EXPLORING THE PROPORTIONS OF MIDDLE-BYZANTINE CHURCHES: A PARAMETRIC APPROACH

Iakovos Potamianos
James Turner
Wassim Jabi

College of Architecture and Urban Planning
The University of Michigan
2000 Bonisteel Blvd.
Ann Arbor, MI 48109-2069
U.S.A.

Abstract

This paper examines two theories regarding the design principles of Byzantine churches through the use of 3D computer models produced by a programming language that allows the manipulation of the models parametrically to derive several instantiations by varying key dimensions. This geometry-based programming language, which is part of a larger solids modeling program, proved to be an excellent tool for determining the scope and the limiting cases of each of the two theories and the degree of their interrelationship.

Keywords: Parametric Solids Modeling, CSG, Byzantine Churches.

1. Introduction

The purpose of this paper is to critically examine two theories regarding the design principles of Byzantine churches through the construction of a parametrically-defined three-dimensional computer model. The first theory, advanced by Marinos Kalligas, concerns the development of the size and proportions of architectural elements according to liturgical requirements regarding the proper succession of images that are perceived by a person entering the church. The second theory, advanced by Nikos Moutsopoulos, concerns a system of proportions that applies to a large number of Middle-Byzantine churches. According to Moutsopoulos, this system of proportions may be safely employed for the restoration of ruined churches from which only a few key architectural elements survive.

To test these theories, we derived the algebraic relationships that describe the system of proportions advanced by each of them. We then tested the intersecting cases wherein both theories hold true. Using the resulting mathematical formulas we defined a three-dimensional computer model of a Byzantine church that parametrically changes in size and proportion by varying a few key dimensions. The model was implemented using a solids modeling language developed at the College of Architecture and Urban Planning at the University of Michigan. Since this project serves as an early example of its application, the programming language is presented alongside the test case.

2. Solids Modeling

Of all the solids modeling tools in use today for architectural applications, the most powerful and useful are the regularized geometric set operations, often called the Boolean operators. Although surface, wire frame, sweep, and primitive instance modeling remain central to most 3D modeling programs (DesignCAD, DataCAD, Micro Station), the ease of use and natural analog of cutting, gluing, and interference provided by the set operations have helped promote their popularity in programs such as AutoCAD and form•Z.

Most solids modeling programs allow one to repetitively apply union, intersection and difference operators to instances of solid primitives or user-generated, B-rep solids to create a target shape: that is, complex compositions can be created through successive operations applied to pairs of transformed shapes. Two unfortunate drawbacks to this process exist. First, it is usually impossible, or at least awkward, to back up a step or two if a mistake has been made. It may only be possible to correct an error through further applications of the set operations (patch a hole, cut off a column, etc.) AutoCAD and form•Z both provide robust “undo” capabilities, but given the fact that solid model building is normally a hierarchical process usually resulting in many modeling branches, backing through the operations may be neither efficient nor sufficient. Second, the procedure provides no mechanism for fine tuning a shape; it is impossible to modify the original parameters, such as coordinate transformation values, to generate alternative shapes.

```
begin part
      $cube          base   0.0 0.0 0.0  1.0 3.0 2.0
      $cube          union  1.0 0.0 0.0  3.0 3.0 1.0
      $cylinder(24)  A-B    2.5 1.5 0.0  2.0 2.0 1.0
end
```

