Roof Modeling Using Architectural Semantics Paradigm

by

Murali Paranandi
Austin Knowlton School of Architecture,
The Ohio State University,
Columbus, Ohio

Abstract

This paper presents an approach to developing the computer aided architectural design systems investigating architectural semantics paradigm and void modeling representation as a method. A prototypical system called FRED(Facile Roof Editor & Designer) was developed incorporating structural logic and characteristics of roof in its basic representation and its operational behavior constrained by distinct attributes of a roof. Design of Hip, Pitch, Multi-level, and Flat roofs in Solid and Shell forms was made possible by extracting from an existing building or creating them as independent entities. The implementation successfully demonstrates that incorporating architectural semantics in the basic representation of a CAD system allows architects to create and test roof morphology fairly quickly, accurately, and fluidity for ideation.

Introduction

With the boom of commercial 3-D CAD systems during the recent years, sophisticated solid and surface modeling technology is within the reach of architects. Most of the existing CAD systems available for use by architects have the ability to construct and visualize fairly complex 3-D models and have proved to in enhance productivity. Thus far, their use has been mainly limited to representation (of a finalized or refined idea), ignoring the equally or more essential ideation (finalizing or refining an idea). This may be an adequate service for architectural profession, but certainly not for architecture. This paper presents an approach to extending the capabilities of CAD systems to facilitate the process of ideation.

Typically advocates of CAD systems in architecture have been projecting the “productivity gains” as the key factor for justifying their use. We need to interpret the word productivity to include the creative component, beyond just the savings in cost and time. Presently, the design component is viewed as something better done on yellow trace with a soft pencil. Computers have been perceived as a suitable medium to represent the developed design solutions, i.e. to produce documentation for construction and client communication. It is important to understand the reasons for this tendency.

1.1. Can Computers be Creative Tools?

Typically, ideation involves visual thinking driven by a theme or a concept, and perception of various alternatives to choose from before finalizing an idea. Paul Laseau discusses how perception affects creativity and how graphics can play a vital role in the process [Laseau, 1975]. He also suggests that if image before us can become an integral part of our thinking, perceptions can be changed rapidly while retaining a grasp of the total problem. This process calls for a fluid way of producing the visual imagery as an extension to our thinking as we develop our perceptions. David Pye [Pye, 1964] clarifies perception and creativity and their critical relationship to each other: "... An inventor's power to invent depends on his ability to see analogies between results...".

The ability to produce and visualize a large variety of alternative solutions seems to be central to creative design process. The question is "Can computers play a role here?". Computers work on numerical models and are extremely fast in crunching numbers and displaying the corresponding changes in numbers in a graphic form. Designers, on the other hand, work on the basis of thematic models. It is not only unreasonable but undesirable to expect designers to change this over to a numerical model overnight. Current CAD systems are a testimony for the failure of this approach as they force the computers graphics logic and algorithm on designers thinking. The attempts to make the designers otherwise (especially using numerical model) have failed miserably! Given this only choice we are left with is to make the computers understand designers themes and actions. This requires development of a mechanism that translates human designers' actions into computer "crunchable" numbers. If we can develop such a mechanism, it appears that computers do have a definite potential to be creative tools.

1.2. Architectural Design semantics approach

Yessios [Yessios, 1986] proposes that a design oriented CAD system should be based on a yet to be fully developed theory of architectural modeling. It should facilitate and allow building models to behave like buildings when they are operated upon. It should also facilitate the incorporation of previous architectural experiences and design theories without restricting innovative paths of discovery. These capabilities should be fully integrated into the internal data structures rather than constructing external lists and procedures that may make it computationally expensive. Yessios group has done some work in this regard which has resulted in the development of void modeling techniques and an implementation of a prototypical system called ARCHMODOS (The Architectural Modeling, Design, & Drafting System of The Ohio State University) in mid 1980's. This work is followed by a number of subsequent graduate thesis projects as extensions this philosophy and applied void modeling techniques in such areas as Lay-Out Of Roads, Facilities Management, Multi-Storied Building Modeling, Design Of Architectural Space Through Void Modeling Representation [Spreacker, 1989], and Stair Modeler [Wong, 1993] etc.

The author takes inspiration from this approach and proposes the following model for a design oriented architectural modeling system as shown in Figure 1.

The proposed model recognizes that architectural design involves composition of various morphological elements such as walls, windows, doors, roofs, stairs etc. and that their structure and logic are unique and derive from their individual characteristics. This system has a robust 3-D graphics engine at its heart and several front end modules incorporating architectural domain specific logic and structure. The front end modules can feed the resultant geometry to the robust graphics system for geometric refinements and development. This information can then be used as a point of departure for the production end systems for various production needs.

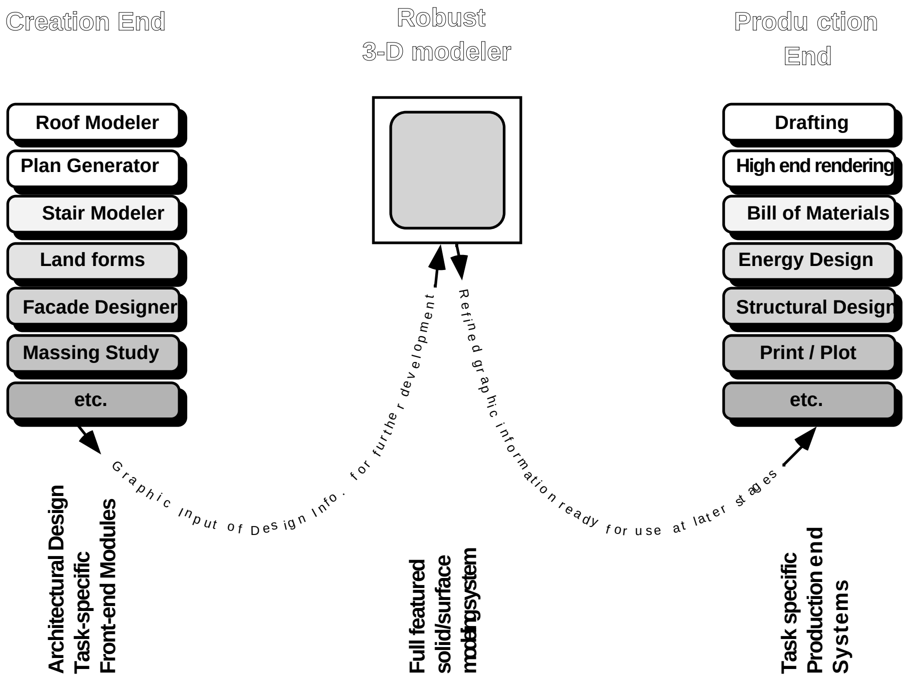

Figure 1. A proposal for an Architectural CAD System.

