

Rich Knowledge Parametric Tools for Concrete Masonry Design Automation of Preliminary Structural Analysis, Detailing and Specifications

Andres Cavieres¹, Russell Gentry² and Tristan Al-Haddad³

¹Georgia Institute of Technology, College of Architecture, Atlanta, Georgia, United States, andres.cavieres@gatech.edu

²Georgia Institute of Technology, College of Architecture, Atlanta, Georgia, United States, T.Gentry@coa.gatech.edu

³Georgia Institute of Technology, College of Architecture, Atlanta, Georgia, United States, tristan.alhaddad@coa.gatech.edu

Abstract

Our research focuses on how new computational tools and design methods can support the design process of masonry buildings. Specifically we discuss the potential of parametric modeling technologies to promote innovation by embedding knowledge on masonry construction as both generative rules and checker functions. Their goal is to inform designers on the feasibility of their intents on early stages of the design process. For this purpose we will adopt a methodology and notation called Building Object Behavior (BOB) to identify and translate construction knowledge towards the implementation of masonry parametric objects. A curved masonry wall is used as case study for the development of several parametric prototypes and their possible implications for collaboration are discussed.

Keywords: BIM, Knowledge-Based Design; Parametric Modeling, Masonry Building Design

Some Difficulties in Masonry Design

Research in masonry construction have promoted important innovations on new masonry unit types, structural analysis methods and more efficient construction processes for years (Beall 2000; Beall & Jaffe 2003; Ramamurthy & Kunhanandan 2004). However there is still a perception that the limits of masonry design are not being challenged by the architects. Current technical innovations are not being extensively transferred into architectural design practice and in most cases new masonry buildings continue to adopt conventional and rather conservative solutions.

There are several causes for this problem but two of them seem to be the key limitations that are currently threatening the competitiveness of this material. The first relates with an increasing number of misconceptions and prejudices among architects regarding the high cost of masonry construction and its limited formal possibilities. Such misconceptions are mainly due the lack of knowledge, especially among young architects about different masonry types and their architectural possibilities.

It can be argued that challenging masonry limits requires a considerable level of expertise, which is in contradiction with the lack of knowledge. Furthermore, expertise on masonry design is, as in any other type of domain, rare and expensive. Construction of complex or unusual configurations is naturally considered risky and complicated without expert knowledge and judgment. To avoid this problem the easy choice is the adoption of conventional solutions or simple variations of well known typologies that reduce risks and uncertainties. In this manner possible difficulties are kept under control by adopting prescribed design formulas and specifications.

A second relevant reason besides the above mentioned lack of technical knowledge is the reduced number of design tools currently available to represent and explore new masonry configurations in a more efficient and productive way. In the absence of such tools what becomes costly is the amount of effort architects have to put on modeling and detailing a building that could have hundreds or probably thousands of masonry units.

Our research aims to address this issue focusing on the development of rich-knowledge parametric modeling tools under a Building Information Modeling (BIM) framework. For that purpose we explore some methodologies and examples developed for the structural steel and pre-cast concrete industries to

capture and embed their domain knowledge into parametric modeling tools. The final goal is to improve the design and construction processes by supporting the creation, testing and evaluation of a higher number of design alternatives from the beginning.

The Limitations of Current CAD Technologies

Masonry construction implies the placement and bonding of individual modular units into a continuum. The geometric characteristics of this continuum as well as the specific patterns adopted for the placement of units are part of a sequence of decisions made by architects and engineers while trying to solve a set of simultaneous issues. Structural stability, functionality, aesthetics and satisfactory performance are some of the fundamental requirements that any building must satisfy as part of a design problem. To provide an integrated solution is not easy, and whenever a solution seems to be reached the chances for exploring other valid alternatives are going to be reduced.

However, good solutions often emerge from exhaustive comparison of different models from a wider pool of alternatives. Unfortunately designers usually succumb to the temptation of believing that the first options could be good enough; or they simply stick to what they already know, avoiding in this manner further explorations.

