A Knowledge-Based Approach to Problem Formulation for Product Model-Based Multidisciplinary Design Optimization in AEC

Benjamin Welle
Center for Integrated Facility Engineering (CIFE)
Department of Civil and Environmental Engineering
Stanford University, Stanford, CA 94305
bwelle@stanford.edu

John Haymaker
Design Process Innovation
San Francisco, CA, 94107
johnrhaymaker@gmail.com

Abstract

The cost-effectiveness and accuracy of a multidisciplinary design optimization (MDO) process is highly dependent on designers' ability to flexibly formulate the optimization problem for specific challenges. Designers need to rapidly modify how object parameters are assigned to groupings of objects in the product model. Our research has developed a Reference-Based Optimization Method (RBOM) to enable this type of flexible problem formulation. However, the responsibility still falls on the designer to manage the problem formulation and MDO process, which can lead to inefficient and costly design decisions. By means of artificial intelligence, in particular knowledge-based systems, these potential barriers to MDO adoption in the Architecture, Engineering, and Construction (AEC) industry could be mitigated, resulting in more efficient design processes and, ultimately, energy-efficient built environments.

Introduction

Managing and reducing the environmental impacts in design is urgently important. For example, reducing the carbon footprint of buildings is a focus of building stakeholders and the architecture, engineering, and construction (AEC) community. The American Institute of Architects (AIA) in the Architecture 2030 Challenge (AIA 2011) and the Federal Government in the Energy Independence and Security Act (FEMP 2007) both call for net-zero energy (NZE) consumption for new building designs by the year 2030. Maximizing energy performance, however, has proven elusive to industry for many years. The precedent-based design processes that are commonly used in industry do not allow for the design exploration required to meet NZE requirements (Clevenger 2009). Performance-based designs are complex multi-criteria problems that require more structured and systematic definition and exploration of design spaces (Papamichael and Protzen 1993). Designers need more information about the performance trends and interactions of the potential design spaces available to them, particularly during the conceptual design stage, when design decisions have largest impact on building performance (Struck, de Wilde et al. 2009).

To achieve anything approaching NZE buildings, designers need to generate far larger spaces, and systematically vary all the parameters in question gradually, in order to be able to comprehend and visualize performance trends and interactions (Mourshed, Kelliher et al. 2003). However, a number of tool and process limitations result in narrow explorations of design spaces. One limitation is that the designers' tools usually generate static design alternatives and are not intended to help define and explore design spaces (Shea, Aish et al. 2005, Mora, Bédard et al. 2008). A second limitation is that these tools do not produce information that is represented in a form that facilitates multidisciplinary analysis (Wang, Rivard et al. 2005, Holzer, Tengono et al. 2007). However, even when these limitations are overcome, designers must tread carefully, as design spaces quickly become unwieldy or infinite (Woodbury and Burrow 2006).

To demonstrate the magnitude of the design spaces facing conceptual building designers, consider the simple example of a rectangular building with 2 windows on each side and 10 desired construction types. A designer may want to determine the impact of independently modifying the construction type for all the windows, the windows on each orientation, and each individual window. The possible alternatives in each of these scenarios are 10, 10,000, and 100,000,000, respectively. Each of these three scenarios represent three different problem formulations with their own vast alternative and impact spaces that designers cannot adequately explore using conventional methods. Designers need efficient methods to define and explore alternative spaces that specifically address the questions they want to answer. No overarching methodology to *enable* and *guide* effective formulation and execution of optimization problems currently exists. This paper will discuss a method we developed to *enable* effective problem formulation for product model-based multidisciplinary design optimization (MDO) and how artificial intelligence (AI) and knowledge-based systems could *guide* it.

Product Model-Based MDO

MDO is a growing engineering discipline concerned with the formalization of iteration and coordination between groups working on the design of complex engineering systems and sub-systems and with creating an environment conducive to these formal methods (AIAA 1991). At its core is the notion that design is a goal oriented decision-making process driven by performance feedback (Malkawi 2004) and the application of MDO methods to support thorough investigation of design spaces in AEC holds much promise (Shea, Aish et al. 2005, Caldas 2008, Geyer 2009).

