Development of a Computer Aided Conceptual Design Tool for Complex Electromechanical Systems

Noe Vargas-Hernandez, Jami J. Shah, and Zoe Lacroix

Department of Mechanical and Aerospace Engineering
Arizona State University
Tempe, Arizona 85287-6106

Abstract

The objective of this paper is to report on the development of a Computer Aided Conceptual Design (CACD) Tool for the Design of Complex Electromechanical Systems (CEMS). Few methods and tools exist for supporting the conceptual design stage; this is a paradox considering that decisions at this stage have the greatest influence in the cost and characteristics of the final product. Two main issues were identified: Standardization and Compatibility. The first refers to the variety of ontologies (such as functions and behaviors) proposed by several authors. These ontologies may work on their own, but they don't "talk" to each other. Even if this issue is solved, there is no compatibility between elements from different tasks (e.g. function to behavior). The proposed approach solves these two issues without sacrificing generality.

Introduction

The objective of this work is to develop a Computer Aided Conceptual Design (CACD) tool for the design of Complex Electromechanical Systems (CEMS) using catalogs of standard elements. Conceptual design is a critical stage of the Product Realization Process (PRP). It is in here where decisions made have the greatest impact in the final product design; paradoxically, little CAD support exists for this stage. Currently, most of the conceptual design is done in a "cocktail napkin" (i.e. low level representations); some available CACD tools are either too abstract (i.e. not implemented) or too specific (i.e. for a particular product type). CAD, CAM, and CAE tools in general focus on the later stages of design (i.e. embodiment, detail), when a concept already exists. What makes difficult to automate conceptual design processes is that design information is continuously evolving; current knowledge representation is neither standardized nor integrated. This situation is magnified with the increased complexity of the products (such as CEMS).

Background

CEMS

Two significant changes can be noticed on how design is conducted nowadays: Combination of multiple disciplines (e.g. mechanical, electro) and increase in complexity (i.e. intricacy of relations and number of elements). Of particular interest are Complex CEMS, common in everyday life, ranging from cell-phones to airplanes. CEMS combine elements from mechanical and electro (i.e. electric and electronic) domains. As technical systems, CEMS transform energy, material, and signal to perform a technical task (Pahl and Beitz 1996). How complex the system is depends on the intricacy of the relations among its elements and disciplines, and the number of elements involved (i.e. size). A complex system can be abstracted (i.e. reduce its complexity by emphasizing essential characteristics) from various points of view, for example, by Domain (e.g. the hydraulic system of an airplane – mechanical domain), by Flow (e.g. the autopilot control system – signal flow), by Function (e.g. the propulsion system – propulsion function), or by any combination and/or selection of interest (e.g. turbine # 2). Any element of a CEMS might be viewed from different abstractions, also, each element could be further divided into sub elements (i.e. a subsystem) or be part of a higher level system (i.e. a supersystyem). If the system is abstracted adequately, it is possible to divide a CEMS into a hierarchy of subsystems and supersystems of higher and lower complexity levels, respectively.

Conceptual Design

The sequence of activities of a system design process depends on various factors, such as the type of design (i.e. novel, evolutionary, etc.), designer's experience (e.g. novice or experienced), approach followed (i.e. Design Method chosen), and the decisions made during the process. A design method is an "identifiable way of working" (Cross, Dorst, and Roozenburg 1992) that formalizes a procedure and externalizes design thinking. Many methods have been developed, especially since WWII, for which two main trends can be identified: Descriptive models (Dixon 1988; Finger and Dixon 1989, Ullman and Diettrich 1987), which reflect what designers actually do, and Prescriptive models (Pahl and Beitz 1996; Hubka and Eder 1988), which prescribe an algorithmic or systematic procedure to follow. Design methods can be also classified into Creative methods, which promote creative thinking (e.g. Brainstorming, 635, C-Sketch) and Rational methods, which encourage a systematic approach (e.g. morphological charts, objectives tree, functional decomposition) (Shah 1998).

