*I. J. Communications, Network and System Sciences*, **2008, 3, 207-283**
Published Online August 2008 in SciRes (http://www.SciRP.org/journal/ijcns/).

# An Agent-Based Multimedia Intelligent Platform for Collaborative Design

**Quan LIU, Xingran CUI, Xiuyin HU**
*School of Information Engineering, Wuhan University of Technology,*
*Wuhan, P.R. China*
*E-mail: {quanliu, xingran, huxiuyin}@whut.edu.cn*

*Received on April 14, 2008; revised and accepted on August 26, 2008*

## Abstract

Collaborative design can create added value in the design and production process by bringing the benefit of team work and cooperation in a concurrent and coordinated manner. However, distributed design knowledge and product data make the design process cumbersome. To facilitate collaborative design, an agent-based intelligent CAD platform is implemented. Intelligent agents are applied to the collaborative design. Adopting the JADE platform as framework, an intelligent collaborative design software (Co-Cad platform for short) is designed. In this platform, every man, design software, management software, equipment and resource is regarded as a single agent, the legacy design can be abstracted to be interaction between agents. Multimedia technology is integrated into Co-Cad platform, communication and identity authentication among collaborative designers from different areas are more convenient. Finally, an instance of collaborative design using Co-Cad platform is presented.

**Keywords:** Agent, Co-Cad Platform, Collaborative Design, JADE, Multimedia Technology

## 1. Introduction

Product design is becoming a collaborative task among designers or design teams that are physically, geographically, and temporally distributed. Plenty of product modeling tools and engineering knowledge from various disciplines spread around different design phases, making effective capture, retrieval, reuse, sharing and exchange of these heterogeneous design knowledge a critical issue [1]. An ideal product design environment which is both collaborative and intelligent must enable designers and manufacturers to respond quickly to commercial market pressures [2].

Compared with current standalone CAD, the collaborative CAD is “not generally accepted” because of both technical and non-technical problems [3]. As an emergent approach to developing distributed systems, agent technology has been employed to develop collaborative design systems and to handle the problems and overcome the limitations [4]. The way in which intelligent software agents residing in a multi-agent system interact and cooperate with one another to achieve a common goal is similar to the way that human designers collaborate with each other to carry out a product design project. Thus, we believe that a collaborative design environment implemented by taking an agent-based approach will be capable of assisting human designers or design teams effectively and efficiently in product design [5].

In order to make collaborative design more convenient, multimedia information is absolutely necessary. Integrating multimedia technology into collaborative design platform can express information more quickly and naturally than letters.

This paper presents an ongoing project on the application of intelligent agents to collaborative design. In the project, an intelligent CAD platform (Co-Cad platform) is implemented, adopting multi-agent technology, collaborative design technology and multimedia technology. The platform uses the proposed JADE as middle-ware and integrates multimedia system. Based on the platform, intelligent agents can interact with each other via network. Thus, the ideal digital collaborative design is implemented.

The rest of the paper is organized as follows: Section 2 gives an overview of the work related to our research. The basic principle and key method are described in Section 3. Section 4 introduces the implementation of Co-Cad platform. A case study is presented in Section 5. Finally, a number of concluding remarks are made in Section 6.

## 2. Related Works

Web-and agent-based approaches have been dominant during the past decade for the implementation of collaborative product environments. This section provides a brief updated review of the applications of agents and web-based technologies to collaborative design engineering.

An earlier review of multi-agent collaborative design systems can be found in reference [6]. Shen et al. [7] provide a detailed discussion on issues in developing agent-oriented collaborative design systems and a review of significant, related projects or systems. The interesting aspects of PACT include its federation architecture using facilitators and wrappers for legacy system integration. SHARE [8] was concerned with developing open, heterogeneous, network-oriented environments for concurrent engineering, particularly for design information and data capturingand sharing through asynchronous communication. SiFAs [9] was intended to address the issues of patterns of interaction, communication, and conflict resolution using single function agents. DIDE [10] was a typical autonomous multi-agent system and was developed to study system openness, legacy systems integration, and geographically distributed collaboration.

Co-Designer [11] was a system that can support localized design agents in the generation and management of conceptual design variants. A-Design [12] presented a new design generation methodology, which combines aspects of multiobjective optimization, multi-agent systems, and automated design synthesis. It provides designers with a new search strategy for the conceptual stages of product design, which incorporates agent collaboration with an adaptive selection of designs.

