Künstl Intell (2012) 26:87–90
DOI 10.1007/s13218-011-0120-0

DISSERTATIONEN UND HABILITATIONEN

Engineering Coordination

A Methodology for the Coordination of Planning Systems

René Schumann

Published online: 21 September 2011

Abstract The reuse of code and concepts is an important aspect when developing multiagent systems (MAS) and it is a driving force of agent-oriented software engineering (AOSE). In particular, the reuse of mechanisms like coordination is fundamental to support developers of MASs. In this article we address the selection of effective and efficient mechanisms to coordinate plans among autonomous agents. The selection of coordination mechanisms is, up to now, not covered sufficiently. Therefore, we present the ECo-CoPS approach that defines a structured process for the selection of coordination mechanisms for autonomous planning systems, where the local autonomy and existing planning systems can be preserved.

Keywords Multiagent system · Coordination · Planning systems · Selection process

1 Introduction

The reuse of code and concepts is an important aspect when developing multiagent systems (MAS) and it is a driving force of software engineering in general and in AOSE in particular. Probably, in the field of AOSE the most recognized types of reuse are agent frameworks and FIPA standards [2]. In this article we address the reuse of higher level concepts. In particular we focus on the reuse of coordination mechanisms for plans among autonomous agents.

R. Schumann (✉)
Business Information Systems, University of Applied Sciences Western Switzerland (HES-SO), Rue de Technopôle 3, 3960 Sierre, Switzerland
e-mail: rene.schumann@hevs.ch

This work is of particular interest for someone who has to coordinate different planners. If those systems already exist, a typical approach would be to wrap them in a new layer where we apply a coordination mechanism. Therefore we address software agents and their coordination as a suitable technique, because coordination has been intensively studied in this field. Each agent represents a particular planner, that generates a sub-plan of the overall plan.

We can find those situations in companies or in collaboration of companies, where different planning systems are in use, each solving a particular sub-problem. Due to the ongoing research in the field of ambient intelligent systems the need to coordinate intelligent heterogeneous systems sharing the same environment will be an issue in other domains, as well.

We will consider an example from a manufacturing site of a company. Goods have to be produced, packed and transported. Therefore three different planning problems have to be solved. For each problem a planning system exists that should not be replaced or modified. The dependencies between those problems is shown in Fig. 1. A coordination in this scenario ensures that all local planning problems, as well as, the overall manufacturing plan of the company are feasible. Each planning system is represented by a software agent that acts on their behalf. These agent have to interact according to a specific coordination mechanism to achieve the coordination of the planning systems.

Coordination is an interesting topic for researchers coming from the fields of distributed artificial intelligence, eco-

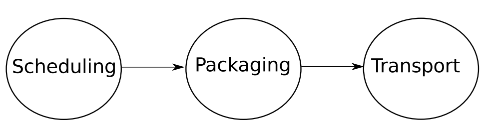

Fig. 1 Dependencies between sub-problems

nomics and game theory. While various coordination approaches have been presented, it has to be stated that there exists no method that leads to optimal results for any given coordination problem. Consequently, a suitable coordination mechanism has to be identified for each coordination problem individually. Up to now, there exists no process for the selection of a coordination mechanism, neither from the field of engineering of distributed systems nor from AOSE.

To overcome this drawback we present the ECo-CoPS approach that supports the developer selecting an effective and efficient coordination mechanism for a given situation. What the developer needs is a process how to select a mechanism. To define and establish a process three steps are necessary: the process has to be defined, it has to be validated, and finally it has to be evaluated. In the first step the process is designed and detailed. In the second step, the validation, it is shown that the process is feasible and can support the developer. In the final step the process has to be empirically evaluated on a sociologically sound base.

The ECo-CoPS approach is described below and in more detail in [6]. Additionally it has been validated by applying it to three case studies, two from logistics [6] and one from ambient intelligent systems [7]. Here we focus on the definition of the ECo and CoPS processes.

