Fall 2007

Management of an intelligent argumentation network for a web-based collaborative engineering design environment

Man Zheng

MANAGEMENT OF AN INTELLIGENT ARGUMENTATION NETWORK FOR A WEB-BASED COLLABORATIVE ENGINEERING DESIGN ENVIRONMENT

by

MAN ZHENG

A THESIS

Presented to the Faculty of the Graduate School of the

UNIVERSITY OF MISSOURI-ROLLA

In Partial Fulfillment of the Requirements for the Degree

MASTER OF SCIENCE IN COMPUTER SCIENCE

2007

Approved by

Dr. Xiaoqing(Frank) Liu, Advisor

Dr. Ming C. Leu

Dr. Maggie Cheng

ABSTRACT

Conflict resolution is one of the most challenging tasks in collaborative engineering design. In the previous research, a web-based intelligent collaborative system was developed to address this challenge based on intelligent computational argumentation. However, two important issues were not resolved in that system: priority of participants and self-conflicting arguments. In this thesis, two methods are developed for incorporating priorities of participants into the computational argumentation network: 1) weighted summation and 2) re-assessment of strengths of arguments based on priority of owners of the argument using fuzzy logic inference. In addition, a method for detection of self-conflicting arguments was developed. In the end, the proposed methods based on a real solar car project are validated. Incorporation of priority of participants and detection of self-conflicting arguments has improved the capability of managing an intelligent argumentation network for the web-based collaborative engineering design system developed in the previous research.

ACKNOWLEDGMENTS

I owe a debt of gratitude to all those who have helped me with this thesis. First I want to thank my advisor, Dr. Frank Liu, who supported me in this research. With his professional suggestions and help, I could do this research more efficiently and easily. Secondly, I would also like to express my gratitude to Dr. Ming Leu and Dr. Maggie Cheng for serving on my thesis committee and for taking time to review this work.

This project is supported by the Intelligent System Center (ISC) in the University of Missouri-Rolla. Participant 4 participates in the case study of solar car design using the collaborative engineering design system.

TABLE OF CONTENTS

Page

LIST OF ILLUSTRATIONS

Figure Page

LIST OF TABLES

1. INTRODUCTION

Modern product design is a complex process involving multiple roles such as designers, manufacturers, suppliers, and customers. Collaborative Decision Support Systems (CDSSs), which are interactive computer-based systems, facilitate the solution of ill-structured problems by a group of decision makers working together as a team [1]. These systems increase the effectiveness of decision groups by interactively sharing information between team members and the computer. Many conflicts arise during a design project, and the designers concentrate on a conflicting issue and attempt to resolve it. Others have narrowed this type of interactive computer-based system down to a specific system that manipulates arguments provided by decision makers.

The principal objective of this project is to enhance the conflict resolution accuracy of an existing argumentation-based CDSS by incorporating the different priorities of decision makers and by detecting self-conflicting arguments.

1.1. COLLABORATIVE ARGUMENTATION ENVIRONMENT

Many computer supported collaborative argumentation systems (CSCA) have been developed. These systems are used in not only design but also many other field such as philosophy, law, education and military. However each system is different from the rest in the underlying argumentation theories. They apply either simple graphs or trees and simple logic programming to present the argumentations. All these systems lack of organizing the participants' arguments and participants have difficulty to forward further arguments. Complex interactions in CSCA involves many factors such as domain and argumentation knowledge, training in CSCA tools, user interface design, motivation to use CSCA and design of arguments. Every factor is supposed to be paid equal attention. Every factor can possibly affect the argumentation process and may have an important impact on the decision making.

1.2. ARGUMENTATION-BASED CONFLICT RESOLUTION

Previous collaborative engineering systems have shown that a lack of physical presence severely impairs the accuracy of decision making when decision makers discuss a design project remotely, especially over the internet, however those systems are not able to resolve the problem by putting forward an effective conflict resolution module. If any conflict arises during the design process, decision makers have a minimal chance of resolving it.

In the argumentation-based CDSS [2] [3] which this project is base on, arguments were organized into a hierarchical structure. Given the hierarchical structure of arguments, some computational approach finds its way into previous research and contributed an effective resolution for the conflicts arising from arguments. In [2] [3], each argument was categorized into two stands, supporting or attacking another argument. Generally, there were multiple design positions for each design issue, and each argument either supported or refuted a position. A numeric weight was assigned to every argument; this denoted the strength of the argument and thus affected the weight of position it was attached to. These values were then input into a fuzzy inference engine that produced a total weight for every position under the given issue. When compared, these output values gave the result of the conflict.

During the ongoing research, researchers theorized that additional enhancements could incorporate more properties of arguments, leading to a more accurate result. The priority of each decision maker is a useful factor that can be taken into consideration. Another important issue is self-conflicting arguments—obviously, the existence of self-conflicting arguments impairs the correctness of the final decision. Therefore, finding an approach to detect and remove them before using the computational approach can enhance the correctness of final decision.

2. RELATED WORK

2.1. COLLABORATIVE DESIGN SYSTEM

A traditional Computer-Aided Design (CAD) system only allows a single user to do design while a collaborative CAD system allows multiple designers to work together on a design. Early research projects in collaborative CAD design systems [5] [6] [7] [8] have successfully addressed some engineering design issues in collaborative environments. They were developed on local area networks, which are platform dependent, and they were not web-enabled. It is hard to use them to support designers in locations thousands of miles away to collaborate in heterogeneous platforms. There have also been research efforts toward enabling traditional CAD systems for collaborative design. For example, a Computer Supported Cooperative Work (CSCW) system [7] was developed using C++ and implemented on AutoCAD for collaborative design. It has a generic model of collaborative design. Another such system is DOME [8], which was built by integrating existing single-user CAD systems using CORBA and C++.

The increasing power of the Internet makes collaborative CAD feasible for a global team. Recently, several web-based CAD systems have been developed to allow multiple users from geographically distributed locations to share their design models over the Internet. These systems fall into three categories. The first category of web-based CAD systems, including C-DeSS [8] and CDFMP [10], integrates web-based multimedia tools, such as online chat and online meeting, with web-based solid model displays so that designers from different locations can share their design ideas over the Internet. However, users cannot develop and edit their solid models online. The second category of web-based CAD systems, including the Internet design studio [11], WCW [12], WebCAD [13], and NetFEATURE [14], allows multiple users to share their design over the Internet, but the users cannot develop their common models concurrently. The earlier web-based collaborative design system developed a couple of years ago has the capabilities of both categories [15]. The third category of web-based CAD systems, including CSM [16], CollabCAD [17], and Alibre Design [4], focuses on collaborative solid modeling.

.

However, all of the existing web-based collaborative design systems provide very little or no support for exploring design alternatives and identifying the best design alternative through intelligent argumentation from multiple perspectives. There is a clear need to develop a fundamental theoretic method of intelligent computational argumentation based conflict resolution that can be implemented on a web-based collaborative engineering design system.

2.2. ARGUMENTATION MODEL

Philosopher Stephen Toulmin developed a very influential model of argumentation [18] that has guided the development of software tools and systems for support of detection and resolution of conflicts in many knowledge domains. Argumentation is a process of arriving at conclusions through discussions and debates. Toulmin's work promoted a more informal approach in dealing with argumentation than formal logic. In the area of engineering design, several argumentation-based conflict resolution methods and systems have been developed based on Toulmin's model. The first of them, gIBIS (graphical IBIS), represents the design dialog as a graph [19]. While being capable of representing issues, positions, and arguments, gIBIS failed to support representation of goals (requirements) and outcomes. IBE [20] extended gIBIS by integrating a document editor. REMAP (REpresentation and MAintenance of Process knowledge) [21] extended gIBIS and IBE by providing the representation of goals, decisions, and design artifacts. As opposed to these systems, Sillince [22] proposed a more general argumentation model. His model is a logic model where dialogs are represented as recursive graphs and the rules of both rhetoric and logic are used to manage the dialog and to determine when the dialog has reached a closure. Alexander [23] described the incorporation of Toulmin's approach into a software product that represents features of arguments in a visual hierarchy to aid analysis of positions taken by proponents and opponents of particular design methods. The biggest challenge with these systems is that sizes of their argumentation networks are often too large to comprehend, and therefore it is very difficult to use them to help make design decisions because they are qualitative and not computational. In addition, they cannot deal with uncertainties associated with argumentations. A computational argumentation method is developed for capturing and analyzing software design rationales [24]. Parsons and Jennings [25] proposed a framework, based upon a system of argumentation, which permits agents to negotiate among themselves to establish acceptable ways for problem solving. QuestMap [26] is a Computer Supported Collaborative Argumentation (CSCA) tool developed to support legal argumentation by equipping users with language needed to construct and analyze arguments. The disadvantage of this tool is its lack of decision making capabilities. HERMES [1] was developed to aid decision makers reaching a decision, not only by efficiently structuring discussion rationale but also by providing reasoning mechanisms that constantly update discourse status in order to recommend the most supported alternative. Its disadvantage is that its weighting factor is not effective as it is not related to its position.