Performance-based design supported by product models, also called building information models (BIM), allows practitioners to flexibly and efficiently generate and modify geometric and semantic models. The successful use of product models for analysis, however, requires some method to pass information between the product model and the analysis application that meets the needs of the user. There is widespread support for a product model-centric approach to MDO in literature (Townsend, Samareh et al. 1998, Mourshed, Kelliher et al. 2003, Lazzara 2008).

Flexible Problem Formulation

To further investigate the potential of MDO for AEC, the authors implemented a simple classroom case study for its structural integrity, energy consumption, daylighting, and initial capital and life-cycle costs (Flager, Welle et al. 2009). The results supported the assertion that the use of MDO in AEC could compress design cycle time, increase design knowledge, and yield substantive product quality and performance gains. The automation process implemented between the selected design and analysis applications, however, lacked flexibility. The chosen structure of exchange requirements constrained the data workflows. An exchange requirement (ER) is "a set of information that needs to be exchanged to support a particular business requirement at a particular stage of a project" (buildingSMART 2009). ERs may be used for data transformation, reduction and simplification, translation, and/or interpretation (Bazjanac and Kiviniemi 2007). The exchange requirement structure for the case study only allowed the optimizer to modify the construction type for all the exterior windows at the same time. The construction type for various facade orientations or for individual windows could not be optimized independently. Such rigid systems require major modifications to handle new problems that differ from the original one by only a few minor variations in the decisions the MDO is being asked to support (Wang, Rivard et al. 2005), or in the use of a problem formulation that either cannot support the design challenge or supports it inefficiently (Berends and Tooren 2008). MDO formulations must support project-specific goals (Geyer 2009), and this functionality must be supported by the selected automation process. Failure to do so will negatively impact MDO cost-effectiveness and accuracy.

Literature defines problem formulation in the context of MDO as pertaining primarily to the automation process *downstream* of executing the analysis applications (O. and C. 1998, Isaacs, Sudhakar et al. 2003). For example, literature identifies the need for flexibility in problem formulation construction in the selection of various optimization algorithms and sequences (Eason and Wright 1992, Mourshed, Kelliher et al. 2003, Kroo 2004), the configuration of complex branching though a visual programming interface (Haymaker, Kunz et al. 2004), and in connecting, replacing, deleting, and adding processes to the problem (O. and C. 1998). However, the limitations in problem formulation capabilities encountered in our case study were due to decisions made *upstream* of the analysis applications, where geometry and other required inputs are generated and structured for analysis. No MDO literature addresses this component of problem formulation.

Therefore, we proposed an additional requirement for flexible problem formulation called a *dynamic exchange requirement structure*. In product model-based MDO, ERs are passed between the product model, the optimization interface (herein Optimization GUI), and/or the analysis application(s). How an ER is passed between these various stages of an MDO framework may vary, resulting in different ER assignment strategies. Figure 1 shows how the ER "Construction Type" may be assigned to different subsets of "Window" objects independently in a product model using three different ER assignment strategies.

The ability to rapidly reconfigure ER assignment strategies between optimizations without the need for software development results in a dynamic exchange requirement structure. With a methodology and some additional resource investment during the initial automation process to enable this functionality, overall costs in the application of MDO to support multiple problem formulations may be minimized. *No MDO methods currently exist to support a dynamic exchange requirement structure, limiting the practical applicability of MDO to AEC.* To fill this research gap, we developed, implemented, and validated an MDO method that enables flexible problem formulation through a dynamic exchange requirement structure called the Reference-Based Optimization Method (RBOM) (Welle and Haymaker 2011).

