Distributed Product Development of a Fuel-Injection System Using Multi-Agent

Zuhua Jiang [1], Shiwei Fu [2], Burkhard Lege [3]

[1] School of Mechanical Engineering, Shanghai Jiao Tong University,
Huashan Road 1954
Shanghai 200030, P. R. China

[2] Kingdee international Software Group
Shenzhen 518057, P. R. China,

[3] Konstanz University of Applied Sciences, Brauneggerstr 55
D-78462 Konstanz, Germany

Abstract. Multi-agent modeling has emerged as a promising discipline for dealing with decision making processes in distributed information system applications. One of these applications is the modeling of distributed design and analysis processes which can link up various designs and simulation processes to form a virtual consortium on a global basis. This paper proposes a multi-agent cooperative framework for the development of a fuel-injection-system including a fuel-injection-system and consisting of more than 90 parts. The meta-model of management agent and actor agent for the development of the fuel-injection-system is presented, and the architecture of the distributed multi-agent system for the development of a fuel-injection-system is discussed. The prototype system and some key agents in the distributed product development are introduced.

1 Introduction

Design is increasingly becoming a collaborative task among designers or design teams that are physically, geographically, and temporally distributed [1]. The complexity of some products, e.g. the fuel-injection-system, makes it hard for a single designer to complete the whole design task. The development of a fuel-injection-system includes the product design, the structural analysis and the performance simulation. Design is a team effort in which groups of designers with different intent and background knowledge work together.

The characteristics of the development of a fuel-injection-system are independent team work and global control, as well as necessary negotiation between the teams. Most tasks can be worked on in different teams respectively. But all results should be controlled for the adherence to the given technological requirements and cost

Jiang Z., Fu S. and Lege B. (2006).
Distributed Product Development of a Fuel-Injection System Using Multi-Agent.
In *Proceedings of the 2nd International Workshop on Artificial Neural Networks and Intelligent Information Processing*, pages 45-53

estimation during the long development period. In addition a lot of negotiation efforts will be undertaken among the different teams to solve many kinds of conflicts.

Multi-agent systems are the product of a recent evolution in the field of software engineering that is leading towards the development of decentralized, distributed software systems composed of autonomous entities that interact and share information with one another. ANUMBA [2] presents a multi-agent system for collaborative design in the construction sector. This system supports interaction and negotiation between the different agents that represent various participants that are usually engaged in a typical collaborative project design. Autonomy and co-operation are the important behavioral attributes in multi-agent systems [3]. The approach is perfectly suited for the development of a system for the distributed development of fuel-injection-systems where expertise from different sources, and in different physical locations, is encapsulated in separate modules that must then be integrated to address a user inquiry. This paper is to investigate the feasibility of such an approach and to develop a prototype system.

2 Meta-Model of Agent in the Distributed Fuel-injection-system Development using Multi-agent

An agent is capable of (1) perceiving and acting at a certain level, (2) communicating in some ways with other agents, (3) attempting to achieve particular goals or perform particular tasks, and (4) maintaining an implicit or explicit model of its own state and the state of its world [4]. Brustoloni's taxonomy of software agents [5] begins with a three-way classification into regulation agents, planning agents, or adaptive agents. A regulation agent reacts to each sensory input as it comes in, and always knows what to do. It neither plans nor learns. Planning agents plan either in the usual AI sense (problem solving agent) or by using case-based reasoning or operations research-based methods. The adaptive agents not only plan, but also learn. LIU [1] present some agents in the architecture of a multi-agent design environment, which includes design tool agent and communication agent.

Because designer is the active, and the computer and software are only the tools to support design activities, some basic design tools and communication tools are used as the fundamental support part of agent, not an independent agent in this paper. Two classes of agents are defined and used in the system for the development of fuel pumps: management agents and actor agents. Management agents are responsible for the control and negotiation in the design process. The actor agents include among others the design agent, the analysis agent, the simulation agent and. These agents are situated on the different layers. The hierarchical relation limits the authority of the agents in this environment. All kinds of computer tools will be used for the actor agents and management agents to support communication and database management, as well as system maintenance.