## 3. The Basic Principle and Key Method

### 3.1. Basic Principle

Usually, each agent is regarded to be a physical or abstract entity. Distributed in the network environment, each agent is independent and can act on itself and the environment, manipulated part of the environment reflection and react to the changes in the environment. More importantly, through communication and cooperation with other agents, it can perform mutual work to complete the entire task. Man, design software, management software, as well as equipment and resources can be viewed as agents. Legacy design activities can be abstracted to be informational communication between agents, which includes not only communication between homogeneous agents but also the communication between heterogeneous agents through technology-aided design software. We can further abstract them to informational exchange. Theoretically, all of the information can be digital. In other words, if we provide a suitable platform for interaction, digital collaborative design can be realized.

This paper seeks to build a multi-agent middleware, through which, man, software, manufacturing equipment within this collaborative organization can carry out informational communication. Different types of agents can communicate with their agent middleware through their respective forms of communication and achieve information interaction with agents in other organizations. Thus, it enables man to relieve from various types of software and achieve collaborative design efficiently and effectively.

In light of the above principle, this paper seeks to construct a collaborative design software platform for agents—Co-Cad platform, and integrates multimedia technology into this platform. Co-Cad platform, using pure Java language, is independent and flexible. In the network it realizes the collaboration between designers—“You see what I see”. Each designer’s operation will be reflected in others’ platform. Designers in different places can exchange their ideas on the interactive design in the form of video chat. Video chat can be the most direct way to confirm the identity of the others, which guarantees the safety of the collaborative design, it is also the fastest and most natural form of information expression.

### 3.2. The Multi-agent System Based on JADE

JADE is a middleware that facilitates the development of multi-agent systems in compliance with the FIPA (Foundation for Intelligent Physical Agents) specifications. Each running instance of the JADE runtime environment is called a container. The set of active containers is called a platform. A single special main container must always be active in a platform and all other containers register with it as soon as they start. Figure 1 illustrates the above concepts through a sample scenario showing two JADE platforms composed of 3 and 1 container respectively. JADE agents are identified by a unique name and, provided they know each other’s name, they can communicate regardless of their actual location. A main container differs from normal containers as it holds two special agents: AMS and DF. The AMS (Agent Management System) provides the naming service and represents the authority in the platform. The

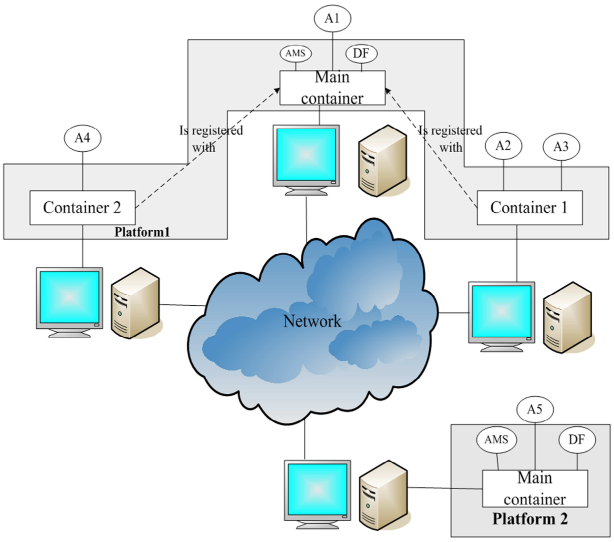


**Figure 1. Containers and Platforms.**

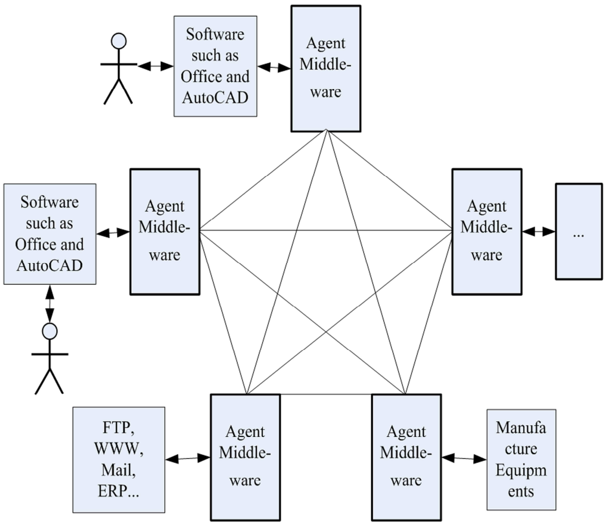


**Figure 2. Communication model.**

DF (Directory Facilitator) provides a yellow pages service by means of which an agent providing the services he requires in order to achieve his goals.