This phenomenon of accepting the “already-known” or the quickest solution is known “design fixation” (Purcell & Gero 1996; Buelow 2007) and we argue that the tendency to follow it becomes particularly strong when the number of parts involved in a problem is high. A continuum assembled by several small units, i.e., a masonry wall, a cladding system, a tiled roof or a brick pavement arise as typical situations where the number of parts implies such number of interdependencies that any further exploration beyond what is commonly accepted is taken as an extra-effort. The fixation on conventional configurations is generally accepted as “default” and in some cases it is even justified under claiming of adherence to traditional constructive wisdom.

This over-simplification is evident on conventions architects adopt for representation of masonry elements in CAD systems. The complexity of a masonry wall is reduced by denoting it as a homogeneous volume, i.e., a rectangular block that does not describe the geometry of the composing units nor their bonding pattern. In terms of quantification of units, a simple calculation is made as function of the total area to be covered, the size of the chosen unit and the type of mortar required. In this manner CAD systems are not used to geometrically represent all the components nor consider special cases such as block cuts or custom units usually required in any masonry design.

If a designer intends to explore different types of building forms, unusual bonding patterns or special details, a complete representation of the masonry assembly would be the most appropriate way to get better visualization and understanding. In a conventional CAD system this requirement would force him / her to model all the units one by one, place them on their respective positions and set all the joints, angles and articulations “by hand”. Once the assembly is modeled any modification on design implies an excessively time consuming and error prone manual process that would easily discourage most attempts.

Knowledge Based Parametric Design

There are currently important software development and research being done to solve similar issues of geometric and information complexity. Their first goal is to take advantage of computing power to automate repetitive tasks such as production of specifications and detailing. These developments strongly suggest that traditional CAD methods that represent building assemblies by oversimplifying their geometry are going to be replaced by another one where all relevant parts will be explicitly modeled. The potential benefit of such type of representation lies on the intelligent behavior that parametric components can exhibit.

Indeed this intelligent, knowledge-based behavior is the most promising capability of parametric technologies in design. The reason is that parametric modeling provides mechanisms to embed domain expertise and design intent into a set of topological and geometric relationships, allowing not only automation of low-level tasks but most importantly, to assist and inform designers with relevant knowledge about the validity of current design alternatives (Eastman, Lee & Sacks 2003).

Design knowledge usually is expressed in terms of some system of rules that control the form. According to Robert Aish (Aish 2005) design is fundamentally about the creation of these rules by means of the definition of proper relationships between design components. Many of these relationships are geometric in

nature or can be expressed in geometric terms, containing both knowledge and intentionality that designers have regarding a given problem.

The relation between geometry and knowledge however is not specific of parametric models but property of any design representation. The distinction here lies on the fact that in a conventional representation, such as paper drawings or non parametric CAD models, the knowledge is normally tacit, hidden behind the geometry while the rules that define relationships only exist in the mind of the designer (Katz 2007). In a parametric model instead this order is inverted; the system of rules becomes explicitly and easily available while the geometry represents just an instance of it.

A practical consequence of this inversion is that the rules and constraints that control a geometric form can come from technical requirements of other domains, such as structural analysis, fabrication or construction management, which usually has a secondary or late participation in decision making. It is anticipated that bringing these requirements in the form of knowledge early on conceptual stages of the design process, more feasible and innovative solutions could be explored from the beginning. This is important because is at this level where most sensitive and costly decisions are made in building design (Eastman et al. 2008).

In this manner parametric design offers the possibility of generation of multiple variations and design alternatives that can be produced, optimized, tested and selected systematically according to various methods. Decision making then has the potential to become not only a more systematic and objective process, but also a much more collaborative process due the availability of explicit, readable and shared knowledge.

Upon that the ability to automatically access and manipulate geometric dependent data, making it interoperable and reusable is the key feature of parametric systems that allows the generation of rich building information. A fully detailed 3D model becomes then the base representation for building projects, working as a central data source for architects, engineers and contractors during the entire life cycle of the project (Lee, G., Sacks, R. & Eastman, C. 2005).