2.1 Meta-model of the Management Agent

Management agents are located on the server and manage the local or the whole development group. The actions of the management agent usually include the decision making process and the possibility to perform inquiries about as well as control and supervise the lower layer agents. The task-oriented problem solving relation is a kind of dynamic organized relation that is formed when agents complete the separate tasks for a common design goal. The relation among agents is dynamically changed. As soon as the tasks are fulfilled, the relation is dissolved voluntarily. All the management agents in the fuel-injection-system development contain a design planer, processor monitor, conflict negotiator and version controller. The meta-model of the Management Agent is presented in Fig.1.

When a new task is defined, the new problem solving relation may be formed by a group of new agents [6]. The design task will be split up and distributed to many actor agents by the design planer according to the product development process and the relation among the actor agents. This dynamic set of tasks and agents and their relation among each other are watched and recorded by the process monitor. Whenever a design event happens, the event monitor of process monitor will be triggered.

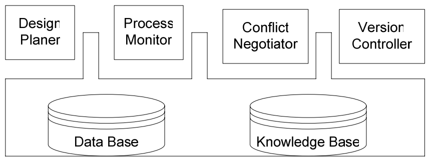

Fig. 1. Meta-model of the Management Agent.

Co-operation between agents has been presented as one of the key concepts which differentiates multi-agent systems (MASs) from other related disciplines and application such as expert systems, distributed computing and distributed object-oriented databases. Such a co-operation is essential for agents to achieve either group related or individual objectives. However, co-operation between agents is often challenged by a limitation of the resources. In such cases, negotiation is a major approach to achieve the co-operation, in which agents attempt to reach a joint decision between the teams of developers on matters of common concern which they are in disagreement and conflict about. The conflicts between competitive individualism and co-operative collectivism are resolved through negotiation [3]. The activities during the design stages involve a lot of negotiation and exchange of information between these design groups.

Several different negotiation mechanisms can be used in MAS, which mainly include rule-based and case-based, as well as game theory and behavioral theory [7]. Here, the rules bases and case bases are used to be the guidance for human interactions by the way of interaction between different agents. A solution must be reached agreeable to all agents otherwise deadlock and failure occurs.

Collaborative design is a process that helps to find satisfying solutions. All the design history records of design tasks performed in the past are managed by the version controller. All the design information is stored in the data base of the management agent. The knowledge in the knowledge base (KB) of a management agent includes the entire design process knowledge map, as well as the rules bases and case bases for negotiation in the local group. When an agent is added to or deleted from the group, the corresponding knowledge of management agent will be modified.

2.2 Meta-model of the Actor Agent

The majority of agents in the fuel-injection-system development are design agents. The analysis agent and the simulation agent also take part in the product development. The design agents and analysis agents, as well as simulation agent, are included in the actor agent, which is managed and controlled by the management agent. But every actor agents is a kind of domain-dependent and semi-autonomous agent, it has its design capacity and task. The meta-model of the actor agent is presented in Fig.2.

All the actor agents have a pre-processor, a core processor and a post-processor. The core-processor in the actor agent has the following abilities:

- Doing routine works just like key part design and performance simulation, as well as the dynamic analysis of an assembly and the structural analysis of key parts by FEM.
- Maintaining and interpreting knowledge related to itself and other agents.
- Interacting with designers, catching the interest and habit of the designers and recording this information in its knowledge base (such as recording the completed cases).

An actor agent gets the information from its management agent with its sensors and then translates it to the internal description of the situation. The pre-processor will standardize the data from the management agent. Supported by the knowledge base and model base, as well as data base, the core function processor will design, analyses or simulate the given tasks. The post-processor will transfer the data from the core function processor, and submit the report to its management agent.

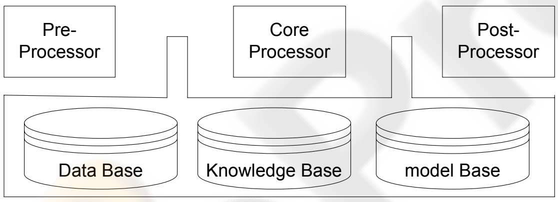

Fig. 2. Meta-model of the Actor Agent.

