



UNIVERSITY OF
CALGARY

Co-ordination in Multi-Agent Systems: An Overview

SENG 609.22 – Agent-Based Software Engineering
Tutorial report
December, 2002

Abstract

“The subject of multi-agents systems shifts attention from any one particular agent to a society of agents and hence, the interactions between agents in the society. There are essentially three forms of interactions: cooperation, coordination and competition. This tutorial is aimed at providing an overview of co-ordination in multi-agent systems. The importance of coordination will be discussed, a brief overview of various coordination techniques, their limitations and sample implementations will be mentioned, and finally some criteria to consider when choosing a coordination technique will be presented.”

Submitted by: Edidiong Uyai Ekaette

Course Instructor: Dr. Behrouz H. Far

Table of Contents

1	<i>Introduction</i>	3
2	<i>The Importance of Coordination</i>	3
3	<i>An overview of Coordination techniques</i>	5
3.1	Organizational Structuring	5
3.2	Contracting	6
3.3	Multi-agent Planning	6
3.4	Negotiation	8
4	<i>Criteria for choosing a coordination technique</i>	9
5	<i>Conclusion</i>	11

References

1 Introduction

In human societies we find inter-relationships of three kinds cooperation, coordination and competition. Cooperation can be seen as coordination among non antagonistic persons, while negotiation is coordination between competitive or self-interested persons. Just as there are myriads of coordination constructs in human societies such as corporations, governments, markets, committees and the like, so also there is no one coordination strategy that works well under all circumstances. This scenario is indicative of what occurs in a multi-agent environment.

Agent coordination is an agent's fundamental capability to decide on it's own actions in the context of the activities of other agents. [3] Therefore, in software agent systems populated by many, heterogeneous, agents coordination is a central concern. Advances in agent-oriented software engineering make it possible to develop complex and distributed systems. Without coordination, agents will unintentionally waste their efforts and resources often failing to accomplish their objectives, individual and collective, and possibly bringing the system to a halt. Essentially, co-ordination is a property of an intelligent agency that ensures that individual agents act in a coherent manner.

2 The Importance of Coordination

In a system of multiple agents, it is desirable to be able to consider the agents collectively, as a society. Just as in human society we have the adage that “no man is an island”, indicating the fact that no one person is sufficient on his/her own so also it applies in the agent world. With the increase in computer networks, it is becoming increasingly rare to find agents that operate on their own. The desire to have agents communicate with each other and share the burden of tasks has led to the inception of a new research and application field known as distributed artificial Intelligence (DAI). DAI draws on concepts from a wide variety of fields including computer science, sociology, economics, philosophy, and management science and can be defined as “the study , construction , and application of multiagent systems, that is systems in which several interacting, intelligent agents pursue some set of goals and perform some set of tasks.”[2] One of the tasks of DAI is to develop mechanisms and methods that enable agents to interact coherently as a unit. There are several reasons why multiple agents need to be coordinated, a few are presented below:

- Preventing anarchy and chaos - With the decentralization in agent-based systems, no one agent has a global view of the entire agency, consequently, agents only have local views, knowledge and goals, which may sometimes be conflicting. To prevent chaos, some arrangement must be put in place ensure that individual and collective goals of the agents are reached in an organized manner.[1]
- Meeting global constraints - there usually exist global constraints which a group of agents must satisfy if they are to be deemed successful. For example, a system of agents constructing a design may have to work with constraints of a predefined

budget. Similarly agents involved in monitoring and reporting faults of varying priority levels on different segments of a network concurrently. Agents need to coordinate their behavior to meet such global constraints [1].