Building Object Behavior

According to Eastman one of the major challenges in the development of such rich-knowledge parametric systems is to find a general method to facilitate the translation of relevant expertise into a set of functional parametric objects. To solve this issue the Building Object Behavior (BOB) description method and notation was developed by his team to assist in this task during the specification of a parametric modeling system for the North American precast concrete industry (Sacks, R., Eastman, C. & Lee, G. 2003). Nevertheless BOB was developed not only focusing on the precast concrete industry, but rather though to be useful on other types of design problems and construction systems. In this manner the general and abstract nature of BOB methodology and notation allowed us to adopt it as basic reference for the definition of parametric behavior of masonry models.

Object behavior in a parametric modeling environment is seen as the ability of a building component or assembly to respond to an internal or external stimulus preserving the original design intent. This response occurs when the system is capable of taking automatic actions in order to “maintain the topological and geometrical consistency of the relationships within and between model objects”. In this manner objects have to be modeled not only as they look but most importantly, as semantic relationships within a specific domain (Sacks et al. 2003).

However a major issue for the implementation of domain specific BIM parametric solutions relates with the problem of how to specify and embed relevant design and engineering knowledge in a parametric modeling system. One of the main difficulties arises from the fact that much of this knowledge is tacit and very hard to be explicitly represented. The tacit nature of design knowledge usually produces different interpretations at the implementation level that may cause several implications downstream the process, such as ambiguity and idiosyncratic adaptation of models.

The fact that a same object behavior can be implemented in different ways raises the question about efficiency, re-usability and scalability of the solution. Therefore there was an urgent need for a method that could help to rapidly and abstractly capture and represent such behaviors prior any software implementation or modeling activity. This method should help to pre-tune and guide parametric object definition in a testing phase, in such a manner that ambiguity and unnecessary complexity could be reduced. According to Lee (Lee

2005) this approach was also found to be especially appropriate for relatively large and complex parametric modeling activities where effective collaboration is a critical issue.

BOB Specification

The building object behavior (BOB) notation was developed to help designers to define a set of parameters and the relationships that represents the main aspects of a building object and its expected behavior within a given domain. According to Lee it is essentially graphical shorthand for sharing the descriptions among different members of a collaborative setting.

In this notation each shape can be labeled as a real-world object with a unique identifier (e.g., room A, wall W1, a column and so on). The graphical conventions adopted by the notation are the same of engineering drawing; however the drawings do not intend to be an accurate representation of shapes. Only the topological structures are important since the goal is to capture and clearly communicate relevant relationships existing within or between objects. All the accuracy will be parametrically determined by design afterwards (Figure 1).