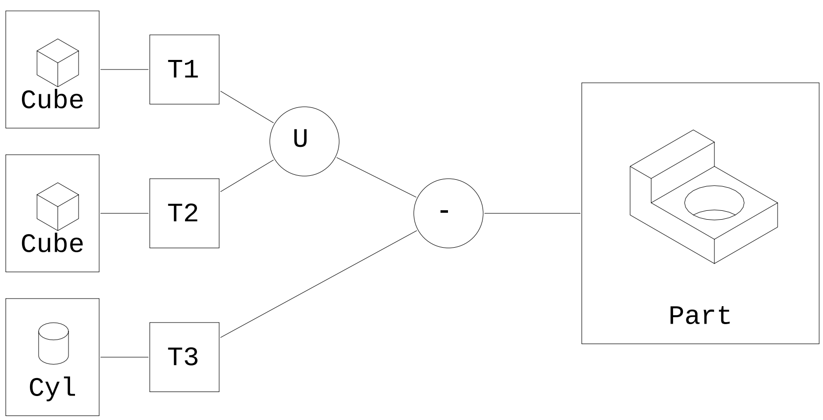

Figure 1. CSG example.

2.1 CSG Modeling

The introduction and wide-spread use of the concept of a constructive solid geometry (CSG) in the early 1980s (Requicha 1980) relieved part of these problems. The CSG approach recognizes that the modeling of a geometry is orderly and hierarchical in nature, and can in most cases be programmed as a series of geometry-transformation-operation statements. Many commercial CAD/CAM systems for engineering and manufacturing (Aries, Autotrol, Cadam, Catia, Patran, Euclid, Bravo) provide a CSG geometry interpreter, and the international standards IGES and STEP include a CSG geometry type as a basic entity. In most cases a CSG script can be edited, either graphically or textually, allowing solid models to be developed in a step-wise fashion similar to computer programming (Figure 1).

2.2 Parametric Solids Modeling

An important addition to a CSG modeler is to allow parametric control of dimensional relationships. This capability permits the creation of variational geometry by allowing single variables or complex expressions to be used instead of fixed dimensions.

```
set thick  1.0
set width  3.0
set diam   2.0
set NPT    24
begin part
      $cube          base  0.0           0.0     0.0 thick width 2.0
      $cube          union thick         0.0     0.0 width width thick
      $cylinder(24)  A-B   thick+width/2 width/2 0.0 diam  diam  thick
end
```

Figure 2. CSG example with parameters.

```
set thick  0.5
set width  4.0
set diam   3.5
set NPT    36
begin part
      $cube          base  0.0           0.0     0.0 thick width 2.0
      $cube          union thick         0.0     0.0 width width thick
      $cylinder(24)  A-B   thick+width/2 width/2 0.0 diam  diam  thick
end
```

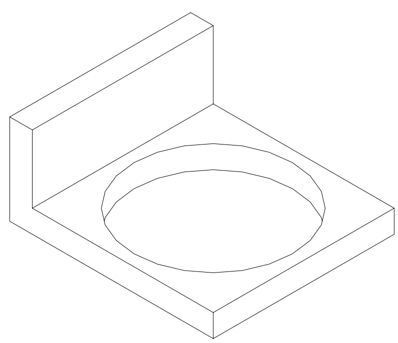

Figure 3. CSG example with different parameters.

Once a solid model is scripted through assignment of un-instantiated parameters instead of fixed values, its features may be adjusted by modifying associated expressions and values of variables.

2.3 Procedural Language for Solid and Polygonal Model Generation

As a subset of GEDIT (Turner 1993), a solids modeling program, and an extension to CSG and parametric modeling, we have developed a simple modeling language named GEDIT Programming Language (GPL) which is parametric, provides CSG capabilities, and is procedural. The language is C-like, allows only floating point data types, but does support complex numerical and logical expression evaluation with the usual mathematical functions such as sqrt, mod, sin, and cos. Conditional branching is supported through simple if, else statements, and may be extended with else if statements. Looping is available through for, while and do-while statements. I/O is supported through simple read and print statements.

In addition, GPL allows for geometric data types; geometric functions such as volume, area, and centroid; and coordinate transformations. Two-dimensional polygons and planar graphs can be created through move, draw and vector statements. Geometries (of the same type - either all polyhedral, or all polygonal) may be combined in a manner similar to a typical CSG language using a set operation statement which contains geometry name, operator, and transformation values. The geometries can be either user-defined (created through GEDIT commands) or primitives - cubeoid, cylinder, sphere, hemi-sphere, torus, cone, rectangle, circle, triangle. Furthermore, tessellation values used to generate non-planar surfaces can be controlled by the program. The language can be used to program math-based functions, but its strength is in the support of the creation of parametric and procedural geometries.