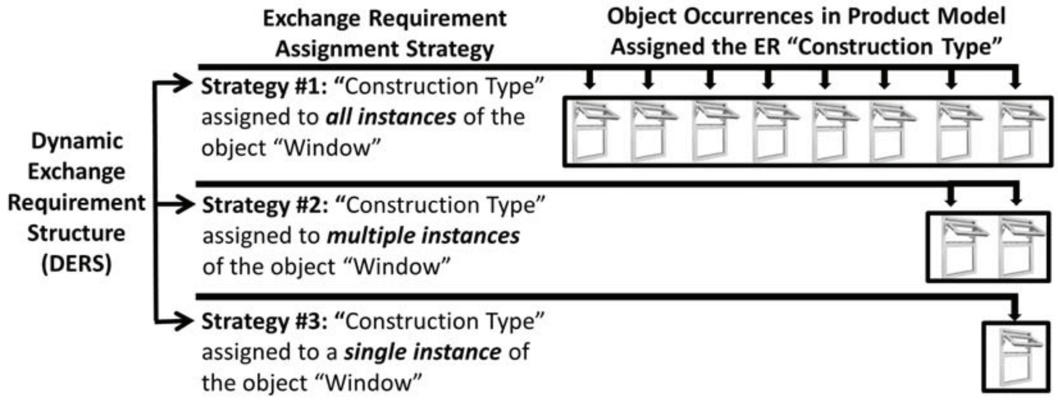

Figure 1: *A Dynamic Exchange Requirement Structure enables designers to assign optimization parameters to objects in a product model using different ER assignment strategies.*

Reference-Based Optimization Method

Our research proposes one specific method to support flexible problem formulation through a dynamic exchange requirement structure called RBOM. The method uses the product model concepts Object ID, Object Type, Object Group, and Object Attribute (Eastman 1999). The ultimate goal of RBOM is to help users match Object IDs from the product model with Object Attributes required for analysis in the desired configuration. It uses the concept of *References* to enable various mapping strategies to achieve this goal. A *Reference* is a mechanism to isolate unique instances of Objects for a given Object Type. The primary RBOM *References* are Global, Grouping, and Detailed. The *Reference* Grouping is further subdivided by the user to describe the unique nature of the group, for example Grouping:Orientation, Grouping:SpaceType,

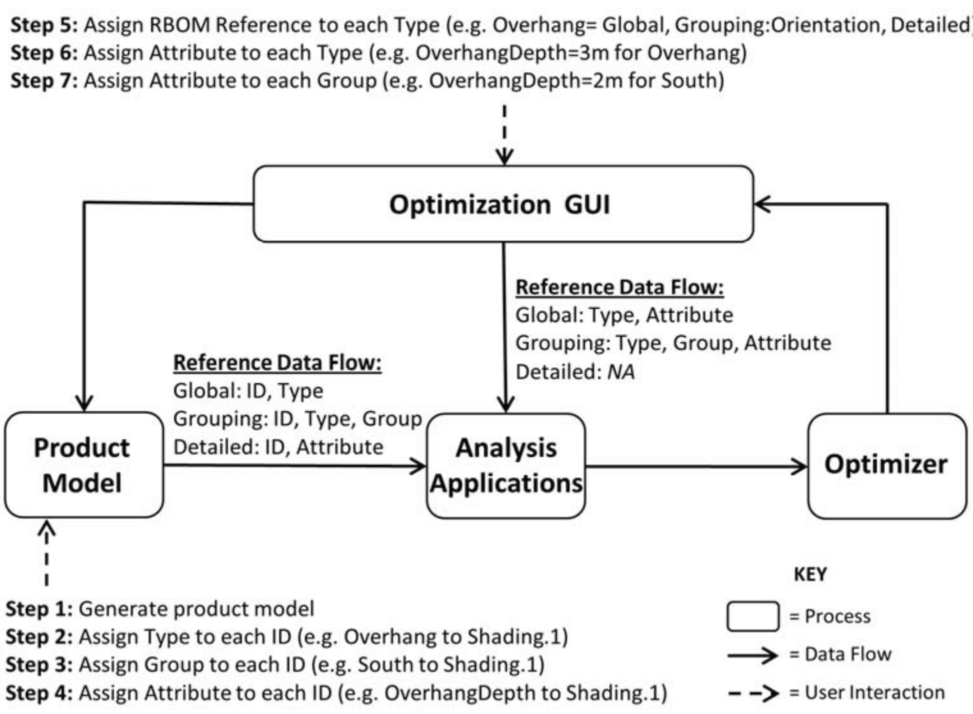

Figure 2: *RBOM guides the user through 7 steps for dynamically assigning exchange requirements.*

Grouping:FloorLevel, etc. Using these *References*, Object Atributes are assigned to either all instances of a given Object Type (Global), subsets of instances of a given Object Type (Grouping), or a single instance of a given Object Type (Detailed), matching the requirements of the three ER Assignment Strategies in Figure 1. The steps the user executes in RBOM, along with the corresponding data flows that result for each *Reference*, are shown in Figure 2 for a general product model-based MDO process.