2. Roof Design as a Case study

The roof is a significant architectural element. Metaphorically, it symbolizes shelter which is one of the primary functions of architecture. Morphologically, it is one of the most dominating elements that defines both exterior form of the built form as an object as well as the interior enclosure as space.

2.1. Design of FRED (Facile Roof Editor & Designer)

A prototypical system called FRED is designed and implemented using architectural semantic paradigm. In this section issues and thinking that affected the design of FRED are discussed followed by a brief description of its features and functionality in the next section. The motivation for designing FRED derives from the need to assist architects in creating three dimensional roof morphology which can be used by engineers and roofers for further dimensional refinement such as designing roof trusses, framing, tiles, etc. Four types of pitched roofs as identified in Figure 3 are supported.

2.2. Issues to be addressed in Roof Modeling

First, let us start by observing some of the shortcomings of existing CAD systems, in the area of roof design:

Creation: Roof forms have unique structures that are difficult to express using the solid modeling primitives (such as cubes, prism, cylinders etc.), or surface modeling primitives (such as SPLINES, Meshes) supported by most available CAD systems.

Manipulation: Manipulation of the roof forms is often cumbersome using typical geometric editing operations. For example, once a roof form is modeled, moving a point will not guarantee the desired slope or thickness of the roof.

Interface: When designing roofs architects are concerned about positioning ridge lines, deciding the overhangs, and the slopes of the roofs, and would ideally like to grab the roof planes, ridge and eaves lines and tweak them around to study the functional, spatial, and morphological implications. Existing CAD systems logic based on Cartesian geometry, and computer graphics algorithms does not facilitate this.

Understanding the Structure of the Roof forms:

"Every three dimensional form is born from its plan as a tree born from its roots"

-Daniele Barbaro, Lapatrica Della Perspecttiva, 1569 [Hersey, 1992]

Roof forms have a strong connection to the floor plan. In most cases, the shape of the floor plan defines the plannimetric morphism (foot print) for the roof form. The interior spatial requirements define the volumetric morphism (section). Vandervort's "base map-sketch" metaphor for design of roofs [Vandervort, 1989] reinforces this idea. As a first step, he suggests making a "base map" of the building with associated information on a piece of graph paper. One of the ways he suggests doing this is to trace over the architect's floor plan. Next, he suggests experimenting with various ideas using this base map.

"...on your basemap, analyze.... place tracing paper over your map and sketch your ideal roof design. Then with a scale ruler, calculate what its actual dimensions will be...."

Influence of the floor plan: Certain design issues (such as deciding the covered areas, the shaping and sizing of the overhangs, positioning of skylights, courtyards, dormers etc.) are better addressed in the "plan mode".

Influence of the third dimension: Once the outline is in place, the issue of what areas are to be covered is decided. The important question of how... still remains to be addressed. Most designers would like to explore various alternatives at this stage to fulfill several formal, spatial, and functional requirements. This process needs to be fluid.

2.2. Choice of Representation Scheme for FRED

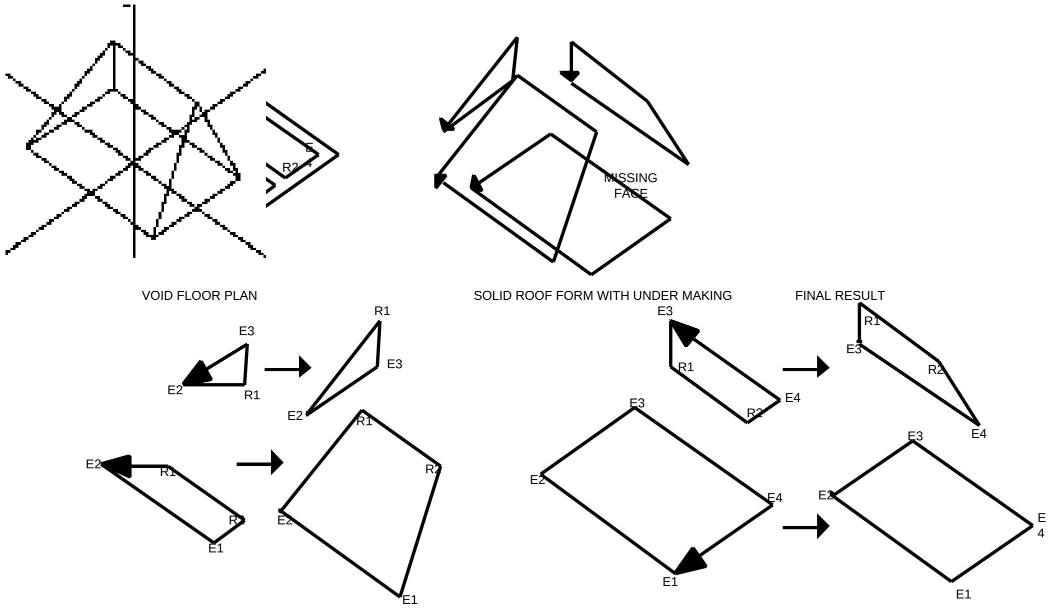

Figure 2. Illustration of FRED's internal representation of roof schema: The second row shows one to one correspondence between void (left) and solid structures (right) .

FRED uses void representation internally that accommodates the "base map-sketch" metaphor where the 3-D roof forms are controlled by a 2-D schema.

2.3. FRED's Constructs and Interface

As shown in Figure 3.0, the constructs of the roof forms derive from a vocabulary architects are familiar with. The creation and modification operations are based on typical sketching procedures architects use.

A combination of "interactivity and automation" is proposed and explored as an approach for architectural design oriented CAD systems to compliment strengths of the man and the machine.

2.4 Sketching Approach

Linton and Sutton [Linton, 1993] describe a sketch as a form of graphic shorthand, a rough drawing that represents major features of an object in space. Sketch conveys the essence of a visual idea, simplifies a complex subject and serves as a spontaneous and quick method of representation.Figure 4. Process of generation of roof forms in FRED .

