A Holonic Multi-agent Co-ordination Server

Andreas Gerber, Christian Russ
German Research Centre for Artificial Intelligence (DFKI)
Stuhlsatzenhausweg 3, 66123 Saarbrücken
e-mail: {agerber, russ} @dfki.de

Abstract
This paper presents a holonic co-ordination server which is made up of a recursive hierarchy of three different agent types: On the top stands an agent providing matchmaking services and representing the co-ordination server to the outside. It passes incoming requests to a subordinated agent type that is equipped with co-ordination mechanisms such as auctions, negotiations and coalition formation mechanisms. For each incoming request, this agent spawns an instance of a third agent type which executes the protocol of the co-ordination mechanism chosen in the request. The holonic structure of the co-ordination server helps to reduce complexity while allowing a high grade of adaptability and flexibility.

Motivation

In today's markets, business entities are forced to interact with other market participants flexibly in order to stay competitive. The trend towards supply webs (Laseter 1998) shows that market participants are forced to form flexible business partnerships that require more interactions with more autonomous business entities than ever before.

Agents offer the advantage that they can automatically and flexibly react to changes in the environment since they can autonomously perform tasks on behalf of their users. Since the interactions between business partners in virtual enterprises or in electronic markets together with their interrelations can get too complex for humans to handle efficiently encapsulating business entities (e.g. buyers/sellers, suppliers/producers/retailers) within agents has been suggested.

For handling the interactions efficiently, an agent-based co-ordination infrastructure is needed that provides a set of co-ordination services (e.g. matchmaking services) as well as co-ordination mechanisms (e.g. auctions, coalition formation mechanisms, profit division mechanisms). It brings together potential partners with common, or complementing, goals and enables them to co-ordinate their activities by using the provided co-ordination mechanisms.

In this paper, we will describe the structure of a holonic co-ordination server that fulfils these requirements.

Holonic Multi-agent System

In many domains a task that is to be accomplished by an agent can be hierarchically decomposed into particular subtasks. Thus, the task's completion may require the distribution of the subtasks to some subagents as well as the combination of their results. To model this combined activity the concept of *holonic agent* or *holon* has been introduced (Gerber, Siekmann, and Vierke 1999).

The concept is inspired by the idea of recursive or self-similar structures in biological s stems. Analogous to this, a holonic agent consists of parts which in turn are agents (and maybe holonic agents). The holonic agent himself is part of a whole and contributes to achieve the goals of this superior whole. Along with agents, holonic agents share the properties of autonomy, goal-directed behavior and communication. But a holonic agent possesses capabilities that emerge from the interaction of subagents. A holon may have actions at its disposal that none of its subagents could perform alone.

Three forms of association are possible for a holon: first, subagents can build a loose federation sharing a common goal for some time before separating to regulate their own objectives. Second, subagents can give up their autonomy and merge into a new agent. Third, a holon can be organized by a special agent named *head* of the holon which moderates the activities of the subagents and represents the holon to the agent society for all interaction processes with other agents.

Multi-Agent Systems (MAS) are well suited for dealing with complex tasks (e.g. planning tasks) that can be divided into several subtasks. Each subtask is then represented by an agent that autonomously solves the task. MAS exhibit the features of stability and robustness since one agent can often take the role of an other agent that has been delayed or suspended for some reason. Furthermore, agents in MAS are characterized by their capability of exchanging messages to achieve coordination and cooperation.

MAS consisting of holonic agents are called *holonic multi-agent system (H-MAS)* (Bürckert, Fischer, and Vierke 1998). In a H-MAS, autonomous agents may join others and form holons. But they are also free to reconfigure the holon, to leave the holon and act autonomously, or to form new holons with other agents. Holonic MAS share the advantages of MAS, but also provide additional advantages. Holonic agents are highly flexible at solving their tasks, having the ability to deal with inevitable change, since they are self-organizing and decentralized. Finally, as an advantage of analysis and building a system, holonically structured MAS exhibit a mapping of conceptual view and operational implementation. The implementation reflects the conceptual structure.