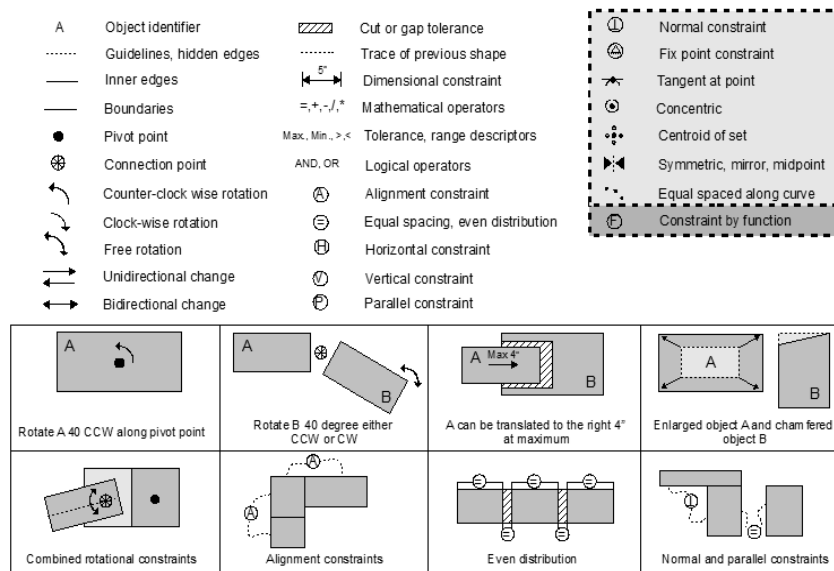


Figure 1: Set of primitive constraint types and extended set declarative geometric constraints. White rectangle set extracted from Lee et al.(2006). Clear grey extracted or based on Bettig (2003). The lower diagrams show examples of parametric behaviors declared using BOB notation as defined by Lee (2006).

In this research the implementation of knowledge of masonry design was done through the development of parametric prototypes using an existing parametric CAD system, called GenerativeComponentst^m |¹. Curved walls were adopted as case study to be defined parametrically using BOB, in such a way that solutions can be easily extended towards more conventional flat orthogonal walls. So far this process has followed three out of four base steps considered by BOB: 1) Elucidation, 2) Translation and 3) Implementation. The fourth point, validation is pending until a more extensive set of implement models are produced.

Elucidation: A curved concrete block masonry wall was chosen as case study for the definition of parametric behavior using BOB. The election of a curved geometry allowed the implementation of general solutions that can be easily adapted towards more common flat orthogonal walls. The first step describes the expected behavior of both a single concrete masonry unit (CMU) and an entire wall assembly according to

1 DigitalProject and Revit are being considered for future developments.

masonry construction guidelines. For this purpose we followed the technical specifications and design recommendations in a series of technical detailing documents provided by the National Concrete Masonry Association of North America (NCMA, 2002, 2003, 2004, 2005, 2006).

The elucidation process starts by identifying the most relevant relationships existing in a design problem. Such relationships do not need to be explicitly geometric but at least have to provide the basis for a geometric interpretation. Therefore the potential link between design knowledge and its parametric representation lies on the clear identification of the building components that might change and how they should do so.

Translation: At the most fundamental level the geometrical representation of domain knowledge operates as relationships between elemental components of the Euclidean geometry, that is, points, lines and surfaces. The set of primitive constraint types and constraint declarations showed above have to be applied to a simplified representation of the of a CMU unit, and after that to the entire curved wall assembly.

Figure 2 shows the sequence of geometric and declarative constraints that defines the behavior of a single unit. The unit itself is treated as a hierarchical assembly of points, lines and surfaces. Upon these basic elements higher level geometric entities are built.

In higher level the wall assembly is defined following a top-down modeling approach. That means that the overall skeleton or control rig is created before the propagation of the above specified CMU unit. The initial input is a surface that describes that wall geometry. This surface works as supporting element for the wall assembly and at this level the bonding patterns and the spacing between courses, headers and mortar thickness is defined.

For this purpose horizontal section curves are specified (controlled by horizontal constraint). The distance between curves should be the height of the CMU head plus a given mortar thickness (equal spacing constraint). Once the course curves are specified a two-dimensional array of points has to be defined by propagating points along each curve at constant spacing (equal spacing along curve).

The propagated points work as reference for the propagation of vertical segments (vertical constraint) which represent a two-dimensional array of masonry headings for a doubled-curved wall based on corbeling courses. Each pair of these vertical segments provides the basic four input points (two start points and two end points) for the propagation of rectangular polygons that work as place holder for the final insertion of CMU blocks (see previous section).

However that positioning of the place holders should not be even but rather follow a specified bonding pattern type. This requirement is accomplished by the specification of a higher level function that combines several geometric and declarative constraints within an algorithm. Therefore a special function based on iterations has to be defined at the implementation level in order to create such bonding pattern.