One of the most important features that JADE provides for agents is the ability to communicate. The communication paradigm adopted is the asynchronous message passing. Each agent has a sort of mailbox (the agent message queue) where the JADE runtime posts messages sent by other agents. Whenever a message is posted in the message queue the receiving agent is notified. If and when the agent actually picks up the message from the message queue to process it is completely up to the programmer.

## 4. Implementation of Co-Cad Platform

### 4.1. Communication Model

Under the currently popular mode of communication services, if A is going to discuss about certain parts of the design with B, A firstly produces his own design using computer-aided design software, such as Auto CAD, and then uploads it to the FTP server, while B using the same design software uploads his own design to the FTP server. Then they download each other’s program. Through traditional telephone or E-Mail and even more network communication tools, they exchange their opinions and finally reach a consensus. When the design is completed, it is uploaded onto the FTP server, and then the WWW server will issue the note that the components design is completed for other people to use. Under this model, the exchanging and sharing of productive data don’t fit the requirements of network and it is at a low level of intelligence. It cannot meet the needs of increasingly complex product design.

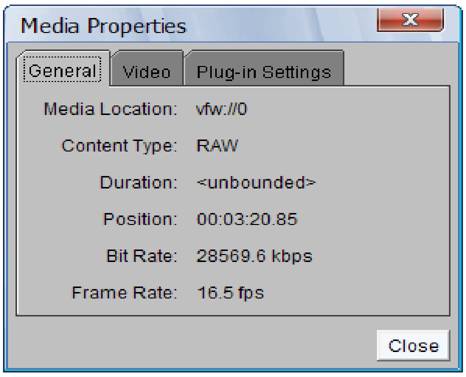
In order to overcome the above shortcomings, the platform’s interactive process design is based on Multi-Agent communication model (Figure 2).

First, A starts his agent middleware and through agent middleware he can know that currently B is also online. Of course, they may have already reached the agreement that software communication through certain middleware. A starts his software, such as Auto CAD, and notifies B. Both of them use the voice and video program and Auto CAD to make the real-time interactive design through agent middleware. After that, the agent middleware will submit the design to the FTP server automatically for others to use, and automatically issues the news that components of the design is finished in the WWW server. In this paper, the middleware is to provide a platform for interaction, but the middleware itself is an agent, they must first interact with other middleware to complete their task. The agent communication protocol adopted in Co-Cad is within JADE platform. The own protocol of JADE facilitates the development of multi-agent systems in compliance with the FIPA specifications.

### 4.2. Design and Integration of Multimedia System

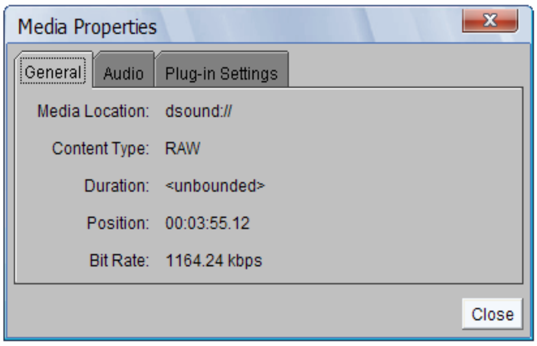
Web-based video conferencing system has been widely used in remote collaborative design system, but these systems are using ready video conferencing system to bring about long-distance transmission. Video conferencing systems are separated from remote collaborative design system, which is inconvenient to integrate with the collaborative design system seamlessly. Due to some restrictions of video conferencing system, the remote collaborative design system can not stick together very well.

