial global planning (GPGP) [Decker and Lesser, 1995]. GPG makes use of five techniques for coordinating activities:

- **Updating non-local viewpoints**: Agents have only local vies of activity, and so sharing information can help them achieve broader views. In his TAEMS system, Decker uses three variations of his policy: communicate no local information, communicate all information, or an intermediate level.
- **Communicating results**: Agents may communicate results in three different ways. A minimal approach is where agents only communicate results that are essential to satisfy obligations. Another approach involves sending all results. A third is to send results to those with an interest in them.
- **Handling simple redundancy**: Redundancy occurs when efforts are duplicated. This may be deliberate - an agent may get more than one agent to work on a task because it wants to ensure that the task gets done. However, in general, redundancies indicate wasted resources, and are therefore to be avoided. The solution adopted in GPGP is as follows. When redundancy is detected, in the form of multiple agents working on identical tasks, one agent is selected at random to carry out the task. The results are then broadcast to other interested agents.
- **Handling hard coordination relationships**: 'Hard' coordination relationships are essentially the 'negative' relationships, thus those that threaten to prevent activities being successfully completed. Thus a hard relationship occurs when there is a danger of the agents actions destructively interfering with one another, or preventing each others actions being carried out. Hen such relationships are encountered, the activities of agents are rescheduled to resolve the problem.

In **conclusion**, the distributed sensing domain is inherently data driven: in our case, new data about recent injured people appearing on the map must be processed by the system. The main problem with the domain is to process information as rapidly as possible, so that the system can come to conclusions about the assignments and paths for rural agents and helicopters to take in time to be useful.

Advantages

Distributed multi-agent planning has several advantages. It can scale well with increased number of agents. It can also handle dynamic environments where agents might be added and removed during planning.

- Systems with **highly dynamic behaviour**:
  - All plans can be adapted to dynamic changes in the environment, therefore our satisfies **flexibility** requirements.
  - However, if an agent changes its local plan, it has to inform other agents (e.g. those that were waiting for a partial result)
- Efficiency

  - If different agents work on the same/similar sub-problems, they will notice that fact in their local plans and reassign their tasks appropriately.

Disadvantages

The drawback is that each agent needs to maintain a model of the plan of every other agent. If models are not exchanged periodically, this information can become inconsistent over a period of time.

Coalition formation

There are situations in which collaborative agents need to actively exchange messages in order to coordinate their actions and carry out a task that cannot be performed by a single agent or pursue a global goal in an efficient way. A way to solve this situation is to form coalitions, a collection of individual work in which each agent decides to cooperate with other agents by usually bringing different complementary tasks. Once this task is completed the payoff is distributed, the coalition is disbanded and agents continue to pursue their own agendas.

- The design of coalitions could be **centralised**, that is much computationally efficient, or **distributed**.
- On the one hand, the coalitions formation could be **external**, where an external agency makes the optimal decision for the coalition onces the requestor has defined the properties of coalition to the external agency (i.e. requestor will advertise the skills (the capabilities) and price (the cost)). On the other one, the coalition formation could be **internal**, where the coalitions are established by group interactions doing multi-lateral negotiation tasks. In that case, the identification of tasks can be static (where the user gives at the beginning) or dynamic (where the own agents generates its at run-time).

Disadvantages

1. It may be possible that the distribution of coalitions into agents may **not** be **homogeneous**.
2. The selected distribution may not be the best in the sense that it could be possible that two agents have taken the same coalition and there is no guarantees on the quality of the distribution.
3. In order to select who have a coalition, it is necessary a high number of messages exchanged between the agents.

Contract Net

Contract Net is a *task-sharing protocol*[1]. The cooperation mechanism that Contract Net implements is a benevolent negotiator cooperation mechanism with explicit information exchange in which agents compete to be assigned tasks to perform. The Contract Net protocol is composed by five stages:

1. **Recognition:** An agent recognises a task, necessary for accomplishing some goal, that he considers that a better option, rather than performing it in isolation, would be to get help from other agents, for performing it.

2. **Announcement:** The agent with the unsolved task generates and broadcasts an announcement for this task, which contains a full specification of the task.

   This specification must at least include:

   - A full description of the task.
   - Constraints that have to be taken into account for the task mean-goal (e.g. deadlines, quality requirements, etc.).
   - Meta-task information (i.e. relevant information concerning to the mean-goal, such as important aspects evaluated, preference on attributes, etc.).