Decision-based design methods using utility analysis [27], negotiation protocols [28][29], value aggregation [30] and others [31] have made important contributions to engineering design. Argumentation-based approach as described in this thesis can be a significant methodology to collaborative decision-making in engineering design, esepecially at the conceptual design stage where concept design alternatives have been generated and a decision-making is needed to select the best alternative before a detail design. The proposed approach has the following unique features: 1) It can capture the design rationale from all members of the design team; 2) It is close to real-world team design because people use arguments to express their views and the rationale behind them in the collaborative engineering practice; and 3) It is easy to implement as a web-based system, which is very important for collaborative engineering design involving perople in geographically distrabuted locations. The main advantage of this approach is its informaity, which is also the main motivation to develop an intelligent computational model and a decision-making method based on this model, in order to establish a solid foundation for this approach.

2.3. INCORPORATION OF PRIORITY AND SELF-CONFLICTING ISSUE

Priority has been used to resolve conflicts in many engineering fields for a long time in practice. For example, in [32], researchers use a priority order to solve an aircraft conflict design. In [33], researchers develop a dynamic prioritized conflict resolution algorithm in a multiple access broadcast network where there is the possibility of a collision when two or more nodes transmit at overlapping times. Although priority assessment has been applied to many engineering fields, incorporating priority into an argumentation network remains challenging.

In [34], Belnap pointed out that self-conflicting argument should not result in defeating other arguments. In other researches such as [35], self-conflicting was not considered as a positive factor.

3. ARGUMENTATION BASED CONFLICT RESOLUTION IN THE COLLABARATIVE ENGINEERING DESIGN ENVIRONMENT

In this section, the collaborative engineering design environment in which we develop the enhanced conflict resolution model presented in this thesis is introduced and an accomplished argumentation based conflict resolution model [3][4] to lay a foundation.

3.1. ARCHITECTURE OVERVIEW

This design environment is based on the client-server architecture as shown in Figure 3.1. On the client side, the system provides user interfaces for solid modeling, whiteboards for design alternatives, argumentation based conflict resolution, and chat rooms for real-time information exchange. On the server side, it manages client communication, and argumentation network. Its graphical user interface is shown in Figure 3.2.

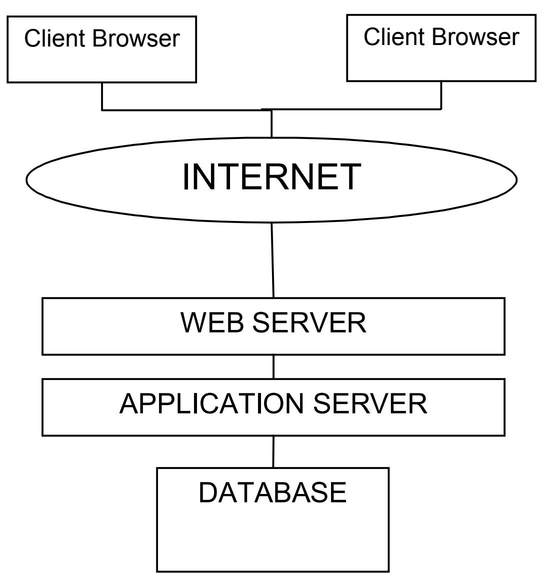


Figure 3.1. Server Client Architecture

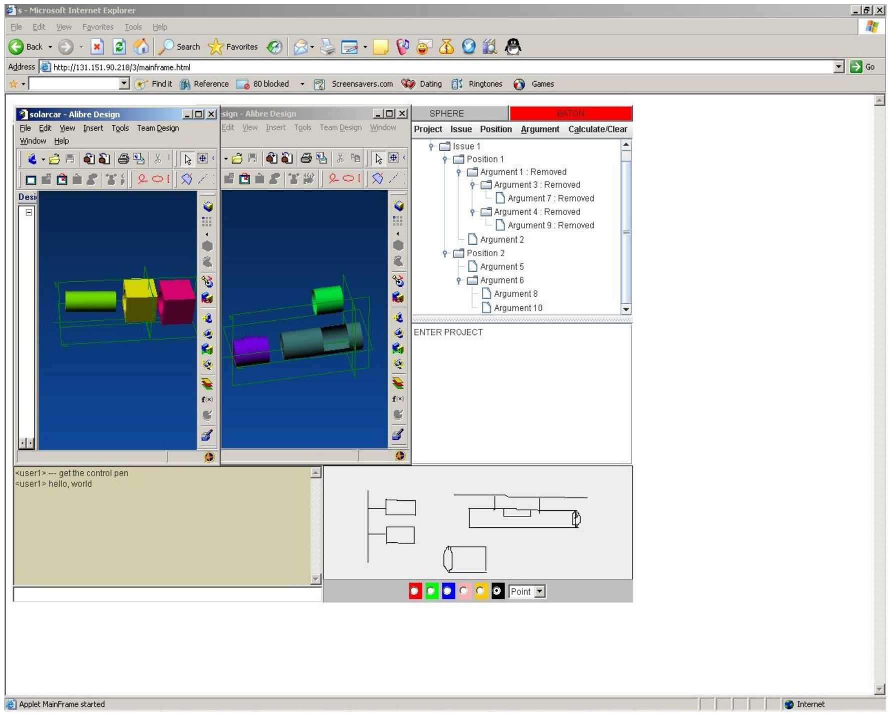


Figure 3.2. User Interface of an Intelligent Collaborative Engineering Design System using Computational Argumentation System

3.1.1. Client Browser. Designers can start an IE at any computer and type in the correct IP address and interface for clients will be requested. Designers are able to share information and discuss through the interface as shown in Figure 3.2.

3.1.2. Web Server. A web server is a computer program that is responsible for accepting HTTP requests from clients, which are known as web browsers, and serving them HTTP responses along with optional data contents, which usually are web pages such as HTML documents and linked objects (images, etc.). Apache server is chosen as the web server because it is highly compatible with the operating system and other servers, its ability to handle server-side programming, and publishing is very considerable.

3.1.3. Application Server. Application server which includes conflicting resolution module runs backstage. The regular functions of application server also include broadcasting the messages to each client and management and coordination of each client so they are capable of working collaboratively and simultaneously.

3.1.4. Database. Database is used to store the information of arguments and designers. Database is connected to the application server. As there are four kinds of data entries i.e. Project, Issue, Position and Argument, four database tables have been used to store the respective information. The relationship between the four tables is shown in Figure 3.3.