Therefore, it seems to be a natural way to represent many organization forms in e-commerce (e.g. virtual enterprises, supply webs) by a H-MAS because the holonic agent-based structuring supports their flexible and fluid formation as well as their dissolving. Furthermore, as the partners of a supply web the sub-agents of a holon have to pursue at least one common goal and thus show a common goal-directed behavior.

A Holonic Co-ordination Server

The co-ordination server provides to the business agents in a supply web or electronic marketplace a generic platform with services, such as auction mechanisms, that enable them to co-ordinate their interrelated activities in a decentral fashion.

Our co-ordination server is designed as an agent that can be easily accessed by other agents for registration, requests, etc..

The architecture of this holonic co-ordination server is built up of a three level hierarchy. Each level contains a distinct class of agents which are specialized in different ways. The co-ordination mechanisms are encapsulated into the second and third levels. In the following, we describe the functionalities of the three agent types that make up the hierarchy (see Figure 1).

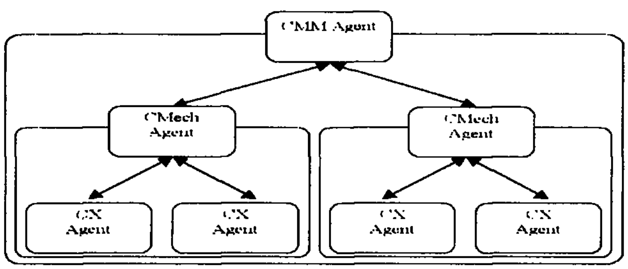

Figure 1. The holonic co-ordination server

The CMM Agent

The *co-ordination matchmaker agent (CMM)* constitutes the top of our hierarchy. It represents the co-ordination server to the agents in the external market environment. The CMM agent holds up-to-date data of all co-ordinations run by subordinated agents and thus can match requests about what service/item can be offered by using which co-ordination mechanisms. To get the information about the co-ordination processes it stands in contact with its subordinated *co-ordination mechanism agents (CMech).*

The co-ordination mechanism agents are a meta-control authority for all running, and planned, co-ordination processes which are performed by so-called service providing *co-ordination execution agents (CX)* . The CMech agents get the current information about the running co-ordination processes (start, termination, variations, etc.) from the CXs and propagate this information together with the information about the planned co-ordination processes u pwards to the CMM agent.

The CMM agent couples requesting agents (which want to find or start an co-ordination for a given item) with the CX agents. It stores the information about the CX agents in a database and updates the information depending on the status reports of the CMech agents. After it has matched the request with its current database, the CMM agent simply returns a ranked list of relevant CX or CMech agents to the requesting agent. The requesting agent then has to contact and negotiate with the relevant CX or CMech agents to get the services/items it desires. This direct interaction between the requesting agent and selected CX or CMech agents is performed independently from the CMM agent (see figure 2). This avoids data transmission bottlenecks, and even if a failure of the CMM agent does occur, running co-ordinations would still work Furthermore, the CMM agent has additional functionalities, e.g. to build a history about co-ordinations, their results and the satisfaction of customers over time and goods.