- Using distributed expertise, resources or information coherently – Just as in human societies, different agents have specialized capabilities (expertise), and/or resources and it is desirable to be able to ask for or use these experts or resources in a conflict free manner. In an air traffic control system for example, the supervisory sub-system requires separate pieces of information for event monitoring, situation assessment, diagnosis, prioritization and response generation.[2] In such systems, the goal is to be able to employ diverse capabilities to solve problems that are not only large but also multi-faceted [2].
- Sharing the results of problem solving or planning: Sometimes during task decomposition, individual tasks are not independent of each other: some tasks can only be carried out after others have been completed, or the manner in which a task is carried out might depend on the results of another task. Moreover, during the execution of their plans, features of the environment that were not known at planning time, that unexpectedly change, can trigger changes in what an agent should do.
- Efficiency: The ability to coordinate activities, to share information and resources increases the efficiency of a system as a whole.

Coordination does not imply cooperation: An effective competitor will coordinate decisions to maximize its advantage against an opponent, such as a company timing a product promotion to undercut a rival. Nor does it imply reciprocation: An agent can coordinate its actions with another agent unaware of its presence, as when an automobile driver passes a second driver whose mind is entirely elsewhere.[3] So, coordination may be achieved without communication [4], provided the agents involved possess models of each others' behaviors. However, more frequently agents may have to communicate with each other – they have to make known their goals, intentions, results and state to other agents. Should they refuse to make known their goals (partly or totally) or intentions the situation would clearly be one of competition requiring other participating agents to postulate on the goals of the competitors. Coordination has been used to address several DAI and distributed computing issues including:

- Network coherence: maximizing how well a distributed system of agents work together;
- Task and resource allocation among agents;
- Recognizing and resolving disparities or conflicts in goals, facts, beliefs and behaviors of agents.
- Determining the organizational structure of an agent set-up, i.e. defining the roles, responsibilities and chains of authority between agents.[1]

Although the above list is not exhaustive, it gives an idea of the kinds of problems that are solved using coordination techniques.

Coordination is not only a concern to agent researchers. As trends move towards distributed computing systems, parts of open distributed systems, which are analogous to agents, need to be coordinated. Simply said, if agents are to become the new paradigm for computing systems, we cannot overlook the need for research into agent coordination techniques.

3 An overview of Coordination techniques

Nwana, Lee and Jennings [1] classified coordination techniques in four broad categories:

1. Organizational structuring
2. Contracting
3. Multi-agent planning
4. Negotiation

Following is an overview and brief critique of each category:

3.1 Organizational Structuring

The simplest way of ensuring coordination is to predefine each agent's responsibilities, capabilities, connectivity and the control flow of the system as a whole. Durfee et al define this as the pre-defined long-term relationships between agents [5]. Hierarchical organizations abound, yielding the classic master/slave or client/server coordination technique, used typically for task and resource allocation among slave agents by some master agent. This technique is implemented in a number of ways:

- The master agent plans and distributes fragments of the plan to the slaves. The slaves may or may not communicate among themselves, but must ultimately report their results to the master agent. An example of this is a network management system, where there are subordinate agents resident on each managed device which report failure cases to a manager agent.
- Blackboard negotiation exploits the classical blackboard architecture [2] to provide a coordinating base. In this scheme the blackboard's knowledge sources are replaced by agents who post to and read from the general blackboard. The scheduling agent (or master agent) schedules the agent's reads/writes to/from the blackboard. This approach may be used when the problem is distributed (diversity of problem-solving techniques), a central scheduling agent is present, tasks have already been assigned, *a priori*, to agents (independence of expertise), there is a common interaction language and there is an incremental solution generation [1,2].

The master/slave strategy exerts much control over the slave's actions and consequently on the problem solving process. However, an excess of such control mitigates against all the benefits of DAI - speed (due to parallelism), reliability, robustness, concurrency, autonomy etc [1]. In the blackboard scenario, the direct agent-to-agent communication may result in severe bottlenecks and places a constraint on the language and ontology of participating agents. The resultant blackboard systems are therefore essentially homogenous and small. This scheme is employed by Werkman in his Designer Fabricator Interpreter (DFI) system [7].

Durfee et al [6] points out that such centralized control as in the master/slave technique is contrary to the basic assumptions of DAI. It presumes that at least one agent has a global view of the entire agency – in many domains, this is an unrealistic assumption.