The complexity of developing a system makes necessary a rational and methodical approach. Since prescriptive models have an emphasis in logical selection, full understanding of events, and rational choice, they appear as a natural choice for complex electro-mechanical systems design. Several models of the development of a system have been proposed, offering a structured top-down approach. Examples of these models are Pahl and Beitz (1996) and VDI2221 (1987); both based on the Analysis-Synthesis-Evaluation triad model. This triad model is repeated throughout the process model for finding solutions. Although there is no unique (i.e. universally accepted) design process model for systems design, most prescriptive models divide the design process into three main stages (Pahl and Beitz 1996; Hubka and Eder 1988): Conceptual, Embodiment, and Detail. The design processes can be viewed as a pipeline of tasks. The information travels through the pipeline by successive transformations. Every task transforms an input into an output state (Shah and Wilson 1989) to be used in subsequent tasks.

Conceptual Design of CEMS

CEMS combine elements from electro (electronic and electric) and mechanic (linear and rotational) domains to transform energy to perform a technical task. Instances of CEMS range from airplanes to home appliances such as washing machines, air conditioning, etc. The design of a CEMS, as any other product, follows three main phases: Pre-Design, Design, and Post-Design. Pre-Design is where market study and customer requirements are defined, Design is where requirements are evolved into a detailed design, and Post-Design, where the product is manufactured, used, maintained and disposed or recycled. During the design phase the system being developed follows a three-step cycle of analysis, design and synthesis. This cycle is iterated for conceptual, embodiment, and detail stages, increasing the level of detail with each iteration. During analysis, the overall system is decomposed into manageable (i.e. easy to solve) elements. During Design, each element is either designed or selected. Depending on the complexity of each system, the element can be defined as a subsystem with its own development cycle. During Synthesis, all designed elements are integrated to form a product. During the conceptual cycle, the analysis task is functional decomposition, the design task is the element design or selection, and the synthesis task is the behavior analysis of the concept.

At each stage (Conceptual, Embodiment, and Detail) of the design process, the following cycle is iterated: Analysis, Design, Synthesis, and Evaluation. Analysis studies the elements of a system and its interrelationships. During Analysis, the system is decomposed in manageable (i.e. a solution can be easily found) parts; for example, during functional decomposition the overall function is decomposed into a functional structure (Pahl and Beitz, 1996). Once the system is decomposed, each element needs to be designed (i.e. solved); during functional decomposition, each element is functionally defined by its inputs (e.g. energy, material, signal) and outputs. The systems engineer (who must be knowledgeable in all involved domains) assigns the elements to the design teams and oversees the overall system design. Depending on its complexity, an element could be treated as a subsystem and be further decomposed. The design of an element can be of different types. Shah and Wilson (1989) identified four classes: Novel (i.e. from first principles), Evolutionary (i.e. modifying a current design), Parametric (i.e. following an already characterized design procedure), and Selection (i.e. searching standard components from catalogs). During Synthesis, each individually designed element is put together to form the solution system. The effects are understood by studying its physical behavior (e.g. CAD layout, CAE simulation, vibration, assembly, etc.). The resulting synthesized design is then evaluated against one or more criteria defined by the designer, for example, cost, manufacturability, reliability, etc. Most of the times, two or more criteria are at conflict (e.g. quantity and quality vs. cost reduction), and optimization is needed to find a solution (Eschenauer, Koski, and Osyczka 1990; Gero 1985).