   This information will be used for the applying agents in order to fit the desired quality expected from the requesting agent.

3. **Bidding:** Agents that receive the announcement and evaluate with a cost function their desire in being assigned the announced task. An agent bid must contain a tender, specifying the conditions in which he can execute the task. Some factors that might an agent consider for the cost function are:

   - Capability of performing the task
   - Cost vs benefits relation obtained from performing the task
   - Other tasks assigned or in bidding process (waiting to be be assigned to).

4. **Awarding:** The announcer agent must choose between the bids submitted and inform the bidding agents of the result of the process.

5. **Expediting:** The successful contractor expedites the task. This might be a recursive process, by generating sub-contracts for the contracted task (e.g. Initial-Publisher: CoordinatorAgent; First-Contractor/Second-Publisher: RuralAgentCoordinator; Final-Contractor: RuralAgent).

Contract Net describes a MAS[2] as a network of nodes, where each node is an agent. Where each agent is capable to implement both roles involved in Contract Net, manager[3] and contractor, in a dynamic way. So when a node has receives a task (this node is a contractor from the point of view of the agent who awarded him this task), it might perform it or decompose it into sub-tasks and announce them (becoming manager from the point of view of the interested agents in the sub-tasks).

Node modules of Contract Net

[1] Task-sharing: activities are distributed among the agents of the system.

[2] MAS: Multi-Agent-System.

[3] Manager: Announcer, publisher, etc.

- **Local database:** Knowledge base, information on the state of negotiations and the state of the solution of tasks.
- **Interface module:** Sends/receives messages, deals with the communication with the other nodes.
- **Task processor:** Executes the tasks assigned to the node.
- **Contract processor:** Studies new offered tasks, submits bids, formalises contracts.

Domain-Specific evaluation contracts Contract Net works with domain specific information, so the announcement message prompts to the potential contractors to use domain specific information for the task evaluation procedures. And the manager considers the bids using as well, domain specific evaluation procedures.

Efficiency Modifications

- Replacing broadcast for focused addressing or group addressing announcements. This can be implemented using learning techniques, rather than domain knowledge. Even direct contracts might be performed in some cases.
- Proactive bidding for type of tasks that a contractor can perform to his supposed future manager.

Issues for implementing Contract Net Contract Net cooperation mechanism can become a very complex communication structure, with many different cost functions implemented in all nodes, that have to be running continuously. Also the information flow can become extremely high and complex if tasks are correlated due to the constrains and performance measures.

Main issues for implementing Contract Net

- Task specification
- Evaluate and specify tasks quality
- Selection between offers
- Differentiate between offers based on multiple criteria

Advantages

- It robust to failures of agents because they can subcontract their tasks.
- Permits to subdivide the task in sub-tasks on which agents are specialists in.
- Does not require external controllers, the agents can communicated all against all and exchange rolls when necessary (easy to implement on P2P agent structure types).
- Much simpler than other methods, when the cost function is clear and the task subdivision is done externally. Less complex strategy is needed from the contractor side of the communication, due to the fact that only is required to apply the cost function and retrieve its result to the manager.

Disadvantages

- Cost functions and task division can lead to very complex functions, highly bounded to problem knowledge.
- Can involve many unnecessary communication if not controlled.
- Highly dependent on the reliability of the task subdivisions and the cost functions.
- Much more explicit communication is needed rather than other methods.

Auctions

This negotiation method is characterised by being based on competition among self-interested parties in order to allocate resources. Into this cooperation mechanism, agents might behave as two different kind of participants.

First, the **auctioneer**, which is unique and makes the "offer", according to what is being negotiated, its role might be:

- A seller who's looking to sell goods at highest possible price.
- Someone willing to subcontract out contracts at lowest possible price.
- A buyer who wants to buy a good at lowest price.

And on the other hand, the **bidders**, which are at least more than one agent, and, analogously to the auctioneer, their roles might be:

- Buyers looking for goods at lowest possible price.
- Contractors who want contracts at highest possible price.
- Sellers looking forward to sell their goods at highest possible price.

For this kind of negotiation mechanism to make sense, agents must fulfil two properties:

- Self-interested: they look forward to get highest payoffs possible.
- Rational: larger playoffs had preference over lower ones.