3 Architecture of the Distributed Fuel-injection-system Development Using Multi-agent

There are several research projects that focus on the application of agent and multi-agent systems for collaborative design. The ACE project [8] undertaken at the US Army Corps of Engineers construction Engineering Research Laboratories (USACERL), investigated how to support collaboration among members of the design team by providing an infrastructure for a community of cooperative design agents that assist the users. The PACT project [9] demonstrates the applications of agents in collaborative distributed design problems in which the project team members are distributed over multiple sites, cut across various engineering disciplines, and deploy different heterogeneous subsystems. There are several common ways of structuring the agent community within MAS. The choice of structure will impose relationships between agents that will fundamentally affect the way they communicate and negotiate. ANUMBA [2] concludes four ways of structuring the agent community, which is

- organizational structuring
- contracting,
- multi-agent planning and
- peer to peer negotiation.

Here the organizational structuring way is used, because only in this architecture there is one agent that has a global overview of the full task.

The central management agent controls the whole development process in the system. After getting the design requirements, the central management agent will plan the development process, and activate and link some agents in some domains. It will also it monitor and negotiate the product development procedure in order to achieve an appropriate cooperative development system. All the design conflicts from lower lever agent will be solved according to the given technical and financial requirements, until a satisfactory solution is achieved.

The actor agent is a kind of domain-dependency agent. It has some special knowledge and abilities and can help designers in a special domain. The actor agent files knowledge of the way how to design certain products based on the individual strategies and preferences of the different human designers involved. It is constructed to "understand" the representation of a design state, and contributes in a manner that leads to successful solutions. The strategies used by the agent are based on deterministic algorithms. In the current implementation, most agents are semi-autonomous, but are triggered by the messages from the management agent. The architecture of the fuel-injection-system development using a multi-agent system is presented in Fig.3.

The product design contains the design and configuration of muzzle and injector. The product design focuses on the key parameter calculation, and the product configuration focuses on the detailed design using similar product cases from the case base.

The assembly simulation is used to query the interference information for the product structure. The dynamic analysis and the analysis by FEM are used to check the maximum force to the key parts when the product is working, using special simulation software.

Performance simulations contain the information about the injection performance simulation and the matching simulation. The injection simulation is used to check the injection process and some performance regulation, and matching simulation is used to evaluate the match effects with diesel.

The management agent for the injector controls the design action of the plug design agent, the outer-valve design agent, the CAM design agent, the plug-spring design agent and housing design agent. It also solves the design conflicts among these design agents. The management agent for muzzle has similar functions as the injector's agent.

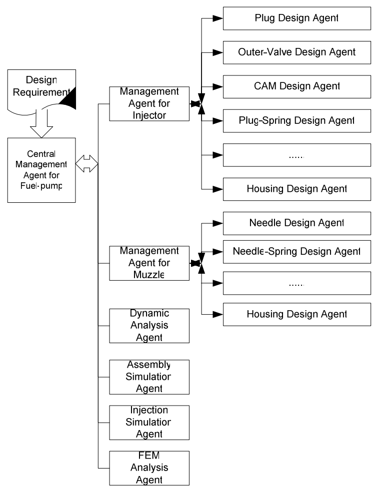

Fig. 3. the architecture of distributed fuel-injection-system product development using multi-agent.

4 Prototype System for the Development of a Fuel-injection-system using Multi-Agent

Fuel-injection-system development process includes three stages, which are the concept design stage, the configuration design stage, and the analysis stage. Some tasks can be performed concurrently, if the working condition and relative parameters of the actor agent is prepared well. The co-operation among different agents is controlled by the management agent using message.

4.1 The Concept Design Stage

According to the technical requirement from the customer, the design task will be split by the central management agent into several subtasks and be sent to the collaborative agents. At the concept design stage, most design work will be done by the design agents. But it is controlled by the management agent for the injector and the management agent for the muzzle.

In the design process for the injector, the diameter of the plug is a key parameter. After the diameter of the plug is confirmed, the design agent for the plug-spring and the design agent for outer-valve can start to work. The design agent for CAM can select the basic profile for motion. All these arithmetic tasks will be communicated to the management agent for the injector.

