

# THE LINK BETWEEN AGENT COORDINATION AND COOPERATION

Angela Consoli<sup>1</sup>, Jeffrey Tweedale<sup>2</sup> and Lakhmi Jain<sup>1</sup>

(1) *Knowledge-Based Intelligent Engineering Systems Centre, School of Electrical and Information Engineering University of South Australia.*

[angela.consoli@unisa.edu.au](mailto:angela.consoli@unisa.edu.au)

(2) *Airborne Mission Systems, Defence, Science and Technology Organisation, Adelaide.*

**Abstract:** Agent coordination is the ability to manage the interdependencies of activities between agents while agent cooperation is the process used for an agent to voluntarily enter a relationship with another to achieve a system derived goal. We describe and show the concepts of Coordinative Cooperation and Cooperative Coordination using examples. These concepts demonstrate the ability for intelligent agents to distinguish between cooperation from coordination and vice-versa. Both concepts can be integrated into a process, using a cognitive cycle to explain the interaction between coordination and cooperation. Furthermore, this paper will discuss how the coordination/cooperation loop is initialised and can be affected by Coordinative and Cooperative events. We recommend suggestions on how these concepts can be designed and implemented in a multiagent system (MAS) and introduce AC<sup>3</sup>M, which is a prototype of this cognitive loop.

**Keywords:** Multi-agent, coordination and cooperation.

## 1 Introduction and Motivation

Multi-agent systems (MASs) enlist in a variety of formats and architectures that exploit specific behaviours and reasoning. These behaviours and reasoning can be compared to the study of Classical Management Theory. A pioneer in Classical Management Theory, Taylor (1947) describes and defines the cornerstone principles of management. These principles include the theory of Research and Analysis, training, cooperation and coordination. He considers Research and Analysis in terms

of time management studies and functional analysis that are generally conducted prior to implementing new or improved work processes [1]. Training is conducted after the implementation to allow for workers to correctly utilize the new process. However, it is coordination and cooperation that are considered as the functions or set procedures that allow for smooth learning during training [1]. Therefore, it is these two functions that we suggest are subjects that are open to further automation.

This paper contains four sections. The section 2 introduces the concepts of the personification of agents and the definitions of agent coordination and cooperation. Section 3 describes the concept of a cognitive loop that exists between coordination and cooperation. These concepts are known as Coordinative Cooperation and Cooperative Coordination. Section 4 describes the implementation of these concepts and followed by a discussion of a prototype of this cognitive loop.

## 2 Coordination and Cooperation of Agents

### 2.1 Personification of Agents

The development of agents has seen a shift from autonomous goal-setting agents to MASs that are responsive to and reason with other agents. The personification of agents has allowed the development of intelligent agents to facilitate human-computer interaction that emulates human behaviour. In this way, intelligent agents are provided with more social abilities. Personification increases trustworthiness and credibility of an agent. It also increases user's engagement, for example a learning environment [2]. Personification of agents can be seen as the *coherent believable, stable and typical cluster of traits and attitudes that are reflected in the agent's behaviour* [3].

Bratman (1990) considers that Practical Reasoning is an extension of the personification of agents. He acknowledges that the practical reasoning in agents is the weighting conflicts of competing options, where each is determined from an agent's desires, values, cares and beliefs [4].

Rao and Georgeff (1995) define BDI agents where the *beliefs* are represented as information that is updated by means of actions. An agent's *desires* are its motivational state, and *intentions* are the agent's deliberative component [5]. Wooldridge (2002) further extends the BDI framework by including practical reasoning algorithms plus the concepts of conventions, commitments and plans [6].

### 2.2 Agent Coordination

The term coordination has been used in explaining concepts in the areas of psychology, sociology, biology, management, finance and information technology

[7], [8]. A working definition is *the act of managing interdependencies between activities performed to achieve a goal* [9].

Agent coordination provide coherency and focus to a MAS. Coherency is important as it aids an agent or MAS to behave and act desirably as a unit [11]. Coherency ensures agents do not conflict, waste effort and squander resources while trying to accomplish their required objectives. The process of coordination enables agents to focus on the task being performed. This unified approach to task decomposition assists the MAS to complete the common goal [10].

## 2.3 Agent Cooperation

A universally accepted definition of cooperation is *acting together with a common purpose* [12]. Cooperation can result from two specific influences; explicit and implicit influences of norms and values. The former is the influence from either a leader or referee. The latter is the influence is from the norms and values that are common between actors [13].

When an autonomous agent enters a relationship with another agent voluntarily, it is said to be cooperating. An agent will generally acquire a goal of another if there is some positive motivational outcome [14], [15].