FRED seeks to be a system that utilizes the built-in roof semantics and constructs. It bridges the gaps between various abstract states of roof design process depicted interactively by the user by sketching using minimal line input.

The following are a few of the design assumptions made for constraining the behavior of roof forms while generation and modification.

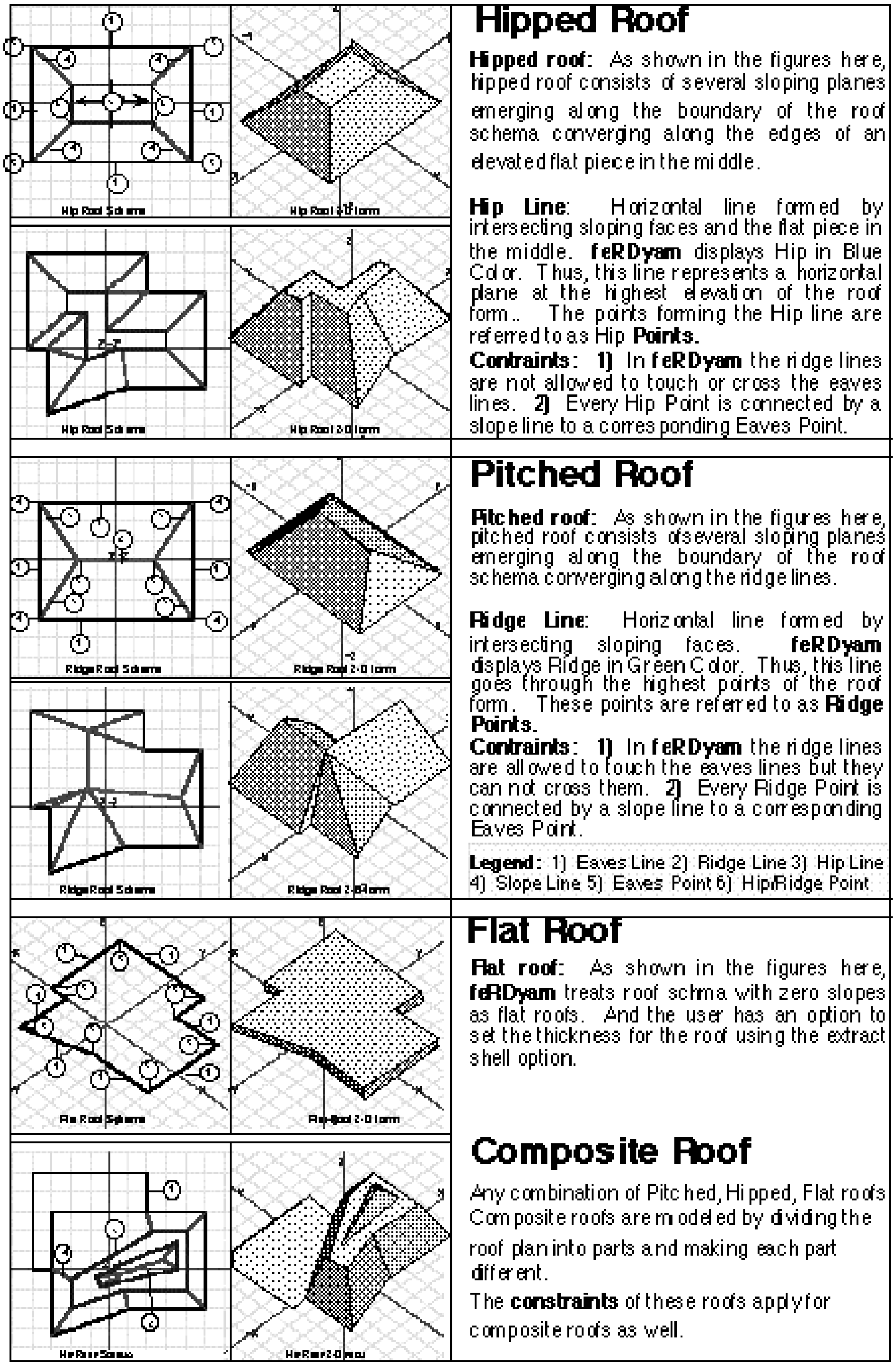

Figure 3. Types of roofs supported by FRED.

Roof Schema:

- There exists only one and only one Ridge line for a given floor plan;
- Slope lines and Pitch lines are bounded by the eaves lines;

• Slope lines consist of two and only two points; one of them is a Eaves point and the other has to be either Hip or Ridge point;

• None of the lines in the roof schema may cross over themselves or any other lines.

3-D Roof form:

• For each curve in the Void representation there exists a corresponding face in Solid representation with an associated slope information; and

• The shell roofs have uniform thickness.