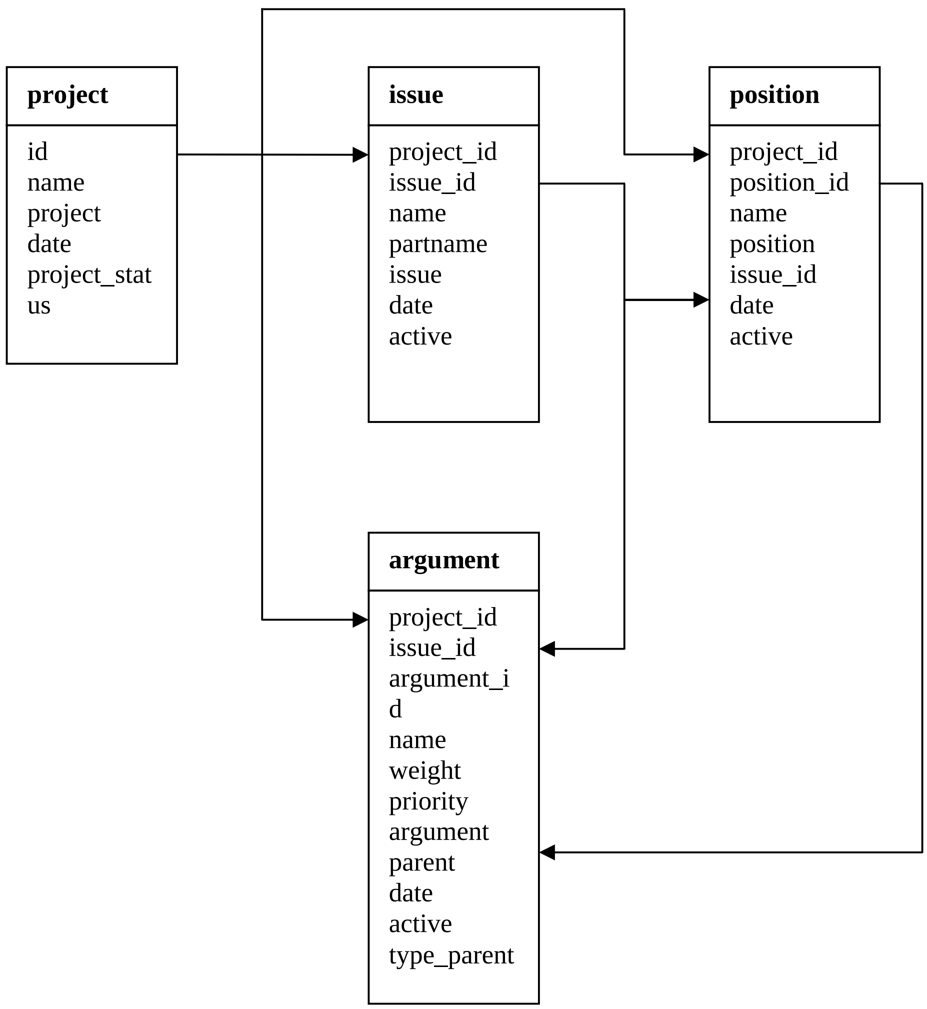


Figure 3.3. Database Relation

3.2. CONFLICT RESOLUTION BASED ON COMPUTATIONAL ARGUMENTATION

In the intelligent argumentation subsystem for conflict resolution, the dialog for a design issue is captured as a weighted directed graph called a dialog graph [24], as shown in Figure 3.4. The nodes denoted by circles are *Positions* i.e. the design alternatives, and the nodes denoted by rectangles are *Arguments*. An arc represents a relationship (attack or support) from the originating argument node to the terminating argument or position node. The weight assigned to an argument is the argument strength. It is the measure of an argument's degree of attack or support of either a position or another argument in the design dialog graph [24]. The weight value is a real number between -1 and 1. A positive number denotes support and a negative number denotes attack while zero denotes indecision. The strength of the argument is viewed as a fuzzy set and linguistic labels are used to represent the strength. Linguistic labels are used as Strong Support, Median Support, Indecisive, Medium Attack and Strong Attack to denote the strength of an argument or a position. A fuzzy inference engine is developed for argument reduction. The fuzzy inference engine has two inputs and one output. The inputs are the strengths of the argument to be reduced and the argument right above it. The output of the fuzzy inference engine is the reduced strength of the argument. The complexity of the network is reduced level by level using a fuzzy inference engine to the point where every argument under a position connects to it directly.

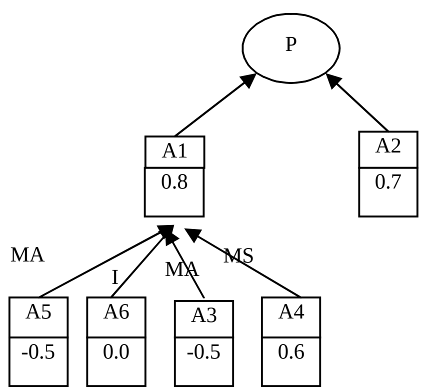


Figure 3.4. Argumentation Network

3.2.1. Fuzzy System and Defuzzification. In this fuzzy inference engine, there are 25 fuzzy rules. Apply fuzzy membership functions SS (Strong Support), MS (Medium Support), I (Indecisive), MA (Medium Against) and SA (Strong Against) respectively to two input variables X and Y. A weight value W_i is assigned to each fuzzy rule by taking the minimum of the membership function values associated with that entry. The output variable *Z* also has five fuzzy membership functions associated with it i.e. SS, MS, I, MA and SA. Specific values are assigned to these fuzzy sets, i.e. SS = 1, MS = 0.5, I = 0, MA = -0.5 and SA = -1. V_i is used to denote all the specific value for each entry fuzzy rule. Therefore, the range of W_i and V_i is respectively from 1 to 25. The system output is computed as follows:

$$output = \frac{\sum_{i=1}^{25} W_i \times V_i}{\sum_{i=1}^{25} W_i} \qquad (1)$$

3.2.2. Argumentation Reduction Level by Level. This fuzzy inference engine is capable of being applied to the whole argumentation network. The two input variables are respectively one argument and the argument which is one-level above and directly pointed by this argument. Then they move this argument up one-level above and assign the output value as the new strength of the moved argument.

4. INCORPORATION OF PRIORITY OF PARTICIPANTS INTO INTELLIGENT ARGUMENTATION

After the original argumentation network has been reduced to the one-level argumentation network where each argumentation directly points to the position, the original method is sum all the updated strength values together [2][3] and obtain the final favorability factors. However in this paper, instead of simply summation, priority is incorporated into the system at this level.

Each participant is assigned a priority. The priority value ranges from 0 to 1. The higher priority a participant has, the more powerful his/her argument is. A priority represents a participant’s authority in a collaborative work. In the previous research, arguments move up in the argumentation network in the process of argumentation reduction. It is reasonable to assume the priority value of each participant is not changed no matter where this participant’s argument is moved to in the network. Two methods for incorporating priority into an argumentation network are discussed below.

4.1. WEIGHTED SUMMATION

Weighted summation is a simple and easy-to-understand way to assess the impact of priority on the final favorability factor. Previous research computed a position's favorability factor by summing up all the final strengths of its arguments. Now the favorability is computed as a weighted sum of strengths of arguments with priority as follows:

$$\text{Favorability} = \sum_{i=1}^{m} p_i \times w_i \qquad (2)$$

where w_i is strength of argument i and p_i is priority of the participant who raises argument i. As an example, a reduced final argumentation network [2] [3] is shown in Figure 4.1. Assume that the priority of participant A is 1, the priority of B is 0.7, and the priority of C is 0.5. The favorability of position P calculated using equation 1 is 0.78.

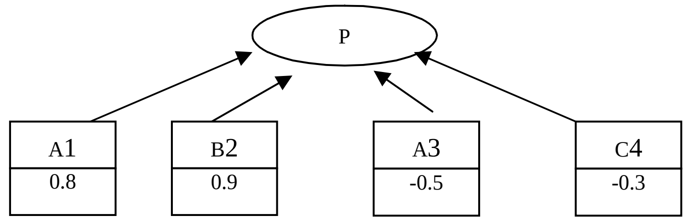


Figure 4.1. The Highest Level where Every Argument Directly Connects to the Position

4.2. REASSESSMENT OF ARGUMENT'S STRENGTH BASED ON PARTICIPANT'S PRIORITY

Another technique to incorporate priority into an argumentation network of the collaborative engineering design system is to re-assess the strength of an argument based on the priority of the participant who raises the argument. It is based on the following priority re-assessment heuristic rules:

- **General Priority Re-assessment Heuristic Rule 1**: If the owner of argument A has a higher priority, the strength of this argument should be higher than its weight alone.
- **General Priority Re-assessment Heuristic Rule 2**: If the owner of an argument has a lower priority, the strength of this argument should be lower than its weight alone.

As the linguistic labels used to represent the degrees of supporting and attacking are Strong Support (SS), Medium Support (MS), Indecisive (I), Medium Attack (MA) and Strong Attack (SA), and the linguistic labels for priority are high (H), medium (M) and low (L), the above two General Argumentation Heuristic Rules can be extended to fifteen fuzzy priority re-assessment rules in a Fuzzy Association Memory (FAM) shown in Figure 4.2.

	H	M	L
SS	SS	SS	MS
MS	SS	MS	I
I	I	I	I
MA	SA	MA	I
SA	SA	SA	MA

Figure 4.2. Fuzzy Priority Re-assessment

Using this fuzzy inference engine, priority can be incorporated and weight to evaluate the strength of an argument. Fuzzy membership functions are used to quantitatively characterize linguistic labels, such as low priority. In previous research, the fuzzy membership function chosen for the weight is the piecewise linear trapezoidal function.

The fuzzy membership function chosen for representing priority is also the piecewise linear trapezoidal function. The three fuzzy sets are Low, Medium and High, and the membership functions are shown in Figure 4.3(A). Figure 4.3(B) shows the five membership functions for the weight fuzzy sets.

Fuzzy inference rules combine two input fuzzy sets and associate with them an output set. The input sets are combined by means of operators that are analogous to the usual logical conjunctives “and”. The fuzzy argumentation rules are stored and represented by a fuzzy association memory (FAM) matrix as shown in Figure 4.4. There are two inputs X and Y. The priority input variable (Y) has three input sets associated with it, which are labeled as “H,” “M,” “L.” The argument weight input variable (X) has five fuzzy sets associated with it, which have been labeled as “SA,” “MA,” “I,” “MS,” and “SS.” The output variable, Z, also has five output sets that are same as the argument strength input sets. Each FAM matrix entry is an output fuzzy set associated with a fuzzy rule. For example, the shaded part in Figure 4.4 represents the rule: “If X is Strong Support (SS) and Y is L (low priority), then Z is Medium Support (MS).”

