CBArch: A Case-Based Reasoning Framework for Conceptual Design of Commercial Buildings

Andrés Cavieres, Urjit Bhatia, Preetam Joshi, Fei Zhao and Ashwin Ram

Georgia Institute of Technology
676 Marietta Street NW
Atlanta, Georgia 30318
andres.cavieres@gatech.edu, urjitsb87@gmail.com, preetamjoshi93@gmail.com, feizhao@gatech.edu, ashwin@cc.gatech.edu

Abstract

The paper describes the first phase of development of a Case-Base Reasoning (CBR) system to support early conceptual design of buildings. As specific context of application, the research focuses on energy performance of commercial buildings, and the early identification of energy-related features that contribute to its outcomes. The hypothesis is that bringing knowledge from relevant precedents may facilitate this identification process, thus offering a significant contribution for early analysis and decision-making. The paper introduces a proof-of-concept for such a system, proposing a novel integration of Case-Based Reasoning, Parametric Modeling (Building Information Modeling), and Ontology Classification. While CBR provides a framework to store and retrieve cases at an instance level, Parametric Modeling offers a framework for rule-based geometric adaptation and evaluation. The ontology is intended to provide a semantic representation, so that new design concepts can be created, classified and retained for further reuse. Potential advantages and limitations of this three-level integration approach are discussed along with recommendations for future development.

Introduction

The problem of how to provide timely access to design knowledge contained in precedents has been subject of study of Cased-Based Reasoning since its origins (Kolodner 1991), (Aamodt and Plaza 1994). A popular approach has been to support the design inception stage, when early decisions usually have a greater impact on the final product (Pearce et al. 1992), (Wang et al. 2002).

In the building design domain, many different systems have been proposed to address this problem. Archie (Pearce et al. 1992), Precedents (Oxman 1993), FABEL (Voss et al. 1994), Design-MUSE (Domeshek et al. 1994), CADSYN and CASECAD (Maher et al. 1995), SEED (Flemming and Woodbury 1995) and CADRE (Hua et al. 1996) are among the most important efforts.

The re-usability of design knowledge has also been studied more recently in the context of Building Information Modeling. One of the main approaches has been to consider rule-based parametric models as the basic representations for embedding design expertise, specially regarding fabrication and construction constraints (El-Bibany et al. 1998), (Sacks et al. 2003).

Our research elaborates on some aspects developed by previous CBR efforts, recognizing at same time important advantages in the integration of CBR with parametric modeling. While CBR provides a framework to store and retrieve relevant cases at instance level, Parametric Modeling offers more appropriate mechanisms for geometric adaptation and evaluation based on explicit rules and procedures (Lee et al. 2005), (Cavieres et al. 2009). In some of the early CBR work that addresses the problem of shape adaptation, the emphasis has been mostly on strategies of parametric dimensional adaptation (CADRE), or topological adaptation of two-dimensional geometry (SEED).

A second aspect of elaboration is the way by which the system attempts to supports knowledge-base growing. CBArch follows the model proposed by SEED, in which the case memory should grow automatically as the design activity proceeds, and new relevant solutions are found. In this way it also considers the integration of a episodic knowledge (precedents) with semantic knowledge (types and relationships), in order to provide more flexible classification and querying capabilities (Flemming and Woodbury 1995), (Fenves et al. 2000). In CBArch, an ontology model is intended to provide this semantic level, so that new design concepts can be created, classified and retained for further reuse. At the current stage of implementation, only a category of building shapes is described in the ontology.

Regarding the context of application, CBArch aims to support conceptual design of commercial buildings from an energy performance perspective. This was considered to be an useful starting point to set-up the framework, as extensive information about energy consumption of commercial building is publicly available on the web.

The model adopted for energy performance evaluation considers three CBR steps: 1) retrieval of real-world building cases, including specific information about properties related to energy consumption; 2) adaptation of retrieved cases into new solutions; and 3) simplified (normative) energy evaluation of new solutions, and their benchmarking against portfolios of real-world cases.

This approach for early energy evaluation in CBArch is currently in progress, reflecting on recent initiatives to promote energy efficient building design by means of normative building thermal load calculation and performance benchmarking.