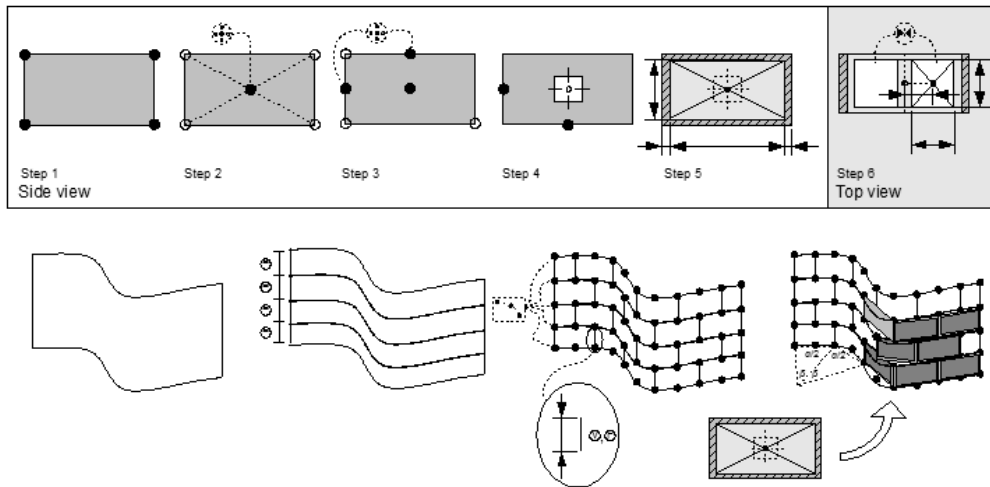


Figure 2 Sequence of aggregated parametric behaviors from CMU blocks to a block wall assembly using BOB notation.

Implementation: Once the basic rules regarding masonry bonding patterns and corbelling curved wall were established using BOB notation, the implementation of parametric building objects was grandly facilitated. However some detailed aspects of the translation had to be solved in combination with the implementation process.

For the running bond pattern a simple algorithm was defined and then implemented as an iteration to create the woven sequence. For the generation of rebar a similar function was created, but also allowing extra input for structural specification of spacing and diameters of each rebar according to load calculations (Figure 3). The generative nature of these scripts allows the designer to freely change the dimensions and shape of the supporting surface while keeping the regularity of the running bond pattern and the specified rebar spacing.

At this point the problem about the structural and constructive feasibility of a non- conventional geometry arose as a main issue. The limits of wall curvature and the inclination of the wall by corbelling consecutive courses were taken as exemplary cases for implementation of structural and construction rules.

Although these rules can be embedded a priori in the same way as generative rules, so that only valid forms can be created by 'enforcement', we considered it an important limitation for free design exploration. Therefore it was foreseen that verification a posteriori by adopting checking functions would solve in a better way both requirements of free exploration and constructive feasibility. The automatics enforcement would be avoided by relying on the fact that the system can inform were undesirable limits are being reached and by letting the design team to decide whether such limits are acceptable or not.