It is possible to find multiple variations or formats of auction mechanisms according to the process followed to achieve a successful result, the basic variations are the following:

- **English auction.** First, the auctioneer announces an opening price (minimum), then, bidders offer higher amounts successively. Highest bid wins the auction.
- **Dutch auction.** It starts with a very high bid, which is repeatedly lowered until a bidder accepts it.
- **First-Price Sealed Bid.** Divided in two phases: first, bidders submit their bid without knowledge about other bids, and second, bids are revealed and winner is chosen. Highest bid wins.
- **Vickrey auction.** A second-price sealed-bid auction. As previous format, consists on two phases: first one to bid, as before, secretly, and second one to chose winner. Highest bid wins but the winner has to pay the second highest price.

Advantages

Independently of auction format, there are some common advantages in all of them:

- There is no need of an already existing market for the product.
- Seller does not need knowledge about how much an item is worth.
- Creates competition among bidders, increasing auctioneer bargaining power.
- Faster than negotiating a price.

Disadvantages

As stated before, in order to be able to implement this kind of negotiation mechanism, agents should meet some requirements. They must be self-interested and rational. Into the context of our project, the whole system pursues a common objective, to minimise deaths and cost, therefore, bidders are not self-interested and there will always be a best choice. *Example: given an injured mountaineer, there shall be an agent which its saving cost is lesser than the rest. Therefore, no other agent shall compete for this contract.*

To conclude, the classic form of this negotiation system does not seem appropriate for our purpose. As our agents may only perform one single bid (it's cost function), therefore, it is not possible to make multiple bidding rounds, as agents shan't compete for their own interest.

Voting

Voting is a cooperative (or benevolent) explicit and negotiator cooperation mechanism which chooses the result of a negotiation using the votes given by all the interested agents. For this mechanism we will follow the social choice rule. This means we will consider as an input the preferences of each agents and as an output all the alternatives ordered according to the social preference. This social choice rule desirably will be calculable, complete, linear, anonymous, unanimous, neutral and independent of irrelevant alternatives. It is impossible to satisfy all this properties at once, so there are several voting mechanisms that will fulfil some properties while relaxing the others.

Simple protocols:

Plurality protocol: One agent, one vote. The most voted alternative wins. This mechanism is simple, efficient and follows the equality principle. Of course, this alternative has some problems, like missing information (second and next preferred alternatives) and the huge effect of irrelevant alternatives.

Anti-plurality protocol: Same as plurality, but counting the least preferred alternative. The one with less votes wins.

Best-worst voting system: Combination of the two before. Each agent gives a positive vote to its favourite alternative and a negative one to it least favourite one. The one with more points wins.

K-approval voting: Each agent selects a subset of k candidates The one with more votes win.

Based on linear orders: Each voter gives a list of all the options, ordered by preference.

Binary: The options are ordered and evaluated in pairs. The overall winner is the chosen option. This method appears to have some problems: the ordering of the alternatives is determinant, and it has a big temporal cost.

Borda: If we have n options, we give n points to the most preferred one, n-1 to the second and so on. The winner is the one with more points. Although this may seem quite fair, it has some problems. For instance, eliminating or adding alternatives may change the outcome. Also, it is very computationally expensive.

Condorcet: As before, each voter provides a list of alternatives ordered by preference. Each candidate is compared to each other, and if an option wins all comparisons it is the winner. In case of a tie, another method is necessary. This method may lead to circular ambiguities.

Complex protocols:

Linguistic votes: Each voter provides a list of all alternatives with a linguistic label (ex:Very good, good, acceptable...). Distances between linguistic labels can be defined and computed to get the overall result.

Uncertain opinions: each agent defines its opinion each alternative with an interval defined over a set of linguistic labels.

Advantages

- It can be an interesting option and not very computationally expensive if we choose a non-complex protocol for cases when we have to choose between a finite number of alternatives and have many individuals to vote for them.

Disadvantages

- For our problem, it is not applicable, since we do not have any case complex enough to apply this mechanism.
- We may have to deal with untruthful or self-interested voters.
- Complex voting protocol could be computationally expensive.

Application of the cooperation mechanisms to our agents

Cooperation among coordinators

The higher level of our cooperation mechanisms shall be used by coordinator agents. Let's briefly define their roles into the cooperation context:

- **Coordinator Agent.** For every injured mountaineer, it shall decide whether it's rural agents or helicopters responsibility to rescue him.
- **Rural Agent Coordinator.** It shall ask rural agents their corresponding rescuing cost, then, it shall report the lowest one.
- **Helicopter Coordinator.** Same as Rural Agent Coordinator, but regarding to helicopters.