In the design process for the muzzle, the diameter and the number of holes for the muzzle is confirmed firstly. Then the design agent for the needle spring can begin to calculate its size and select its heat processing method, and the design agent for the housing can calculate many parameters inside the muzzle house. Fig. 4 is one of interface of the management agent for Muzzle.

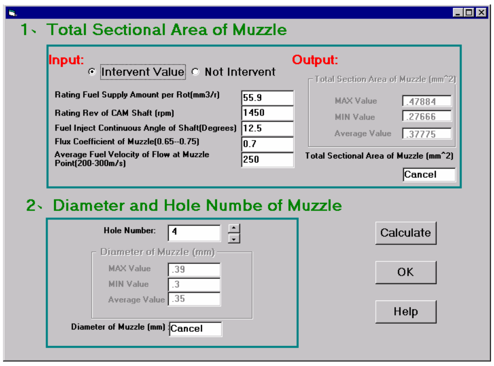

Fig. 4. The Management Agent for the Muzzle.

4.2 The Configuration Design Stage

All the design agents take part in the configuration design stage. Each team member with a different area of expertise will be primarily concerned with his own area of interest. All design agents will take part in the new product development, and will have to meet minimum design standards as defined with their internal knowledge bases. The case-based design method is often used to support designing many parts. But all the results will be verified by many standards and criteria. If the technical performance can not be satisfied, the parameters of part will be re-designed using some formula or rules. All the information of the successful part case will be saved in the case base. Fig. 5 is an interface of design agent, also it display the BOM (Bill of Material) and prototype of product.

After all the parts are modeled using CAD software, for example Pro/Engineering, the assembly work can be done according to the structure style from the concept design stage. The assembly simulation agent can check the space violation and some conflicts in the assembly process (Fig. 6). When the assembly agent finds constraint violation, it will inform collaborative agent to solve problem by coordination among the design agents.

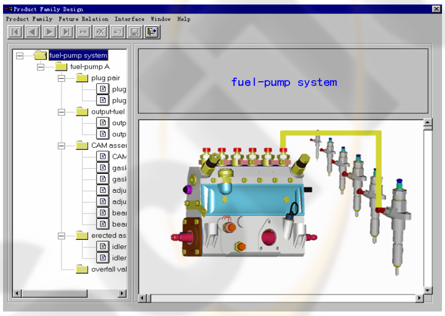

Fig. 5. The Configuration design for fuel-injection-system.

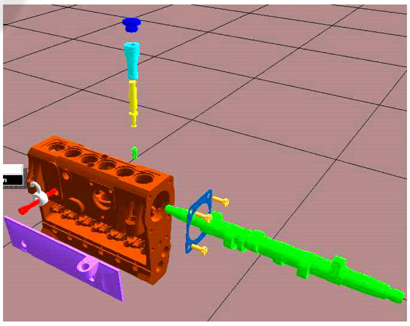

Fig. 6. The simulation of assembly process.

4.3 The Analysis Stage

In the analysis stage, the injection performance analysis is important, because the injection process will affect the work process of the diesel combustion. The dynamic analysis for the injector and the muzzle will be performed, and their results are used for the FEM analysis of some key parts.

The injection simulation agent will calculate many kinds of injection performances during the dynamic injection process, including the pressure of the fuel at the point of the injector and at the outlet of the pump, as well as injection agile and the displacement of the needle during an operating cycle (Fig.7). The whole injection process can be simulated, and all the performances can be drawn using charts.

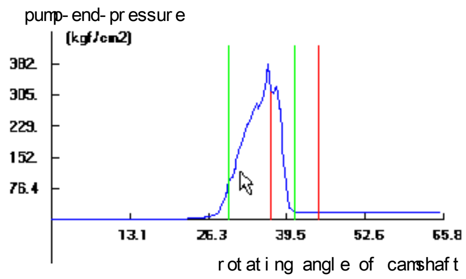

Fig. 7. One of the simulation results of the fuel injection.