3.2 Contracting

A classical coordination technique for the allocation of tasks and resources among agents is the contract net protocol. In many domains, some tasks are inherently synergic, and should therefore be handled by one agent, others are better allocated to different agents. This is because agents have different capabilities and resources and therefore different costs for handling tasks. This approach to task allocation assumes a market structure with a single centralized mediator. Agents may assume any of the following two roles:

- A mediator whose task it is to break a problem into sub-tasks and search for contractors to do them, as well as to monitor the problem's overall solution.
- A contractor who performs a sub-task. Note however, that contractors may further decompose the sub-task and sub-contract them to other agents, therefore acting as managers in this case.

The process used by managers to locate contractors is outlined below:

- A manager announces a task: This will usually take the form of a broadcast message;
- Contractors evaluate the task with respects to their abilities and commitments places a bid on the task if he so wishes ;
- The manager evaluates received bids , chooses a contractor and awards the contract to it;
- The manager awaits the result of the contract.

The classical contract-net protocol assumed a multi-agent system of cooperative agents. Recent improvements have led to better results that are viable among self-interested agents as well as cooperative ones [1].

Huhns and Singh [4] noted that the contract net is best used when:

- The application task has a well-defined hierarchical nature;
- The problem has a coarse-grained decomposition;
- There is minimal coupling among sub-tasks.

The advantages of the contract net include the following

- dynamic task allocation via self-bidding which leads to better agreements;
- agents can be introduced and removed dynamically, it provides natural load-balancing (as busy agents need not bid),
- it is a reliable mechanism for distributed control and failure recovery [1].

Most of the initial limitations of the contract net protocol stemming from the assumption that participating agents are non-antagonistic and benevolent are being addressed by new methods such as negotiation protocols. However, the contract-net protocol is communication intensive and requires proper task decomposition.

This approach has been used in many applications [8,9,10,11] and

3.3 Multi-agent Planning

Another approach to co-ordination in agent-based systems is to engage the agents in the design of a plan that will guide how they work. In order to avoid conflicting actions and interactions, agents build a multi-agent plan that details all future actions required to achieve their goals and then interleave these plans to come up with a working plan.

Multi-agent planning falls in two main classes: the traditional centralized multi-agent planning class and the distributed multi-agent planning class.

In centralized multi-agent planning, there is usually a coordinating agent, whose job it is, to receive partial or local plans from all other agents, analyze these plans to identify conflict points and modify these plans, combining them into one multi-agent plan that is free of conflicts. In the final multi-agent plan, communication commands are inserted to synchronize agents' interactions appropriately [1]. Cammarata et al [12] also employed centralized multi-agent planning in a simulated air-traffic control domain. In this demonstrator, two agents (i.e. airplanes) in a potential conflict scenario chose one of them to act as the coordinating agent. The other agent forwards his plans to the coordinating agent, who then modifies his own plan to avoid the potential collision. This approach suffers from the same shortcomings as the master/slave strategy: control is central.

In distributed multi-agent planning, there are two sub classes:

- Distributed planning for centralized plans

Formulating a complex plan might require collaboration among a variety of cooperative planning specialists; just like generating the solution to any complex problem would [2]. The idea is to share the task of forming the plan, and to share the results of the sub-tasks to come up with an overarching plan. If any of these planners cannot perform its planning subtask with the partially-constructed plan, they can back track and find other choices. A plan sharing approach would require that planning agents to generate plans in parallel, and then share and merge these to converge to a final plan. This way of coming up with a plan still requires that there be proper decomposition of the plan-formation task, which may not always be suitable for all domains.