It is important to mention that the work presented in the paper is exploratory, and its implementation is still at initial stages. The main effort has been the integration of the different applications needed to support each module of the framework. For this reason, most of the data structures and algorithms implemented are fairly simple, needing further revision and development.

The paper structure is as follows. First, the paper introduces the context of application, describing the relevance of conceptual design and the need for early energy performance assessment. Next, the paper describes the main source of building cases adopted in this study. In the following section the paper provides an overview on the architecture of the proposed framework. Then, a more detailed description of each step in the CBR cycle is presented. Finally, potential benefits, limitations and lessons learned from this preliminary stage are discussed, along with recommendations for future work.

Context of Application

The general goal of the proposed framework is to assist architects and engineers in the exploration of alternative building configurations at conceptual level. Conceptual design essentially focuses on the definition of basic geometric models that capture the main characteristics of the design intent. Some of these characteristic are the size, the overall building shape, its orientation and how the main activities are related with each other, and distributed within the site, among others. Many of these characteristic have a large impact in the life-cycle performance of buildings, particularly from a energy consumption perspective.

Thus, the assessment of relevant examples can provide a valuable initial guidance to improve early decision-making, specially in the context of complex buildings, like office complexes, laboratories, schools or hospitals. In this scenario, one fundamental task is the identification and selection of building (sub)systems and features that relate with energy performance (Augenbroe and Park 2005).

To facilitate this task, the authors developed a database containing information about energy consumption of commercial buildings in the United States, based on information provided by the Commercial Buildings Energy Consumption Survey (CBECS 2003). This survey was elaborated in 2003 by the U.S. Energy Information Administration (EIA) and released for public use in 2008. The surveyed data sets comprises 5210 individual records, each containing information over sixty different building properties, including annual energy performance and costs.

With the emerging agenda of sustainable design and high performance buildings, detailed information like the one provided by CBECS is becoming increasingly available. At the same time, building energy performance calculation and benchmarking methods have been rapidly developed. Due to the characteristics of the data available in CBECS and the approach adopted in CBArch, a simple-heat-balance-based calculation method is proposed for energy evaluation of early design alternatives. This method is adapted from the ISO 13790,and its normative building energy calculation and rating standards.

CBArch Framework Overview

A traditional CBR model follows a cycle made of four main steps: 1) Retrieve, 2) Adapt, 3) Evaluate, and 4) Store (Kolodner 1993). Sometimes, the representation of both the retrieved and adapted cases are the same, therefore only one case repository is needed. However, in CBArch two different types of representation are needed; the 'real-world' precedents, represented as feature vectors for similarity assessment, and the parametric representations required for graphical visualization, design adaptation and evaluation. Figure 1 illustrates the CBArch cycle.

For this reason, the data model from the best retrieved match(es) need to be mapped into appropriate parametric representations. The key link between both is the building shape attribute described in the CBECS database. This description is based on a enumeration of basic footprint types denoted by literals, such as “U-shape”, “H-shape” or “Square shape” and so on. The articulation between real-world building cases and their parametric co-relations is done through an ontology of general building shape types. At the beginning these types are few, corresponding to a subset of the enumerated types defined in the CBECS database. One of the goals of CBArch is that this small ontology of shape types must grow interactively as new design solutions are found.

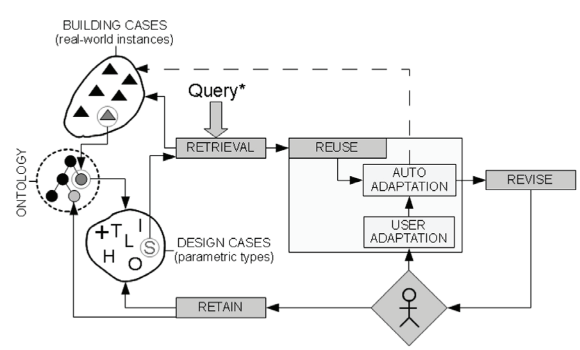

Figure 1: CBArch framework.

Implementation

At the current stage, the implementation focuses primarily on the integration of different software applications, and the corresponding mapping between their representations.