The membership functions for the fuzzy sets SS, MS, I, MA and SA are denoted by F_{SS}, F_{MS}, F_I, F_{MA} and F_{SA} respectively. A particular value x of the input variable X then has membership degrees $F_{SS}(x)$, $F_{MS}(x)$, $F_I(x)$, $F_{MA}(x)$ and $F_{SA}(x)$. For example, with the trapezoidal membership functions shown in Figure 4.3 (B) and a value x = -0.7, it would be:

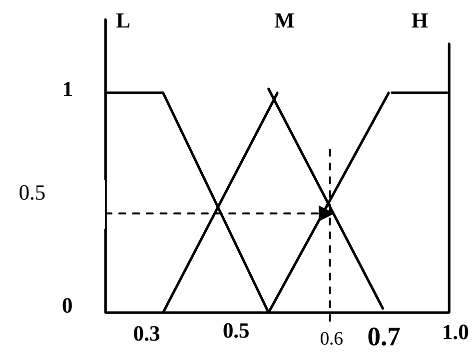
$F_{SS}(-0.7) = 0.0$

$F_{MS}(-0.7) = 0.0$

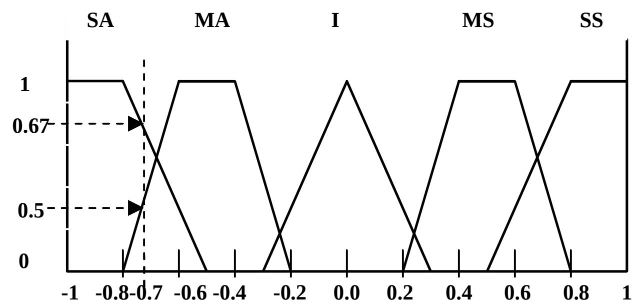
$F_I(-0.7) = 0.0$

$F_{MA}(-0.7) = 0.5$

$F_{SA}(-0.7) = 0.67$



(A)



(B)

Figure 4.3. Membership Functions (A) for Priorities (B) for Strength

	H	M	L
SS	SS	SS	MS
MS	SS	MS	I
I	I	I	I
MA	SA	MA	I
SA	SA	SA	MA

Figure 4.4. FAM Matrix

Similarly, a particular value for y of the input variable *Y* would have membership degree values $P_H(y)$, $P_M(y)$, $P_L(y)$. The value y = 0.6 as shown in Figure 4.5 would result in

$P_H(0.6) = 0.5$

$P_M(0.6) = 0.5$

$P_L(0.6) = 0.0$

Consider x = -0.7 and y = 0.6 as values of the input variables X and Y. A strength value is assigned to each entry in the FAM matrix by taking the minimum of the membership function values associated with that entry. Now consider the FAM matrix entry corresponding to X, a member of the fuzzy set MA, and Y, a member of the fuzzy set M. Figure 4.5 illustrates the membership value for the priority input. The strength w_1 associated with the entry would be computed as:

$w_1 = \min [F_{MA}(-0.7), P_M(0.6)]$

$= \min [0.5, 0.5]$

$= 0.5$

Only those FAM matrix entries that have nonzero membership-function values for both *X* and *Y* will have nonzero strengths associated with them. The shaded entries in the Figure 4.6 show the four activated rules for the values in the example. In addition to w_1, there are three more non-zero weights.

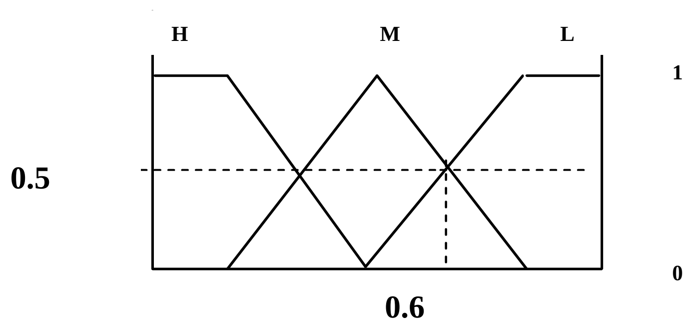


Figure 4.5. Membership Value for Priority Input

w_2 = min [F_{MA}(-0.7), P_H(0.6)]

= min [0.5, 0.5]

= 0.5

w_3 = min [F_{SA}(-0.7), P_M(0.6)]

= min [0.67, 0.5]

= 0.67

w_4 = min [F_{SA}(-0.7), P_H(0.6)]

= min [0.67, 0.5]

= 0.67

The output variable *Z* also has five fuzzy sets associated with it, i.e. SS, MS, I, MA and SA. Specific values are assigned to these fuzzy sets, i.e. SS = 1, MS = 0.5, I = 0, MA = -0.5 and SA = -1. The system output is computed as follows:

$$\text{Output} = \frac{(w1 . MA + w2 . MA + w3 . SA + w4 . MA)}{(w1 + w2 + w3 + w4)}$$

= -0.89

	H	M	L
SS	SS	SS	MS
MS	SS	MS	I
I	I	I	I
MA	SA	MA	I
SA	SA	SA	MA

Figure 4.6. The Fuzzy Association Memory

5. DETECTION OF SELF-CONFLICTING ARGUMENTS

5.1. OVERVIEW

The robustness of an argumentation network is fundamental to making a convincing decision over multiple positions. However, the self-conflicting problem may hamper the robustness of the whole network and cause negative consequences.

The existence of self-conflicting arguments means that several of arguments of a participant are contradictory among themselves. In a complicated collaborative design environment with a number of participants, the self-conflicting problem could take place frequently, and self-conflicting arguments are not easy to detect in many cases. The existence of self-conflicting is such a major issue in a collaborative design environment that it is often difficult to obtain a convincing decision.

If a participant has some self-conflicting arguments in the network, then no matter how powerful this participant is, his arguments will provide some unaccountable and confusing information instead of positively contributing to the argumentation process.

5.2. SELF-CONFLICTING ALGORITHM

Here is a simple example. In the network shown in Figure 5.1, the owner of argument A1 is O_1, A2 attacks A1, A4 supports A2, and A5 supports A4; therefore it can be easily concluded that A5 attacks A1. But if the owner of argument A5 is also O_1, then A1 and A5 are a pair of self-conflicting arguments of owner O_1.

In this simple example, it is easy to detect the self-conflict. However, in a large network with many self-conflicting arguments, they cannot be easily detected by simple human observation. The self-conflicting problem is divided into two categories. The first one is one-to-one self-conflicting, which includes two obviously contradictory arguments belonging to one owner. The second is multiple self-conflicting, a more complicated relationship where a few arguments of one owner are conflicting with each other. This kind of self-conflicting is computationally difficult to discover. It is necessary to develop an effective algorithm to detect and remove self-conflicting arguments, no matter what type of self-conflicting it is. Using an algorithm shown in Figure 5.2, many self-conflicting arguments can be detected by traversing all offspring argument nodes of argument node A.