According to this configuration, it is our choice to use an auction system as our negotiation mechanism. Specifically, a modified version of First-Price Sealed-Bid, in which, instead of choosing the highest bid as a winner, the lowest one is chosen.

Negotiation process

Every time an injured mountaineer appears (or under special circumstances where an injured person can't be rescued by the responsible agent), the auction starts. Rural agent coordinator and helicopter coordinator ask their respective subordinates their rescuing cost, from which they choose the lowest, which shall be their respective bid. Once bidders had sent their cost, coordinator agent shall take the minimum and make the contract with the winner.

Because of the definition of costs (1 per rural agent step and 20 per helicopter rescue), the system shall automatically take into account that it is always (as long as possible) preferable to sent rural agents than helicopters.

Special considetarions

It is necessary to consider special situations in which using a certain type of rescuing agent it is not possible. When this happens, as the winner of the auction is the lowest bidder, it shall be assigned a very high cost as bid. These situations are:

- Not idle agents.
- Not enough places at helicopter.
- Paths completely blocked by avalanches.
- Not enough time steps before mountaineer dies.
- Type of injures not compatible with idle agents.

Cooperation mechanism of RuralAgents with their Coordinator, the RuralAgentCoordinator

The structure presented in the practical exercise for this part of the communication is a hierarchical structure. For this reason we decided that RuralAgents will not be capable to communicate among themselves. Only the RuralAgentCoordinator is capable to communicate with RuralAgents, and they are only capable to respond to his queries. From the approach that we selected RuralAgents have no initiative in the communication process.

The communication system that we selected for implementing this behaviour is Contract Net, we will use the standard FIPA Contract Net protocol.

Additional Considerations

As standard Contract Net is described as allowing roles exchanges among participants. But we decided that only the RuralAgentCoordinator would be able to adopt the manager role in the communication process.

Communication flow process for RuralAgents using Contract Net

We imposed some restrictions to fit the selected protocol with our current hierarchical agents structure.

1. The RuralAgentCoordinator identifies a task that he is not able to perform, or that he prefers to delegate rather than performing it. In our case as the RuralAgentCoordinator can not perform the objectives that he will be given, because he can not perform rescues. He will reach this step from any task that he gets.

2. The RuralAgentCoordinator performs the announcement of the new task with its required information among the RuralAgents on the system.

3. The RuralAgents when receiving the announcement, from the Contract Net description, they should be able to reject an announcement, by not retrieving any answer, but since our system simulates critical life-or-dead situations and we know that RuralAgent agents are a limited resource, we want to always be able to evaluate from the big-picture of the system if a RuralAgent should or shouldn't take a contract. This way we will be able always to force a task to be taken, and we will never have lack of interested RuralAgents. As from what we have exposed our RuralAgents will be always forced to give an answer, they just have to evaluate their cost function on the new task and retrieve the obtained value.

4. When the RuralAgentCoordinator has received all the answers of the RuralAgents informed for the current contract, he will evaluate which of them is the more likely to be assigned to the announced contract. For this reason this system is highly dependent on the correctness of the cost function. This function should map the relation between when a task is difficult to perform or the schedule is crowded.

5. Once the RuralAgentCoordinator has decided who will get the contract, he just has to inform to all the RuralAgents of the result from the Contract Net process.

6. Finally, when the contractor RuralAgent finishes or fails the task he will inform the manager, that will always the RuralAgentCoordinator, of the final result of performed task. In case the RuralAgent encounters an avalanche, he will return a failure result and the contract will end.

Cooperation mechanism of HelicopterAgents with their Coordinator, the HelicopterCoordinator

Negotiation process for assign rescuer

1. First, the HelicopterCoordinator receives the data of an injured mountaineer from the CoordinatorAgent.
2. With this, the HelicopterCoordinator makes an announcement to all the HelicopterAgents with the injured person location.
3. Each HelicopterAgent bids for the rescue, making a proposal providing the following information:

   Current state: free, on duty and number of injured people on board

   Current location

   Estimation of time needed to perform the rescue:

   Helicopter idle at the hospital: 1 step to take off + distance to the injured mountaineer.

   One person on board and no other duty: Distance to the injured person.

