Technical Correspondence

Decision-Making Assistance in Engineering-Change Management Process

D. Habhouba, S. Cherkaoui, and A. Desrochers

***Abstract*—Effective engineering-change management (ECM) is a real challenge in mechanical engineering industry and manufacturing companies. Computer-aided design systems are usually connected to other systems such as ERP or product data management, but currently this integration does not provide effective means to manage engineering change (EC). While communication between multidisciplinary teams working on a project is known to have a significantly positive impact on the ECM, the communication between disciplines is generally performed solely through message exchange. Experts could feel the need to meet to agree on the requested changes, which in turn translates into longer design and manufacturing processes. There is a need for a system that assists human experts in making decisions about ECs. Such a system will considerably reduce the processing time following a change-request procedure. This paper proposes a collaborative tool named EchoMag, which assists designers and experts during the change-management process. The proposed system ensures the coherence of data between the various disciplines involved in the change process. EchoMag also assists experts in making decisions by proposing alternative solutions when change requests are not agreed upon. Software agents were used to implement EchoMag for which a prototype was developed. Results of the implementation are discussed.**

***Index Terms*—Computer-aided design (CAD), concurrent engineering, design process, engineering-change management (ECM), software agent.**

I. Introduction

Designers are sometimes compelled to make important ECs on products. These products could be documents, parts, or processes. While changes are easily verified when they are made in the first stages of the design process, making ECs in the last stages of the design process can affect the production cost seriously. Studies have, in fact, shown that the engineering-change management (ECM) could represent from 70% to 80% of the final product's cost [1].

Existing engineering-data management (EDM) and PDM product data management (PDM) software offer some change management modules. However, these currently are far from satisfying the needs of companies working in a concurrent engineering environment. Forced to implement some ECM approach, some organizations are content with improving their change management procedures, while others decide to develop proprietary systems to help them manage the engineering-change process more efficiently [2].

To carry out ECs, different disciplines involved in change requests usually must meet, negotiate, and approve or reject a proposed change. Furthermore, change requests in organizations using computer-aided design (CAD) tools generate numerous documents and require many signatures, which make the engineering-change process even more complex. Helping automate parts of this communication and negotiation process to find optimal solutions can help save time and money. Effective and intelligent tools for collaboration are necessary to carry out the cooperation between the various disciplines and ensure effective ECM.

The work presented in this paper proposes EchoMag, an agent-based tool that helps efficiently manage the collaboration process between different disciplines and validate the engineering-change requests. The tool can also assist experts in making decisions in case of conflicts. We chose agent technology because it is a good instrument to integrate diverse heterogeneous data belonging to various projects or disciplines. Agent abstraction is also a good way to represent the different experts or disciplines involved in the change validation process.

We propose an architecture that:

1) helps automate the EC process and assist designers in making sound EC decisions;
2) improves communication between disciplines involved in the change requests;
3) extracts information pertinent to EC requests made by designers.

To help validate the concepts and mechanisms of EchoMag, a working prototype was developed and integrated to a typical software environment that may exist in a mechanical engineering industry, namely, CAD tools and the attached databases managed within ERP or PDM systems.

II. Related Works

In spite of the great evolution PDM software have gone through, still some pertinent functionality has not yet been properly addressed or is merely nonexistent in current systems. Such is the case of EC functionality. SMARTEAM and other PDMs allow constraints checking on CAD products; however, users can only check constraints manually. In other words, they offer specification management tools that help user access specifications easily and plan verifications. Moreover, the PDMs do not offer any intelligence or decision-making assistance. Actually, in a concurrent engineering environment, the decision process regarding changes made on products uses a great number of relations between data and processes.

Several strategies are used in manufacturing companies to minimize the negative impact of change requests. Some companies choose, for example, to spend more time checking products before approving them so as to avoid change requests. Some others limit the possibility of making changes to only the early stages of the design [3], [4]. These strategies are unfortunately of little help in multidisciplinary environments handling relatively big projects. Thus, some efforts have been put in developing specialized tools that can help better manage ECs.