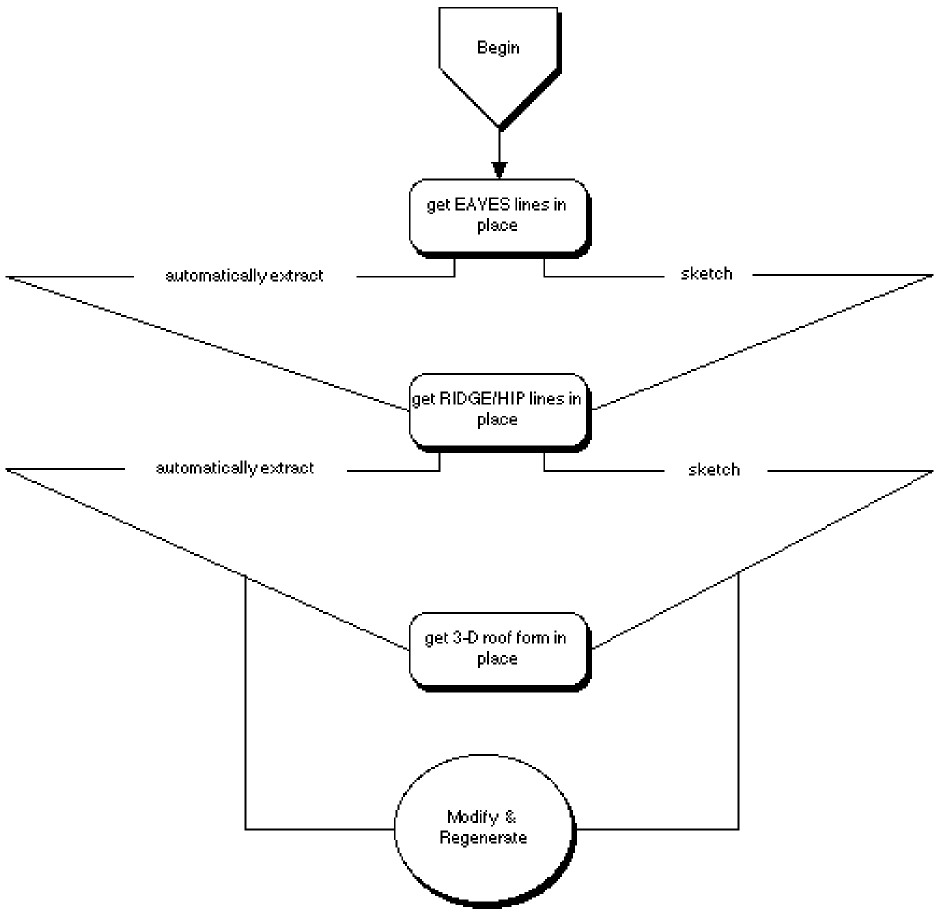

Figure 4. Process of generation of roof forms in FRED .

3. FRED's Features & Capabilities

FRED is designed as a stand-alone application that takes advantage of the existing features of Mac•Mod 844 and adds additional functionality for creation of roof forms. FRED allows the creation and manipulation of four types of roofs as identified in Figure 5 in solid or as shell form either as independent entities or from an existing floor plan.

3.1. Roof Modeling with FRED

Creation of the roof form involves the following steps.

1. Creation of Roof Schema.
2. Derive 3-D Roof form.
3. Modifications and creating variations

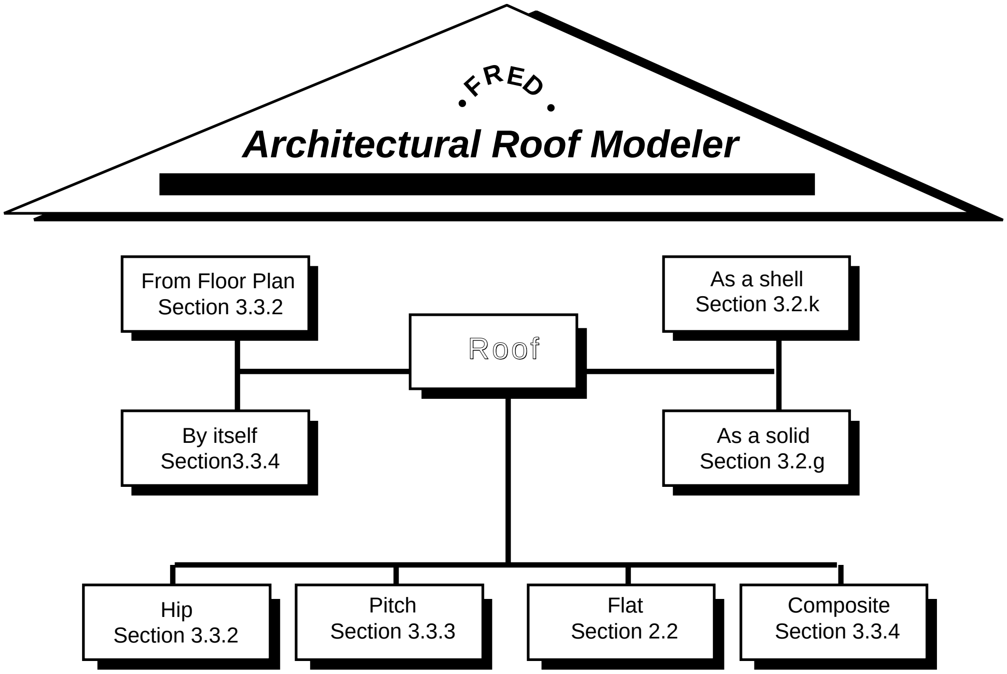

Figure 5. Primary capabilities of FRED.

3.1.a. Creation of Roof Schema

Creation of informationally complete roof schema involves creating the roof plan followed by assigning the heights information. Roof schema can be composed by exclusively sketching, or extracting or a combination of both.

As an Independent Entity: Using Roof Line tool (Figure 6.a), a user can sketch the roof plan. FRED recognizes and assigns proper attributes (which also decide the colors in which the lines are displayed) to individual line segments such as ridge, eaves, HIP, or slope lines even as the user sketches the roof plan using a vector line drawing tool. The user is free to edit (move or delete a point or a line using Figure 6.b or Figure 6.c) the roof schema at any stage during the process. Note that for this process a building floor plan is only optional. However, sketching in reference to a floor plan can help the user make more useful decisions.

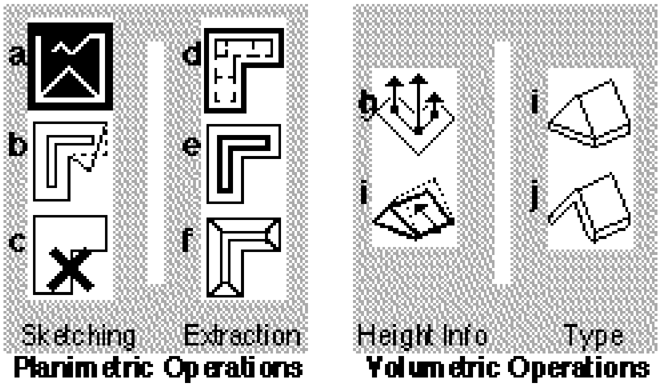

Figure 6. FRED tools.

From an Existing Floor Plan: FRED can trace an existing structured floor plan (created in Mac•Mod 844 using void representation), and derive an outline with a user specified offset to make a base map for the roof (using tools shown in Figure 6.d, and 6.e). The user can use this as the point of departure for the creation of a new roof form. From this point the user can either choose to sketch or use FRED automatic tools or a combination of both to complete the reminder of the roof plan.

3.1.a.1 Heights Information

By default, FRED interprets the height of the building models as the height for the EAVES lines and Hip line's height is set as half of floor height. Additionally, once the roof plan is in place, the user can 1) by assign heights at the curve, line, or point level (Figure 6.h), 2) assign the Slopes (Figure 6.i) for individual curves (planes).

3.1.b. Creation of 3-D roof form

Solid Roof: FRED's roof derivative tool (Figure 6.i) can automatically derive a 3-D solid roof form from an informationally complete roof schema.