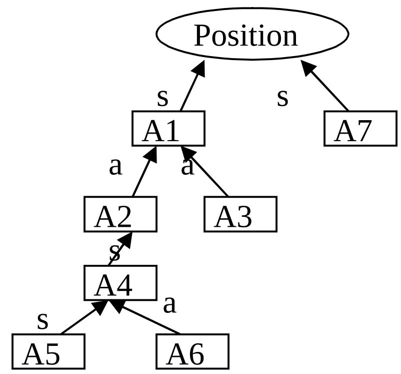


Figure 5.1. A Simple Example to Illustrate Self-conflicting

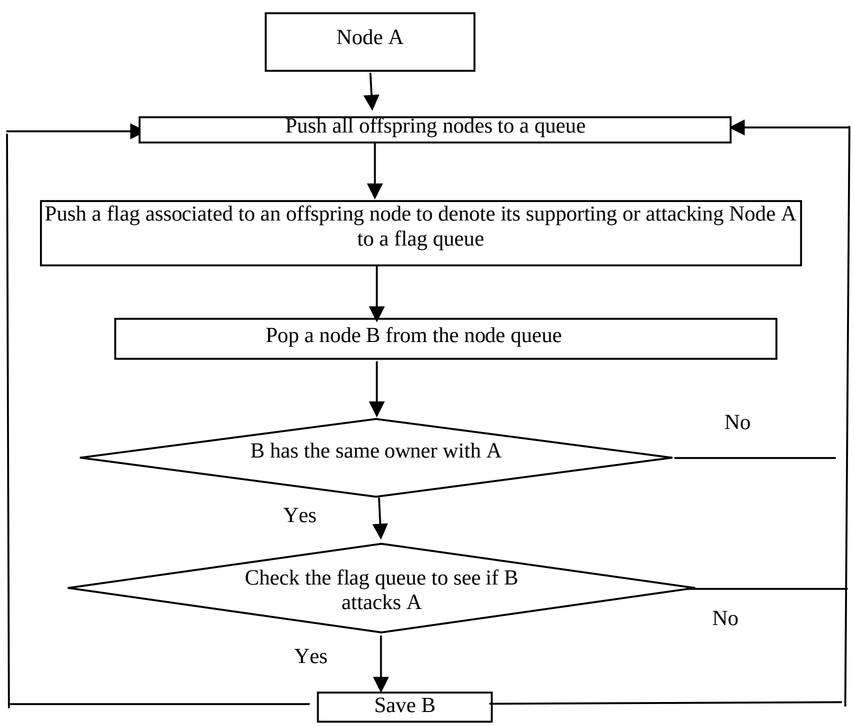


Figure 5.2. The Algorithm to Resolve the Self-conflicting

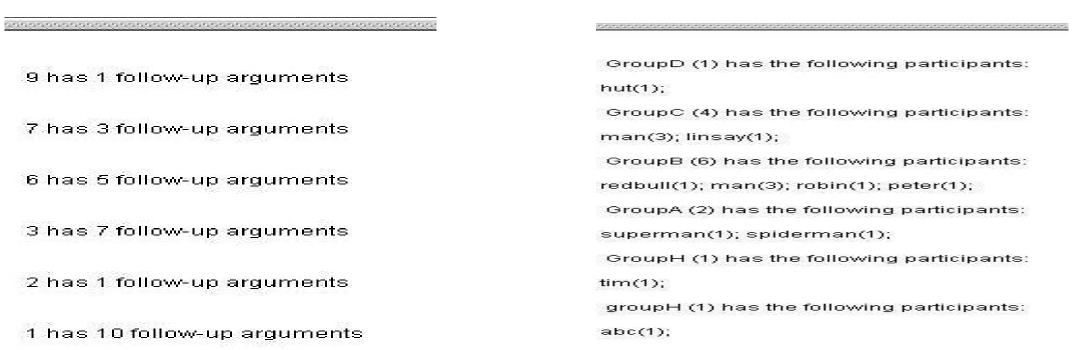
6. QUANTITIVE ANALYSIS OF ARGUMENTATION NETWORK

6.1. OVERVIEW

Normally, when people start participating in a large complicated argumentation network, they do not even know where to start. Therefore, it is necessary to provide a little statistical information about the network to help users comprehend it. This research proposes to provide two types of statistical information about an argumentation network: owner-oriented and argument-oriented.

6.2. TWO QUANTITIVE ANALYSIS TOOLS

Owner-oriented information indicates participation of each participant and its relation with other owners. It shows how many arguments one participant owns and which group this participant belongs to. Figure 6.1(B) shows an example of how this system presents owner-oriented information. Argument-oriented information shows which arguments are popular. Normally, a popular argument has many more follow-up arguments supporting or attacking it. Figure 6.1(A) shows an example of how this system presents argument-oriented information.



(A) (B)

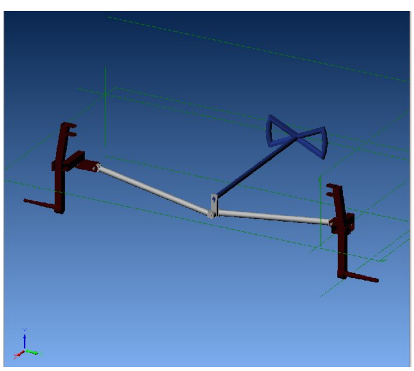
Figure 6.1. Quantities Analysis Tools (A) Participant Oriented Information (B) Argument-oriented Information

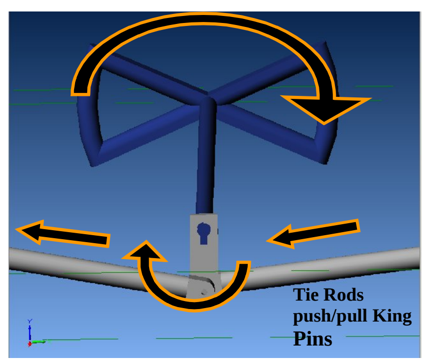
The self-conflicting argument detection technique and quantitative analysis tools are implemented in a collaborative engineering design system. After the detection of self-conflicting arguments, the design team is able to discuss which one needs removing from the argumentation network.

7. ILLUSTRATIVE EXAMPLE AND SENSITIVITY ANALYSIS

7.1. SOLAR CAR MECHANISM

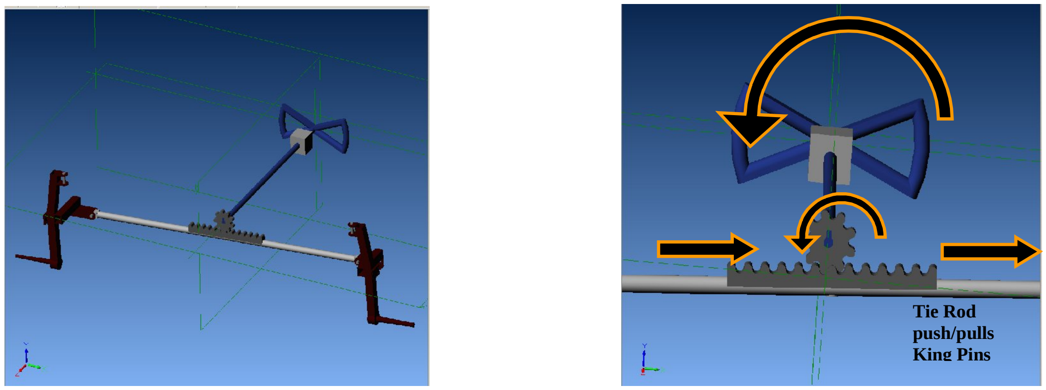
UMR's Solar Car Team, a student design team, which won the competition in 2003 and placed fourth in the 2005 American Solar Challenge, is confronted with many challenging issues including resolving various design conflicts. One of the design tasks is the redesign of the solar car steering system. The steering is located on the front two wheels of the solar car with the rear wheel being the drive wheel. For the upcoming '08 race, all of the teams will be required to utilize a steering wheel for their car. This was quite a change from the previous cars which all used a push pull cable steering system. As a team, they quickly reduced their design choices to a simple push pull lever design (Figure. 7.1) and a rack and pinion system (Figure. 7.2). Both designs seemed to have pros and cons associated with each of them, making the decision between the two, tedious. Design 1 would be much easier to manufacture, but posed a safety issue with the rate of turn being most sensitive at the apex, where minor adjustments would be made at highway speeds. Design 2 did not have this safety issue, but the difficulty of manufacturing a steel rack and pinion and the added weight of a steel design lead the team to finally decide upon design 1.





(A) (B)

Figure 7.1. Design 1 (A) Regular (B) Zoom-in



(A) (B)

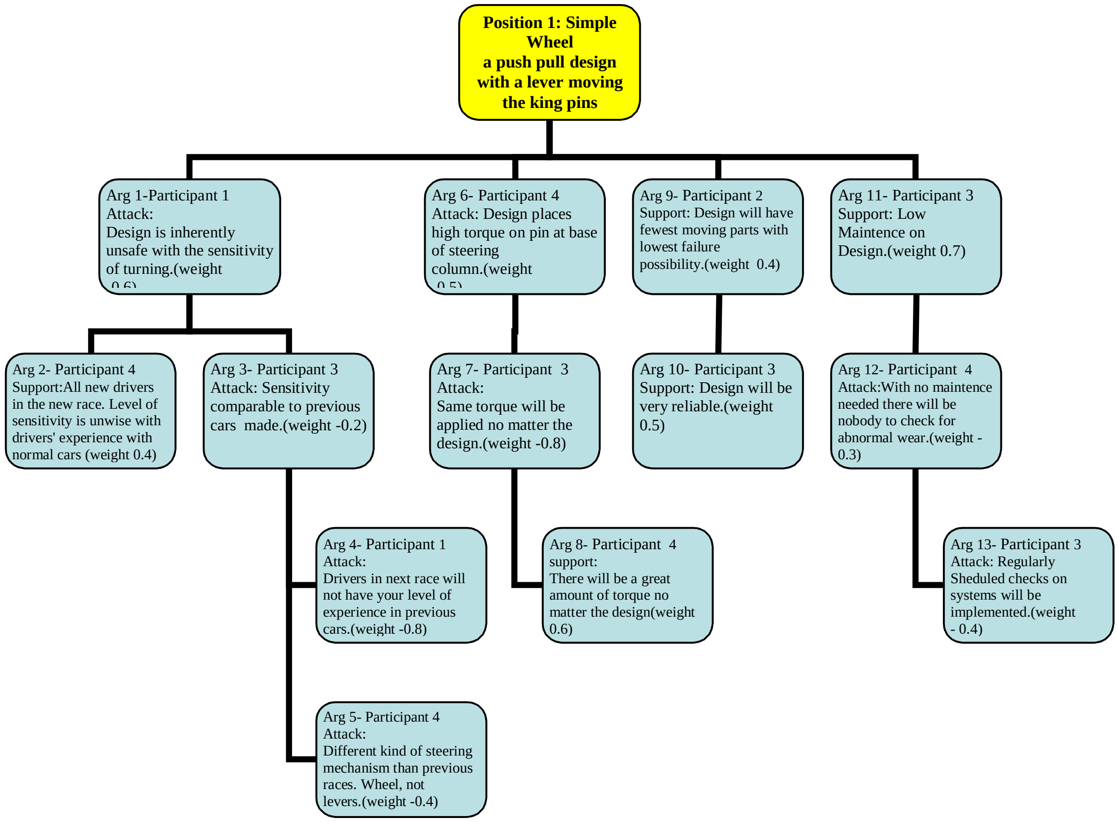
Figure 7.2. Design 2 (A) Regular (B) Zoom-in

There are four participants on this design team. They are Participant 1, Participant 3, Participant 2, and Participant 4. Participant 2 is the vice president of manufacturing who is more experienced in the design aspect of the solar car team. Participant 3 is an experienced member of solar car team who is replacing Participant 2 in the upcoming race year. Participant 1 is one of the advisors for the solar car team. Participant 4 is a new participant on this design team. Based on responsibilities and experience of participants, their priorities are assigned as in Table 7.1.