State of the Art

Knowledge Representation

Design knowledge, in its broadest meaning, refers to all information (i.e. objects, concepts and relationships) assumed to exist in a design. Knowledge can exist in many shapes and representations, but is more useful (i.e. easier to manipulate and aid in reasoning) for the designer when formally represented. Representation refers to how the information is stored and presentation to what the designer sees. A knowledge representation schema is an explicit specification of a conceptualization (i.e. an abstract view of the world that we wish to represent). KR schemas consist of vocabulary elements and syntax (i.e. rules) on how to create relation structures. In each task, a vocabulary is used in the reasoning process to create a relation structure that represents the design at that stage. There exist various KR schemas types: Physical, Graphical, Semantic, and Analytical; some examples are shown in Table 1. Physical schemas are presented as physical models and prototype mockups and the information is stored physically; an example is a Lego® set, where the vocabulary elements are the building blocks and the structures are the constructs. Graphical schemas are represented by scaled drawings where the information is stored geometrically; an example is CSG, where the vocabulary elements are the primitive shapes, and the structures are the Boolean constructs. Analytical schemas are presented as diagrams or equations and the information is stored in data structures and mathematical expressions; examples are Bond Graphs (BG) (Thoma 1975) and Function Converters (Summers et al. 2001). In BG, the vocabulary is a set of BG elements and the structures are the BG systems. Semantic schemas are presented as texts and tables and the information is stored as data tables, lists, and structures; an example is catalog components, where the vocabulary are the design variables and the structures are the variable lists for each component. Some schemas combine two or more types, for example, Graph Grammars (Schmidt et al., 2000) are a mathematical method to manipulate graphs, and Exemplars (Summers 2000) are bi-partite graphs representing geometric, algebraic, and semantic entities.

Table 1 – Knowledge Representation During Conceptual Design

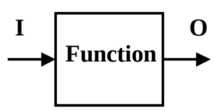
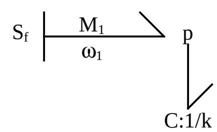
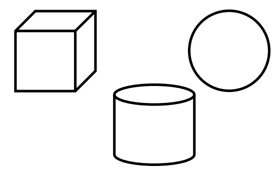
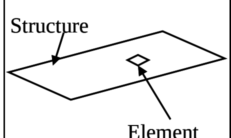
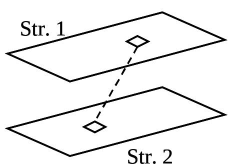
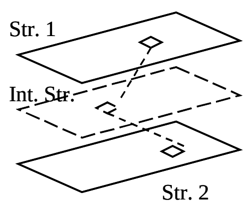
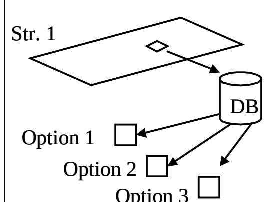
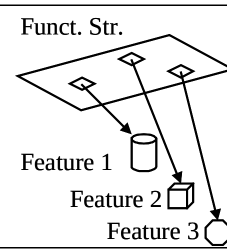
Tasks →	**Functional Decomposition**	**Component Selection**	**Behavior Analysis**	**Layout Analysis**
Vocabulary Elements	Functions	Attributes	Mathematical Expressions	Geometric Shapes
Relation Structure	Functional Structure	Lists	Diagrams, Equation Systems	Sketches, 3D Mockups
Schema Example	Function Converters	Classes of Objects	Bond Graphs	Constructive Solid Geometry (CSG) Primitives
Diagram		Object 1 Attribute 1 Attribute 2 Attribute 3 : :		

Table 2 – Current Research in KR in Conceptual Design

SCHEMA	**CONSTRAINTS**	**TRANSITION ELEMENTS**	**EXHAUSTIVE SEARCH**	**FEATURES**
EXPLANATION	Connect elements from 2 structures using constraints	Use transition elements to connect function and form	An Element in structure 1 is searched for an equivalent in structure 2	A function is fulfilled by an specific shape
DIAGRAM 				
ADVANTAGE	Preserves original structure	Flexibility: for each trans. Elem. More than 1 form may exist	Catalogs help capture design knowledge and experience	For each function there is a form
LIMITATIONS	Difficult to decide which entities to connect	Difficult to define every trans. Elem.	Once obtained, elements in structure 2 must be connected manually	Once the shapes are obtained, these need to be connected
REFERENCES	Deng, Britton, and and Tor 2000	Schmidt et al. 2000; Eder 1995; Jensen 2000.	Chakrabarti & Tang 1996; Li et al. 1996; Brady and Juster 1995	Schulte & Weber 1993; Brunetti & Golob 2000

Survey of Available Systems

Several methods for KR in conceptual design exist nowadays at various stages of implementation. Table 2 presents a summary of the most representative approaches. Constraints have been used to connect two structures; this approach has the advantage of being non intrusive, but constraints are difficult to assign and maintain. Transition elements require an intermediate layer between structures; connecting layers is easy, but defining the elements is not. Exhaustive search finds all possible matches for a particular element; this approach allows for flexibility and captures design knowledge, but the selected option has to be manually connected to the original element. Features can be assigned to a particular function, hence, based on a functional structure a set of features can be obtained; the disadvantage is that the shapes obtained need to be connected manually.