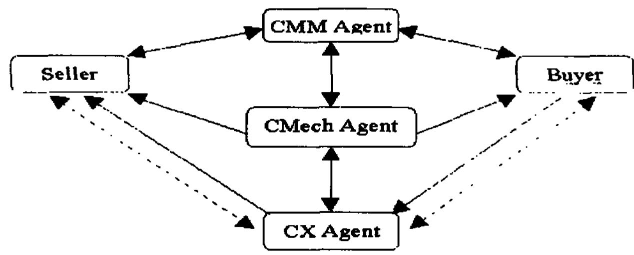

Figure 2. Agent relationships by using the co-ordination server

The CMech Agent

The next hierarchy level of our co-ordination server is built up of CMech agents which are created with regard to the kind of mechanisms the co-ordination house wants to offer. There are CMech agents for all kind of co-ordination mechanisms, e.g. for the English-, Dutch- or Vickrey-auction (Fischer, Russ, and Vierke 1998). If e.g. many auctions are running at the same time, one single agent can no longer handle all the requests in an efficient way. Thus, the CMech agent does not execute any co-ordination on its own. Instead it creates new CX agents which are speciall designed to deal with a specific instance of a co-ordination mechanism imposed by the CMech agent, e.g. an auction for a certain good. If the existing CX agents are overloaded the CMech agent can start a new CX agent which handles the next set of co-ordination tasks what results in load balancing through agent networks (Gerber 2000). Each of the CMech agents administrates a variable number of CX agents which perform the same co-ordination mechanism. Thus, these agents can be bundled into one holon per mechanism. The holon is represented by an CMech agent which is the head of the holon and co-ordinates the distribution of the co-ordination tasks to its subagents.

The CX Agents

The third level of the hierarchy contains CX agents which are able to execute multiple co-ordinations of the same mechanism. Only if such an agent is overloaded, another agent of this type is created and may be started at/on another place/computer in the co-ordination server network, guaranteeing a good performance. If an co-ordination has ended the CX agent having performed the co-ordination task will push the information about that co-ordination task to the CMech agent and to all participants of that co-ordination task. The CMech agent forwards this information to the CMM which stores the information for later use, e.g. statistic computations. After all co-ordinations of the CX agent have ended this agent might be terminated if it is not needed anymore.

The realisation of this co-ordination server as a holonic MAS has several advantages. Firstly, the holonic structure allows to group independent, autonomous CMech and CX agents, probabl owned by different system designers and located at different servers, by their functionalities to one virtual auction server with clearly specified information flows between its components. This conceptual structure is open as well as flexible and enables as a second advantage the flexible adaptation to dynamic changing workloads. a property called *operator abstraction*: When many auctions of the same type are started one agent might not be able to handle all these tasks alone. Other agents of the same type could already running on other servers could join the virtual auction holon or could be created on different servers. Then, the tasks can be split up between them in order to realize load balancing. To retain control over all spatially as well as organisationally distributed auction tasks superior co-ordination instances are necessar at the first and second level of the virtual auction server in order to enable a global control over these tasks.

Interactions between the Agents of the Three Levels of the Co-ordination Server

We will describe the interactions between the agents b taking a look at an auction that could be initiated by a seller and executed by the co-ordination server (see figures 2 and 3):

In the following we want to describe ho an auction can be started and performed by an agent society. A seller starts an auction by sending an appropriate message to the CMM agent. This message contains information about the item to be auctioned off and in addition to that the seller's preferences concerning the co-ordination/auction mechanis to be used and the auction monitoring services.

Then the CMM agent triggers an appropriate CMech agent which spawns a CX agent for executing the auction.

After that the CMech agent transmits the address of the spawned CX agent back to the superior CMM agent.

The CMM stores the information about the started auction in its database and sends the address of the CX agent executing the auction to the seller, if the seller wants to have auction monitoring (see the dotted line in figure 2) access to the CX agent. Otherwise the CX agent automaticall sends information about the auction state in intervals to the seller.

While executing an co-ordination task, CX agents forget that they are part of a holon and act as single agents. Thus it is possible for co-ordination participants to contact the directly, this avoids the need to parse all messages from the head of the holon (CMM) down to the CX agents. Therefore, the CX agents do not only have to push the results of executed co-ordination tasks to their upper CMech agent but also to all participants in that co-ordination.

Hence a buyer that is by chance interested in buying this specific ite asks the CMM agent if such an item is currently auctioned off. The CMM agent says yes to this question and sends the address of the corresponding CX agent stored in its database to the buyer. After that the buyer registers at the CX agent and monitors as well as bids in the auction.

After the auction has finished the CX agent reports its outcome to the superior CMech agent that informs the bidders about it.