GPL maintains a single "current" geometry which can be operated on, measured, transformed, read, and stored; the set operations include the current geometry as one operand. The transformation and measurement statements also operate on the current geometry. Like PLASM (Paoluzzi 1991), which advertises itself as a "calculus on polyhedra," the purpose of GPL is to allow for an algorithmic definition of a single variational geometry. However, unlike GLIDE (Eastman 1977), which forced the designer-programmer to work through Euler operators at the level of faces, edges and vertices, GPL encapsulates its geometric data structures and allows one to easily manipulate and combine complete three-dimensional solid shapes.

GPL, PLASM and GLIDE assume that designers can program; that is, the designer can determine the steps necessary to create a shape, and then map those steps into the corresponding statements of the programming language. This assumption is somewhat more acceptable today than it was in 1977 when GLIDE was introduced to be "... easily used by a designer with minimal computer experience" (Eastman 1977, Eastman 1993); the architectural designer who is also a competent computer programmer is rare. GLIDE allowed its users to eventually work at parametric kit-of-parts level, but each shape needed to be constructed with low-level geometry and topology operators.

Instead of including a complete formal description of the language, a representative programming example for creating treads of a single stair run is given. Readers who are familiar with computer programming will notice the similarity to the C language.

3. Test Case: Exploring the Proportions of Middle-Byzantine Churches.

In order to put GPL to the test and explore the internal consistency and interaction of two theories regarding the proportions of Byzantine churches, we used GPL to define the procedural and parametric code for the construction of a three-dimensional solid model of a Byzantine church. The model definition was based on a set of algebraic formulas that we derived from the geometric constructions found in (Kalligas, 1946) and (Moutsopoulos, 1963). What follows is a discussion of the two theories, a description of our derived system of proportions, and concluding remarks.

```
read "Enter BOTTOM elevation:"  bottom
read "Enter TOP elevation:"  top
do
  if (top<=bottom)
    print "ERROR: top elevation less than bottom elevation."
  endif
while (top<=bottom)

set height top-bottom
#
# Riser height.
set TRUE 1
do
  do
    read "Enter RISER height (inch):" riser
    if ((riser*(1./12.)>height)
      print "ERROR: riser too large."
    endif
  while (riser*(1./12.)>height)
#
# Number of treads.
  set ntreads round(height/(riser*(1./12.)))-1
  print "Number of treads = " ntreads
  getok "OK?" ok
while (ok!=TRUE)
#
# Tread values.
read "Enter tread DEPTH (inch): " depth
read "Enter ACTUAL tread depth (inch): " actual
read "Enter tread WIDTH (inch): " width
read "Enter tread THICKNESS (inch): " thick
#
# Create each tread.
set riser  riser*(1./12.)
set width  width*(1./12.)
set thick  thick*(1./12.)
set depth  depth*(1./12.)
set actual actual*(1./12.)
set riser  height/ntreads
for I 1 ntreads-1 1
   $cube union depth*(I-1) 0. bottom+(I*riser) actual width thick
endfor
#
# Write solid geometry.
putgeo "Enter stair run name: "
print "DONE"
```

Figure 4. Sample Stair Procedure.

```
Enter BOTTOM elevation: 0
Enter TOP elevation: 9
Enter RISER height (inch): 8
Number of treads =       13.00
OK? ok
Enter tread DEPTH (inch): 9.5
Enter ACTUAL tread depth (inch): 12
Enter tread WIDTH (inch): 36
Enter tread THICKNESS (inch): 1.5
Enter stair run name: stair
DONE
```

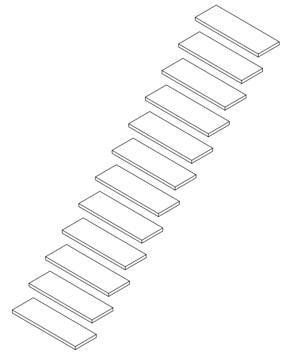

Figure 5. Sample program execution.