The allied concurrent engineering (ACE) is a process in which different companies or disciplines can work together to design a product. The ACE combines the two concepts of virtual collaboration [5] and concurrent engineering. The ACE approach effectively shares data and

Manuscript received November 30, 2009; revised April 21, 2010; accepted June 22, 2010. Date of publication November 18, 2010; date of current version April 19, 2011. This work was supported in part by the Consortium for Research and Innovation in Aerospace in Quebec (CRIAQ) and in part by the Natural Sciences and Engineering Research Council of Canada (NSERC) under the Project IP2CM. This paper was recommended by Associate Editor W. Shen.

D. Habhouba and S. Cherkaoui are with the Computer and Electrical Engineering Department, Université de Sherbrooke, Sherbrooke, QC J1K2R1, Canada (e-mail: dounia.habhouba@usherbrooke.ca; soumaya.cherkaoui@usherbrooke.ca).

A. Desrochers is with the Mechanical Engineering Department, Université de Sherbrooke, Sherbrooke, QC J1K2R1, Canada (e-mail: alain.desrochers@usherbrooke.ca).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSMCC.2010.2059013

processes of a product throughout its development cycle. The project is divided into two phases. The first phase develops the methodology of change management. The second phase develops the management change system. Chen *et al.* [6] studied the model of reference for the EC before developing their own model. The ACE's main objective is to effectively integrate all the resources of a company. In other words, the changes are propagated to other business systems. The success of the integration process depends on the integration of the different information systems of the organization. The implementation of ACE is an integration solution for the company rather than a module in an already existing system [7], [8]. Integrating all the systems of a company is not an easy task. Developing a module that extracts information from different system so as to verify consistency could be easier.

Tang *et al*. [9] explain in their paper how to use the design structure matrix (DSM) for change-propagation management in single and multiple domains. Multiple dependencies are possible between two items in a domain. Identifying these dependencies is very important to predict the impact of a change. The purpose of this kind of change propagation analysis is to assist decision makers in their decision regarding ECs. Interdomain change propagation consists of tracing the impact of the change starting from product to processes. The interdomain DSM could be browsed in two ways, from product to processes or from processes to product. Authors propose an implemented prototype that gives a good idea on all the dependencies of a system and the impact of an EC. Their system takes as input all the items interacting in the system, the DSM representation, direct dependency capture, and indirect dependency capture. As an output, the system gives all the affected items when an instigated item is specified (EC is triggered by this kind of item) and all the possible propagation paths. Even if DSM identifies possible impacts and propagations of a change, it does not provide assistance to expert to agree on a change or propose alternative solutions when change requests are not agreed upon.

Web-based ECM [10] is a web-based architecture for implementing the ECM process. Instead of using documents to generate change requests, designers fill request forms on the Internet. This approach offers a lot of advantages. First, the quantity of documents generated is considerably reduced; second, the treatment time of a change request is reduced; and finally, the data generated is automatically shared and transmitted to the disciplines involved. However, the tool does not offer any intelligence or decision-making assistance.

The SIMNET project [11] proposes a change-management approach based on a parameter concept. In this project, engineering decisions determine engineering variables and the most elementary elements of the engineering variables are "parameters." Clarkson *et al*. [12] propose an analysis of change behavior based on the mathematical models to predict the risk of change propagation. Their change prediction method consists of: 1) doing an initial analysis; 2) creating a product model; 3) completing dependency matrices; and 4) computing predictive matrices. The analysis of change propagation in their approach can cause serious delays in project design.

Eckert *et al*. [13] present an analysis of potential causes of change requests and the effects of changes. Changes could be initiated by customer requirements, certification requirements, innovations, problems with past designs, new customer requirements, or recent innovations. The major source of emergent problems is the number of parts in a product and the relationship between them. In fact, the parts in complex products could be highly interconnected. Eckert and Clarkson's work presents an interesting analysis on change processes that could help researchers design automated solutions for change propagation. Their work, however, does not propose a practical tool for change management.

In [14], the authors propose, to deal with the complexity of engineering management, a system based on entities and constraints, where constraints give an idea of the design intent.

All the applications cited earlier help inform the right person at the right moment that a change has been made on a design document. However, they do not propagate changes automatically and do not assist decision makers in conducting EC decisions. In what follows, some relevant works based on software agents for helping streamline processes in engineering environments are presented. Most of which, however, do not address change management specifically.