At the core of the framework is a parametric modeling software, used for the generation of building design information. This application works as the front-end environment in which the user initializes the query, visualizes the retrieved cases, and performs adaptation and evaluation.

In this research the authors used GenerativeComponents (GC) parametric modeler and its API to build connections with the other modules. These modules include the real-world building database built from the CBECS data sets (stored in a MySQL database), the parametric prototype database (stored in a Access database), the building shape ontology (defined in OWL-DL) and the ontology reasoner (Jena and Pellet API's).

Each module has a specific use in one the main four steps of the CBR cycle. The retrieval is reviewed first, including a description of the main inputs. Adaptation (reuse) is presented next, with emphasis on topological adaptation. Then, a candidate approach for energy evaluation (revise) is introduced in the context of early energy performance assessment. Finally, a temptative model for storage step (retain) is presented, including instance and concept retainment.

Retrieval

The retrieval is the first step in the CBR cycle. The similarity criteria for selection of cases depend on the balance between availability of data and the level of detail required for a given phase of the design process. At the same time, case properties of interest at the conceptual stages can vary according to each problem, set of goals or specific design practices.

Based on this observation, only a small subset of all the properties described in the CBECS database were considered useful for similarity comparison. Therefore many of the sixty features available were filtered out to optimize the search.

Figure 2 shows a dialog window containing the main inputs required. Initially, only twelve features are needed to provide an useful the similarity evaluation. The inputs correspond to a minimum set of properties required to query for cases with an expected similar energy profile. Building shape, main building activities, total area, total operating hours per week, number of employees during main shift and isolation from the buildings correspond to this basic description of building requirements. Census division corresponds to a a standard subdivision of the territory of United States, and it provides a rather coarse description of geographic location and whether conditions.

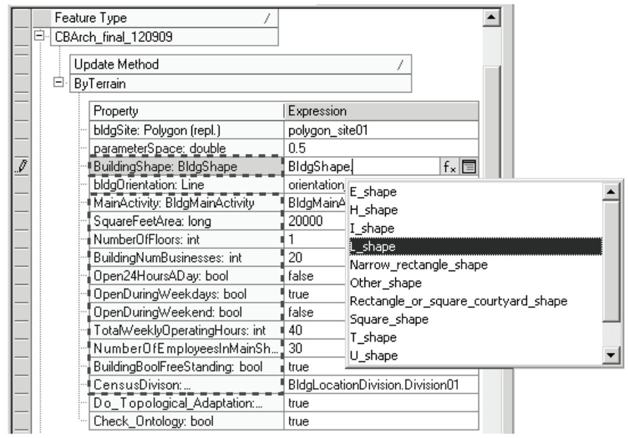

Figure 2: User query with 12 features (dashed areas).

The retrieval method developed in CBArch was based on *k*-nearest neighbors algorithm (*k*-NN) using Euclidean distance as main similarity metric. This algorithm was chosen because the source cases in the CBECS database are described mostly in terms of continuous variables, and therefore could be treated as feature vector representations.

However some key attributes, such as building shape, main building activity or geographic location cannot be treated in the same way. A challenge faced during the implementation was to find quantifiable ways of comparing them. For example, to evaluate the similarity between two different building shapes requires both domain knowledge and certain amount of subjectivity. To solve this problem, architects were consulted and asked to provide a look-up tables containing percentages of similarity based on their judgment and expertise. These look-up tables were then used as heuristic rules by the retrieval algorithm in order to deal with non-easily comparable features. In a similar fashion, other heuristics include:

A) Exponential Scaling: used for features like number of floors. A building with 2 floors can be considered different from a building with 5 floors, but another building with 20 floors is not as different from one that has 27-30 floors. Thus as the base value of measurement (number of floors) increases, the significance of the gap decreases.

B) Magnification (Linear Scaling): used for filtering some features like the total number of operating hours per week. This gives us a way to use purely numerical values. Other cases are thus considered to be too far away from the test case.

C) Direct Testing: In cases of truth values or fixed valued functions over the feature vectors, direct testing was used. If the values match, then the similarity is positive otherwise zero.