As Co-Cad platform is completed developed in Java language, in order to make the seamless integration with



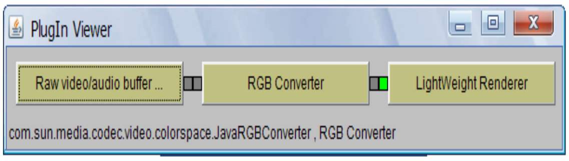
Media Location: vfw://0
Content Type: RAW
Duration: <unbounded>
Position: 00:03:20.85
Bit Rate: 28569.6 kbps
Frame Rate: 16.5 fps

**Figure 3. The video properties.**



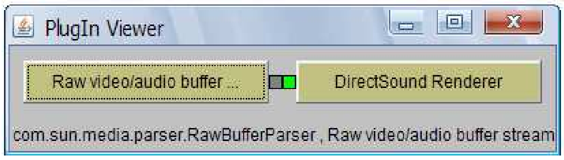
Media Location: dsound://
Content Type: RAW
Duration: <unbounded>
Position: 00:03:55.12
Bit Rate: 1164.24 kbps

**Figure 4. The audio properties.**



com.sun.media.codec.video.colorspace.JavaRGBConverter , RGB Converter

**Figure 5. PlugIn viewer of video.**



com.sun.media.parser.RawBufferParser , Raw video/audio buffer stream

**Figure 6. PlugIn viewer of audio.**

multi-media audio and video system possible, this paper applies multimedia technology to achieve video chat function and adopts the JMF (Java Media Framework) as development environment. To handle multimedia files and equipment in Java language, it's a must to install JMF installation package.

### 4.2.1. Media Capture

Time-based media can be captured from a live source for processing and playback. For example, audio can be captured from a microphone or a video capture card can be used to obtain video from a camera. Capturing can be thought as the input phase of the standard media processing model.

A capture device might deliver multiple media streams. For example, a video camera might deliver both audio and video. These streams might be captured and manipulated separately or combined into a single, multiplexed stream that contains both an audio track and a video track.

### 4.2.2. Media Processing

In most instances, the data in a media stream are manipulated before they are presented to the user. Generally, a series of processing operations occur before presentation:

1) If the stream is multiplexed, the individual tracks are extracted.
2) If the individual tracks are compressed, they are decoded.
3) If necessary, the tracks are converted to a different format.
4) Effect filters are applied to the decoded tracks.

The tracks are then delivered to the appropriate output device. If the media stream is to be stored instead of rendered to an output device, the processing stages might differ slightly. To capture audio and video from a video camera, process the data, and save it to a file:

5) The audio and video tracks would be captured.
6) Effect filters would be applied to the raw tracks.
7) The individual tracks would be encoded.
8) The compressed tracks would be multiplexed into a single media stream.
9) The multiplexed media stream would be saved to a file.

It includes a viewer that displays a graphical overview of a processor's tracks and plug-ins. This graph enables you to monitor the media flow during playback, capturing, or transcoding. Figure 3, Figure 4, Figure 5 and Figure 6 display the media properties and plug-ins currently.

## 5. Case Study

The only software requirement to execute the platform is the Java Run Time Environment version 1.4. All the software is distributed under the LGPL license limitations and it can be downloaded from the JADE web site http://jade.tilab.com/. Having uncompressed the archive file, a directory tree is generated whose root is jade and with a lib subdirectory. This subdirectory contains some JAR files that have to be added to the classpath environment variable.

The process of collaborative design using Co-Cad platform to carry out a product design project is performed

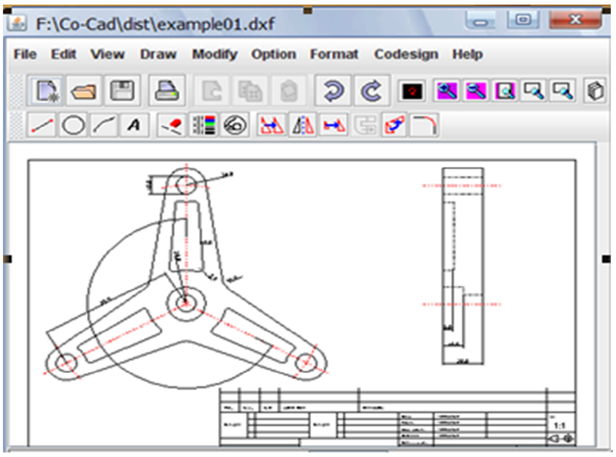
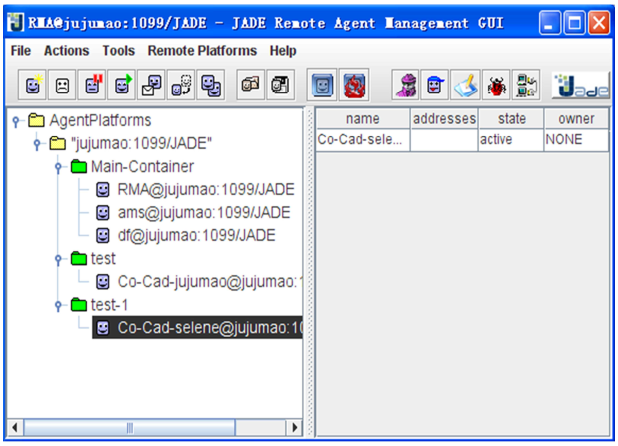