More than one *Reference* may be assigned to a single Object Type. For example, if the user wants to isolate a group of Object Types located on the third floor of the south orientation, rather than creating a new *Reference* (i.e. Grouping:OrientationandFloorLevel), the user could assign two Groups in Step 3, then assign the two *References* Grouping:Orientation and Grouping:FloorLevel in Step 5.

Knowledge-Based Systems for RBOM

Engineering design automation involves: (1) identifying the relevant design knowledge; (2) providing a formalism for representing and processing the knowledge; and (3) implementing the formalism in a computer environment (Reddy, Gupta et al. 1992). We have just presented one specific technique for MDO design automation to provide the user the flexibility to formulate problems as necessary to meet their design challenge. However, few designers have the level of knowledge or heuristic capacity to handle the multiple sources of complexity during the MDO implementation process, resulting in its limited use in AEC. Knowledge-based systems have the potential to provide methods to assist designers in effectively managing MDO processes (Jiaoying, Feng et al. , Arora and Baenziger 1986, Boyle 1989).

Research has identified a wide range of AI applications for MDO, including evolutionary algorithms for parameter optimization and design space exploration (Bäck and Schwefel 1993), optimizer selection (Rogers and Barthelemy 1986), optimizer performance diagnosis (Arora and Baenziger 1986), Artificial Neural Networks (ANN) to improve the speed of GA-based building simulation environments (Magnier and Haghighat 2010), constraint specification and management (Gelsey, Schwabacher et al. 1998), objective specification (Boyle 1989), design variable modifications (Arora and Baenziger 1986), analysis preparation (Andrews Vogel 1990), and distributed computing (Girimonte and Izzo 2007). Many of these applications are supported by knowledge-based systems.

Knowledge-based systems, or expert systems, are computer programs containing knowledge about a narrow domain for solving problems within that domain, and consist of a knowledge base (domain knowledge expressed as general facts, rules and heuristics) and an inference mechanism (reasoning engine) (Pham and Pham 1999). Two types of knowledge bases can be used in MDO: (1) an existing knowledge base about a class of design problems developed over time and (2) a generated knowledge base that is developed for a particular design problem (Arora and Baenziger 1986). Remembering and reusing past events through Case-Based Planning (CBP) (Humm, Schulz et al. 1991) is one particular method that would strengthen an AEC knowledge base due to frequent similarities in energy performance trends for a given building type in a given climate. MDO knowledge bases must be integrated over various design domains to achieve robust knowledge bases (Curran, Verhagen et al. 2010). The use of knowledge-based systems for design optimization has many benefits, including improvements in cost and reliability and the documentation of the entire design process, from how goals are set to design assumptions to why decisions are made, among others (Stephanopoulos 1990) .

Several of the aforementioned AI applications for MDO deal with the issue of problem formulation. The application of AI to the component of problem formulation that RBOM addresses through a dynamic exchange requirement structure has never been discussed in literature, though the potential applications are significant. Though RBOM enables the *ability* to flexibly formulate information flows throughout an optimization to meet the needs of the user, it still relies on designers to understand what their needs are and make appropriate decisions as to when to apply certain *References* and when not to. These decisions are not always clear, frequently quite complex, and are impacted by specific design objectives, building physics, schedule and budget constraints for the project, as well as the resources available to continually assess the ongoing MDO process and be able to identify if the selected problem formulation was the right choice or not based on some set of criteria (which may evolve over time during the course of the design project as well).

For example, it may be decided by a design team that the appropriate RBOM *Reference* for a particular daylighting study is the Detailed *Reference*, for they believe optimizing for the construction type for every window in the building is preferable given the building footprint, climate, and diverse external shading profiles for the site. The team may be confident with the decision based on building physics alone, but soon realize that the size of the design space is 100,000 possible alternatives with an estimated simulation time for the optimization of around 15 days given the computing resources at their disposal and the long simulation times of their selected analysis applications. This time requirement may not meet their projected schedule constraints. Even if the MDO process time requirements fit within the schedule constraints, there may be diminishing returns to occupying their computing resources for that long given a set of alternative uses they could allocate them to. Additionally, after several days of running the Detailed *Reference* MDO, the results of the optimization may start to show that using a Grouping *Reference* based on facade orientation would have been sufficient in determining the primary impacts of the design alternatives on performance objectives, but at a fraction of the computing resources their initial decision incurred. A decision will need to be made whether to stop the optimization, start a new one, or use the results of the partially completed optimization.