After a successful retrieval, the attribute values of a selected case gets partially replicated into the data structure of a parametric object. For instance, all the information requested at the query stage gets replicated in the parametric model. Additional information such as building materials, façade properties, glazing percentage and sun protection devices necessary for energy performance evaluation are instantiated as well.

One important assumption was that this additional level of detail brought by the retrieval process can eventually provide useful insights, increasing the awareness about features and issues not taken in consideration previously by the user (Domeshek et al. 1994).

Adaptation (Reuse)

Adaptation is performed after a best match or a list of best matches is retrieved. Then, a parametric model gets instantiated by automatic selection from a library of parametric templates. These templates are adjusted to fit the dimensional characteristics of the retrieved case, as well as the main building orientation defined by the user. However, in most situations an instantiated case will not perfectly fit the geometric constraints of a different site. In such situations the user or the system are expected to proceed with some kind of shape adaptation.

There are two basic approaches for shape adaptation, namely geometric adaptation and topological adaptation (Schmitt et al. 1997). Another important assumption made in this project is that sometimes non-geometric parameters may drive the topological / geometrical features of a building, but the opposite is normally the case. For example, many empirical calculations used to provide an initial estimation of energy are based on geometric dependent properties, such as ratios between linear dimensions, volumes and surface areas (e.g. glazing percentages as function of wall area).

Both types of adaptation can be performed either by the user himself or by some automatic procedure. Other non-geometric properties relevant to evaluation are expected to be manually modified from the retrieved data, or added by the user as needed. In this paper we focus on automatic procedures for geometrical and topological adaptation of building shapes, as drivers for the adaptation of non-geometrical parameters.

Geometric Adaptation The geometric adaptation is the simplest shape adaptation method. The set of rules to achieve geometric adaptation can be summarized by means of three basic geometric transform operators: Move, Rotate and Scale. In CBArch the adaptation process starts with geometric adaptation, so to accommodate the parametric building representation by some combination of these operators. However, despite that geometric adaptation makes much sense from a design perspective, and it is part of the natural capabilities of parametric modeling systems, the current implementation of CBArch does not explore this approach in depth. The reason lies on the fact that geometric adaptation does not produce new shape types, but only dimensional variation of existing types. Because this kind of variation can be achieved procedurally, there is no special need for retaining a geometric adapted form.

Topological Adaptation A more interesting scenario is provided by topological adaptation. The reason is that new shape configurations have different meanings from a building design perspective, implying new architectural concepts which are worthier of keeping and reusing.

The main algorithm involved in topological adaptation follows a simple heuristic procedure based on an adjacency list representation of the building shape (Figure 3).

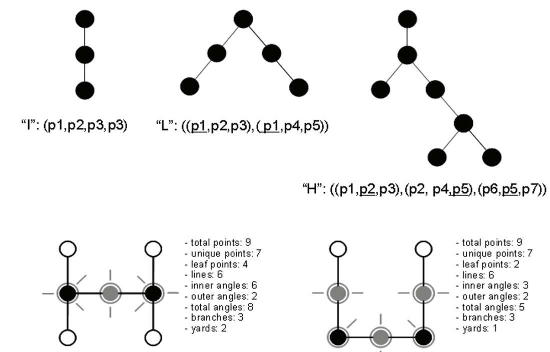

Figure 3: Adjacency list representations for a simplified building shape topology.

Any part (space component) of the building mass model that is failing to fit within the boundaries of the site must be re-positioned at the closest empty spot available within its original branch. This rule corresponds to a basic design criteria according to which only compatible spaces should be put together. Since there is no information available regarding specific sub-space types and uses, it is assumed that spaces in the same branch are compatible. If there is no empty spot available in the same branch, then a new branch has to be created.

The topological adaptation algorithm repeats these two steps iteratively until a solution is found for the same floor level. Otherwise, the search procedure continues on the next floor level above the original position, and son on. Such re-accommodation leads to a change on the configuration of the building layout, whenever a new branch has to be created at ground level to accommodate a misfit space. The system keeps track of connectivity relations between graph nodes for later classification of shapes by the ontology reasoner. This also includes the recognition of cycles for identification of shapes such as "Square_with_courtyard" (Figure 4).