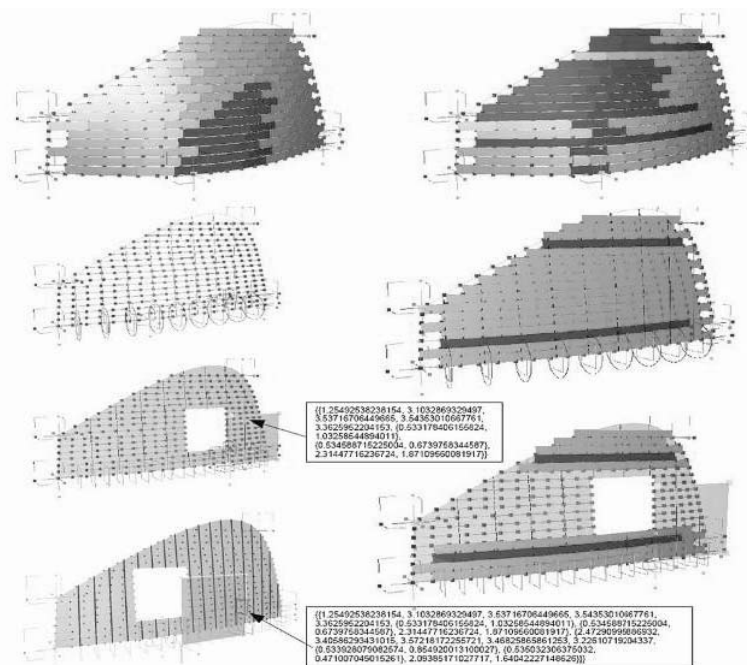


Figure 3: Examples of generative rules and checker functions. Corbelling and angles between adjacent units are calculated on demand by activation of special corbelling or angle checker functions. Masonry running bond pattern and vertical reinforcement are generated based on predefined patterns and automatic calculations of load and stresses.

Validation: Currently most of our research effort is being put on the three initial stages required for parametric definition of masonry objects. The validity of the prototypes done so far is currently being tested by the authors by including them as part of small scale design exercises. The goal is to have several working prototypes that include more design and construction rules from the NCMA technical specifications and guidelines. The second goal is to generate effective interfaces for collaboration with other specialists, specifically for structural analysis based on spreadsheets or customized interoperability with domain specific

analysis tools. Those interfaces are intended for design refinement and optimization, by taking advantage of commonly specified object's behaviors.

Once a comprehensive set of design rules and guidelines are implemented then more extensive design exercises will be done to test the effectiveness of the prototypes under a real-world design process work flow. The evaluation will look that the behavior displayed by parametric assemblies reflects the knowledge and design intent embedded on them at the beginning by producing the appropriate response and output.

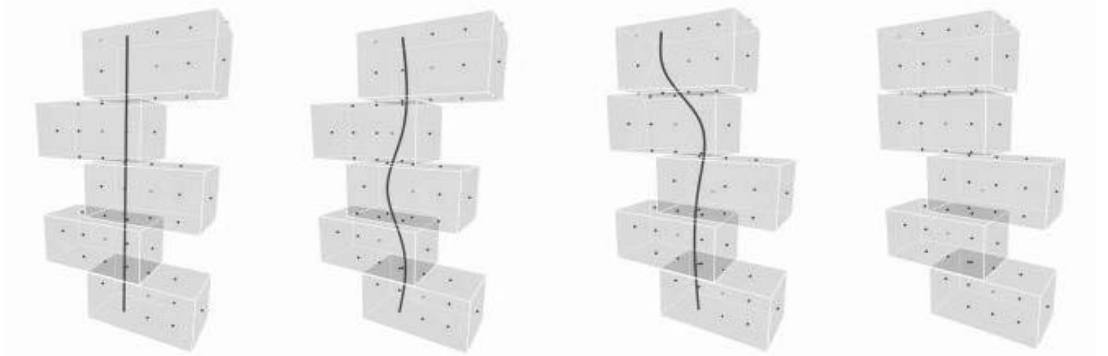
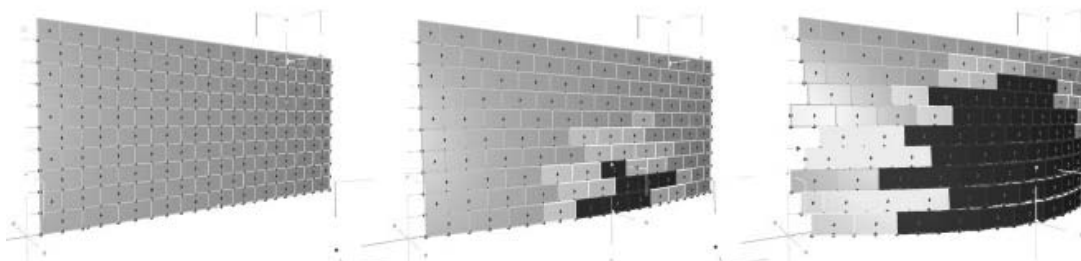


Figure 4: Example of checker function (in development) to determine maximum bending for vertical reinforcement according to wall curvature.