2 Reuse of Coordination Concepts in AOSE

The reuse of coordination mechanisms has been supported by mediating agent infrastructures like TuSCoN [5]. In these infrastructures coordination mechanisms are embedded into the environment as coordination artifacts that can be used by agents. This is a valuable approach for reusing coordination mechanism. But as pointed out before there exist no dominant coordination mechanism for all situations. So, even if the agent infrastructure can provide a large set of different coordination artifacts, the selection problem is still an open issue. The ECo-CoPS approach is orthogonal to coordination artifacts. The usage of coordination artifacts faces some problems for the coordination of autonomous planning agents. It is necessary for the agent to reveal all planning relevant information to the infrastructure, to enable the artifact to coordinate the plans. This is a serious limitation for the application of coordination artifacts, because they limit the autonomy of the agents. We consider here that determining future actions is a vital part for the agents' autonomy.

Other researchers have addressed the reuse of organization forms [3] and interaction protocols [1] in MAS. They have built up classification schemes according to specific criteria and created a repository of existing solutions that have been classified by their criteria. By classifying the current situation at hand possible organization forms or protocols can be retrieved from the repositories. This approach alone is not sufficient selecting a suitable coordination mechanism, but we are going to integrate it in our work presented below. Also it allows for further automation e.g. by case-based reasoning tools, that recommend candidate mechanisms [8].

3 The ECo-CoPS Approach

The main idea of the ECo-CoPS approach is to preserve existing planning systems, and to enable the coordination among them by using software agents. Each agent represents one planning system. An agent can interact with the planning system but does not have insights of the system. The ECo-CoPS approach addresses the selection of a coordination mechanism for inherently distributed autonomous planning systems. This selection is guided by the ECo (**E**ngineering **Co**ordination) process. An important step of the ECo process is the prototypical implementation of candidate mechanisms. Therefore the implementation step is supported by the CoPS (**Co**ordination of **P**lanning **S**ystems) process and the CoPS framework. Both guide and ease the implementation of coordination mechanisms.

3.1 The ECo Process

The ECo process consists of five steps that can be executed in an iterative manner. These steps are: model the coordination problem, elicit coordination requirements, select appropriate coordination mechanisms, implement selected approaches, and evaluate candidate mechanisms to identify the best one for the current situation. The process is outlined in Fig. 2. In the modeling phase the coordination problem and each planning problem (sub-problem) is modeled with a specific level of detail to describe the necessary criteria for local and global feasibility of plans and aspects to be

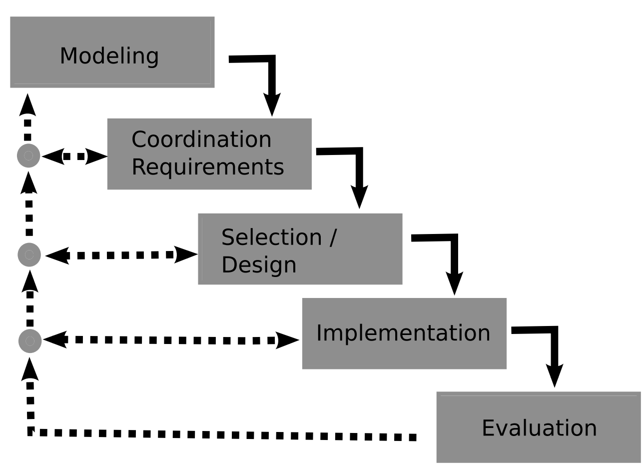

Fig. 2 The ECo process model

optimized. In our example the different planning problems: scheduling, packing and transportation have to be modeled. Moreover, the dependencies between those problems have to be specified. Based on the modeling techniques used for planning systems, this step uses a set-constraint based approach to model the coordination problem at hand.