Weidong *et al*. [15] propose a flexible and open architecture based on software agent to support distributed or centralized collaborative design. In this architecture, synchronous and asynchronous collaboration are allowed. There is no support for ECM. The CAboCAD system [16] is an open design environment in the architectural domain. The research team used the software component technology to implement it. Each building component or assembly is designed as an independent component. A component agent method is used as a basic system module. Designers can manage the different component agents and exchange information using a web-based interface manager. The system does not address the problem of ECM.

In [17], the authors modeled the EC system as a set of components. Each component propagates changes to components linked to it using some rules. In such systems, the problem of loop detection should be addressed. When a change occurs, it is propagated to other components and may go back at the end to the same starting component. It would, thus, require a module for detecting and resolving loops.

Intelligent agents have also been used to develop computer-aided process planning and scheduling systems. The agent approach was recognized as being an effective one as it ensures an adaptation and dynamism in the planning process. Integrated, distributed, and cooperative process planning system [18] is an example of a distributed and cooperative planning system based on software agents. The planning tasks are divided into three levels; initial planning for studying the feasibility of the manufacturing operations, decision making generates an optimal list of alternative plans considering the availability of the resources and detailed planning, and the last level determines the machines and tools finally selected for manufacturing. Unfortunately, no prototype was developed to validate this concept, and the work does not address ECM. Kornienko *et al*. [19] also addressed the planning process as a constraints satisfaction problem. Their system generates an optimal distributed plan satisfying all the constraints. Among the various agent encapsulation approaches defined in their paper [20], Shen *et al*. define two distinct approaches to using agents in planning. In the functional-decomposition approach, agents are used to encapsulate functions such as planning, resources allocation, transportation management, and material handling. In the physical-decomposition approach, agents represent physical entities such as: workers, machines, tools, products, parts, etc. However, to integrate systems within an organization (CAD systems, databases, ERP, etc.), the functional decomposition approach is the most commonly used.

The objective of PLACID platform [21] (platform for the innovating and distributed design) is to assist the cooperative design while satisfying some constraints. PLACID is based on software agents, it contains four types of agents: *application agent* (or microtools), a *mediating agent*, *system agent*, and *interface agent*.

It can be noted that all agent-based systems cited earlier provide help to share knowledge between the various disciplines of an organization so as to ultimately help manage conflicts. However, the problem of change management automation is not handled explicitly. The automation of negotiation process for change management is also not explicitly taken into account in these works.

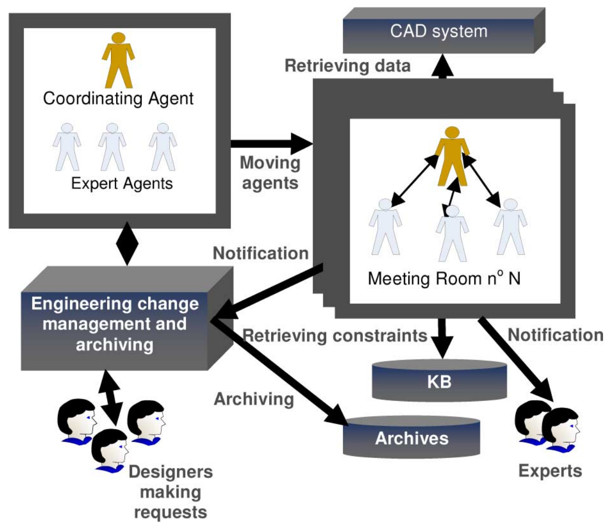

Fig. 1. Architecture overview.

EchoMag multiagent proposed system-assist designers to make decisions about change requests. It offers alternative solutions if some constraints cannot be satisfied. EchoMag is also independent from the CAD system used. The proposed system and the CAD tool are connected via an interface that can be adapted to the existing CAD system.

III. Architecture Overview

A. High Level Architecture

EchoMag structure is composed of three modules as briefly described in [22]. The first module; the *Engineering Change Management and Archiving* module, manages and archives all change management requests made by designers. The second module is the *Multiagents System*, whose purpose is to check whether the proposed changes can be adopted or not. The third module, the *Constraints Management* module allows the definition of design and manufacturing constraints on the products and their corresponding CAD models. These constraints are kept in a repository or knowledge base (KB) to be subsequently accessed by agents.

Before further detailing the system structure and functionality, we will introduce some definitions used in EchoMag architecture (see Fig. 1):

Coordinating Agent (CA): Coordinates the work of expert agents.