3.1 Presentation of the Two Theories.

Marinos Kalligas advanced a theory outlining the way in which the Byzantine church was designed in connection with its original function. Kalligas claims that the design of the church nave and narthex were directly related to their function. In the narthex, a space which precedes the nave, stayed the catechumens (those in the process of initiation) and those of the faithful that were sinners and by ecclesiastical verdict were not allowed to enter the nave. During the most sacred part of the liturgy the doors between the narthex and the nave were closed so that no sinner or uninitiated could participate in or hear the proceedings.

Kalligas claims that a number of architectural elements were designed in a coordinated fashion to visually enhance the distinction between narthex and nave but also to carefully place the most sacred iconographic representations according to their significance. As shown in figure 6, precisely at the moment that one reaches the threshold of the church the only thing one sees is the main apse which is outlined through the "beautiful gate," i.e. the gate between the narthex and the nave. Usually, the "beautiful gate" is completed upward by an arch concentric (if projected at the same plane) with the upper half-sphere of the apse. At the lower part of the apse, and before it, a low parapet or balustrade was originally located which was transformed later into the iconostasis or screen. Windows are located approximately in the middle of the apse and, a small distance above, in the hemispherical part of the apse, one usually sees the representation of Mary (Figure 7). This is significant because Mary usually represents the “mediator” or the “savior of the sinners” and as thus is the first image seen by the catechumens. What one sees from this point onward changes continuously as one moves from the entrance through the narthex. At the middle of the narthex one only sees the base of the dome, while on either side of the central apse two similar, but smaller apses open up. At this point, however, one cannot see the main feature: the apex of the dome, where Christ is represented. In order to be able to see the image of Christ one has to proceed further up to the "beautiful gate," exactly at the point where the nave proper begins and its entire width is revealed.

The successive images are tightly interrelated into a harmonious composition and interdependence. The image that first appears is like a preparation, an introduction to what is going to follow. It then develops and culminates in a brilliant climax at the moment the entire space is revealed from the apex of the dome to the entire width of the nave. In this movement from the entrance, through the narthex, and toward the center of the church, is based the total aesthetic effect of the Byzantine church. If one follows carefully the movement of the eye while entering, one discovers that the general direction of this movement defines a line that in the beginning is almost horizontal, and as it proceeds, it turns upward; the eye seeks to discover where the light comes from and for this reason it moves from the darker areas toward the brighter ones. The significance is in the catechumens' anticipation to see, to be illuminated in the real and the metaphorical sense. The catechumens are called "becoming illuminated" and in the prayers on behalf of the catechumens words that are derivatives of light appear time and again. When, finally, the catechumens are allowed to enter into the nave, one could imagine the kind of openness and relief of the soul proceeding toward the light (the dome being the brightest area of the nave). Now the catechumens would be able to raise their eyes toward the dome to see what they most yearned for.

The second theory by Nikos Moutsopoulos is based on observations of the relative proportions of the various architectural components of the church interior. The transverse section is, according to Moutsopoulos, the most important section for the shaping of the church, because it is mainly in this that the various heights within the church are determined:

a. The height of the four columns,
b. The height of the bases of the four semi-cylindrical barrel vaults (stemming from the center of the nave),
c. The height of the apex of the dome from the floor, etc.