In the elicitation step requirements are identified that have to be fulfilled by a coordination mechanism to be applicable for the given coordination problem. In our example one coordination requirement is to ensure the consistency between the plans, which means that the dependencies between them have to be checked. These requirements can be formally described using the terms and concepts introduced in the modeling step. Therefore a formal validation to check if a requirement is satisfied by a coordination mechanism is possible if needed.

The third step is the selection phase. Coordination mechanisms have to be identified that can satisfy the coordination requirements. The result of this step is a set of candidate mechanisms that can effectively coordinate the planning systems, i.e. they are applicable in the sense that they satisfy the coordination requirements. If this set is empty a suitable mechanism has to be designed.

To identify this set of mechanisms it is necessary to reduce the set of candidate mechanisms, because checking if a mechanism satisfies the coordination requirements can be time consuming. Therefore existing mechanisms have been characterized in accordance to a classification scheme defined after several case studies. Suitable classes can be retrieved by describing the scenario at hand. This idea has been proven valuable in the related work discussed above and can help to reduce the effort of a detailed analysis. To characterize a mechanism/scenario the following questions have to be answered:

- Does an allocation problem have to be solved?
- Are the local objective functions comparable?
- Are the different planning problems homogeneous?
- Does a global objective function exist?
- Is information hiding necessary?
- Do interdependencies exist between sub-problems?

The retrieval of probably interesting classes of mechanism is currently done by hand. An automated support of this step can be of particular interest if the classification scheme is going to be extended, the classes in the repository grow, or a particular repository for mechanisms is used. The coordination mechanisms contained in the retrieved classes need to be investigated in more detail. Their compliance with the coordination requirements have to be checked. This can be time consuming, because typically it has to be shown that a specific requirement is satisfied by a mechanism respecting the features of the situation at hand.

In the particular example we use a simple method that ensures feasibility of plans, which is result passing, i.e. the plans for scheduling, packing and transportation are generated according to the sequence given by the dependency graph in Fig. 1. Another more sophisticated approach is to allow the agents to negotiate about shifting the production and packaging of some orders to create synergies between different orders. In particular this negotiation-based approach bases on the idea that requirements towards local plans and plan suggestions are exchanged between agents that represent dependent planning systems until they have reached an agreement.

To evaluate the effectiveness of the remaining candidates they have to be implemented. Implementation is supported by the CoPS process and framework. These implementations can be used to test the efficiency of the mechanisms with real-world like data, to identify the most efficient one. In the evaluation for the case study used in this paper it could be shown that the negation-based approach leads to plans inducing lower costs and also the plans generated by this approach are more stable, in the sense that the variance is lower. The entire case study can be found in [6].

3.2 The CoPS Process

The CoPS process is a sub-process of the ECo process. It structures the decision making during the implementation of a coordination mechanism. The CoPS process addresses decision making on the global level, i.e. among all agents, and on the local level, that have to be made for each agent individually. The CoPS process is shown in Fig. 3. The global process step is the definition of commonly accepted conversation protocols. It is global in the sense that all agents have to agree on the same conversation protocols to allow for an effective coordination. All other steps of the CoPS process have to be done for each agent, and effects only the local decision making; therefore they are referred here as local. First, each entity has to define its conversation policy. This policy is defined by the organizational entity responsible for the planning step. It is implemented by the agent that represents the particular planning system. A conversation policy is a set of "restrictions on communication based on the content of the communicative act" [4, p. 121].

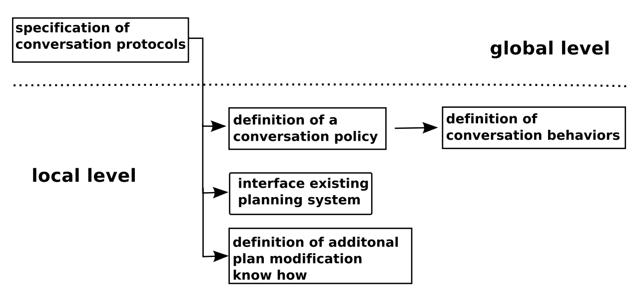

Fig. 3 Overview of the CoPS process

Within a conversation strategy a planning entity has to encode which concessions it is willing to make to whom, for instance. A conversation strategy is implemented in a number of conversation behaviors. A conversation behavior is executed in a particular state of the conversation, i.e. a state in the conversation automaton of one of the participants of the conversation.