Expert Agents (EA): Has an expertise in a given discipline within a mechanical engineering field.

Meeting Room: A computer where different agents meet and negotiate a solution to a problem.

Knowledge Base: All the specifications and constraints on CAD products.

EchoMag verifies automatically if there are some models that have been modified in the database. It compares the new and the last version of the product to check if there has been a modification. In this case, an engineering change request (ECR) is automatically generated and the verification starts. Users could choose to run EchoMag outside the working hours so as to avoid conflicts if designers are working on products at the same time. This can also improve execution time of the application if the network is free from data traffic.

As illustrated in Fig. 1, when ECRs are formulated, the system first verifies if there are any archived solutions available for the requests at hand. If so, existing solutions are sent to the designers. Otherwise requests are sorted by project and priority for further treatment. This work is performed in the *Engineering Change Management and Archiving* module.

When further treatment is to be performed for project, a *CA* is generated for the project by the *Multiagents System* module. The *CA* creates the relevant *EA,* according to the disciplines involved, in the ECRs at hand. The *CA* then chooses a *Meeting Room* to manage the negotiation process among *EA*. *Meeting rooms* are located in a computer capable of supporting heavy calculations during the negotiations.

EA need then to move to the meeting room. In agent-base technology, there are two types of possible mobility:

Strong: The agent migrates with its state of execution. After a strong migration, the agent continues its execution exactly at the point it was stopped before migrating.

Weak: The agent maintains his data state while moving from a machine to another. His data is composed of the values, which can be sent over the network and then be retrieved by the agent in the destination machine. It is the programmer who decides which variables will be part of the state data.

In EchoMag, strong mobility is used. In the *Meeting Room*, every *EA* receives information about the change request from the *CA* and loads the appropriate constraints from the *KB*. Information in the KB should be well maintained so as to get correct solutions. *EA* then extract all relevant information needed from the CAD system to verify the constraints. If every constraint is respected, the change request is accepted. If there are some conflicts, a negotiation process is launched by the agents to propose alternative solutions to the designers. Notifications are then sent to the concerned human experts and solutions are archived for future use.

The proposed system not only verifies constraints on models, but it also helps designers and human experts make sound decisions about the ECRs. The link between EC management validation process (change validation by EA) and data stored in the KB and CAD system is automated.

Finally, EchoMag does not suffer from a tight integration to the CAD system used; an interface, that is adapted to the used CAD system, is developed for a specific CAD to help extract information used by the agents from the CAD system easily. For example, in EchoMag developed prototype, the system was connected to the CATIA CAD system via an interface where calls to the CATIA API methods are defined. The same interface can be adapted to another connecting CAD tool.

B. Detailed Architecture

The detailed architecture of EchoMag contains additional submodules as shown in Fig. 2. When EchoMag is in operation, one or various ECRs are made by the designers. The *Archiving* submodule first verifies if there are any archived solutions available for the requests at hand. It sends available solution results for previously solved ECRs and sends other ECRs for which known solutions do not exist to the *Constraint Gathering submodule* (CGM).

The CGM sorts ECRs by project. The *Change Request Allocation submodule (CRAM)* creates a CA for each group of ECRs of a single project using the *Agent Creation submodule*. Each CA sorts its own ECRs by priority and retrieves views on models involved using the *Views Manager*. It creates EAs corresponding to the views retrieved.

The CA communicates with the negotiation manager to know if the chosen *Meeting Room* dedicated to negotiation is not overwhelmed and if it is possible to launch a negotiation process. Once the negotiation finishes, CAs send results to the designers and experts. Results are then stored with help from the *Archiving submodule* (see Fig. 2).

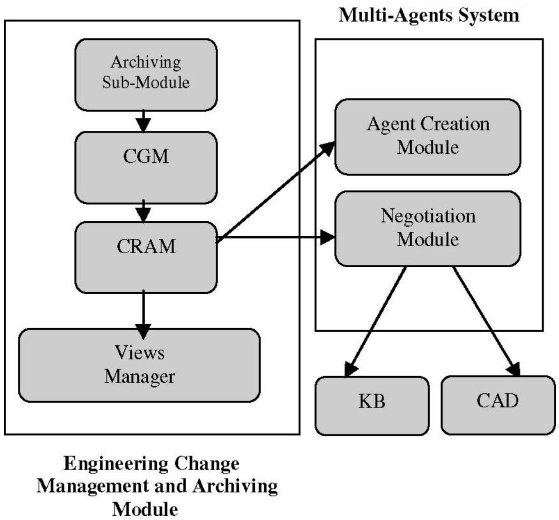

Fig. 2. Detailed architecture.