Shell Roof: FRED's Shell Derivative tool (Figure 6.j) allows the user to derive a shell form from the solid form with a specified uniform thickness.

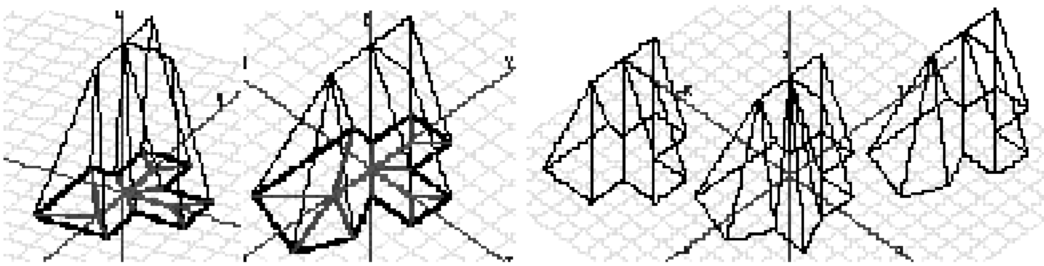

Figure 7. Modification of roof forms by editing the roof plan.

3.1.c. Modification Of Roof Forms

The user can create a roof plan and extrude into a 3-d solid form (boundary representation) . Although this solid form can be edited upon like any other solid, i.e. using Mac•Mod's geometric editing operations at all topological levels, it will not guarantee a proper structure of the roof. The following methods were devised to facilitate this process.

By Editing Roof Plan: Therefore, FRED allows the user to edit the roof plan and see its effect in the 3-D extrusion. This back and forth process can go on as long as the user desires to explore the possibilities. Basically, on any line in the roof plan points / segments can be inserted / deleted / moved. FRED allows saving/loading just the solid roof from information or roof schema information or both. This allows the user to save and retrieve various alternatives for comparative analysis.

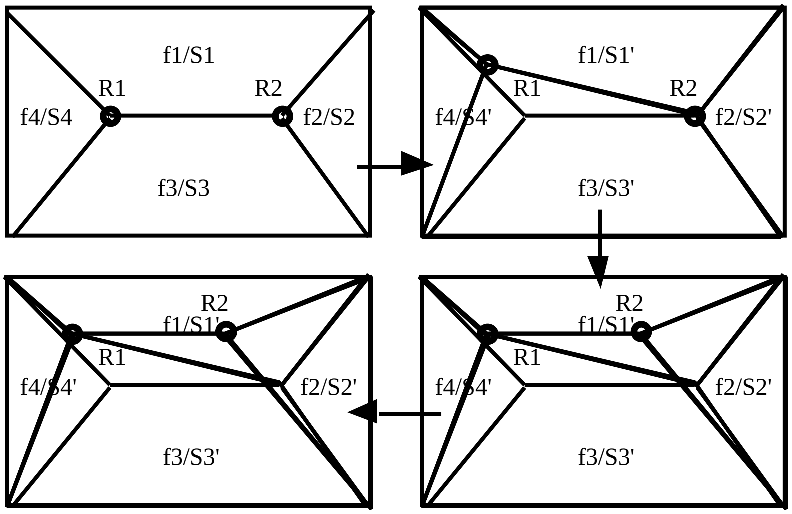

Figure 8. Deriving the ridge points by resolving the plane intersections(3 faces at a time).

By Slope Assignments: FRED maintains a slope value for each curve in the roof schema based on the heights information. The user can specify any desired value (current prototypical implementation allows values between 0° to 89°) for any given roof curve. FRED will then use the slope information to resolve the ridge points as shown in Figure 8. If a given RIDGE point is shared by more than 3 curves, the user is limited to specify slope values for only up to 3 curves (since more than 3 planes are not guaranteed to share a point). Also, when using this method to model roofs, FRED assumes that EAVES lines lie in the same horizontal plane. Once the slope assignments are done user can choose to generate the 3-D roof form by double clicking. From this point onwards, the user can modify slope values for any curve/s and regenerate the 3-D roof form to observe the changes.

3.1.c.1 Triangulation

Due to the nature of Roof Modeling operations, it is often possible to create a non-planar roof form. FRED provides an option to triangulate the faces to deal with situations like that.

3.1.d. FRED Output

The 3-D roof form created in FRED is meant to be imported into a robust solid modeler for further visualization and contextual refinements. Then the refined 3-D graphic data can be exported other task-specific analytical modules for such tasks as determining the roof framing, structural analysis etc. Currently the system uses the Mac•Mod file format. Translators such as DXF and 3DGF are under works for better connectivity with various commercial programs.

Chadwick [Chadwick, 1991] suggests that the engineering and the construction of the roof follows the determination of its form.

“The shape of the roof greatly influences the overall appearance of the house. Whatever shape is chosen from the different types that may be constructed, the main concern is to provide a roof that is functional, durable, and appealing to the eye.”

The interested reader is referred to Kubicek’s [Kubicek,1986] Opus 2x4 : A Wood Framing Construction Computer System, for how three dimensional graphic data such as the one produced in FRED can be processed to determine the framing sections, dimensions, and joinery. Also, Al-Rijleh’s [Al-Rijleh, 1987] Knowledge-Based Expert System For Design Of Roof Trusses is relevant in understanding how structural design can be done using FRED’s output.

3.2. Creating Roof forms Using FRED

Floor plans along with 3-D building models as generated in Mac•Mod 844 can be imported into FRED and can be used as a point of departure for creation of roof forms. Figures 9 and 10 illustrate the generation of a Hip roof from a floor plan. This involves

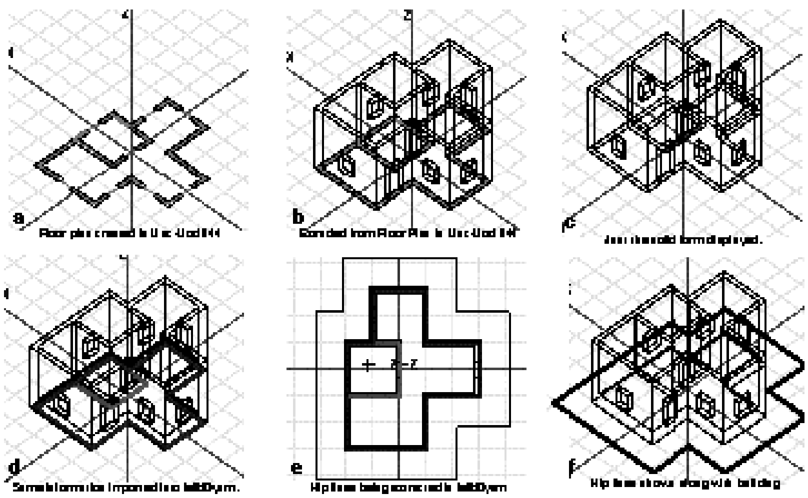

Figure 9. Building models are imported and used as a point of departure.