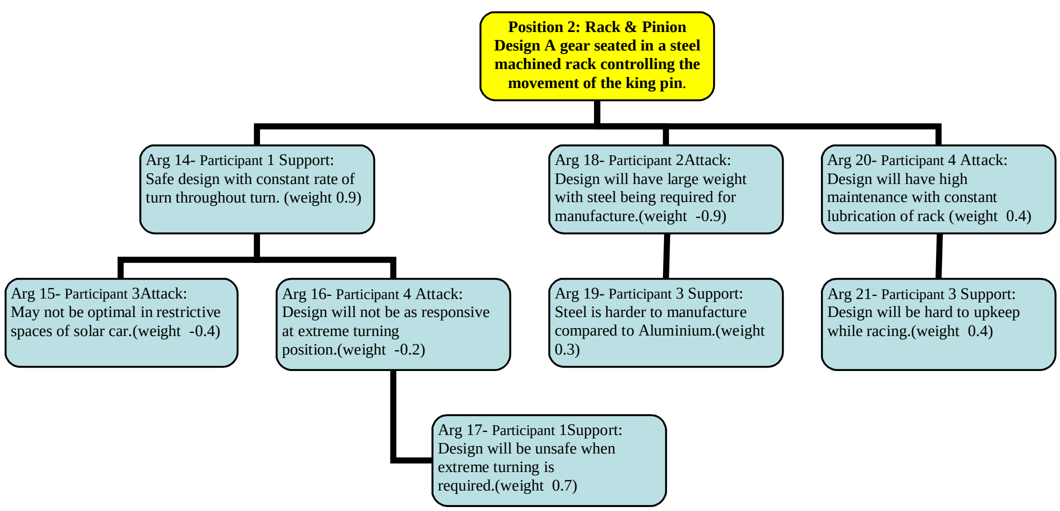
Table 7.1. Each Participant's Priority

Participant 1	0.5
Participant 2	0.6
Participant 3	0.5
Participant 4	0.3

When the Positions are entered, the participants can enter arguments either supporting or attacking the positions. Its corresponding argumentation tree is shown in Figure 7.3.



(A)



(B)

Figure 7.3. Designs (A) 1: a Pusher Design (B) 2: Rack and Pinion Design

7.2. SELF-CONFLICTING ARGUMENTATION DETECTION

As shown in the Figure 7.4, under position 1 Participant 4's two arguments--"Design places high torque on the pin at the base of the steering column" and "There will be a great amount of torque no matter the design"-- are self-conflicting with each other. And under position 2, Participant 1's arguments—"Safe design with constant rate of the turn throughout the turn" and "Design will be unsafe when extreme turning is required"—conflict with each other. Then the system is able to precisely detect the self-conflicting arguments as shown in Figure 7.4

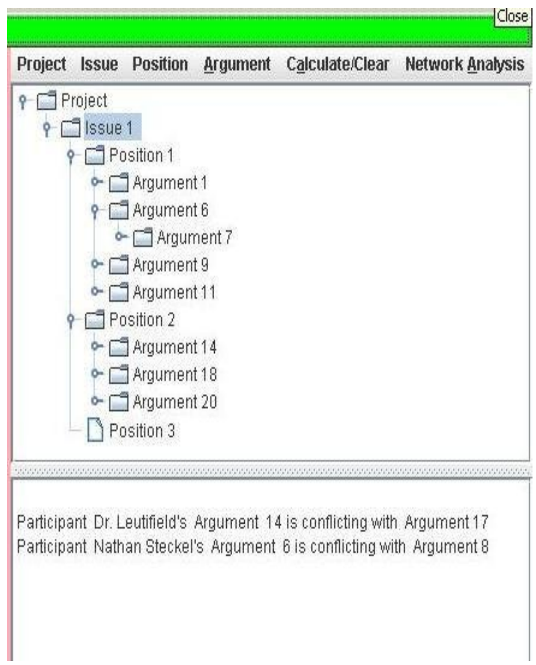
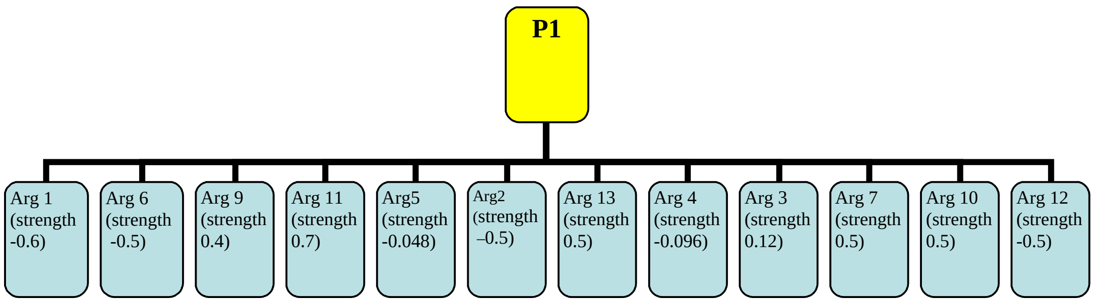


Figure 7.4. Self-conflicting Detection

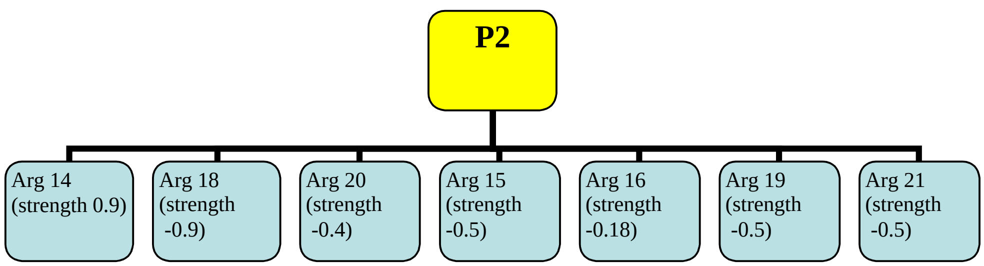
After the detection of self-conflicting, participant can decide which argument needs to be removed and which one needs to stay. Here in this example, the four team members decide to get rid of argument 17 and argument 8.

7.3. ARGUMENTATION WITH PRIORITY ASSESSMENT AFTER REMOVAL OF SELF-CONFLICTING ARGUMENTS

The priority is incorporated at the highest level of argumentation network. Without incorporation of priority, after reducing to the highest level, the updated weight for each argument should be like the following Figure 7.5:



(A)



(B)

Figure 7.5. Reduced Argumentation Trees (A) for Position 1 (B) for Position 2

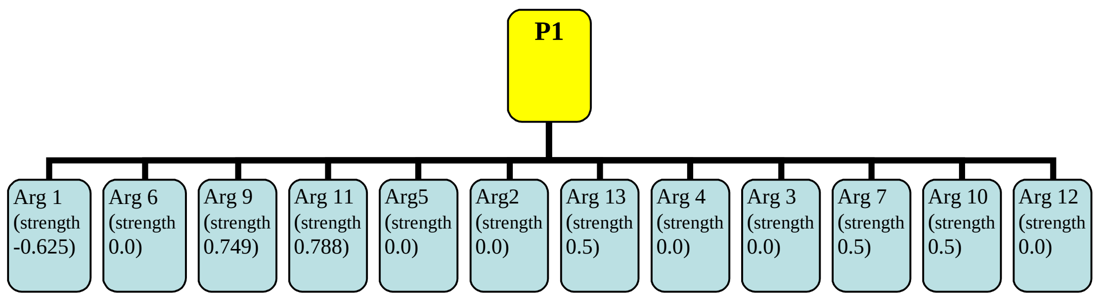
After using fuzzy inference engine to incorporate priority at the highest level, the strength value of each argument should be like Figure 7.6.

Finally, all the value attached to each argument is added up to obtain the favorability factor of each position.