Wooldridge and Jennings (1999) highlight two major assumptions required for agent cooperation to succeed. Firstly, because cooperation can fail, an adequate contingency plan is required and agents must initiate the social processes. Agents are required to start interaction processes that will instantiate and help accomplish cooperation [14].

## 2.4 Coordinative Cooperation

Swarts (2004) argues that coordination is based on two events: the ratification of a proposal and the revision and subsequent ratification of proposals [16]. Like contracts, any form of revision must be met with full and voluntary agreement with all parties. Such agreement is shown as cooperation. The quality of coordination depends on a group's ability to not only communicate and store their ideas, but also to share these ideas with others [16]. The latter event can be seen as Coordinative Cooperation. A working definition would be:

*“... The act of managing interdependencies of actors or objects such that an actor/object will voluntarily enter a relationship with another actor/object and adopt the goal of that actor/object for the common purpose of achieving a common goal.. ”*

For Coordinative Cooperation to be recognised, an event occurs where agents are coordinated into tasks. This event is known as a *Coordinative Event* and is defined as an action that has resulted from coordination.

## 2.5 Coordinative Cooperation

The assumption of the research focuses on the integration of coordination and cooperation in either order and postulates that the same logic is used in either process. Therefore a working definition of Cooperative Coordination is:

*“... When an agent voluntarily enters into a relationship with and adopts the goal of another agent such that the interdependencies between the agents’ activities are managed to achieve the goal...”*

Using Tulken’s (2001) approach to cooperation, cooperative coordination can therefore be pre-empted by two important cooperative characteristics (the explicit influence of a leader and the implicit influence of norms and values). Like Coordinative Cooperation, Cooperative Coordination possesses a certain event that must occur for coordination to be recognised. This event is known as a *Cooperative Event*. In this case, a cooperative event is defined as an action that has resulted from cooperation.

The main difference between Cooperative Coordination and Coordinative Cooperation is hierarchy. In addition, the assumption for both concepts is the requirement that cooperation is voluntary. If this assumption is then broadened and cooperation fails, then the voluntary relationship can also fail. It is mandatory that cooperation must continue until it can be reached and based on this argument, the same approach can be applied in forming voluntary relationships.

## 3 Implementation

### 3.1 Defining a Formal Model of Coordinative Cooperation

This research has progressed sufficiently to define a model capable of improving Coordinative Cooperation and Cooperative Coordination using the same approach as Wooldridge and Jennings (2001) when implementing Practical Reasoning in agents. Consider the following: a group of agents where a Coordinative Event has occurred.

Table 1: Algorithm of Coordinative Cooperation

```

1   Algorithm: Coordinative Cooperation
2
3   get the plan through getPlan(AG+)
4   Coordinate the first agent through Allocate(AG+, ag(1), CE)
5
6   while not succeeded(Pl) and getEvent(CE)
7
8       if D(ag(1), ag(n)) then /*Same Norms*/
9           if I(ag(1), ag(n)) then /* Same Values*/
10              if Committed(ag(1), ag(n), Pl) then
11                  /* Voluntarily agreed, thus cooperation*/
12                  ag(role) = getAgentRole(ag(n))
13                  /* Enter agent into Cooperation element & remove
14                  from set of agents*/
15                  ag(n) → CA
16                  ag(n) = ag(n+1)
17              end if
18          end if
19      end if
20
21  end while

```

There are six important steps to this algorithm. Firstly is the function `getPlan()`, where this function retrieves the plan specific for the leader agent. The second point is the function `Allocate()`. This is where the coordination occurs and sets the Coordinative Event. Thirdly is `getEvent()` where this function flags whether the Coordinative Event is still activated. Once cooperation has been achieved, the Coordinative Event is then empty.

Fourthly are the functions `D()` and `I()`. These functions determine if the desires and intentions of the leader and another agent are the same. Once these are found to be the same, the function `Committed()` is then called. This function will determine if the agent will voluntarily agree to the plan of the leader agent. If this function returns true, then it can be said that agent  $n$  has voluntarily agreed, thus is cooperating. Therefore agent  $n$  is given its role by the function `getAgentRole()` which is based on their desires and intentions and then placed into the set  $CA$ , or Cooperative Agents. Finally `succeeded()` will be set to true once all agents have been placed into the set  $CA$ . This indicates that cooperation has been achieved as all agents have voluntarily agreed to the plan of the lead agent,  $Ag^+$ .

The same approach can be used with Cooperative Coordination. This algorithm also uses the same functions as Coordinative Cooperation, but there are differences. The main difference is that the function `Allocate()` is used once cooperation has been established. This function is also used for all agents, not just for one as in the case of Coordinative Cooperation. However, the concept of cooperation is the same; all agents must have returned true on the function `Committed()`.