extracting Eaves lines interactively by eye balling it, or by specifying an offset distance (Figures 9.e, 9.f). Similarly, Hip lines can be extracted from the Eaves lines (Figures 10.H1, 10.H2). FRED can generate possible slope lines and display a preview for the user for examination (Figures 10.H3, 10.H4, 10.H5). The user has the option to 1) accept them as they are, or 2) use part of them and edit the rest; or 3) choose to sketch the slope lines interactively. Figure 9 shows a case where user chose to keep major portion of slope lines suggested by FRED and rejected some (Figure 10.H6). The user can assign the height information in a variety of ways using the Query tool (by assigning slopes at curve levels, and or assigning heights at point, line level). In Figure 10, the system defaults were used and a solid roof form was extruded (H7) followed by extraction of a shell roof form (H8).

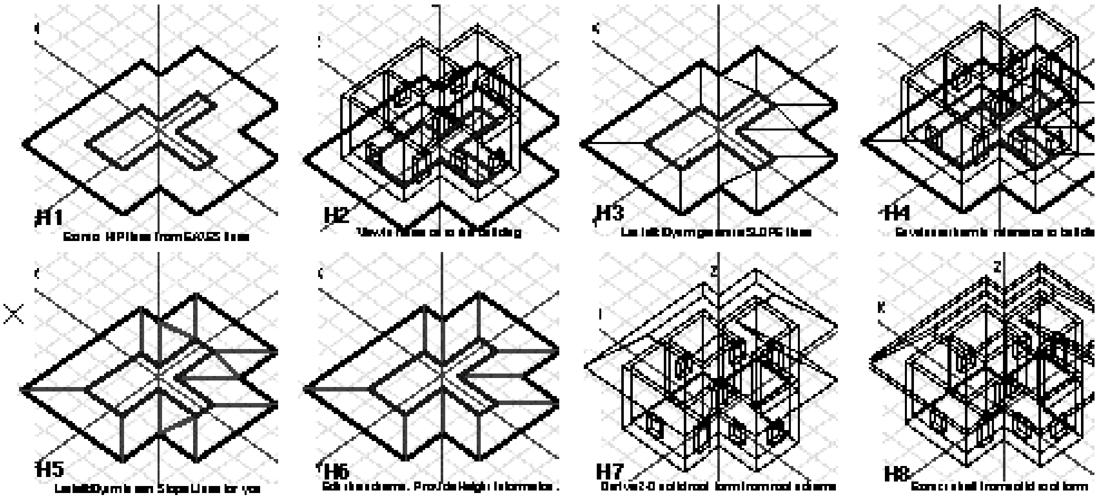

Figure 10. Creation of a HIP roof form in FRED.

Figure 11 shows generation of a Pitched roof form using a combination of procedures involving: automatic extraction (for EAVES lines: R1), sketching (for the Ridge lines: R2), controlled automation (letting FRED generate slope lines and editing them: R3-4), Height assignment (for RIDGE lines: R5), automatic roof extrusion following by automatic extraction of shell form.

Figure 11 shows generating a composite roof calling for more user participation to accommodate idiosyncratic explorations.

4. Extensions and Conclusions

FRED, the prototypical implementation presented in this paper, represents a partial implementation of architectural roof modeling methods. FRED aims at being a brainstorming tool for architects during ideation stage. As demonstrated by the examples in section 3, it makes the process of generating and modifying the roofs quick and fluid. Also, FRED uses a terminology that is familiar to architects and operations that are intuitive for the architect's thinking styles.

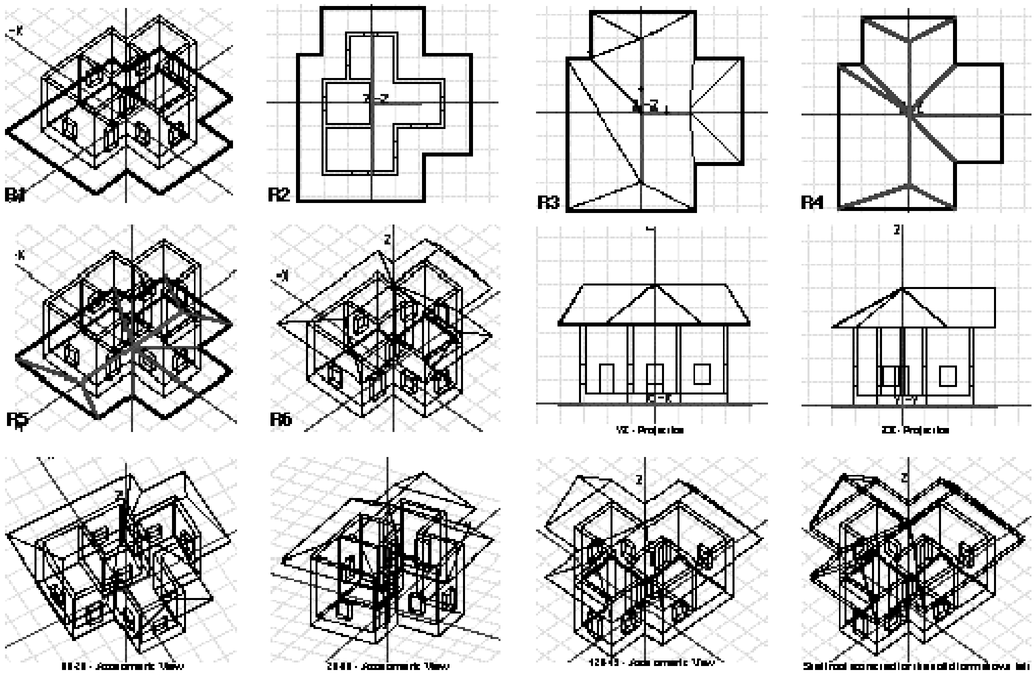

Figure 11. Generation of a Pitch Roof form in FRED.