This example highlights just a few of the challenges designers face in appropriately selecting a problem formulation method. Artificial intelligence research in the fields of computer-aided process planning (CAPP) (ElMaraghy 1993) and reconfigurable manufacturing systems (RMS) (Ismail, Musharavati et al. 2008) has a significant potential to help manage the wide range of uncertainties design teams face when deciding how to structure and manage their MDO process due to their common goal of maximizing process efficiency given a set of constraints, goals, and objectives. The same principles that motivate CAPP and RMS to identify and enable an ideal assembly line or manufacturing process given a range of technological or economic constraints to create a product also motivate the identification and enabling of the most efficient data structure and data flow for an MDO process in AEC (or any other industry) to create a product. RBOM provides a data structure that *enables* this type of problem reconfiguration to happen once the preferred problem formulation is identified, however it is the process of intelligent reasoning *to determine what the problem formulation should be* in the first place that still remains a major challenge for the industry. Rapid reconfigurability or redesign (Boyle 1989) of an MDO process is critical to the widespread adoption and effectiveness of MDO in today's design environments given the high first-costs of implementation. Knowledge-based systems that have the intelligence to determine cost-effective responses to unpredictable changes in design requirements given a set of goals, constraints, and objectives applied to the MDO process itself rather than the alternative generation and analysis process hold much promise.

The development of an MDO problem formulation knowledge base for energy-efficient building design is a formidable challenge. When considering just the domain of passive thermal performance, which refers to all non-mechanical energy flows in a building, potential domain knowledge that could be used for a baseline knowledge base includes the following:

Building Physics

- Daily, seasonal, and annual temperature profiles
- Daily, seasonal, and annual solar profiles
- Daily, seasonal, and annual wind profiles
- Daily, seasonal, and annual external shading profiles
- Internal load profiles (occupant, lighting, and equipment)
- Conditioning requirements
- Ventilation requirements
- Operating schedules
- Building shape, dimensions, and other geometric parameters
- Building construction types
- Passive thermal design strategy (passive heating and cooling, thermal mass, natural ventilation, etc.)
- Control strategies
- Case-based performance evaluations

Project Constraints and Objectives

- Budget
- Schedule
- Energy code and sustainable design rating system compliance

MDO Process Constraints and Objectives

- Design and analysis application simulation times, accuracy, and reliability
- Optimizer performance
- Design variable and performance constraint performance relative to design objectives

While research has suggested that the use of AI, in particular knowledge bases, is well suited to the field of engineering design (Arora and Baenziger 1986) and the early stages of building design (Sabouni and Al-Mourad 1997) due to the prevalent use of rules-of-thumb developed over the years, this same tendency to rely on precedent-based design, typically using antiquated rule-of-thumb based on empirical data from buildings built over 30 years ago, is a major barrier to successfully realizing high-performance built environments in AEC. Caution must be used to ensure that any knowledge bases developed and

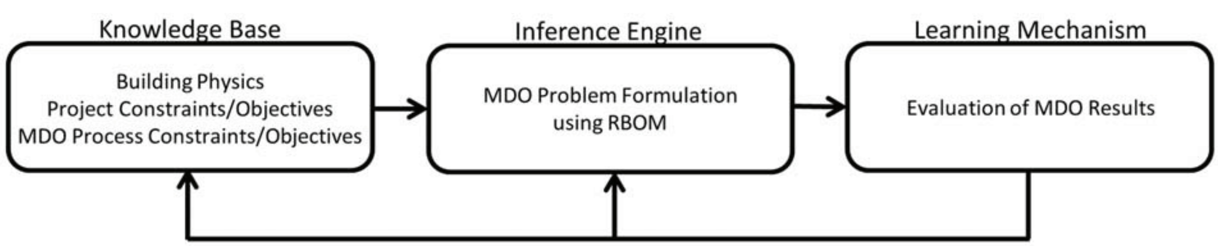

Figure 3: *RBOM guides the user through 7 steps for dynamically assigning Exchange Requirements.*

leveraged for MDO problem formulation in the domain of energy-efficient buildings employ effective learning mechanisms for the continual improvement of both domain knowledge and the inference engine. Figure 3 shows a high-level overview of how a knowledge-based system for passive thermal MDO problem formulation could operate using RBOM.