Fundamental Issues

Various problems can be identified in current conceptual design methods. First, information from a previous task is difficult to reuse, for example, when deriving a behavior structure from a functional structure. Second, the result of each task is a structure unconnected to other task's structures; for example, a functional structure is unconnected to behavior structure. This is shown in Figure 1. Third, some information may be lost while reinterpreting an output to an input; for example, a simplified behavior may be used to represent a function. Fourth, The effect of changes is difficult to trace back to previous design tasks, for example, how a change in the physical layout will affect the functionality.

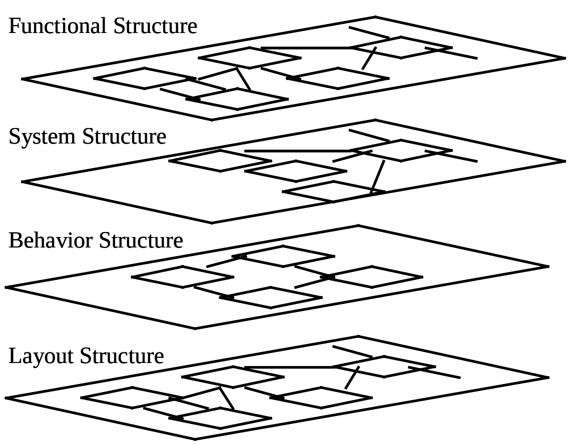


Figure 1 – Unconnected Structures in Conceptual Design

Standardization and Compatibility

Two main issues are identified for the development of a CACD tool. The first issue refers to the definition of element's ontologies. How to define standard sets of basic functions, behaviors, and components with all the needed qualities (e.g. simple to use, complete, technically correct)? Further, how to define mappings between elements of these three sets. The second issue refers to the structure (i.e. syntax) of the vocabulary. How to connect elements to form valid design structures, and how to connect elements between different structures?

Requirements for a CACD Tool

Various concepts were learned from the survey of methods and tools in conceptual design. First, catalogs allow flexibility; designers can define their own vocabulary elements using basic elements. Second, having independent structures is not the real problem, the improvement should focus on developing inter structure relationships. In general, what is missing is a schema that represents the knowledge from the various tasks during conceptual design. The KR schema will require an ontology (i.e. a formal specification on how to represent entities and relationships). The ontology is composed of a vocabulary (i.e. a listing of elements) and a structure (i.e. allowable relationships between elements). This is shown in Figure 2. The vocabulary should be able to represent various task's vocabularies (e.g. function, behavior), and allow for user-defined elements. For structure relationships, the schema should allow the independent creation of structures for each task in conceptual design (i.e. intra-structure relationships), and make possible to relate elements among structures from different tasks (i.e. inter-structure relationships).