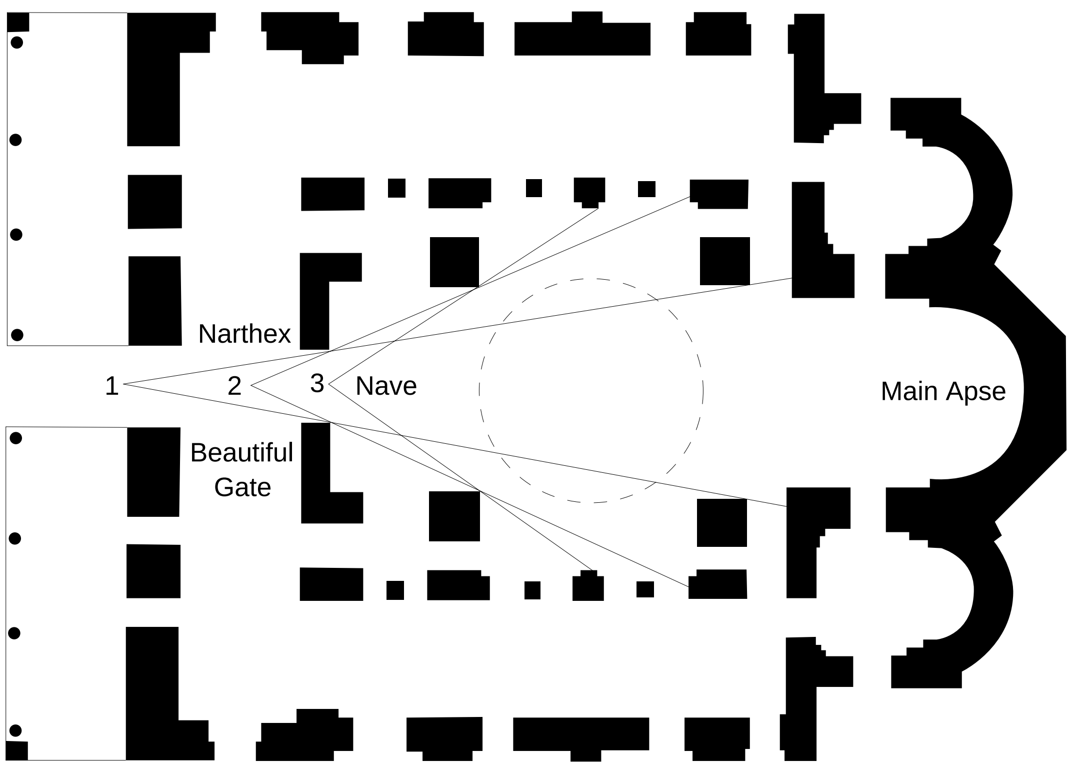

Figure 6. Plan of a Byzantine church showing lines of sight from (1) entrance, (2) narthex, and (3) nave. (According to Kalligas, 1948)

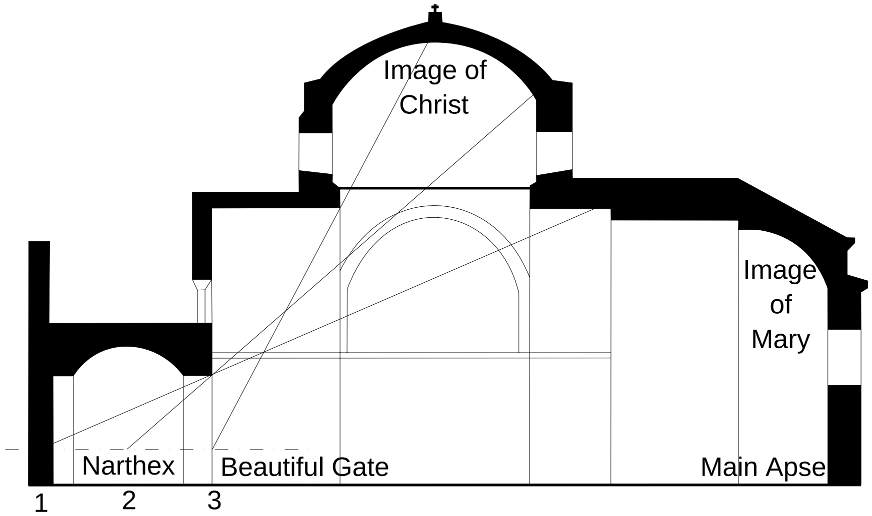

Figure 7. Longitudinal section of a Byzantine church showing lines of sight towards (1) the main apse, (2) the base of the dome and (3) the apex of the dome. (According to Kalligas, 1948)