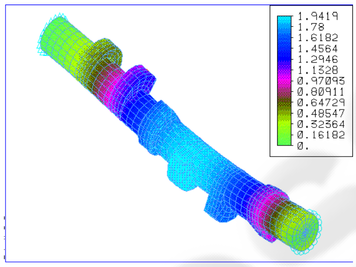

Fig. 8. FEM analysis result of the cam shaft.

If the injection performance is satisfied by the user, the modeling work for the movement of injection can be done by the dynamic analysis agent. All the forces and torques what the user want can be calculated, and its movement can be simulated. Using the forces and torques from mechanical dynamic analysis, as well as some constrains to structure, the stress of some key parts can be calculated by the FEM analysis agent. Fig. 8 is to introduce one of the FEM analysis result of the cam shaft.

Each agent can also communicate directly with its management agent by the internet. All the agents exchange design data and knowledge via a local network or the internet via the management agent.

6 Conclusions

A key aspect of collaborative working between the multi-disciplinary teams involved in the complicated product development is to facilitate the flow of information across the heterogeneous software tools in use. This paper aims to investigate the use of intelligent agents to facilitate collaborative design and focuses initially on the development and design of a fuel-injection-system. This approach overcomes the problem of geographically distributed teams.

Designers and design agents play the important role in the fuel-pump system development. Actor agent is present to cover the function of regular part design and assembly simulation, as well as dynamic analysis and performance simulation. Management agent is used to control and negotiate the design process and parameter transfer. All the computer and software are used to be the fundamental part of actor agent and management agent, not the independent agent.

Organizational way of structuring the agent community is used to manage the product development and to realize the design task. The actor agents are controlled and triggered by the messages from the management agent. Some agents are semi-autonomous, and some co-operation activities are limited. Pro-activeness is not discussed in this paper, and actor agent can not exhibit goal-directed behavior by taking the initiative.

Acknowledgements

The author is most grateful to National Nature Science Foundation of China (No. 2003CB317005) and Shuguang Program of Shanghai Educational Committee (No. 05SG15) for financial support that made this research possible. The author also thanks the Konstanz University of Applied Science for the support of the cooperation between the two universities in Shanghai and Konstanz.

References

1. Hong liu, Mingxi Tang, John Hamilton Frazer, Supporting dynamic management in a multi-agent collaborative design system, Advances in engineering software, 35(2004), 493-502
2. C. J. Anumba, O. O. Ugwu, L. Newnham, A. Thorpe, A multi-agent system for distributed collaborative design, Logistics Information Manage, 2001, 14(5/6), 355-366
3. C. J. Anumba, Z. Ren, A. Thorpe, O. O. Ugwu, L. Newnham, Negotiation within a multi-agent system for the collaborative design of light industrial buildings, Advances in engineering software, 34(2003), 389-401
4. K. S. Barber, J. Kim, Toward flexible tolerant intelligent manufacturing: sensible agents in shop-floor control, artificial Intelligence in Engineering Design Analysis and Manufacturing, 14(2000), 337-354
5. Brustoloni J C, Autonomous agents: characterization and requirements, Carnegie mellon technical report CMU-CS-91-204, Pittsburgh: Carnegie Mellon University, 1991
6. Liu H., Lin Z. K., A collaborative design approach in MADS, Proceedings of the Fourth International Conference on CSCW in Design 1999, 297-303.
7. Kraus S., Multi-agent negotiation under time constraints, Artificial Intelligence, 1995, 75(2),297-345
8. McGraw K. D., Lawrence P. W., Morton J. D., Heckel J., The agent collaboration environment: an assistant for architects and engineers, Proceedings of the Third Congress on Computing in Civil Engineering, Anaheim, California, June 17-19, 1996, 739-745.
9. Cutkosky M. R., Englemore R. S., Fikes R. E., Genesereth M. R., Gruber T. R., Mark W. S., Tenebaum J. M., Weber J. C., PACT: an experiment in integrating concurrent engineering system. In: Huns M. N., Singh M. P., editors, Readings in Agents, San Mateo: Morgan Kaufmann Publishers, 1998 56-65