C. Negotiation Process

EAs must negotiate to find a satisfactory solution to a given ECR problem. Their work is managed by the CA. Some characteristics of the negotiation process embedded within EchoMag are listed in the following.

1) Independence of expertise (agents are working independently of each other).
2) Diversity of expertise for ECRs handling (change requests vary a lot one from another).
3) Controlled process (the CA supervises the negotiation process as well communication between EA).

EchoMag uses the concept of the blackboard negotiation strategy. Each expert agent writes some parameter values that satisfy its own constraints on the blackboard and the other agents check whether the proposed solution could be satisfactory for them. Agents cannot negotiate beyond a maximum time setup at EchoMag.

At the end of the negotiation process, the first solution satisfying all the constraints is sent to the designers. The first solution found is the cheapest solution in terms of the material and manufacturing costs. The cost is calculated with a utility function. The utility function is set in EchoMag at the installation phase for a particular engineering company and depends on material and manufacturing process data. It can, of course, be adjusted over time. Material and manufacturing costs are known in abstract. Utility functions can then be calculated by each EA for a particular solution proposed.

Finally, if no solution is found, the designers are invited to formulate another proposal of ECR or proposals of additional solutions.

IV. Prototype Implementation and Results

To test and validate the functionality of EchoMag, a prototype embedding all functionality described earlier was developed. The prototype was then used to manage EC for the project of designing an aircraft's scaled wing. The wing project is quite simple but it nevertheless validates the main functionality of our system. The wing consists of four main parts; the upper skin panel, the lower skin panel and the leading, and trailing spars. Three disciplines are involved; weight and balance, structure, and manufacturing. The example presented here uses simple rules of design but the work done in the backend by agents is not very simple. Coordination and negotiation of agents use a complex algorithm to reach a satisfactory solution. A set of ECR management scenarios was then experimented. Hereafter are the details of a particular ECR example.

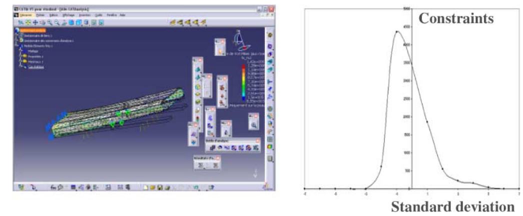

Fig. 3. FEA of the model used and constraints distribution.

A designer makes an ECR to modify the thickness of the upper skin panel from 5 to 1 mm. In the multiagent system, four agents are created; CA, a weight and balance expert agent (WBEA), a manufacturing expert agent (MEA), and a structure expert agent (SEA). Constraints are also formulated for each corresponding disciplines.

Constraints are generally formulated by the experts and their expression is validated by a compiler. A constraint is formulated using model types. In a previous work [23], the authors discussed the benefits of models' typing when it comes to automating constraints verification, even when a large number of constraints corresponding to different disciplines are involved.

For this project example, a single constraint was defined for each discipline. It is clear that in real-life projects, several constraints of additional complexity can be defined. The purpose of this simple project is, however, to validate the proposed concepts while facilitating the analysis of the operation of the system.

The constraints for the project are defined in the following:

Weight and balance constraint

$$\text{Mass\_wing} <= 1 \text{ kg.} \tag{1}$$

To verify this constraint the WBEA needs to calculate all the parts contained in the wing.

Cost constraint

$$10\$ <= \text{Cost\_wing} <= 20\$. \tag{2}$$

The MEA needs to verify the manufacturing process used and calculate the cost constraint. MEA verifies also the sheet-metal constraint. If the thickness of a part sheet metal is modified, other parts using the same sheet metal should be modified too. MEA notifies users in the case of the violation of this constraint.