The upper two levels of the hierarchy consist of holonic agents whose lifecycles are not limited to any point of time. The lowest level includes only agents which do not have to exist all the time and can be created and terminated dynamically.

Generally, by using a holonically structured hierarchy, all incoming and outgoing messages have to be transmitted to the head (Gerber, Siekmann and Vierke 1999), i.e. the CMM agent in our server structure. But sending all messages to the head of the holon results in a very narro bottleneck for the system, whereby all co-ordination tasks are slowed down. Thus our CMM agent is only used for performing co-ordination and information tasks within the holon.

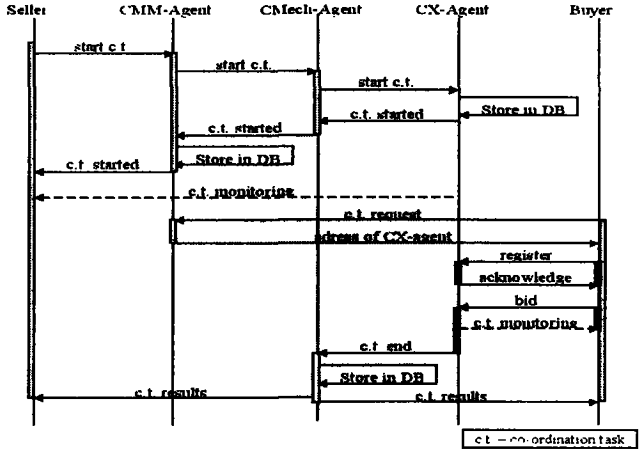

Figure 3. Interaction between sellers, buyers and the co-ordination server agents

Market-based Supply Web Co-ordination Mechanisms for the CMech Agents

Several market-based allocation mechanisms are suitable for the co-ordination of supply web activities, e.g. the allocation of a set of tasks to a set of supply chain agents. Such mechanisms are the simulated trading algorith (Bachem, Hochstättler, and Malich 1992) , the Vickrey auction (Vickrey 1961), and the matrix auction for multiple heterogeneous item (Gerber, Ruß, and Vierke 1999) . We compared their suitability, using the allocation of transportation tasks to a set of truck agents within the multi-agent scheduling system MAS-MARS (Fischer, Müller, and Pischel 1996) in a co-operative and a competitive setting.

Market-based Co-ordination Mechanisms

Simulated Trading Simulated trading (ST) (Bachem, Hochstättler, and Malich 1992) is a randomized algorith that realizes a market mechanism where the participating contractors optimize a task allocation by successivel selling and buying tasks. Trading is done in several rounds, each of which consists of a number of decision cycles. In each cycle, the participants submit one offer to sell or buy a task. At the end of each round the stock manager, the central coordinating instance, tries to match the sell and buy offers of the contractors such that the costs of the global solution decrease.

In the sealed-bid second-price or Vickrey auction (VA) (Vickrey 1961) every bidder submits a sealed bid for the item to be auctioned off to the auctioneer. The bidder who submitted the best bid receives the item for the second highest bid made. In contrast to traditional auctions like the English and Dutch auctions, this procedure is truth revealing, i.e., it forces the bidders to submit bids that equal their true valuations for the items.

Matrix auctions (MA) (Gomber, Schmidt, and Weinhardt 1998) are - in contrast to ST and VA - applicable for the simultaneous assignment of multiple items or

tasks to bidders. The auctioneer announces in a matrix-k-auction (MA-k) the k offered items to the bidders who, in turn, calculate their valuations for each potential combination of items and report them to the auctioneer.

From the transmitted bids or reported valuations of the bidders the auctioneer identifies the optimal allocation of all k items. The price for each assigned subset of items equals the second-highest bid in the matrix column for that set of items. Like the Vickrey auction, the matrix auctions are truth-revealing.