In this transverse section (Figure 8) the most characteristic ratios are the ar/B and C/B, where ar is the height from the floor to the apex of the dome, C is the height from the floor to the apex of the barrel vaults and B is the narrow interior dimension of the nave. It was observed in the church of the monastery of Kaisariane (end of 11th century) that an isosceles triangle is inscribed in the interior with ab as a base (the narrow side B of the plan) and point 5 , at the apex of the dome, as the third point. The most important aspect of this observation is not the inscription of the isosceles triangle a5b, since three points always define a triangle, but that the legs of the isosceles triangle, that is, the sides 5a and 5b pass through the apex of the transverse barrel vaults (3 and p) and, indeed, touch the slanting decorative strip, which characterizes formally the beginning of the drum of the dome. It was also observed that from point x, the intersection of the diagonals of the rectangle abcd, passes a horizontal line that determines, with good approximation, the height of the four columns of the nave, a point that is located almost always at the midpoint of the apex of the barrel vaults from the floor and is, therefore, at C/2 from the floor. From the points 4 and 6 of the intersection of the diagonals of the basic rectangle abcd and the triangle a5b (4 is the point of intersection of a5 and db, 6 is the point of intersection of b5 and ca), passes a horizontal plane that defines either the bases of the barrel vaults or the decorative strip that characterizes them aesthetically.

The verticals that pass through the points 4 and 6 also pass either through the axis or through the edges of the column shafts. Another observation is that point 8, the intersection of the diagonals of the circumscribed great rectangle abqr that passes through the apex of the dome, is very close to point 7 of the intersection of aq and dt which is always located on the straight line ef. Finally, particularly in the example of Kaisariane, in the construction of smaller parts the inscription of rectangles is based on ratios that follow the rule of the "golden section". These observations have been verified for a great number of churches of the 11th and the 12th centuries. The above observations, however, in conjunction with the proportions of the plan (Figure 9), are of particular interest not only for the chronological classification of a monument of the type under discussion, but especially in a case of reconstruction or restoration of a church that has been destroyed but of which the following elements are known.

a. Plan outline;
b. The height and the position of one column; or
c. The height of the apex of a barrel vault.