Structure constraints

$$\text{SF} \geq 1.2 \tag{3}$$

$$\text{SF} = \frac{\text{Yield strenght}}{\text{Maximum stress}}. \tag{4}$$

Fig. 3 shows the finite element analysis (FEA) of the wing. FEA consists of a computer model of the design that is stressed and analyzed for specific results. It is possible to verify if the proposed design is respecting specifications and constraints prior to manufacturing or construction. Modifying an existing product or structure is used to qualify the product or structure for a new service condition. In case of the structural failure, FEA may be used to help determine the design modifications to meet the new condition.

The safety factor is defined as the ratio of yield strength and maximum stress. In the constraint, the FEA is satisfied if the safety factor is over 1.2. Results of FEA are used by the structure agent to verify the safety factor.

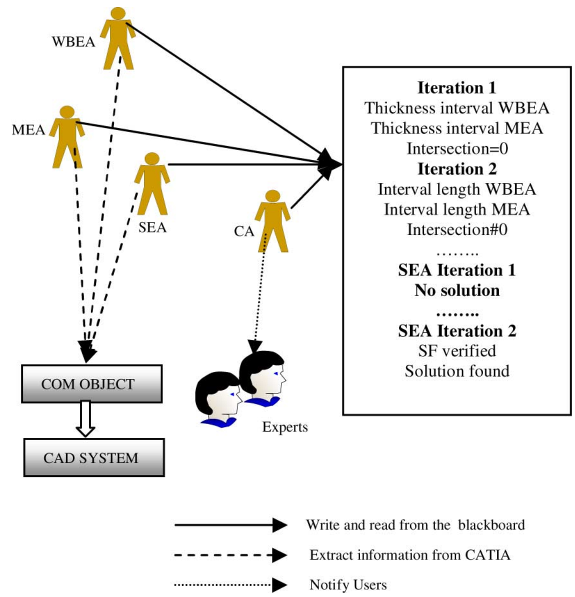

Fig. 4. Agents negotiating.

Once the FEA is completed, the results are exported to an Excel file. A module has been developed within the system to remove absurd values of constraints and obtain the correct maximum stress. The bar graph of the distribution is like a bell and then it shows clearly that the distribution follows a normal distribution (see Fig. 3). We noted that significant values are almost always in the range [−3∗standard deviation 3∗standard deviation], so we decided to remove all solutions for constraints that are beyond this range. This gives a set of reasonable constraints.

EchoMag fetches the KB and the database associated to the project for the information it needs to perform verifications.

Views: Each part is associated to some disciplines or views. This information is useful for CA to know how many EA should be created.

Parts: EchoMag must know some important information about parts (related products, type, geometry, status, and views). This information could be very useful in calculations. For example, it is a useful tool to know which other parts must be considered in the constraint verification process.

Weights: It is important to define weight so as to drive negotiation process correctly. EA should know which parameters they are allowed to change first.

Users: This information is needed for notification purposes. An email is sent to the users involved after finishing the negotiation process.

Material library: Various grades of aluminum were defined in the CAD system; Aluminum-6063-O, Aluminum-6063-T1, Aluminum-6063-T4, Aluminum-6063-T5, Aluminum-6063-T6, Aluminum-7178-O, Aluminum-7178-T6, Aluminum-7178-T76. The different grades of aluminum were stored in the database in ascending cost order so as to enable the MEA to find the cheapest solution first.

To manage the ECR in the scenario, the following operations illustrated in Fig. 4 are performed. First, EchoMag creates the CA. The CA takes as input parameters part name and change request information. It extracts the three views on the part from the database, weight and balance, and manufacturing and structure views. It then creates the WBEA, MEA and SEA.

Fig. 4 shows the most important iterations of the scenario. To simplify calculations, we consider that the upper skin panel of the wing has a rectangular format.