4.1. Extensions

As a prototypical application, FRED is still far from a complete system for architectural roof modeling. Its purpose is to setup a direction for architectural modeling. Nevertheless, with few more additions, FRED can be an effective interactive brainstorming tool for the exploration of roof forms. This section discusses how enhancements could improve FRED in two important areas. The first, is enhancing the constraint mechanism. The second, is to provide additional primitives and features to work with.

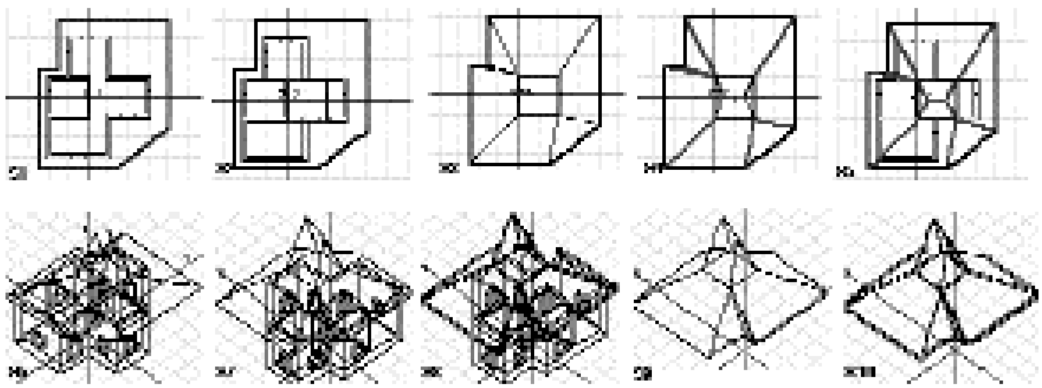

Figure 12. Process of creating a composite roof in FRED.
For clarity, fig. c, d show just the roof portion without the building.

4.1.a. Enhancements To Constraint Enforcement

Enforcing additional constraints to maintain certain desired properties of the roofs can make the design process more efficient. The following are possible areas for improvement.

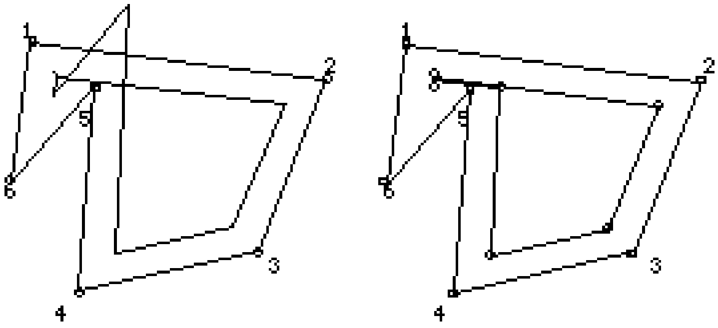

Figure 13. Deriving the Hip Line for concave floor plans.

4.1.a.1 Deriving HIP / RIDGE lines:

Currently, while deriving HIP lines, if FRED detects that HIP lines cross over any other lines, it flags an error and prompts the user to resolve the situation. Instead, it is possible to make the system assist further by deriving a combination of RIDGE line and HIP line using subdivision methods. An example of such a scenario can be seen in Figure 13.. Existing algorithms need to be expanded to handle this.

4.1.a.2 Check For Crossing Faces

It is possible that the faces cross over one another while adjusting the ridge points for slopes. Again, the existing algorithms can be refined to check for this condition and resolve it in an efficient manner.

4.1.a.3 Junction Of Walls With The Roofs

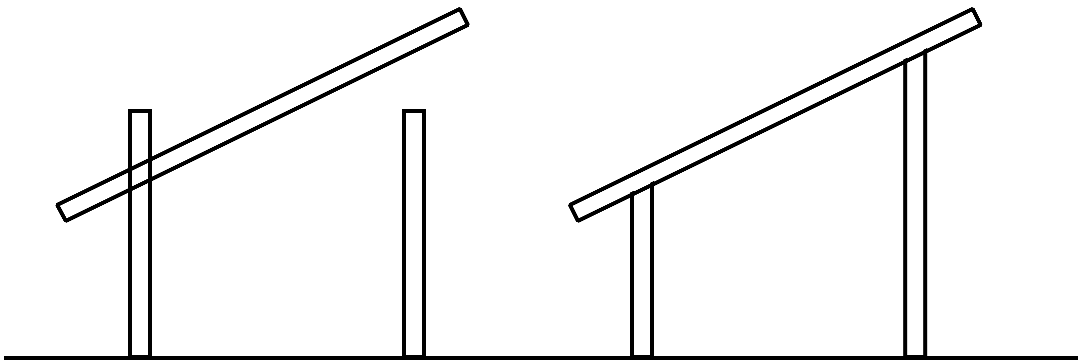

Figure 14. current implementation (left), future extension (right).

Currently, a FRED generated roof intersects with the building walls. The wall extrusion procedures can be modified such that they do not intersect with the roofs at the junction. Tobin's work in Constraint-based modeling [Tobin, 1991] is relevant in solving this problem.

Alternatively, this can also be accomplished as a post process by automating the extension of vertical segments of the walls to flush with the soffit of the roof. This involves keeping track of the roof faces on the soffit, and the side faces of the building at the time of generation and extending the segments of the later to flush with the former.

4.1.a.4 Options in Slope Assignments:

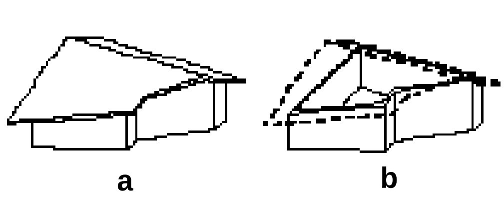

Figure 15.
(a, b) Ridge lines constraining the Eaves lines; walls adjusted to flush with roof soffit
(c, d) Eaves lines kept horizontal; walls remain as they are.

While assigning heights by slopes, the current implementation keeps the eaves lines horizontal, moving the ridge points in the vertical plane. This results in non-planar faces and the user needs to resort to triangulation. A future extension will include several options for deriving the point heights by slopes, one of which is shown in Figure 15.

As shown in Figure 13, the individual eaves points are adjusted based on the perpendicular distance from the ridge line to maintain the slope (planarity) for the face. This method needs to be accompanied with the constraint on the walls to flush with the roof soffit as discussed in the previous section.