Suitability of the Mechanisms for Supply Web Co-ordination Tasks

In the co-operative setting, the truck agents belong to one company and have no interest in optimizing their individual profits. The auctioneer agent represents the company and ries to minimize the overall cost per order (see Figure 4). An analogous setting in the supply web domain would be the intraorganisational optimization of the supply flow by a retail company.

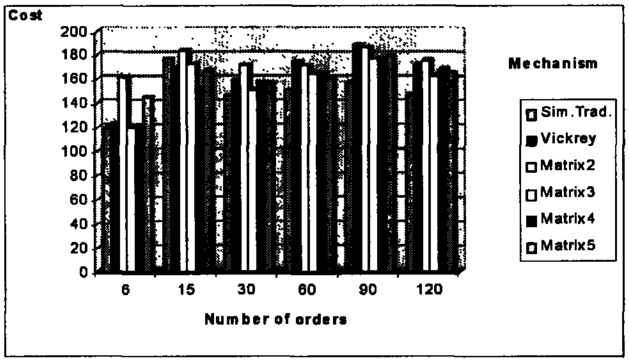

Figure 4: Overall cost per order

In the competitive setting, the truck agents represent independent self-interested forwarders. They compete on an open electronic market for the assignment of transportation tasks in order to optimize their route plans and in this way their surplus per order. The auctioneer agent does not represent a shipping company but acts as an independent broker. You can think of an analogous setting in the supply web domain where some independent companies form a short-term supply path, a virtual company or co-operate with other companies within the framework of a strategic alliance.

In (Gerber, Ruß, and Vierke 1999) , we empiricall examined the scalability and tractability of the market-based co-ordination mechanisms mentioned above b comparing their processing time and allocative efficienc for order sets of different size. Thereby, the allocative efficiency of the mechanisms is measured in terms of cost, and surplus.

In the cooperative setting - where cost per order is the crucial issue -the simulated trading procedure produces generally the best results with tractable effort. The simulated trading procedure is proved to be most efficient for large order sets where much optimization can be done. MA-3 and MA-4 perform slightly better than the remaining protocols. Hence, ST would be the protocol of choice for the auctioneer. Nevertheless, MA-3 achieves acceptable results as well.

In the competitive setting - where individual agents tr optimize their benefit - the individual surplus of the agents taking part in the co-ordination process has to be compared (Figure 5). In this setting ST is not applicable. Here, the MA-2 procedure ensures a maximal payoff for the self-interested agents and outperforms dominantly all other mechanisms, followed by the Vickrey auction.

In summary, especially ST, MA-2, and MA-3 are ver suitable for supply web co-ordination tasks and therefore are integrated into our supply web co-ordination architecture.

With respect to the tractability of the mechanisms, the evaluation showed that ST, VA, MA-2, and MA-3 can be rated as scalable, while MA-5 and, for large order sets, MA-4 do not provide better results, but loose tractability, indicating that matrix auctions where six or even more items are traded in parallel are not expected to be particularly efficient.

Due to the space restriction we could not present all results; more results of the evaluation can be found in (Gerber, Russ, and Vierke 1998) .

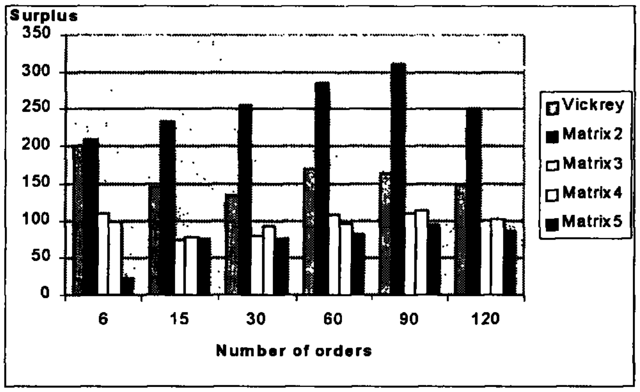

Figure 5: Overall surplus per order

Conclusion & Further Work

In this paper, we have presented a three-part agent-based co-ordination infrastructure for supply webs. We have already implemented this infrastructure as well as suppl chain agents in Java. First results in coordinating the suppl activities of the agents by the use of the co-ordination server and several co-ordination mechanisms, as the matrix auction (Gerber, Russ, and Vierke 1998), simulated trading (Vierke 2000) and the extended contract net protocol (Vierke 2000), are very promising.