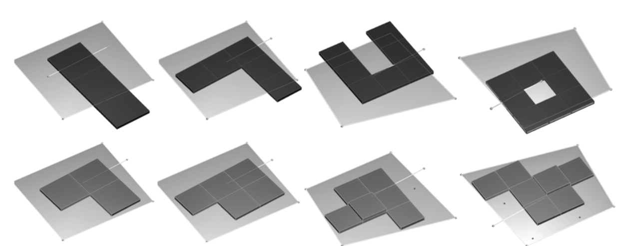

Figure 4: Series of automatic topological adaptation in CBArch. Top row: standard retrieved shapes. Bottom row: adapted shapes.

Evaluation (Revise)

In a design domain like architecture, evaluation is a complex process that ranges from subjective assessment of qualitative aspects such as aesthetics, to more objective quantitative performance analysis.

In this framework, once the building design has been retrieved from the database and adapted considering local constraints, a quick quantitative energy performance evaluation to the building is proposed. This can be achieved by performing a simplified normative calculation of the building annual energy consumption for a given set of design parameters, and then could lead to adjustments in specific design parameters.

We propose to adapt the ISO 13790 (ISO 2008) approach to calculate building thermal loads for this task. A web-based toolkit (Augenbroe et al. 2010) of this normative assessment approach has been implemented to quickly estimate building energy performance at the early design stage.

The real-world building database in CBArch consists of energy-related configurations associated with construction materials for walls and roofs. We first of all assign typical material properties (U-value, solar transmittance, shading factor, etc.) to these configurations, and then calculate the annual thermal load intensity of the parametrically instantiated case using these parameters. In addition, the evaluation module also gives energy performance ranking of the design case compared with real-world buildings in the database.

In order to achieve a more energy efficient design, the user can provide an expected energy performance ranking percentile (e.g. better than 80% of the real-world buildings). If a given design does not meet the expectation, CBArch system or the user could search through different combinations of design modifications on (1) heat insulation improvement, (2) solar heat gain reduction and (3) infiltration reduction to provide a list of design revisions that will lead to energy efficiency. The user then selects the best design based on his expert assessment of the evaluation results.

Store (Retain)

The current implementation focuses exclusively on the outcomes of topological adaptation as the solutions to be classified and stored. This is because topological adaption may lead to new building shape configurations that can be further reused and adapted.

The retain stage therefore is two-folded; it has to store newly created configurations as concepts pertaining to the category of building shapes, as well as the specific instances that exemplify such concept. For that purpose CBArch makes use of a small ontology of building shapes modeled in OWL-DL language using Protégé ontology editor (Figure 5). CBArch also integrates with Pellet reasoner for automatic classification and storage of new shape types. The integration of CBR with OWL-DL ontologies for classification of cases was initially inspired by the jCOLIBRI framework (Recio-García et al. 2006).

The main motivation behind a building shape ontology was to define a high level representation for control and administration of the design rules contained in parametric libraries. This shape ontology is based on a simplified model of building topology, intended to capture an intuitive understanding of standard building layouts and their possible derivations.

In order to classify a shape, the ontology reasoner requires a description of the connectivity structure of the graph representation underlying the parametric model. This structure is parsed by the system on demand, and used to derive the set of properties required to perform automatic classification.

Since pre-existing standard shape types are specified as non-disjoint classes, multi-inheritance of new shapes is allowed. In this way, a new shape can be classified either under existent standard type, or a sub-type of one or more existing standard types. If the shape is well evaluated by the designer, it can be stored in the ontology and made available for further reuse.