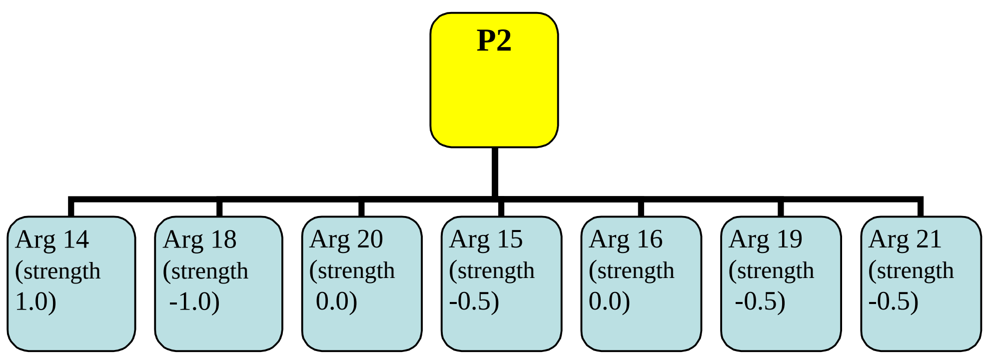
For weighted summation method, each participant’s priority is used as weight to sum up all the strength value.

The designers can view the results by clicking the Calculate Result button. Two results are available in the system, one is weighted summation priority incorporation, and the other is fuzzy inference engine priority incorporation. The two results of the conflict that will be displayed by the system are as Figure 7.7.

Both Results says that Position 1 would be a better option for replacing the previous failed Latch Mechanism. The Solar Car team at UMR is ready to use Design 1, which says that the Conflict Resolution module gave a result consistent with the expected one.

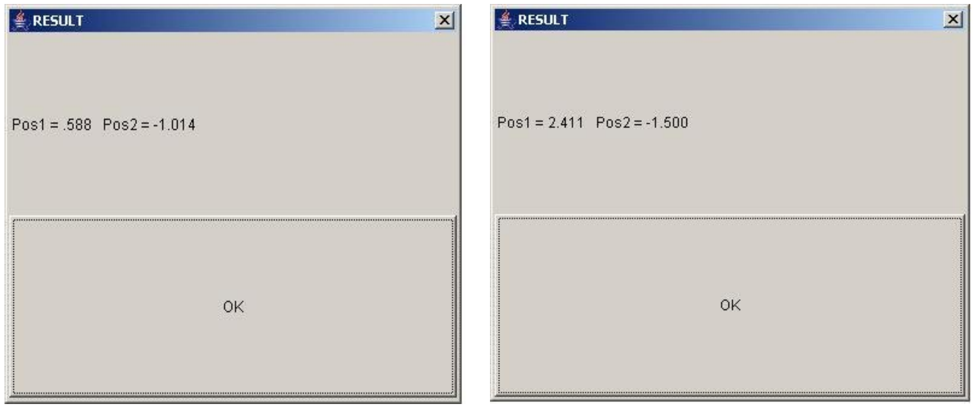


(A)



(B)

Figure 7.6. After Incorporating Priority (A) Position 1 (B) Position 2



(A) Weighted Summation (B) Fuzzy Priority Re-assessment

Figure 7.7. Favorability Factors with Priorities in Table 7.1

7.4. INCORPORATION OF PRIORITY

Two methods are applied to incorporate priority into the system – weighted sum and fuzzy inference engine. The priority value is assigned according to the experience and skill of each participant. In this section, how the priority impact on the system is validated. On position 1, Participant 1 holds an attacking argument while Participant 2 and Participant 3 hold supporting arguments. On position 2, it is the other way around (shown in Table 7.2). Participant 4 holds attacking arguments both positions.

How the favorability of each position changes is presented when the priority changes in two different situations.

Table 7.2. Each Participant’s Stand in this Project

	Position 1	Position 2
Participant 1	Attack	Support
Participant 2	Support	Attack
Participant 3	Support	Attack
Participant 4	Attack	Attack

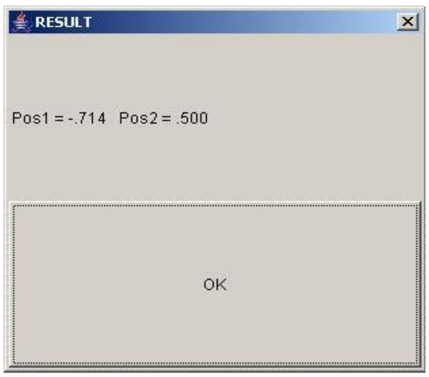
Situation 1(as shown in Table 7.3): Participant 1's priority is extremely high. The priorities of Participant 2 and Participant 3 are is extremely low. The increase of the priority value of Participant 1and the decrease of priority of Participant 2 and Participant 3 are supposed to be followed by the augment of the favorability factor of position 1 and the dwindling of favorability factor of position 2.

Table 7.3. Priority for Situation 1

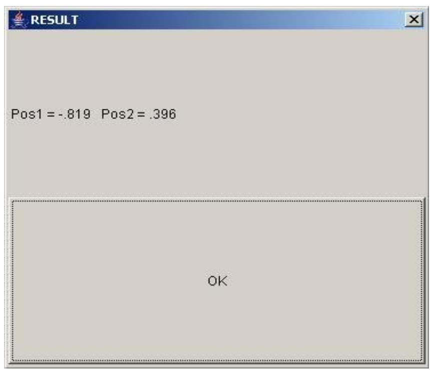
Participant 1	0.9
Participant 2	0.1
Participant 3	0.1
Participant 4	0.3

Expected result: The favorability factor of Position 2 is highly boosted while the favorability factor of Position 1 is decreased.

Obtained result: The values in Figure 7.8 clearly indicate the increase of favorability of position 2 and the decrease of favorability of position 1 compared with those in Figure 7.7 for both methods of incorporating priorities.



(A)



(B)

Figure 7.8. Favorability Factors for Situation 1 (A) Fuzzy Priority Re-assessment (B) Weighted Summation

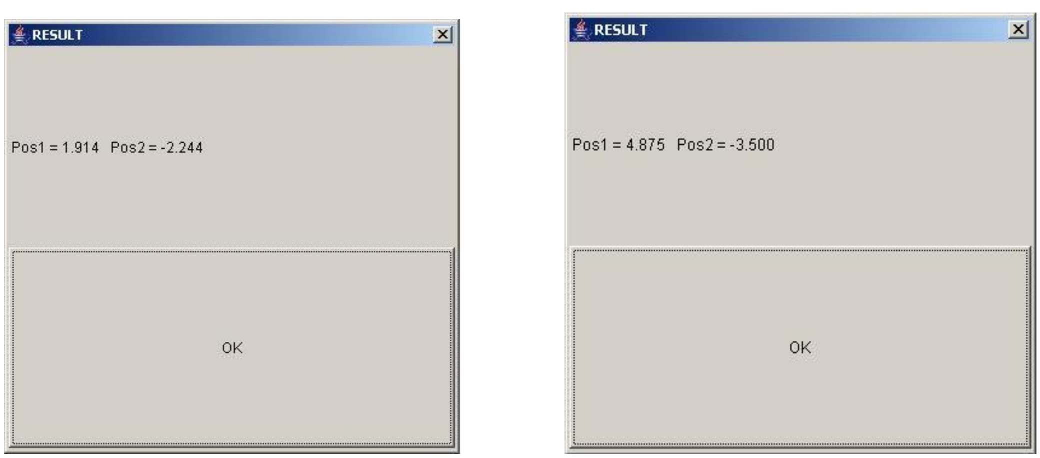
Situation 2(as shown in Table 7.4): The priority of Participant 2 and Participant 3's priority is extremely high; Participant 1's priority is extremely low. The increase of the priority value of Participant 2 and Participant 3 and the decrease of the priority value of Participant 1 are supposed to be followed by the increase in the favorability of position 1 and the decrease of favorability of position 2.

Expected result: the favorability factor of Position 1 is highly boosted while the favorability factor of Position 2 is decreased.

Obtained result: The values in Figure 7.9 clearly indicate the decrease of favorability of position 2 and the increase of favorability of position 1 compared with those in Figure 7.8 for both methods of incorporating priorities.

Table 7.4. Priority for Situation 2

Participant 1	0.1
Participant 2	0.9
Participant 3	0.9
Participant 4	0.3



(A) (B)

Figure 7.9. Favorability Factors for Situation 2 (A) Weighted Summation (B) Fuzzy Priority Re-assessment

As can be seen, change in priority values may lead to very different favorability values and hence the final. From the above analysis and examples, it can be concluded that the incorporation of priority is important to decision making between alternatives.

8. CONCLUSION AND FUTURE WORK

The enhanced argumentation model described in the thesis is an evolution of the previous one. It takes more important factors of arguments into consideration while the previous system only presents and implements the novelty idea of reduction of arguments in the argumentation network and introduces the fuzzy inference engine.

The main contribution of this thesis includes several significant improvements to the argumentation model and subsystem of a previously existing intelligent collaborative engineering design system. Firstly, the priority of participants is incorporated into the argumentation model using two different techniques: weighted summation and priority-based argument strength re-assessment. Secondly, this research develops an effective approach for detection of self-conflicting arguments. Several new analytical tools help participants analyze argumentation in a large, complex argumentation network. The proposed methods and technique are validated using a solar car design case study. It demonstrates their effectiveness.