WBEA must verify the weight and balance constraint. It extracts mass information on all the parts of the wing, calculates thickness interval, and writes it on the blackboard

$$\text{Thickness} \le \frac{(1\text{ kg} - \text{Mass\_children\_parts}/\text{Aluminium\_density})}{\text{Width\_Upper\_skin\_pannel\_Modified} * \text{Length\_Upper\_skin\_pannel\_Modified}}.$$

MEA verifies the cost constraint, it also uses all the parts of the wing to calculate a possible set of solutions for thickness and writes them on the blackboard

$$\text{Thickness} \le \frac{(20\$ - \text{Cost\_children\_parts})}{\text{Lenght\_Upper\_skin\_pannel\_Modified} * \text{Width\_Upper\_skin\_pannel\_Modified} * \text{Cost\_Upper\_skin\_pannel\_Modified}}$$

$$\text{Thickness} \ge \frac{(10\$ - \text{Cost\_children\_parts})}{\text{Lenght\_Upper\_skin\_pannel\_Modified} * \text{Width\_Upper\_skin\_pannel\_Modified} * \text{Cost\_Upper\_skin\_pannel\_Modified}}$$

CA reads their propositions and calculates the intersection between the two solution sets (Iteration 1). As there is no intersection, it proposes to keep the thickness desired and change the length of the wing.

WBEA uses the volume formula of the upper skin panel to propose an interval for length and writes it on the blackboard

$$\text{Length} \le \frac{(1\text{ kg} - \text{Mass\_children\_parts}/\text{Aluminium\_density})}{\text{Thickness\_desired*width\_Upper\_skin\_pannel\_Modified}}.$$

MEA calculates a set of solution for length and writes it on the blackboard

$$\text{Length} \le \frac{(20\$ - \text{Cost\_children\_parts})}{\text{Thickness\_desired*Width\_Upper\_skin\_pannel\_Modfied*Cost\_Upper\_skin\_pannel\_Modified}}$$

$$\text{Length} \ge \frac{(10\$ - \text{Cost\_children\_parts})}{\text{Thickness\_desired*Width\_Upper\_skin\_pannel\_Modfied*Cost\_Upper\_skin\_pannel\_Modified}}.$$

CA finds a common set of solutions for length. However, the structure constraint is violated. The safety factor is not respected. First, SEA tries to change material from Aluminium_6063_O to Aluminium_6063_T1, but the constraint is still not respected (see Fig. 4, SEA Iteration 1). Then, for each length value in the interval found by WBEA and MEA, SEA changes the material. Finally, the constraint is respected for a length of 100 mm and the material Aluminium_7178-T76 (Iteration 2).

Since material and length could be changed, the cost and the weight of the wing could probably change. WBEA and MEA verify their constraints again. And finally, they are respected.

According to the different views, the CA notifies experts involved in the change request. It sends by email the blackboard file containing the proposal agreed upon. The proposal could be accepted or rejected by human experts. Agents help experts decide, but the final decision is made by humans. The solution found is archived in the KB.

V. Conclusion and Perspectives

This work proposes a collaborative tool named EchoMag that assists designers and experts during the change-management process. The proposed system ensures the coherence of data among the various disciplines involved in the change process. EchoMag also assists experts in making decisions by proposing alternative solutions when change requests are not agreed upon. Software agents were used to implement EchoMag for which a prototype was developed. The prototype version of EchoMag helped validate the concepts of the system and shows how it is possible to assist disciplines in collaborating toward managing ECRs.

It is clear, however, that the EchoMag's work relies on the premise that other additional works (this time performed by humans) need to be performed systematically. The previous works relate to facts that 1) typing that must be used systematically for all parts designed and 2) constraints relative to all involved disciplines need to be expressed in the KB. In EchoMag, the data affected by the constraints may be in early conceptual design or concrete at the end of the process.

While the prototype version of EchoMag helped validate the concepts of the system, our future work will target a study of the scalability of the design in projects with a significant number of parts, constraints involved.

Acknowledgment

The authors would like to acknowledge project collaborators at Bombardier Aerospace and Pratt & Whitney for their insights.

References

[1] K. G. Mcintosh, *Engineering DATA Management: A Guide to Successful Implementation*. McGraw-Hill, 1995, 279 pages.

[2] G. Harhalakis, "Engineering changes for made-to-order products: How an MRP II system should handle them," *Eng. Manage. Int.*, vol. 4, pp. 19–36, 1986.

[3] C. Terwiesch and C. H. Losh, "Managing the process of engineering change orders: The case of the climate control system in automobile development," *J. Prod. Innov. Manage.*, vol. 16, pp. 160–172, 1999.

[4] G. Q. Huang and K. L. Mak, "Current practices of engineering change management in UK manufacturing industries," *Int. J. Oper. Prod. Manage.*, vol. 19, pp. 21–37, 1999.