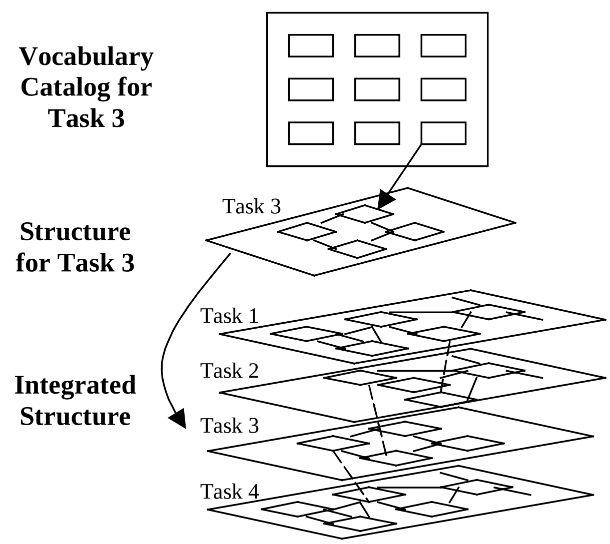


Figure 2 – KR Schema Requirements

Proposed Approach

General Overview

It is not the objective for the proposed CACD tool to substitute already available tools for any of the tasks; on the contrary, exchange of design structures is encouraged (with the development of parsers or translators) between CCAD and other specific tools. What this CACD tool will do is maintain the functional "essence" of the design during three basic modes of operation: Structure, Mapping, and Validation for the Structure Builder shown in Figure 3. During Structure mode a designer can develop a structure (e.g. functional, behavior, or component structure). During Mapping mode, the designer can first develop a functional structure and then interactively define behavior and component structures. It could be also possible to first develop a behavior or component structure and then derive a functional structure. In Validation mode, a designer could modify an existing behavior or component structure and observe the effect in the functional structure.

The proposed system architecture is shown in Figure 3. The user interacts with the system through a GUI. Through the structure builder the user interacts with the catalog in the three modes previously explained (i.e. structure, mapping and validation) to create the structure instance. As explained before, this tool doesn't intend to substitute available tools for other tasks; the translation parser imports and exports the structure instance information to and from other data formats. Parsers will need to be defined for each particular exchange case.

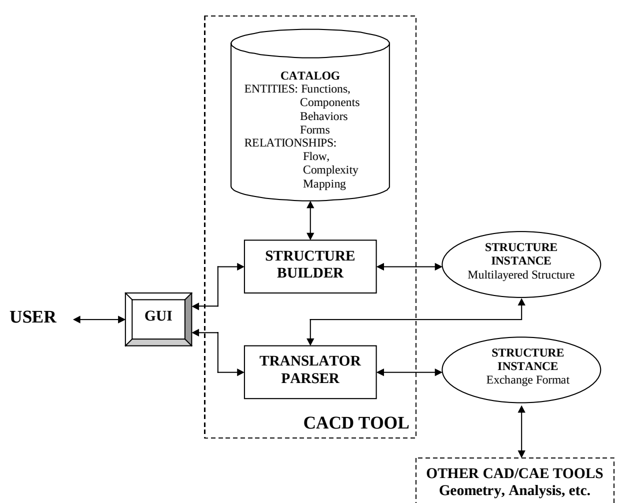


Figure 3 – Proposed System Architecture

Catalogs

Various authors have defined different ontologies of functions. Most of the times, these ontologies can be cross-referenced with slight changes (e.g. join vs. connect). The taxonomy proposed in this work relies on a key concept: Bond Graphs (BG). It was found that for every basic function element a corresponding BG behavior element exists (Vargas-Hernandez 2002). This is pictured in Figure 4. Based on the basic set of BG behavior elements, a basic set of functions was defined. Further, it was found that for each basic BG behavior element, a basic component element exists. Combinations of these basic elements will result in user-defined elements that are easier and practical to use and can be stored in catalogs.

The vocabulary module could function in two modes: definition and use. During definition mode the designer chooses which elements to include in a vocabulary for a given task. The element's attributes contain the necessary design information as they are entered in the catalog. The vocabulary catalog could be implemented using XML, which allows for the logical diversity of the attributes.