The desirable future work includes improving the self-conflicting detecting mechanism and involving more properties of arguments such as requirements of design. Currently the system is capable of detecting self-conflicting arguments according to their stands. In the future, the designers might run across more complicated self-conflicting situations such as the self-conflicting arguments all positively contribute to the argumentation network and the controversial decision of which arguments should be removed. To achieve this, a mechanism must be designed to analyze the self-conflicting arguments and offer a more accurate solution. With the involvement of more objective properties of arguments the system can gain more information in order to assign the weight and priority more objectively and eventually a mechanism for assignment of weight and priority needs to be invented and implemented.

BIBLIOGRAPHY

[1] Karacapilidis, N., and Papadias, D. (1998), "HERMES: Supporting Argumentative Discourse in Multi-Agent Decision Making," Proceedings of the 15th National Conference on Artificial Intelligence (AAAI-98), Madison, WI, AAAI/MIT Press, pp. 827-832

[2] Liu, X. F., Raorane, S., Zheng, M., and Leu, M. C., "An Internet Based Intelligent Argumentation System for Collaborative Engineering Design," *Proceedings of the 2006 IEEE International Symposium on Collaborative Technologies and Systems*, *(CTS 2006)*, Las Vegas, Nevada, May 14-17, 2006, pp. 318-325.

[3] Xiaoqing(Frank) Liu and Samir Raorane, Ming C. Leu, "A Web-Based Intelligent Collaborative System for Engineering Design," a chapter in a book edited by W. D. Li, S. K. Ong, Andrew Y. C. Ne, and Chris McMahon: Collaborative Product Design & Manufacturing Methodologies and Applications, Springer,2007.

[4] Alibre Design(2005), "Alibre Design software," http://www.alibre.com

[5] Sriram, R. (2002), Distributed and Integrated Collaborative Engineering Design, Sarven Publishers, 2002.

[6] Reddy, R., Srinivas, K., Jagannathan, V., and Karinthi, R. (1993), Computer Support for Concurrent Engineering-Guest Editors' Introduction, IEEE Computer, 26(1): 12-16.

[7] Zhou, J. and Lin, G. (1999), "Implementation of Collaborative Design Environment Based on Single User CAD Systems," Third International Conference Knowledge-Based Intelligent Information Engineering Systems, 3 Aug.-1 Sept. 1999, pp.78 – 83.

[8] Pahng, G.-D.F., Seockhoon B. and Wallace, D. (1998), "A Web-based collaborative design modeling environment," Proceedings of the Seventh IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises (WET ICE '98), 17-19 June 1998, pp. 161 – 167.

[9] Klein, M. (1997), "Capturing Geometry Rationale for Collaborative Design Enabling Technologies," Infrastructure, Proceedings Sixth IEEE workshops on for Collaborative Enterprises, June 18-20, 1997, pp. 24 – 28.

[10] Zhang, H., Wu, H., Lu, J. and Chen, D. (2000), "Collaborative Design System for Performance," Proceedings of Academia/Industry Working Conference on Research Challenges, 27-29 April 2000, pp. 59 – 63.

[11] Siddique, Z. (2004), "Internet Design Studio", Proc. of ASME 2004 Design Engineering Technical Conferences and Computers and Information in Engineering Conference, Salt Lake City, Utah, Sept, 2004.

[12] Wang, L, Wong, B B., Shen, W., and Lang, S. (2001), "A Web-based Collaborative Workspace using Java 3D", Proc. of the Sixth International Conference on Computer Supported Cooperative Work in Design, July 2001, pp. 77-82.

[13] Peng S., Tang, M. and Dong, J. (2001), "Collaborative Model for Concurrent Product Design," The Sixth International Conference on Computer Supported Cooperative Work in Design, 12-14 July 2001, pp. 212 – 217.

[14] Lee, J. Y., Han, S. B., Kim, H. and Park, S. B. (1999), "Network-Centric Feature-Based Modeling," Proceedings of the Seventh Pacific Conference on Compute Graphics and Applications, 5-7 Oct. 1999.

[15] Chang, H-C., Liu, X. F. and Lu W. F. (1999), "A WWW Based Collaborative System for Integrated Design and Manufacturing," International Journal of Concurrent Engineering: Research and Application (CERA), 7 (4): 319-334.

[16] Chan, S., Ng, C. and Ng, V. (1999), "Real-Time Collaborative Design of Complex Objects on the Web," IEEE SMC '99 Conference Proceedings, 1999 IEEE International Conference on Systems, Man, and Cybernetics, Volume: 2, 12-15 Oct. 1999, pp. 120 – 125.

[17] CollabCAD (2005), "CollabCAD software," http://www.collabcad.com.

[18] Toulmin, S. (1958), The Uses of Argument, Cambridge University Press, 1958.

[19] Conklin, J. and Begeman, M. (1988), "gIBIS: A Hypertext Tool for Exploratory Policy Discussion," Transactions on Office Information Systems, 6(4):303– 331.

[20] Lease, M. and Lively, J. L. (1990), "Using an Issue-Based Hypertext System to Capture Software Life-Cycle Process," Hypermedia, Vol. 2, No. 1, pp. 34 – 45.

[21] Ramesh, B. and Dhar, V. (1992), "Supporting Systems Development by Capturing Deliberations during Requirements Engineering," IEEE Transactions on Software Engineering,18(6):498 – 510.

[22] Sillince, J. (1997), "Intelligent Argumentation Systems: Requirements, Models, Research Agenda, and Applications," in Encyclopedia of Library and Informati Science, edited by Allen Kent, Vol. 19, No. 22, pp. 176 – 217, Marcel Dekker.

[23] Alexander, I. (2003). Modelling argumentation, Toulmin-style. April 15, 2005. http://easyweb.easynet.co.uk/~iany/consultancy/papers.htm

[24] Sigman, S. and Liu, X. F. (2003), "A Computational Argumentation Methodology for Capturing and Analyzing Design Rationale Arising from Multiple Perspectives," International Journal of Information and Software Technology, Vol. 45, No. 3, March 2003.

[25] Parsons, S., and Jennings, N. R. (1996) "Negotiation through Argumentation -- A Preliminary report," Proceedings of the 2nd International Conference on Multi Agent Systems, pp. 267-274.

[26] Morge, M. (2004), "Computer Supported Collaborative Argumentation", Proc. of the 4th Workshop on Computational Models of Natural Argument, Spain, 2004, pp. 69-72.

[27] Thurston, D. L. (2006), "Multi-attribute Utility Analysis of Conflicting Preferences," in book Decision Making in Engineering (editors: K. E. Lewis, W. Chen and L. C. Schmidt), ASME Press, pp. 125-134.

[28] Lu, S. C-Y., Zhang, J., Wang, C., and Grobler, F. (2005), "Modeling Design Processes and Stakeholder Perspectives to Support Collaborative Engineering Negotiation," International Journal of Computer Applications in Technology, 23(1): 2-12.

[29] Middlekoop, T., Pepyne, D. L., and Deshmukh, A. (2006), "Analysis of Negotiation Prototols for Distributed Design," in book Decision Making in Engineering (editors: K. E. Lewis, W. Chen and L. C. Schmidt), ASME Press, p 265-280.

[30] Jin, Y. and Danesh, M. R. (2006), "Value Aggregation for Collaborative Desig Decision-Making," in book Decision Making in Engineering (editors: K. E. Lewis, W. Chen and L. C. Schmidt), ASME Press, pp. 291-299.

[31] Lewis, K. E., Chen, W., Schmidt, L. C. (2006), *Decision Making in Engineering Design*, ASME Press, New York.

[32] Archambault, N. Durand, N.(2004), *Scheduling heuristics for on-board sequential air conflict solving*, Volume: 1, pp - 3.1-9 Vol.1.

[33] Charles U. Martel, W.Melody Moh Teng-Sheng Moh (1996),*Dynamic Prioritized Conflict Resolution on Multiple Access Broadcast Networks*,IEEE Transactions on Computers ,Volume 45 , Issue 9, pp. 1074 – 1079.

[34] Belnap, N. D., Jr. "A Useful Four-Valued Logic," In *Modern Uses of Multiple-Valued Logic*, M. Dunn and G. Epstein, editors, 1977, pp. 8–41.

[35] Bochman, A., "Propositional Argumentation and Causal Reasoning," International Joint Conference on Artificial Intelligence, 2005, pp. 388-393.

VITA

Man Zheng was born in Xi’an, China on December 27th, 1979. She graduated from Xi’an No.6 High School, China and later received her B.S. degree in Computer Science from Shaanxi Normal University in June 2002 and M.E degree in Software Engineering from Zhejiang University. Since August 2005, she has been a master student in the Computer Science Department in the University of Missouri-Rolla. She has been a research assistant in the Computer Science Department under Dr. Frank Liu. She received her Masters in Computer Science degree at University of Missouri-Rolla in December 2007.