Conclusions

Sustainable building design requires a new paradigm of performance-driven design processes. Design teams must leave behind antiquated precedent-based processes to engage in rapid multidisciplinary design and analysis to truly optimize the multidisciplinary performance of their designs. The variability in the challenges faced by designers requires that they be able to flexibly formulate the structure of the optimizations. This paper describes one method to enable such flexible problem formulation.

Choosing the correct problem formulation remains a challenge for designers, even given this new flexibility. Ironically, this is where precedent-based knowledge can have an important constructive role in the formulation of optimization problems. Appropriately structured knowledge bases, intelligent methods to uses this knowledge in formulating good optimization problems, and advanced techniques in improving the knowledge bases over time are important developments for lowering the cost and improving the effectiveness of MDO in AEC.

Acknowledgements

This research is funded by the Center for Integrated Facility Engineering (CIFE) and the Precourt Energy Efficiency Center (PEEC) at Stanford University.

References

AIA, AIA Architecture 2030 Committment Homepage, 2011. Accessed in January 2011, from: http://www.aia.org/about/initiatives/AIAB079458.

AIAA 1991. Current state of the art in multidisciplinary design optimization.,

A. Andrews Vogel 1990. Automated domain decomposition for computational fluid dynamics. Computers & Fluids 18: 329-346.

J. S. Arora and G. Baenziger 1986. Uses of artificial intelligence in design optimization. Computer Methods in Applied Mechanics and Engineering 54: 303-323.

T. Bäck and H.-P. Schwefel 1993. An Overview of Evolutionary Algorithms for Parameter Optimization. Evolutionary Computation 1: 1-23.

V. Bazjanac and A. Kiviniemi, Reduction, simplification, translation and interpretation in the exchange of model data., D. Rebolj (Ed.) 24th Conference on Bringing ITC Knowledge to Work, Faculty of Civil Engineering, University of Maribor, Moribar, Slovenia, 2007, pp. 163-168.

J. P. T. J. Berends and M. J. L. v. Tooren, MDO Design Support by Integrated Engineering Services within a Multi-Agent Task Enviornment, 26th International Congress of the Aeronautical Sciences, Anchorage, Alaska, 2008, pp. 1-18.

J. M. Boyle 1989. Interactive engineering systems design: A study for artificial intelligence applications. Artificial Intelligence in Engineering 4: 58-69.

buildingSMART, Information Delivery Manual (IDM) Homepage, 2009. Accessed in January 2011, from: http://www.iai.no/idm/.

L. Caldas 2008. Generation of energy-efficient architecture solutions applying GENE_ARCH: An evolution-based generative design system. Advanced Engineering Informatics 22: 59-70.

C. Clevenger, and Haymaker, J., Framework and Metrics for Assessing the Guidance of Design Processes, 17th International Conference on Engineering Design, Stanford, CA, 2009.

R. Curran, et al. 2010. A multidisciplinary implementation methodology for knowledge based engineering: KNOMAD. Expert Systems with Applications 37: 7336-7350.

E. Eason and J. Wright, Implementation of Non-Hierarchic Decomposition for Multidisciplinary System Optimization, AIAA/NASA/USAF/OAI 4th Symposium on Multidisciplinary Analysis and Optimization, Cleveland, OH, 1992, pp. 1008-1016.

C. M. Eastman 1999. *Building Product Models: Computer Environments Supporting Design and Construction*. CRC Press, Inc.

H. A. ElMaraghy 1993. Evolution and Future Perspectives of CAPP. CIRP Annals - Manufacturing Technology 42: 739-751.

FEMP 2007. Energy Independence and Security Act (EISA) of 2007, Report #: P.L. 110-140 (H.R. 6) Washington, D.C.

F. Flager, et al. 2009. Multidiscplinary Process Integration and Optimization of a Classroom Building. Journal of Information Technology in Construction (ITCon) 14: 595-612.

A. Gelsey, et al. 1998. Using modeling knowledge to guide design space search. Artificial Intelligence 101: 35-62.

P. Geyer 2009. Component-oriented decomposition for multidisciplinary design optimization in building design. Advanced Engineering Informatics 23: 12-31.