	column01	column02	column03	column04	column05	column06	column07	column08	column09	column10
course01	180	180	180	180	179	179	180	180	180	
course02	180	179	179	179	178	178	178	179	179	180
course03	179	179	178	177	177	177	178	178	179	
course04	179	178	178	177	176	175	176	177	178	179
course05	179	178	176	175	174	175	176	177	178	179
course06	179	178	177	175	174	173	174	176	177	178
course07	178	177	175	174	172	172	174	176	177	178
course08	179	177	176	174	172	170	172	174	176	177
course09	178	177	175	172	170	169	172	174	176	178
course10	179	177	176	173	170	168	169	172	175	177

	column01	column02	column03	column01	column02	column03	course01	column01	column02	column03	
ecc01	0.00	1.12	2.17	course01	286.25	286.25	286.25	course01	7.15625	7.15625	7.15625
ecc02	0.55	1.62	2.60	course02	326.25	326.25	326.25	course02	7.231367	5.40928	3.670329
ecc03	0.00	1.06	2.07	course03	366.25	366.25	366.25	course03	9.15625	5.045189	1.054585
ecc04	0.51	1.51	2.44	course04	406.25	406.25	406.25	course04	6.755317	0.049048	-6.36895
ecc05	0.00	0.95	1.88	course05	446.25	446.25	446.25	course05	11.15625	1.344295	-8.20291
ecc06	0.45	1.33	2.20	course06	486.25	486.25	486.25	course06	5.546491	-7.50564	-20.0501
ecc07	0.00	0.83	1.65	course07	526.25	526.25	526.25	course07	13.15625	-3.682	-20.1179
ecc08	0.38	1.14	1.91	course08	566.25	566.25	566.25	course08	3.749832	-16.8342	-36.7195
ecc09	0.00	0.69	1.40	course09	606.25	606.25	606.25	course09	15.15625	-9.73739	-34.1217

Figure 5: Example of function to check angles between adjacent CMU blocks in a curving wall. Grey block means angle values above 179 degrees, so that no cut is necessary according to guidelines. Yellow means that flanges of flanged blocks have to be removed. Red means cuts beyond flanges. A spreadsheet file of angle values is automatically generated (top), as well as spreadsheets containing eccentricities between adjacent rows (bottom left) and automatic load and stresses calculations for vertical reinforcement (bottom right). The hierarchical structure of the parametric assembly can be represented by a Directed Acyclic Graph (DAG) that manages all changes by automatically propagating updated values to dependent components.

Summary and Future Works

Parametric modeling is a core technology of Building Information Models (BIM) that enable the generation of rich building information. This technology promotes more accurate representations of building components and assemblies, facilitating the modeling of complex systems and automating tedious and error prone activities of detailing, scheduling and control of changes. Nevertheless the most important feature of parametric modeling is that it supports the incorporation of technical knowledge on early stages of the design process. In this manner we believe that these tools have the potential to improve design processes by increasing designer's understanding on the implications of his / her decisions and promoting the collaborative exploration of more innovative solutions.

Our research means a step towards such direction, by adapting the Building Object Behavior methodology and notation originally developed for the concrete precast industry into the domain of concrete masonry design.

In an initial stage we started the elucidation of knowledge based on available masonry design guidelines provided by the National Concrete Masonry Association. These guidelines contain both design standards as well as construction best practice. Thus initial samples of construction and structural knowledge were translated into parametric rules using BOB. Several prototypes were implemented including generative rules such as running bond pattern and rebar distribution while the verification of allowable angles between units and corbeling were included as checker functions to be applied a posteriori. This approach was considered useful to avoid the risk of over-constraining and interference to free exploration of alternatives. An important issue will be the development of criteria to define where certain type of rules should be implemented in a generative way or as checker functions, especially considering the implication that they can produce in a design work flow.

Further work has to focus on increasing the number of construction rules to be embedded in both CMU units as well as overall building assemblies (besides walls, masonry pavements, slabs and roofs can be considered).