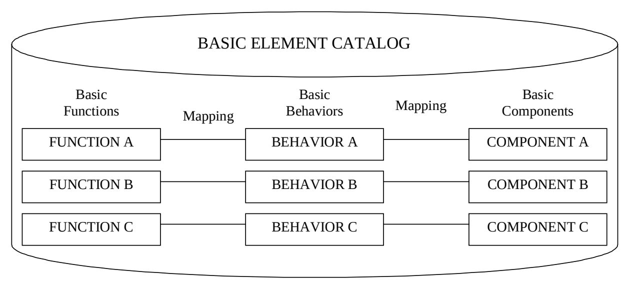


Figure 4 – Proposed Catalog Strategy

Structures

With respect to the structure (i.e. syntax) for the creation of design structures, it was found that in each structure there are two types of relationships: Flow and Complexity. This is shown in Figure 5. Flow relationships connect two elements at the same level of detail (e.g. the output of a function to the input of another). Complexity relationships connect element at different levels of resolution or detail (e.g. function to subfunction or function to superfunction). A third type of relationship exists between structures: Mapping. This relates two entities from different structures (e.g. function to behavior, behavior to component). It was found that the type of data structures needed to represent Flow, Complexity and Mapping relationships are network, hierarchical, and relational respectively. The overall multilayered structure representing the conceptual design requires a multidimensional data structure; because of this and the many constraints needed, XML is being used for the implementation of this CACD tool. It is expected that this CACD tool improve the conceptual design of CEMS by shortening the design cycle time while ensuring the functional feasibility of the design. Design changes can be easily validated through its structure. This concept design structure also represents the design history, which can be used for variation design, or as a way to document and/or understand how a design evolved.

It was found that the there are three types of possible relationships between elements in an integrated structure. Within the same task-vocabulary the relationships can be either of flow type (connects inputs and outputs of compatible flows) or complexity type (indicates the parent or son component). Between different task-vocabularies, the relationship is of type mapping (relates two elements from different vocabularies). It was also found that, because of the various relationship types, various data structures will be present. The complexity relationship needs a hierarchical structure, flow is a network, mapping is relational, and overall, the structure is multidimensional.

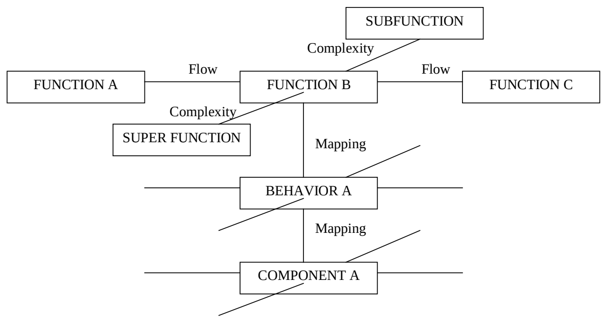


Figure 5 – Proposed Structure Strategy

Conclusions

It can be concluded that the proposed schema presented here offers several advantages relative to current methods and systems. The vocabulary is capable of representing various task vocabularies (e.g. function and behavior) and the user is able to define his/her own elements. The relationships allow the independent creation of structures for each task; also, it is possible to relate elements among structures from different tasks. The proposed schema is superior (reducing time and effort necessary) when compared to current conceptual design methods. The structure created represents the design history of a particular design, and the designer is free to move form one end to the other (e.g. from function to form). Currently the authors are working on the implementation of this schema as a CACD tool. Once implemented, the proposed schema will help designers shorten the time and decrease the effort necessary to generate conceptual designs of CEMS. Various challenges need to be addressed. The characterization of necessary knowledge for each element to be included in the catalog. Maintaining the integrity of complexity and flow relationships. Dealing with multiple uncommitted options while concretizing the design structure.

Future Work

Some avenues for future development of the schema proposed here are: Include knowledge base or case base reasoning to capture design experience during conceptual design. Expand the schema to other vocabulary domains such as chemical, thermal, etc. Include DFX (i.e. Design for Manufacturing, Assembly, Environment, etc.) functionality to improve the quality of the design.

References

Brady, D., Juster, N. P., 1995, “A Computerized Tool to Create Concept Variants form Function Structures”, AI System Support for Conceptual Design, Proceedings of the 1995 Lancaster International Workshop on Engineering Design, pp. 209-226.

Brunetti, G., and Golob, B., 2000, “ A Feature-Based Approach towards an Integrated Product Model Including Conceptual Design Information”, Computer-Aided Design, Elsevier, 32 877-887.