D. Girimonte and D. Izzo 2007. *Artificial Intelligence for Space Applications*. London, UK:Springer.

J. Haymaker, et al. 2004. Perspectors: Composable, Reusable Reasoning Modules To Construct an Engineering View from Other Engineering Views. Advanced Engineering Informatics 18: 49-67.

D. Holzer, et al. 2007. *Developing a Framework for Linking Design Intelligence from Multiple Professions in the AEC Industry*. Springer.

B. Humm, et al. 1991. A system for case-based process planning. Computers in Industry 17: 169-180.

A. Isaacs, et al., Design and Development of MDO Framework, International Conference on Modeling, Simulation, Optimization for Design of Multi-disciplinary Engineering Systems, Goa, India, 2003, pp. 1-5.

N. Ismail, et al., Manufacturing process planning optimisation in reconfigurable multiple parts flow lines, World Academy of Materials and Manufacturing Engineering, 2008.

S. Jiaoying, et al. Artificial intelligence in Computer Aided Design. Computers in Industry 8: 277-282.

I. Kroo 2004. Distributed Multidisciplinary Design and Collaborative Optimization,

D. S. Lazzara 2008. CAD-Based Multidelity Analysis and Multidisciplinary Optimization in Aircraft Conceptual Design. PhD Thesis Proposal, Massachusetts Institute of Technology, Cambridge, MA.

L. Magnier and F. Haghighat 2010. Multiobjective optimization of building design using TRNSYS simulations, genetic algorithm, and Artificial Neural Network. Building and Environment 45: 739-746.

A. M. Malkawi 2004. Developments in environmental performance simulation. Automation in Construction 13: 437-445.

R. Mora, et al. 2008. A geometric modelling framework for conceptual structural design from early digital architectural models. Advanced Engineering Informatics 22: 254-270.

M. Mourshed, et al., ArDOT: A tool to optimize environmental design of buildings, Eighth International IBPSA Conference, Eindhoven, Netherlands, 2003, pp. 919-926.

S. A. O. and T. J. C. 1998. Framework Requirements for MDO Application Development, Report #: AIAA-98-4740, NASA Langley Research Center, Hampton, VA.

K. Papamichael and J. P. Protzen, The Limits of Intelligence in Design, Focus Symposium on Computer-Assisted Building Design Systems 4th International Symposium on Systems Research, Informatics and Cybernetics, Baden-Baden, Germany, 1993, pp. 1-10.

D. T. Pham and P. T. N. Pham 1999. Artificial intelligence in engineering. International Journal of Machine Tools and Manufacture 39: 937-949.

R. R. Reddy, et al. 1992. Heuristic, symbolic logic and knowledge-based approach to the design and construction of buildings. Computers & Structures 43: 1191-1197.

J. L. Rogers and J.-F. M. Barthelemy 1986. An expert system for choosing the best combination of options in a general purpose program for Automated Design Synthesis. Engineering with Computers 1: 217-227.

A. R. Sabouni and O. M. Al-Mourad 1997. Quantitative knowledge based approach for preliminary design of tall buildings. Artificial Intelligence in Engineering 11: 143-154.

K. Shea, et al. 2005. Towards integrated performance-driven generative design tools. Automation in Construction 14: 253-264.

G. Stephanopoulos 1990. Artificial intelligence in process engineering--current state and future trends. Computers & Chemical Engineering 14: 1259-1270.

C. Struck, et al. 2009. An investigation of the option space in conceptual building design for advanced building simulation. Advanced Engineering Informatics 23: 386-395.

J. C. Townsend, et al. 1998. Integration of a CAD System into an MDO Framework, Report #: NASA/TM-1998-207672, Hampton, VA.

W. Wang, et al. 2005. An object-oriented framework for simulation-based green building design optimization with genetic algorithms. Advanced Engineering Informatics 19: 5-23.

B. Welle and J. Haymaker 2011. Reference-Based Optimization Method (RBOM): Enabling Flexible Problem Formulation for Product Model-Based Multidisciplinary Design Optimization, CIFE Technical Report #195, Stanford, CA.

R. F. Woodbury and A. L. Burrow 2006. Whither design space? Artificial Intelligence for Engineering Design, Analysis and Manufacturing 20: 63-82.