### 3.2 Using JACK<sup>TM</sup>

To show these two algorithms working, a development environment can be used. The development environment JACK<sup>TM</sup> allows intelligent agents to be developed and implemented within a multiagent system. The difference between JACK<sup>TM</sup> and other agent-oriented software is the ability to use events, plans and teams to model a MAS, but more importantly, it can use an agent's BDI in making decisions on these events, plans and capabilities. Therefore, Coordinative Events as well as the team's plan can be represented. Consider the case of Coordinative Cooperation using the algorithm defined:

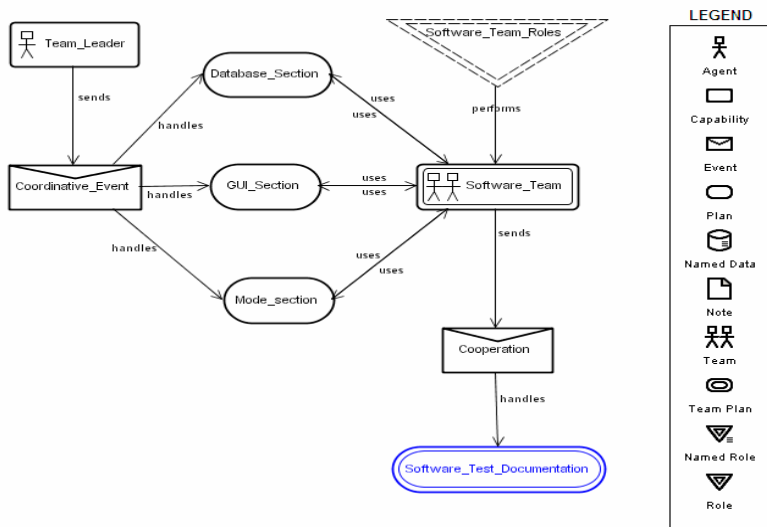


Figure 1. Coordinative Cooperation using JACK<sup>TM</sup> – Overview

## 4 AC<sup>3</sup>M – Modeling Agent Cooperation and Coordination

The concepts discussed have been implemented into a prototype called AC<sup>3</sup>M. This model uses the algorithm described and JACK<sup>TM</sup> design architecture to the concepts. Figure 2 demonstrates Cooperative Coordination. Consider a meeting of people/agents that are needed to be formed to solve a specific goal. There is a lead person/agent where this leader needs each person/agent due to their specific role and expertise that they possess. When an agent agrees to cooperate, a line is drawn from the **Team Leader Agent** to an agent. This indicates a cooperative event has occurred. A line will then be drawn from the agent to their expertise, indicating that

they have been coordinated to that task. The same logic occurs for Coordinative Cooperation; however in that case, when an agent agrees to cooperate, a line will be extended from the agent to the task, indicating cooperation has occurred.

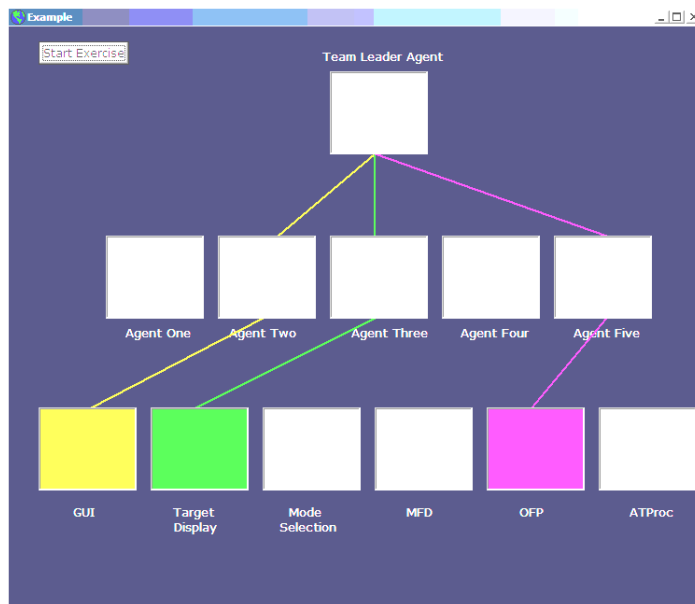


Figure 2. Cooperative Coordination in AC<sup>3</sup>M

## 5 Conclusion and Future Works

Coordination and cooperation are two abilities that can be used to assist each other, although they are commonly treated separately and allowed to compete, causing gridlock or system failure. The concepts of Coordinative Cooperation and Cooperative Coordination occur in everyday life for all forms of life. Combining both concepts into a single agent can help in the further development of intelligent agents but more importantly, in the development of systems that can emulate the human society.