Chakrabarti, A., and Tang, M. X., 1996, “Generating Conceptual Solutions on FUNCSION: Evolutio of a Functional Synthesiser”. Artificial Intelligence in Design ’96, Gero, J. S., and Sudweeks, F. (eds), pp. 603-622.

Cross, N., Dorst, K., and Roozenburg, N., 1992, Research in Design Thinking, Deslft University Press.

Deng, Y. -M., Britton, G. A., and Tor, S. B., 2000, “Constraint-Based Functional Design Verification for Conceptual Design”. Computer-Aided Design, Elsevier, 32: 889-899.

Dixon, J. R., 1988, “On Research Methodology Towards a Scientific Theory of Engineering Design”, Artificial Intelligence for Engineering Design, Analysis and Manufacturing, Vol. 1, No. 3, pp. 316-337.

Eder, W. E., 1995, “Systematic Conceptualization – With Computational Tools?”, Knowledge Intensive CAD, KICAD, Vol1, pp. 205-224.

Eschenauer, H., Koski, J., Osyczka, A. (eds.), 1990, Multicriteria Design Optimization, New York: Springer-Verlag.

Finger, S, and Dixon, J. R., 1989, “A Review of Research in Mechanicl Engineering Desing. Part I: Descriptive, Prescriptive, and Computer-Based Modles of Design Processes”, Research in Engineering Design, (1989) Vol. 11, pp. 51-67.

Gero, J. S. (ed.), 1985, Design Optimization, Academic Press, New York.

Hubka, V., and Eder, W. E., 1988, Theory of Technical Systems, Springer Verlag, Berlin.

Jensen, T., 2000, “Function Integration Explained by Allocation and Activation of Wirk Elements”, Proceedings of 2000 ASME DETC, Baltimore, MD.

Li, C. L., Tan, S. T., and Chan, K. W., 1996, “A Qualitative and Heuristic Approach to the Conceptual Design of Mechanisms”, Engineering Applications on Artificial Intelligence, 9:1, pp. 17-31.

Pahl, G. and Beitz, W., 1996, Engineering Design - A Systematic Approach, Ed. 2, Springer, London.

Salomone, T. A., 1995, What Every Engineer Should Know About Concurrent Engineering, Marcel Decker Inc. New York.

Schmidt, L., Dusselmann, S., Schlick, C., Tegel, O., and Luczak, H., 2000, "A New Approach to Computer-Supported Systematic Design – An Empirical Study". International Design Conceference – Desing 2000, Dubrovnik.

Schulte, M., and Weber, C., 1993, "The Relationship and Function Shape", ICED'93, The Hague.

Shah, J. J., 1998, "Experimental investigation of collaborative techniques for progressive idea generation," Proceedings, ASME Design Theory and Methodology Conference, Atlanta.

Shah, J., and Wilson, P., 1989, "Analysis of Design Abstraction, Representation and Inferencing Requirements for Computer-Aided Design", Design Studies, Vol 10 No.3 pp 471-488.

Summers, J. D., 2000, Development of Domain and Solver Independent Method for mechanical Engineering Embodiment Design (Ph.D. Proposal), Department of Mechanical Engineering, Arizona State University, August, 2000.

Summers, J. D., Vargas-Hernadez, N., Zhao, Z., Shah, J., Lacroix, Z., 2001, "Comparative Study of Representation Structures for Modeling Function and Behavior of Mechanical Devices". Proceedings of 2001 ASME DETC, Pittsburgh, PA.

Thoma, J. U., 1975, Introduction to Bond Graphs and their Applications, Ed. 1, Pergamon Press, London.

Ullman, D., and Dietterich, T., 1987, "Mechanical Design Methodology: Implications on Future Developments of CAD and KBS", Engineering with Computers, Vol. 2, pp 21-29.

Vargas-Hernandez, N., 2002, CAD Support for the Conceptual Design Stage of CEMS, Ph.D. Dissertation Proposal, Arizona State University.

VDI2221, 1987, VDI- Design Handbook 2221: Systematic Approach to the Design of Technical.