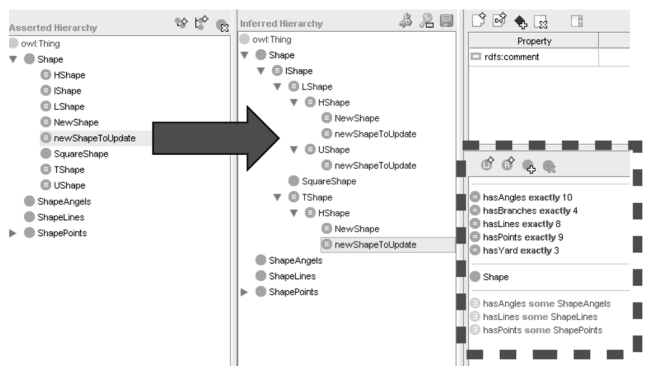

Figure 5: Ontology in Protégé (OWL-DL). Classification of building shapes based on parametric model topology graph.

Discussion

The system as proof-of-concept provides some preliminary results. It successfully retrieves cases from the database of commercial buildings, and instantiates them as parametric representations. It also supports geometric and topological adaptation according to the specified set of rules, and performs classification of basic adapted shapes as new concepts in the building shape ontology when appropriate.

From these initial results, some observations can be made. First, the automatic retrieval and population of technical properties from real world precedents can be considered initially as an useful source of information to facilitate early design-analysis integration, and decision-making.

However, despite some level of usefulness, this approach is limited. Because retrieval is restricted to structural property-value combinations, it does not provide deep insights about the causal relationships and principles behind a given building physical behavior. Indeed, some researchers have criticized this approach as source of superficial knowledge, and therefore have argued for the need of behavioral and functional representation as means to provide deeper design knowledge and reasoning (Chandrasekaran et al. 1993), (Qian and Gero 1996), (Goel 1997), (Brown 2003).

From this perspective, the classification of building (sub)systems under an ontology of functional categories may provide a potential area of research, specially regarding the integration of specialized quantification methods to support early building performance evaluation and optimization.

Similarly, another related application for ontologies in design is the classification and management of parametric models according to the design rules they are intended to satisfy, as specific forms of design knowledge. Currently, parametric modeling environments do not provide support for functional categorization of such rules, restricting in this manner their re-usability.

Regarding topological adaptation, CBArch showed sometimes unexpected results. This fact can be evaluated favorably in the context of conceptual exploration, because it opens the spectrum of possibilities beyond what was initially requested by the designer. However, the representation of buildings in CBArch is still too simplistic, even for conceptual design. Additional levels of design information have to be added to facilitate the elaboration of more complete building models, including specific space requirements, expected activities and associated equipment. This should lead to a richer set of adaptation rules, including the consideration of energy related contextual constraints, such as natural light and natural ventilation patterns, etc.

Further work in CBArch also has to explore better integration with energy evaluation methods. The proposed energy evaluation (revise) module suggests an adaptation of standard normative methods for quick thermal load calculation, and benchmarking against real-world buildings described in the database. This approach is considered to be appropriate for early conceptual design, because it allows a quick estimation of alternatives based on few building design parameters. However, despite the CBECS 2003 data sets were a good starting point for setting-up the framework, it does not provide detailed enough building design information (occupant schedule, window-wall ratio, material properties, etc.). Because of that, many input parameters of the calculation have to be assumed by default. These rough assumptions can be reduced if the new version of CBECS 2007 provides richer energy-related building information.

References

Kolodner, J.L. 1991. Improving Human Decision-Making through Case-Based Decision Aiding. *AI Magazine* 12(2): 52-68.

Aamodt, A and Plaza E. 1994. Case-Based Reasoning: Foundational Issues, Methodological Variations, and System Approaches. *Ai Communications* 7(1): 39-59.

Pearce, M., Goel, A., Kolodner, J., Zimring, C. , Sentosa, L. and Billington, R. 1992. Case-Based Design Support: A Case Study in Architectural Design. *IEEE Expert: Intelligent Systems and Their Applications.* 7(5): p. 14-20.

Wang, L., Weiming, W., Sie, H., Neelamkavil, J., Pardasani, A. 2002. Collaborative Conceptual Design: State of the Art and Future Trends. *Computer-Aided Design* 34(13): 981-996.

Oxman, R.E. 1993. Precedents in Design: A Computational model for the Organization of Precedent Knowledge. *Design Studies* 15(2): 141-157.