- Distributed planning for distributed plans

This is the most challenging version of distributed planning: both the planning process and the results are intended to be distributed [2]. The idea is to provide each agent with a model of other agents' plans. Agents communicate in order to build and update their individual plans and their models of others' until all conflicts are removed [1]. An exemplification of this approach is Durfee's partial global planning (PGP) approach [13]. The PGP framework gives an individual agent the ability to:

1. represent its own planned activities
2. communicate about these plans with others
3. model the collective activities of multiple agents
4. propose changes to one or more global activities to improve group performance and
5. modify its planned local activities in accordance with the more coordinated proposal.[13,2]

Generally, distributed multi-agent planning, is more complex than the centralized form of planning: it requires complex computing and communication resources. Furthermore, some planning techniques, such as PGP are gradual, and therefore their scope of applicability may be limited.

3.4 Negotiation

Negotiation is an important part of the coordination work. There is extensive literature on the topic, because negotiation is a key coordination technique used to address several DAI problems. A basic definition of negotiation is given by Bussman and Muller [14]:

“...negotiation is the a communication process of a group of agents in order to reach a mutually accepted agreement on some matter.” Sycara further points out that to negotiate effectively, agents must reason about beliefs, desires and intentions of other agent [15], and this has led to the development of techniques for the following:

- representing and maintaining belief models;
- reasoning about other agents’ beliefs;
- influencing other agents’ intentions and beliefs.

These later topics lead to the usage of all sorts of artificial intelligence and mathematical techniques including, game theory, logic, belief revisions distributed truth maintenance etc. Negotiation as a technique is used in all of the other approaches already discussed. This overview is not exhaustive; it however, provides a good background for understanding negotiation techniques. Negotiation techniques can be classified in the following broad categories:

- game-theory based negotiation;
- plan-based negotiation
- Miscellaneous negotiation approaches.

Game-theory based negotiation:

Developers of agent-centered negotiation mechanisms focus on the following problem: “Given an environment in which my agent must operate, what is the best strategy for it to follow?” [2]. When agents interact, there is the necessity to think of the other agent’s preferences and strategies. So, whenever one is confronted with a situation in which an agent’s rational decision-making depends on his expectations about what one or more other agents will do, and theirs similarly depends on what he will do, game theory becomes very useful to the analyst [16]. The key concepts in a game-theoretic approach include utility functions, a space of deals, strategies, and negotiation protocols. A deal is a joint plan between agents that would satisfy all of their goals. Each deal had a utility value attached to it. Utility is defined as the difference between the worth of achieving a goal and the price paid in achieving it. Each agent wants to maximize its own utility. The agents discuss a negotiation set, which is the set of all deals that have a positive utility for every agent. There is an implicit assumption that each agent in the negotiation is an expected utility maximiser.

Because, agents are not constrained to be truthful at all time, the negotiation process can be viewed as a two-stage process: the actual negotiation and the execution of the joint plans [17]. Game theoretic techniques also have to take time into consideration during the negotiation process, because time influences the outcome of the negotiations.

The game theoretic approach still has to be further extended to incorporate situations of partial or incomplete knowledge, when the agents do not have full knowledge of their opponent’s preferences or of the full pay-off matrix. Much of the work presumes two agents negotiating, though there is research into n-agent negotiations [1*]. As more agents get involved in the negotiation process, the pay-off matrix becomes bigger and

may become intractable and incomplete. It is hard to see how the game theoretic technique can be used reliably in real applications without exploring uncertainty resolution techniques for multi-agent systems [16].

Plan-based negotiation

Under the section on distributed planning, we considered agents that have differing local plans and need to coordinate their plans to avoid conflicts in the system. The decision of which agent, should wait for another is fairly random and arbitrary in some cases, but need not be. A large amount of work in negotiation is concerned with resolving these kinds of conflicts. Sometimes, for example, the decision of which agent should revise his local plans, can be based on the space of alternative possibilities open to that agent. This is indicative of the level of flexibility that agent has. There are several other negotiation plans open to agents, but almost all of these techniques assume the agents to be honest about the importance of their goals and their level of flexibility.

Miscellaneous negotiation Approaches

Because human interaction always seems to require some form of negotiation, many researchers draw from human negotiation strategies.