The concepts of Coordinative Cooperation and Cooperative Coordination can be broadened by merging the two concepts into one. The concept of a coordination/cooperation loop has been used to demonstrate these principles in the AC<sup>3</sup>M prototype. This system can also be integrated into JACK<sup>TM</sup> so the beliefs, desires and intentions of agents can be incorporated to determine when coordination and cooperation are satisfied.

## Acknowledgements

I wish to extend my gratitude to the RADSIM team at BAE Systems Australia for not only their unconditional support, but for providing me the inspiration in establishing these concepts. I would also like to thank the KES working group at UniSA for their inspirational and somewhat stimulating conversation on all matter concerning agents.

## References

1. Taylor, F.: *The Principles of Scientific Management*, New York, Harper and Row (1947).
2. van Mulken, S., Andre, E., Muller, J.P.: *The Persona Effect: How Substantial Is It?* In: Johnson, H., Nigay, L. and Roast, C. (eds.): *Proceedings of HCI'98 - Human Computer Interaction - People and Computers XIII*, Springer-Verlag (1998) 53-66
3. Castelfranchi, C., de Rosis, F., Falcone, R.: *Social Attitudes and Personalities in Agents*, In: *Proceedings of AAAI Fall Symposium on Socially Intelligent Agents*, AAAI Press, Cambridge, Massachusetts, United States of America, (8-10 November 1997) 16-21
4. Bratman, M.E.: *What is Intention?*, In: *Intentions in Communications*, MIT Press, Cambridge, MA (1990) 15-32
5. Rao, A. S., Georgeff, M. P.: *BDI Agents: from theory to practice*, In: *Proceedings of the First International Conference on Multi-Agent Systems*, San Francisco, The MIT Press, (1995) 384-389
6. Wooldridge, M.: *An Introduction to Multiagent Systems*, Chichester, John Wiley and Sons (2002)
7. Wang, Z., Tianfield, H., Jiang, P.: *A Framework for Coordination in Multi-Robot Systems*, In: *Proceedings of INDIN 2003 – IEEE International Conference on Industrial Informatics*, (2003) 483-489
8. Borgoff, UM, Bottini, P, Mussio, P., Pareschi, R.: *A Systematic Metaphor of Multi-Agent Coordination in Living Systems*, In: *Proceedings of ESM'96 – 10th European Simulation Multiconference*, Budapest, Hungary, (2-6th June 1996) 245-253
9. Malone, T.W., Crowston, K.: *What Is Coordination Theory and How Can It Help Design Cooperative Work Systems?* In: *Proceedings of CSCW'90 – Conference on Computer-Supported Cooperative Work*, Los Angeles, California, United States of America, (1990) 357-370
10. Denti, E., Omicini, A., Ricci, A.: *Coordination Tools for the Development of Agent-Based Systems*, In: Trapp, R (ed): *Proceedings of EMCSR 2002 – 16th European Meeting on Cybernetics and Systems Research*, Austrian Society of Cybernetic Studies, Vienna, Austria, (2-5 April 2002) 671-676
11. McBurney, P., Parsons, S.: *Engineering Democracy in Open Agent Systems*, In: Omicini, A., Petta, P. and Pitt, J (eds): *Proceedings of ESAW 2003 – 4th International Workshop on Engineering Societies in the Agents World IV*, Lecture Notes in Artificial Intelligence, Springer-Verlag, London, UK, (October 29-31 2003) 66-80
12. Hua, Z.: *Study of Multi-Agent Cooperation*, In: *Proceedings of Third International Conference on Machine Learning and Cybernetics*, vol. 5, IEEE, Shanghai, China, (August 2004) 3014-3017
13. Tulken, H.: *Economic Theory and International Cooperation on Climate Change Issues*, In: *Proceedings of International Climate Policy after COP6 – Workshop on Frontiers in*



International Climate Policy Research, University of Hamburg, Germany, (September 24-25 2001)

14. Wooldridge, M.: The Cooperative Problem-Solving Process, *Journal of Logic and Computation*, vol. 9, no. 4, (1999) 563-592
15. D'Inverno, M., Luck, M.: *Understanding Agent Systems*, Springer-Verlag, Heidelberg (2004)
16. Swarts, J.: Cooperative Writing: Achieving Coordination Together and Apart, In: *Proceedings of 22nd Annual International Conference on Design of Communication – The Engineering of Quality Documentation*, ACM Press, Memphis, Tennessee, United States of America (2004) 83-89