Voss, A., Coulon, C-H., Gräther, W., Linowski, B., Schaaf, J., Bartsch-Spörl, B., Börner, K., Tammer, E., Dürschke, H., and Knauff, M. 1994. Retrieval of Similar Layouts – About a Very Hybrid Approch in FABEL, in: Gero, J.S. and Sudweeks, F. (eds.) *Artificial Intelligence in Design '94*, Kluwer Academic, Dordrecht, NL, p. 625-640.

Domeshek, E.A., Kolodner, J.L. and Zimring, C. 1994. The Design of a Tool Kit for Case-Based Design Aids. J.S. Gero, ed. *AID94: Third International Conference on Artificial Intelligence and Design*. The Netherlands: Kluwer Academic Press.

Maher, M.L., Balachandran, M., Dong, M.Z. 1995. *Case-Based Reasoning in Design*. Lawrence Erlbaum Associates.

Flemming, U. and Woodbury, R. 1995. Software Environment to Support Early Phases in Building Design (SEED): Overview. *Journal of Architectural Engineering* 1(4): 147-152.

Hua, K., Fairings, B. and Smith, I. 1996. CADRE: Case-Based Geometric Design. *Artificial Intelligence in Engineering* 10(2): 171- 183.

El-Bibany, H. E. and Paulson B.C, Jr. 1998. A Parametric Architecture for Design, Management, and Coordination in a Collaborative AEC Environment. *Computer-Aided Civil and Infrastructure Engineering* 14(1): 1-13.

Sacks, R., Eastman, C. and Lee, G. 2003. Parametric 3D Modeling in Building Construction with Examples from Precast Concrete. *Automation in Construction* (13): p. 291-312.

Lee, G., Sacks, R. and Eastman, C. 2005. Specifying Parametric Building Object Behavior (BOB) for a Building Information Modeling System. *Automation in Construction*, (15): 758 – 776.

Cavieres, A., Gentry, T.R., Al-Haddad, T. 2009. Rich Knowledge Parametric Tools for Concrete Masonry Design: Automation of Preliminary Structural Analysis, Detailing and Specifications. In Proceedings of the 26th International Symposium on Automation and Robotics in Construction (ISARC), 544-552. Austin, Texas.

Fenves, S.J., Rivard, H .and Gomez, N. 2000. SEED-Config: A Tool for Conceptual Structural Design in a Collaborative Building Design Environment. *Artificial*

Intelligence in Engineering 14(3): 233-247.
Augenbroe, G. and C.-S. Park. 2005. Quantification Methods of Technical Building Performance. *Building Research & Information* 33(2): 159-172.
U.S. Energy Information Administration (EIA). 2003. Commercial Building Energy Consumption Survey. From http://www.eia.doe.gov/emeu/cbecs
Schmitt, G., Dave, B. and Shih, S-G. 1997. Case-Based Architectural Design, in: Maher M.L. and Pu, P. (eds.) *Issues and Applications of Case-Based Reasoning*, Lawrence Erlbaum Associates, Inc. p. 241-260.
ISO (2008). 13790:2008. Energy Performance of Buildings - Calculation of Energy Use for Space Heating and Cooling.
Augenbroe, G., Lee, S., Lee, J., and Zhao, F. 2010. EPSCT: Energy Performance Standard Calculation Toolkit. From http://epsct.org
Recio-García, J.A., Díaz-Agudo, B., González-Calero, P.A., Sánchez- Ruiz, A.A. 2006. *Ontology based CBR with jCOLIBRI. In Proceedings of the Twenty-sixth SGAI International Conference on Innovative Techniques and Applications of Artificial Intelligence*, Springer-Verlag.
Chandrasekaran, B., A. Goel, et al. 1993. Functional representation as design rationale. *Computer* 26(1): 48-56.
Qian, L. and Gero, J. S. 1996. Function-behavior-structure paths and their role in analogy-based design. *AI EDAM* 10(04): 289-312.
Goel, A. K. 1997. Design, analogy, and creativity. *IEEE Expert Intelligent Systems & Their Applications* 12(3): 62-70.
Brown, D. 2003. Functional, Behavioral and Structural Features. *Design Engineering Technical Conferences, Chicago*, Illinois, USA, ASME.