F:\Co-Cad\dist\example01.dxf
File Edit View Draw Modify Option Format Codesign Help

**Figure 7. Single designer's interface.**


RMA@jujumao:1099/JADE - JADE Remote Agent Management GUI

File Actions Tools Remote Platforms Help

- AgentPlatforms
  - "jujumao:1099/JADE"
    - Main-Container
      - RMA@jujumao:1099/JADE
      - ams@jujumao:1099/JADE
      - df@jujumao:1099/JADE
    - test
      - Co-Cad-jujumao@jujumao:1
    - test-1
      - Co-Cad-selene@jujumao:1(

| name | addresses | state | owner |
|---|---|---|---|
| Co-Cad-sele... | | active | NONE |

**Figure 8. Main interface of the agent manager.**

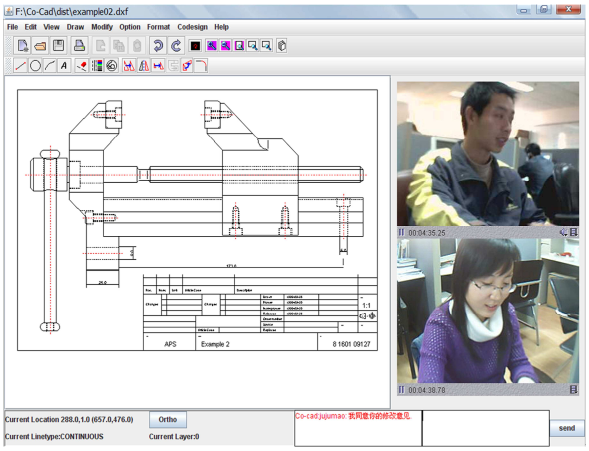
as follows:

Step 1: The host computer used as server launches JADE, and then Co-Cad platform, starts the test container in JADE. Other collaborative designers only need to run Co-Cad.

Step 2: While JADE platform is running, all collaborative designers can start Co-Cad platform and see Co-Cad software interface. Click on the "CodesignStart" submenu, the moment of collaborative design is coming. Figure 7 shows single designer's interface on Co-Cad platform before collaborative design with others.

Step 3: Each designer can launch codesign request and select codesign partner. Only when both designers agreed to other's request, they can go on collaborative design and show the audio and video window. The main user interface of the agent manager in JADE platform is as Figure 8.

Step 4: Collaborative designers can communicate with each other in form of letters or video chat spontaneously and freely. If one of the designers edits or modifies the design, the same operation will show on the other's


F:\Co-Cad\dist\example02.dxf
File Edit View Draw Modify Option Format Codesign Help
APS
Example 2
8 1601 09127
1:1
00:04:35.25
00:04:38.78
Ortho
send

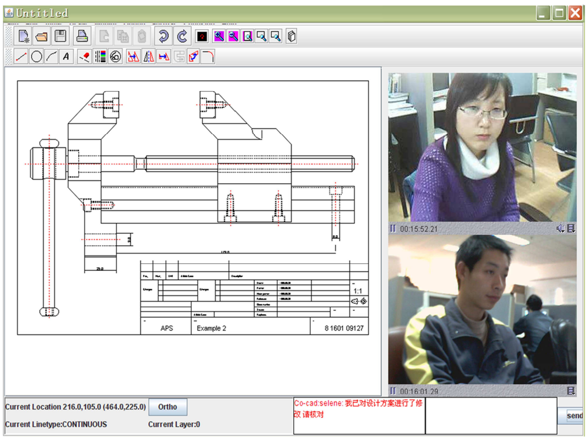
Current Location 288.0,1.0 (657.0,476.0)

Current Linetype:CONTINUOUS

Current Layer:0

Co-cad:jujumao: 我同意你的修改意见.