   No-one on board and on its way to rescue a person: He will respond with two distances: The distance from him to the new injured person ($D1$) and the distance from him to the injured person who was previously planned that was going to be rescued + the distance between both people injured ($D2$).

   Two people on board: Distance to the hospital + 1 step to land + 1 step to take off + distance from the hospital to the injured person.

   Any other case: No bid.

4. Once the HelicopterCoordinator has all the HelicopterAgent bids, he will choose the most promising HelicopterAgent and, in case of tie, the Helicopter with the least amount of takeoffs and landings will be selected. On the assumption that the HelicopterAgent has responded with two distances ($D1$, $D2$), the HelicopterCoordinator will verify for another HelicopterAgent with a less distance than $D2$ and if it did not exist, first will check if it is possible to rescue the last injured person in $D2$, thus enlarging the route of HelicopterAgent, or otherwise rescue the injured person who has the shortest life time left of the old or new injured person. If the selected person is the new injured person, then the HelicopterAgent will rescue him in $D1$ steps and that would produce a suspension of the previous rescue for the former injured person. If the selected person is the former injured person, this means that we can not rescue both of them and then we will minimise the rescue time. It is important to remark that this behaviour will not only rescue closed injured persons thanks to the use of the priority list by life time left of the injured person in the Coordinator Agent.
5. The HelicopterCoordinator will bid to the CoordinatorAgent with its chosen HelicopterAgent and its proposal of rescuing plan.
6. Depending on the result in the CoordinatorAgent auction:

   If HelicopterCoordinator wins: The HelicopterCoordinator will accept the chosen HelicopterAgent's proposal, and it will assign the rescue to it. It will reject all the other proposals.

   If HelicopterCoordinator loses: The HelicopterCoordinator will answer with a rejection to all the HelicopterAgents and the negotiation will end.

7. The chosen HelicopterAgent will perform the rescue and when it's back at the hospital it will send the HelicopterCoordinator an inform notifying the success of the rescue.

Negotiation process for landing or taking off

With our specification, only one landing or take-off can be performed at a time in a cell. This affects the arrival and departure of a Helicopter from a hospital, and the operation of rescuing a mountaineer from a cell.

To control this, we will implement a simple request communication protocol, so the Helicopter wanting to land or take off will ask for permission to the Coordinator.

We will be considering the following cases:

Hospitals:

1. **Two or more helicopters want to land at the same time:** If one or several of the candidates to land have assigned another rescue, the one with less time[4]. will land first. In case of a tie, or that any of the helicopters have another duty, it is indifferent.
2. **Two or more helicopters want to take-off at the same time:** All of the candidates will then have an designated rescue to accomplish, so the one with less time will have the priority to take-off.
3. **One helicopter wants to land and other wants to take off at the same time:** the take off operation will always have priority over the landings. In case there are several helicopters wanting to take off, case 2 will be applied.

Rescues: If two or more helicopters want to rescue two or more injured people from the same cell at the same time, the one wanting to rescue the injured person with less life time will proceed first. In case of a tie, the helicopter with higher priority will be the one that has another rescue to perform later. In case of several helicopters with another pendant rescue, the first will be the one with less time to do it.

[4] Having less time means that the agent is supposed to arrive to the injured person with an smaller margin of time before he dies.

References

[1] Wooldridge, M. (2009). An introduction to multiagent systems. John Wiley & Sons.

[2] Khan Majid Ali ,Boloni Ladislau (2007). Coalition Formation and Teamwork in Embodied Agents. University of Central Florida

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

[4] Antonio Moreno, Agent communication, *Introduction to Multi-Agent Systems Notes (2015)*

[5] Antonio Moreno, Cooperation in MAS (I): emergent cooperation, PGP, GPGP, *Introduction to Multi-Agent Systems Notes (2015)*

[6] Antonio Moreno, Cooperation in MAS (II): Coalition formation, Contract Net, *Introduction to Multi-Agent Systems Notes (2015)*

[7] Antonio Moreno, Cooperation in MAS (III): Negotiation via auctions, *Introduction to Multi-Agent Systems Notes (2015)*

[8] Antonio Moreno, Cooperation in MAS (IV): Voting, *Introduction to Multi-Agent Systems Notes (2015)*

[9] Antonio Moreno, Cooperation in MAS (V): implicit methods, *Introduction to Multi-Agent Systems Notes (2015)*