4.1.b. Additional features

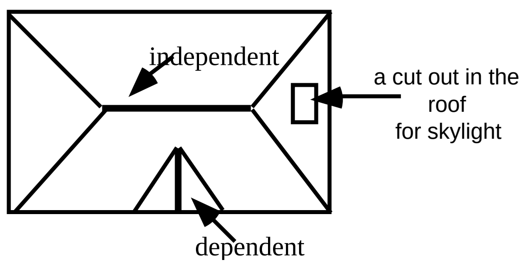

Figure 16 Dormer windows and skylights as primitives.

4.1.b.1 More roof components as Primitives

Certain primitives (like a dormer window, a cut-out for skylight, etc.) can be incorporated in to FRED . This can be easily accomplished by associating these primitives with a ridge line and an eaves line that will dictate the geometry of the primitives. The interface for these primitives can be similar to that of inserting doors, windows in Void modeling.

4.1.b.2 Options to facilitate Clerestories:

Existing algorithms to derive the solid roof from void roof plan can be modified, with very little effort, to generate an individual object per face. The user can assign slopes for each one of these planes and manipulate them in 3-D space one at a time. Figure 5.5 shows an example of this operation.

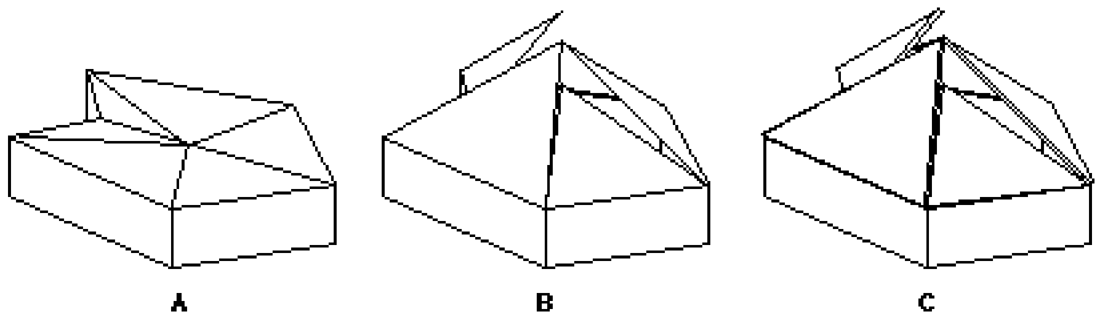

Figure 17. Option to treat each plane as a separate object to allow manipulation of each plane to crate clerestories.

4.2. Conclusions:

Major goals set forth for FRED were accomplished as follows:

- One of the primary objectives was to build a system that has built-in knowledge about the attributes and the relationships of the objects being drawn. FRED provides this ability by recognizing the relationships and assigning proper attributes of the roof structure to the vector lines drawn by the user.

- FRED provides increased productivity by automating such low-level decisions as drawing parallel lines, slope lines, and deriving the solid roof forms along with calculating and storing the slope information.

- Manipulating the slopes of individual faces of a 3-D roof form are provided an intuitive interface for architects. Also the rest of the roof structure is constrained to automatically match the slope.

- FRED takes advantage of the structure of roofs, in deriving the eaves lines, hip lines, and slope lines. All of these individual components are constrained to fit the behavior of the roofs. Also, maintaining the structure of the roof, allows interactive generation and makes manipulation of roofs easier. Plannimetric editing operations are constrained to maintain the structure of the roof and the derivation of the modified roof form was easily accomplished. Also, such features as sketching, graphic editing, combined with query point, query segment, query slope makes the manipulation process fluid.

The main idea tested in designing and implementing FRED is the incorporation of structural logic and characteristics of roof in the representation and operations. This successfully enriched the process of designing the roofs. This proves the validity of Yessios architectural semantic paradigm and the power of void modeling to solve architectural modeling problems for ideation.

References

[1] Al-Rijleh, Muhammad Maher. [1987]. "A Knowledge-Based Expert System For Design Of Roof Trusses". Masters Thesis, Department of Civil Engineering, The Ohio State University, Columbus, Ohio.

[2] Chadwick, Smith, Milner. [1991]. Roof Framing. American Association for Vocational Instructional Materials, 745 Gaines School Road, Athens, Georgia, 30605, U.S.A.

[3] Hersey, Fredman. [1992]. Possible Palladian Villas. The M.I.T. Press. U.S.A.

[4] Kubicek, Frank L. [1986] . "Opus 2x4 : A Wood Framing Construction Computer System". Masters Thesis, Graduate Program in Computer Aided Architectural Design, The Ohio State University, Columbus, Ohio.

[5] Laseau, Paul. [1975]. Graphic Problem Solving for Architects & Builders. CBI Publishing Company Inc., Boston.

[6] Linton, Harold and Sutton, Scott. [1993]. Sketching the Concept: Perspective Illustration for Architects, Designers and Artists. Design Press, New York.

[7] Pye, David. [1964]. The Nature of Design. Reinhold Publishing Corp, New York.

[8] Spreacker, Thomas B. [1989], "Design of Architectural Space through Void Modeling Representations", Master's Thesis, Graduate Program in Computer Aided Architectural Design, The Ohio State University, Columbus, Ohio.

[9] Tobin, Kennith. [1991], "Interactive Constraint-based Solid Modeling As a Design Tool", Master's Thesis, Graduate Program in Computer Aided Architectural Design, The Ohio State University, Columbus, Ohio.

[10] Vandervort Don. [1989]. Patio Roofs & Gazebos. Edited by Feldman, Fran. Lane Publishing Company, Menlo Park, California, U.S.A.

[11] Wong. [1993]. "Interactive Stair Modeler as a Design Tool", Master's Thesis, Graduate Program in Computer Aided Architectural Design, The Ohio State University, Columbus, Ohio.

[12] Yessios, C.I. [1986-a], "The Computability of Void Architectural Modeling", The Computability of Design, Proceedings of Symposium on Computer-Aided Design at SUNY, Buffalo.

[13] Yessios, C.I.[1986-b], "Architectural Modeling", Architecture 844 Lecture Notes-I, Graduate Program in Computer Aided Architectural Design, Department of Architecture, The Ohio State University.

[14] Yessios, C.I.[1986], "What has Yet to be CAD", ACADIA Proceedings.