One of the more useful papers on negotiation is from Bussmann and Muller's negotiating framework for cooperating agents [14]. Drawing from socio-psychological theories of negotiation, they evolve a cyclic negotiation model which is both general and simple. The general strategy is that the negotiation begins with one or more agents making a proposal. Other agents evaluate the proposal, and come up with a list of their preferences which have been violated by the proposal. Everyone updates his knowledge about the preferences of the others and the negotiation cycle resumes.

There are several other negotiation techniques that are similar to Bussmann and Muller's using the blackboard strategy for communication and sharing knowledge for example, Werkman proposes a knowledge-based model of an incremental form of negotiation [7]. Werkman's Designer Fabricator Interpreter (DFI) model is based on various human models of negotiation.

4 Criteria for choosing a coordination technique

Having looked at all the techniques above, even though it is not an exhaustive list, I find it would be useful to list some criteria that ought to be considered before a technique is adopted for any multiagent system development. The list below can be extended and made more comprehensive:

1. The number of agents

Ignoring every other factor, such as heterogeneity and distributivity in the environment, we find that as the number of agents in the system increases, the amount of communication between these agents becomes a central issue, whether or not control is centralized or distributed. If each agent interacts with every other agent, the number of paired interactions will grow quadratically with the number of agents and the complexity of the coordination problem will increase exponentially: If every agent could choose among b actions, each potentially having different impacts on the

other agents, the space of possible combinations would be b^n for n agents [3]. There is no one strategy that will absolutely deal with the problem of quantity, however some strategies will fare better than others. For example, instead of having every agent ask every other agent for help when they need it, a middle agent that has information about the services offered by all the agents in the domain can easily route a request to the appropriate agent.

2. The classes of agents

By classes of agents, I mean the types of agents. In addition to having different goals, beliefs and expertise, agents can also have various communication languages, ontologies, or internal architectures. It is important to take the heterogeneity of the system into consideration when choosing a coordination technique. The Blackboard strategy for example will not suffice in an environment where interacting agents have different communication languages and ontologies.

3. The versatility of each agent

Versatility here refers to the ability of agents to flexibly decide for itself what goals to pursue at a given time and how to pursue them. Coordinating specialized agents that single-mindedly perform a given task, places less strain on the coordination strategy. To handle issues of versatility, the technique must be able to handle run-time plan coordination. One of such techniques is the PGP technique [13].

4. The task environment – distributive or not

If agents are collected together such that tasks originate from one point and are coordinated from that one point, the coordination technique is central. In other tasks environments when agents are highly distributed and tasks are inherently distributed among them, the coordination technique is strained by uncertainty issues, such as - who is or should be doing what [3].

5. The task environment – Presence of uncertainty

Uncertainty persists in many ways in any multi-agent system: There may be uncertainty in the data and sensors received by agents, the pay-off matrix of an agent in a game-theoretic environment may be incomplete, during negotiations, self-interested agents may not necessarily be honest etc. The coordination strategy should anticipate either implicitly or explicitly, the range of conditions under which the solution it provides will be followed. Durfee [3] rightly stated that, “In a task environment in which minor deviations can lead to severe consequences, finding robust solutions can be imperative”.

6. Overhead limitations

Some coordination strategies are more cost intensive than others. Cost here, would include computation requirements, communication overload, time spent, and so on. The pay-off between cost and quality is a decision that faces the developer of a multi-agent system. For example, to make high-quality coordination decisions will place a high demand for information exchange among agents, the absence of which will result in decisions being made without complete or accurate information.

5 Conclusion

This paper has provided an overview on coordination in agent-based systems. Coordination has been seen to be very important in the design and implementation of any multi-agent system. Without good coordination, many of the benefits of the multi-agent paradigm will disappear. There has been extensive research done in this area, and this research continues even now. The challenge I think is to be able to provide a framework that guides the choice of a coordination technique based on the properties of any particular multiagent system and to devise a means of evaluating the complexity and suitability of each technique.