**Figure 9. Interface of Selene's Co-Cad platform.**



Current Location 216.0,105.0 (464.0,225.0)

Current Linetype:CONTINUOUS

Current Layer:0

Co-cad:selene: 我已对设计方案进行了修改 请核对

**Figure 10. Interface of Jujumao's Co-Cad platform.**

platform. As presented in Figure 9 and Figure 10, Selene and Jujumao are designing a mechanism accessory using Co-Cad platform. Figure 9 presents interface of Selene's platform, and Figure 10 is Jujumao's interface. They interact with each other in the form of video and chat. Selene modifies the plan, then the same modification shows in Jujumao's platform. There is a small window used to input letters for chat, where speaking loudly is not politely.

## 6. Conclusions

In this paper, on the basis of the problem identification and the analysis of the requirements for a collaborative design platform, an agent-based platform supporting collaborative design via the cooperation of a network of intelligent agents is presented. On the platform, all agents are written in the Java language using the JADE platform and work together to perform flexible, adaptive and dynamic design tasks in an autonomous and collaborative way. As the platform is still being fully implemented, more experiments are required to be carried out in order to test and improve our platform. In the future testing of our approach and software, we intend to involve designers in the process with real design examples since we believe that a design-oriented approach needs to be taken in order to identify those key tasks that need collaboration and support by software agents.

However, some challenging problems, such as task assignment, conflict detection and conflict solution, need to be carefully addressed and further development efforts are required before the technology can be widely deployed. In our project, on-going efforts are being made to refine the coordination agent and its underlying methodology into detail.

## 7. Acknowledgments

This project is supported by International science and technology cooperation project (NO.2006DFA73180) from China's Ministry of Science and Technology.

## 8. References

[1] J. X. Wang and M. X. Tang, "A multi-agent framework for collaborative product design," in: Z. Shi and R. Sadananda (Eds.): PRIMA 2006, LNAI, Vol. 4088, Springer, Heidelberg, pp. 514–519, 2006.

[2] J. X. Wang and M. X. Tang, "An agent-based system supporting collaborative product design," in: B. Gabrys, R.J. Howlett, and L.C. Jain (Eds.): KES 2006, Part II, LNAI, Vol. 4252, Springer, Heidelberg, pp. 670–677, 2006.

[3] J. X. Wang and M. X. Tang, "Knowledge representation in an agent-based collaborative product design environment," Proceedings of the 9th International Conference on Computer Supported Cooperative Work in Design (CSCWD 2005), Vol. 1, IEEE Computer Society Press, pp. 423–428, 2005.

[4] M. A. Rosenman and F. Wang, "A component agent based open CAD system for collaborative design," Automation in Construction, Vol. 10(4), pp. 383–397, 2001.

[5] S. M. Wu, H. Ghenniwa, Y. Zhang, and W. M. Shen, "Personal assistant agents for collaborative design environments," Published by Elsevier B.V. Computers in Industry 57, pp. 732–739, 2006.

[6] S. E. Lander, "Issues in multi-agent design systems," IEEE Expert 12 (2), pp. 18–26, 1997.

[7] W. Shen, D. H. Norrie, and J. P. Barthès, "Multi-agent systems for concurrent intelligent design and manufacturing," Taylor and Francis, London, UK, 2001.

[8] G. Toye, M. R. Cutkosky, L. Leifer, J. Tenenbaum, and J. Glicksman, "SHARE: A methodology and environment for collaborative product development," in: Proceedings of Second Workshop on Enabling Technologies: Infrastructure for Collaborative Enterprises, pp. 33–47, 1993.

[9] D. C. Brown, B. Dunskus, D. L. Grecu, and I. Berker, "SINE: support for single function agents," in: Proceedings of Applications of Artificial Intelligence in Engineering, Udine, Italy, 1995.

[10] W. Shen and J. P. Barthès, "An experimental environment for exchanging engineering design knowledge by cognitive agents," in: M. Mantyla, S. Finger, T. Tomiyama (Eds.), Knowledge Intensive CAD-2, Chapman &Hall, pp. 19–38, 1997.

[11] M. J. Hague and A. Taleb-Bendiab, "Tool for management of concurrent conceptual engineering design," Concurrent Engineering: Research and Applications 6 (2), pp. 111–129, 1998.

[12] M. I. Campbell, J. Cagan, and K. Kotovsky, "A-Design: an agent-based approach to conceptual design in a dynamic environment," Research in Engineering Design 11, pp. 172–192, 1999.