While the partial prototypes developed so far worked well as independent models, extensive validation tests have to be done under the conditions of complete design processes. This challenge raises the problem of scalability of the model according to computer memory requirements.

Finally, consistent interoperability between design parametric models and engineering analysis tools have to be studied in order to streamline the feedback cycles between engineers and architects. In this scenario it is foreseen that BOB notation and methodology can be an important resource for the implementation of models oriented towards the optimization of many building life-cycle aspects, including fabrication, construction coordination, energy performance, and compliance with performance based building codes among others.

References

- [1] Aish, R. "Notes about Generative Components." SmartGeometry – 23rd Conference of the Education and Research in Computer Aided Architectural Design in Europe ECAADE. 2005. Lisbon, Portugal: Jose Duarte.
- [2] Beall, C. (2000). "New Masonry Products and Materials." Prog. Struct. Engng Mater.
- [3] Beall, C. & Jaffe, R. (2002). "Concrete and Masonry Databook". New York: McGraw Hill.
- [4] Bettig, B. & Shah, J. (2003). "Derivation of a Standard set of Geometric Constraints for Parametric Modeling and Data Exchange." Computer-Aided Design, 33, 17-33.
- [5] Buelow, P.V. (2007). "An Intelligent Genetic Design Tool (IGDT). Applied to the Exploration of Architectural Trussed Structural Systems, in Structural Engineering." Stuttgart, Germany. Universität Stuttgart, Institute of Lightweight Structures and Construction.
- [6] Ramamurthy, K., & Kunhanandan Nambiar E., K., (2004). "Accelerated Masonry Construction: Eeview and Future Prospects,." in Prog. Struct. Engng Mater.
- [7] Eastman, C., (2004). New Methods of Architecture and Building. Proceedings of the 24th Conference of the Association for Computer Aided Design in Architecture Acadia. Cambridge and Toronto, Ontario, Canada: S. Williamson, P. Beesley, N. Cheng.
- [8] Eastman, C., Teicholz, P., Sacks, R. & Liston, K. (2008). "BIM Handbook: A Guide to Building Modeling for Owners, Managers, Designers, Engineers and Contractors." Hoboken, NJ : Wiley.

- [9] Eastman, C., Lee G. & Sacks, R. (2003). "Development of a Knowledge-Rich CAD System for the North American Precast Concrete Industry." Proceeding of the 23rd Conference of the Association for Computer Aided Design in Architecture Acadia (pp. 16). Muncie, IN: K. Klinger.
- [10] Katz, N. (2007). "Parametric Modeling in AutoCAD." (http://aecbytes.com/viewpoint/2007/issue_32.html). AECbytes: Lachmi Khemlani.
- [11] Lee, G., Sacks, R. & Eastman, C. (2005). "Specifying Parametric Building Object Behavior (BOB) for a Building Information Modeling System." *Automation in Construction*, 15, 758 – 776.
- [12] Purcell T., & Gero J. (1996). "Design and Other Types of Fixation." *Design Studies*, 17, 363-383.
- [13] Sacks, R., Eastman, C. & Lee, G. (2003). "Parametric 3D Modeling in Building Construction with Examples from Precast Concrete." *Automation in Construction*, 13, 291– 312.
- [14] National Concrete Masonry Association. (2002). "Typical Sizes and Shapes of Concrete Masonry Units." TEK 2-1A Unit Properties. Herndon, VA.: Cemex.
- [15] National Concrete Masonry Association. (2005). "Grouting Concrete Masonry Walls." TEK 3-2A Construction. Herndon, VA: Cemex.
- [16] National Concrete Masonry Association. (2004). "Modular Layout of Concrete Masonry." TEK 5-12 Details. Herndon, VA.: Cemex.
- [17] National Concrete Masonry Association. (2006). "Concrete Masonry Radial Wall Details." TEK 5-10A Details. Herndon, VA: Cemex.