Our main goal is to extend the developed infrastructure b integrating more co-ordination mechanisms such that the agents can co-ordinate their activities more eff iciently b using them. Therefore, our future activities will mainl consist in the development of a set of agent-based co-ordination and negotiation mechanisms as well as their integration in the co-ordination infrastructure. The developed mechanisms are intended to support the configuration and co-ordination of distributed business processes as well as the (re)allocation of r esources and tasks within supply webs. Moreover, we will investigate their effects on supply chain execution by applying them to simulation scenarios.

The developed agent technology could be directly applied in related areas, e.g. for implementing electronic markets and virtual enterprises. Our work has been supported by the European Union and the SAP subsidiary SAP Retail Solutions.

References

(Bachem, Hochstättler, and Malich 1992) A. Bachem, W. Hochstättler, and M. Malich. Simulated Trading: A Ne Approach For Solving Vehicle Routing Problems. Technical Report 92.125, Mathematisches Institut der Universit¨at zu K¨oln, Dezember 1992.

(Bürckert, Fischer, and Vierke 1998) H.-J. Bürckert, K. Fischer, and G. Vierke. Transportation Scheduling with Holonic MAS — The TeleTruck Approach. In: Proceedings of the Third International Conference on Practical Applications of Intelligent Agents and Multiagents (PAAM'98), 1998.

(Fischer, Müller, and Pischel 1996) K. Fischer, J. P. Müller, and M. Pischel. Cooperative transportation scheduling: an application domain for DAI. Journal of Applied Artificial Intelligence. Special issue on Intelligent Agents, 10(1), 1996.

(Gerber, Russ, and Vierke 1998) C. Gerber, C. Russ, and G. Vierke. An empirical evaluation on the suitability of market-based mechanisms for telematics applications. Technical Report TM-98-02, German Research Center for Artificial Intelligence, 1998.

(Gerber, Ruß, and Vierke 1999) C. Gerber, C. Ruß, and G. Vierke. On the Suitability of Market-Based Mechanisms for Telematics Applications. In Proceedings of the International Conference on Autonomous Agents (Agents'99), 1999.

(Gerber, Siekmann, and Vierke 1999) C. Gerber, J. Siekmann, and G. Vierke. Flexible Autonomy in Holonic Agent Systems. In Proceedings of the 1999 AAAI Spring Symposium on Agents with Adjustable Autonomy, 1999.

(Gerber, Siekmann, and Vierke 1999) C. Gerber, J. Siekmann and G. Vierke. Holonic Multi-Agent Systems, DFKI Research Report RR-99-03, ISSN 0946-008X, Kaiserslautern 1999.

(Gerber 2000) C. Gerber. Self-Adaptation and Scalabilit in Multi-Agent Societies, PhD Thesis, Universität des Saarlandes, Saarbrücken, 2000.

(Gomber, Schmidt, and Weinhardt 1998) P. Gomber, C. Schmidt, and C. Weinhardt. Efficiency incentives and computational tractability in the coordination of multi-agent systems. In Proceedings of the Workshop Kooperationsnetze und Elektronische Koordination, 1998.

(Laseter 1998) T. M. Laseter. Balanced Sourcing: Cooperation and Competition in Supplier Relationships. Jossey-Bass, ISBN: 0787944432, October 1998.

(Vickrey 1961) W. Vickrey Counterspeculation, Auctions, and Competitive Sealed Tenders. In: Journal of Finance 16: 8- 37, 1961.

(Vierke 2000) G. Vierke. TELETRUCK - A Holonic Multi-Agent System for Telematics. PhD Thesis, Universität des Saarlandes, Saarbrücken, 2000.