The access to the local planning system can either be done directly, if the planning system is part of the agent, or by using integration techniques, like web-services, for instance. It might be useful to add local planning-relevant knowledge to the agent, so that the agent can modify the input data of the planning system in a meaningful way. This can lead to reduced interaction times between the planning system and the agent, as the agent can modify the input data in a way that enables the planner to operate more efficiently.

4 Conclusion

In this paper we have presented a methodology for the selection of coordination mechanisms among autonomous planning agents, to facilitate the reuse of existing coordination mechanisms. The ECo-CoPS approach allows for maintaining the local autonomy of the agents, as well as, keeping existing planning systems in use. The ECo-CoPS approach has been applied to three different case studies. In [6] we have shown that the ECo-CoPS approach is advantageous for coordination problems from the logistic domain. In [7] we showed that the ECo-CoPS approach can also be transfered and be useful in the domain of ambient intelligent systems. Also we think that the ECo process can be used as a blueprint for the selection process of other concepts in MAS.

References

1. Bussmann S, Jennings NR, Wooldridge M (2003) Re-use of interaction protocols for agent-based control applications. In: Giunchiglia F, Odell J, Weiß G (eds) Agent-oriented software engineering III (AOSE 2002), Bologna, Italy, vol 2585. Springer, Berlin, pp 73–87
2. FIPA—Foundations for Intelligent Physical Agents: FIPA standard specifications (2002) http://www.fipa.org/repository/standardspecs.html. Accessed: 02/04/11
3. Jonker CM, Treur J, Yolum P (2005) A formal reuse-based approach for interactively designing organizations. In: Luck M, Odell J (eds) Agent-oriented software engineering V (AOSE 2004), vol 3382. Springer, New York, pp 221–237
4. Kagal L, Finin T (2004) Modeling conversation policies using permissions and obligations. In: Eijk RMv, Huget MP, Dignum F (eds) Workshop on agent communication (AC2004), New York
5. Omicini A, Ricci A, Viroli M, Castelfranchi C, Tummolini L (2004) Coordination artifacts: environment-based coordination for intelligent agents. In: AAMAS 2004, vol 1. IEEE Comput Soc, New York, pp 286–293
6. Schumann R (2010) Engineering coordination: a methodology for the coordination of planning systems. PhD thesis, Institute of Informatics, Goethe University. http://publikationen.ub.uni-frankfurt.de/volltexte/2010/8143/. Accessed: 21/03/11
7. Schumann R (2011) Engineering coordination in future living environments. In: Dörner R, Krömker D (eds) Self integrating systems for better living environments (Sensyble 2010). Shaker Verlag, Aachen, pp 125–132
8. Shepperd M (2003) Case-based reasoning and software engineering. Springer, New York, pp 181–198, Chap 9

René Schumann received his diploma degree in 2004 from the University of Oldenburg. From 2004 to 2007 he worked at OFFIS Institute for Information Technology, Oldenburg. From 2007 to 2010 he was employed as a researcher in the information systems and simulation group at the Goethe University Frankfurt. He received his Ph.D. in 2010. From 11/2010 until 05/2011 he was a DAAD Researcher at the National Institute of Informatics, Tokyo, Japan. In June 2011 he joined the Applied Intelligent Systems Lab at the University of Applied Science Western Switzerland (HES-SO).
His research interests are multiagent systems, in particular the fields of coordination, distributed constraint reasoning, and the engineering of multiagent systems (AOSE). Additionally he is interested in techniques for solving planning and scheduling problems.