[5] J. Shah, J. Sadowsky, N. Macia, M. Woodfill, and F. Wilson, "The virtual corporation: Simulating real world collaboration in a university setting," in *Proc. ASME Design Theory Methodol. Conf.*, 1995, p. 20.

[6] Y. M. Chen, W. S. Shir, and C. Y. Shen, "Distributed engineering change management for allied concurrent engineering," *J. Comput. Integr. Manuf.*, vol. 15, pp. 127–151, 2002.

[7] M. Gruninger and M. S. Fox, "The logic of enterprise modelling," in *Modelling and Methodologies for Enterprise Integration*, P. Bernus and L. Nemes, Eds. Cornwall, Great Britain: Chapman and Hall, 1996, pp. 83–98.

[8] P. Bernus, L. Nemes, and T. J. Williams, "A framework to define a generic enterprise reference architecture and methodology," in *Proc. ICARV* 1994, Singapore, pp. 88–92.

[9] D. Tang, R. Xu, J. Tang, and R. He, "Design structure matrix-based engineering change management for product development," *Int. J. Internet Manuf. Serv.*, vol. 1, no. 3, pp. 231–245, 2008.

[10] G. Q. Huang, W. Y. Yee, and K. L. Mak, "Development of a web-based system for engineering change management," *Robot. Comput. Integr. Manuf.*, vol. 17, pp. 255–267, 2001.

[11] K. Rouibeh and K. R. Caskey, "Change management in concurrent engineering from a parameter perspective," *Comput. Ind.*, vol. 50, pp. 15–34, 2003.

[12] P. J. Clarkson, C. Simons, and C. Eckert, "Predicting change propagation in complex design," *J. Mech. Des.-ASME Trans.*, vol. 126, pp. 788–798, 2004.

[13] C. Eckert, P. J. Clarkson, and W. Zanker, "Change and customisation in complex engineering domains," *Res. Eng. Des.*, vol. 15, no. 1, pp. 1–21, 2004.

[14] J. Eustache, R. Maranzana, Y. Lanuel, and Y. Gardan, "Managing complexity in a CAD environment," in *Proc. Change Manage. New Ind. Revolution*, New York, Oct. 7–9, 2001, pp. 104–109.

[15] W. Fang, M. X. Tang, and H. J. Frazer, "Supporting collaborative product design in an agent based environment," *Lecture Notes Comput. Sci.*, vol. 2718, pp. 293–298, 2003.

[16] M. Rosenman and F. Wang, *Autom. Constr.*, vol. 10, no. 4, pp. 383–397, 2001.

[17] A. Mokhtar, C. Bedard, and P. Fazio, "Information model for managing design changes in a collaborative environment," *J. Comput. Civil Eng.*, vol. 12, no. 2, pp. 82–92, Apr. 1998.

[18] F. T. S. Chan, J. Zhang, and P. Li, "Modelling of integrated, distributed, and cooperative process planning system using an agent-based approach," in *Proc. Inst. Mech. Eng., Part B: J. Eng. Manuf.*, 2001, pp. 1437–1451.

[19] S. Kornienko, O. Kornienko, and P. Levi, "Flexible manufacturing process planning based on the multi-agent technology," presented at the 21st IASTED, Int. Conf. Applied Informatics, Innsbruck, Austria, Feb. 2003.

[20] W. Shen, L. Wang, and Q. Hao, "Agent-based distributed manufacturing process planning and scheduling: A state-of-the-art survey," *IEEE Trans. Syst., Man, Cybern.C Appl. Rev.*, vol. 36, no. 4, pp. 563–577, Jul. 2006.

[21] A. J. Fougères, "Agents to cooperate in distributed design," in *Proc. IEEE Int. Conf. Syst., Man Cybern.*, Hague, Netherlands, Oct. 10–13, 2004, pp. 2629–2634.

[22] D. Habhouba, S. Cherkaoui, and A. Desrochers, "Engineering change management and decision-making assistance using software agent," in *Proc. CCECE Can. Conf.*, May 2006, pp. 1694–1697.

[23] D. Habhouba, A. Desrochers, and S. Cherkaoui, "Verification of product technical specifications in CAD systems using software agents," in *Proc. 7th Int. Conf. Comput. Suppported Coop. Work Design*, 2002, pp. 236–240.