From this paper, it is possible to identify four major components which must be present in any comprehensive coordination technique:

1. It is important to try to anticipate all potential actions that agents in a system might possibly take, and to impose restrictions that prohibit undesirable interactions. Such “social laws” ensure that the multi-agent system will not become unmanageable and erratic. This is the same even in human societies.
2. There must be flexibility in the structure so that agents can operate in a dynamic environment: in other words the laws need not be too restraining.
3. There must be social structures or etiquette that, describe how agents should behave towards one another when engaged in coordination processes- this is really like protocols..
4. Agents must have sufficient knowledge and reasoning capabilities to exploit both the available structure and the flexibility.

However, I do not think that a comprehensive technique is what is needed for every system. There are systems in which the structures do not require excellent coordination techniques (the structure and organization is rigidly defined), and tradeoffs should be made where applicable.

References

- [1] Nwana H. S., Lee L., and Jennings N. R., "Co-ordination in Multi-Agent Systems." Lecture Notes in Computer Science; 1198 (c) Springer-Verlag Berlin Heidelberg 1997 .
- [2] G. Weiss, ed., Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence, MIT Press, Cambridge, Mass., 1999.
- [3] Durfee Edmund H., "Scaling Up Agent Coordination Strategies," University of Michigan. IEE July 2001.
- [4] Huhns M and Singh M. P., "CKBS-94 Tutorial: Distributed Artificial Intelligence for Information Systems." Dake Centre, University of Keele. 1994
- [5] Durfee E. H, Lesser V. R. and Corkill D. D., "Coherent Cooperation among Communicating Problem Solvers", IEE Trans Comput, 11, pp 1275-1291 (1987).
- [6] Durfee E.H. ,Lesser V. R., and Corkill D. D., "Trends in Cooperative Distributed Problem Solving", IEEE Knowledge and Data engineering, 1, No.1, pp 63-83, 1989.
- [7] Werman, K. J: "Knowledge-based model of negotiation using shareable perspectives", Proc of the 10th Int. Workshop on DAI, Texas. 1990.
- [8] Parunak H V-D, "Manufacturing Experience with the Contract Net", in Gasser L and Huhns M (Eds): "Distributed Artificial Intelligence 2", Morgan Kaufmann 1989.
- [9] Sandholm, Tuomas. "An implementation of the contract net protocol based on marginal cost calculations." In Proc. Of the National Confr. On AI, pages 256-262, Washington, D.C., July 1993.
- [10] Sandholm, Tuomas. "Negotiation among self-interested Computationally Limited Agents. " PhD thesis, University of Massachusetts, Amherst, 1996. Available at <http://www.cs.wustl.edu/~sandholm/dissertation.ps>.
- [11] Sandholm, T W., "Contract types for satisficing task allocation: I theoretical results.", In AAAI Spring Symposium Series: Satisficing Models, Pages 68-75, Stanford University, CA, March 1998.
- [12] Camarata S, McArthur D and Steeb R: "Strategies of Cooperation in Distributed Problem Solving", Proc of the 8th Int. Joint Conf on Artificial Intelligence, pp 767-770. 1983
- [13] Durfee E. H., Lesser V. R., "Partial Global Planning: A Coordination Framework for Distributed Hypothesis Formation", © IEEE 1991.
- [14] Bussmann S and Muller J., " A Negotiation Framework for cooperating Agents", in Deen S. M. (Ed): Proc CKBS_SIG, Dake Center, University of Keele, pp 1-17. 1992.
- [15] Sycara K: "Multi-agent Compromise via Negotiation", in Gasser L and Huhns M (Eds): "Distributed Artificial Intelligence 2", Morgan Kaufmann 1989.
- [16] Ekaette E.U. "A Survey on uncertain programming and uncertainty management techniques in agent-based systems", University of Calgary. Aug. 2002.
- [17] Zlotkin G and Rosenschein J. S., "Blocks, Lies and Postal Freight: The Nature of Deception in Behavior", Proc of the 10th Int Workshop on DAI. 1990.