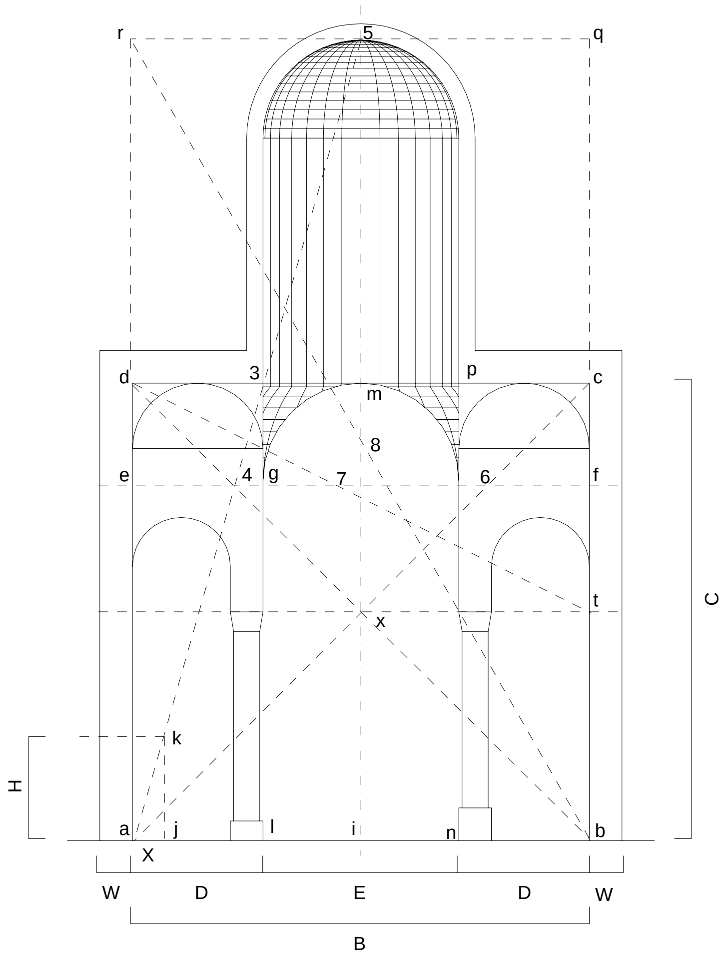

Figure 8. Transverse cross-section of a Byzantine church. (According to Moutsopoulos, 1963).

The reconstruction is easy when based on a geometrical construction that proceeds by connecting vertices and indicating line-intersections. The following measurements are known: (1) The width of the narrow side of the plan ab=B, (2) the distance an, that is, the distance of a from the projection n on the floor of the keystone p of the southern barrel vault (which is also defined by the footprint of the base of the column or pier), (3) ai, that is the distance from a to the midpoint of ab=B and finally (4) bt=C/2, i.e. the height of an extant column. The remainder is constructed as follows: we double the height of column bt to find the height C. We draw db and 3a that intersect at 4, which also lies on ef. We join t and d. The point of intersection of dt and ef is point 7. We draw a7 and extend it to meet the extension of bc at q and the vertical drawn from i at point 8 which when extended meets the horizontal drawn from q at 5 and thus determines the apex of the dome. Other helpful elements (ratios), for the reconstruction of a church of this type, are related to the construction of the similar triangles a5i and a3h. From the ratios C/D, C/B/2, etc. we can easily determine the apex of the dome 5.

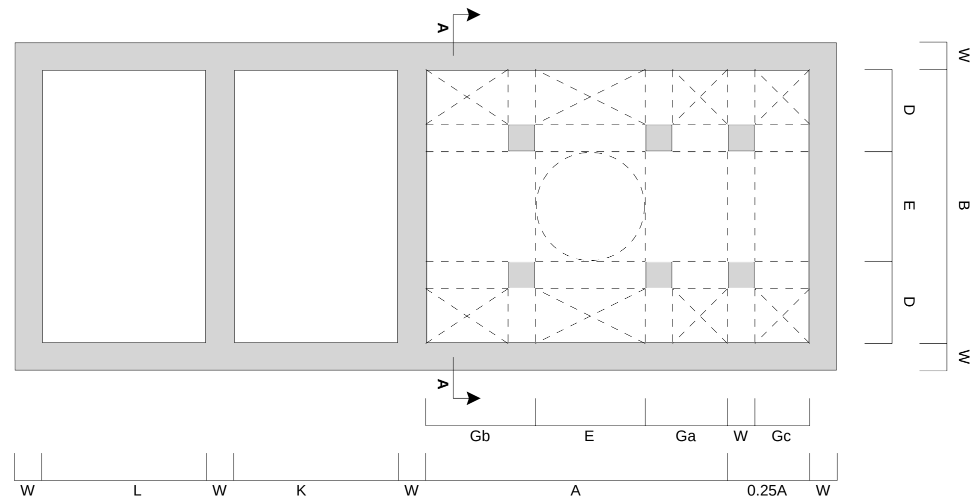

Figure 9. A plan of a Byzantine church. (According to Moutsopoulos, 1963).

3.2 Observations Based on the Analysis with the Aid of the Parametric Computer Model.

The reader may have noticed by the now the geometric nature of this problem. To encode this information in a procedural and algebraic set of relationships, we analyzed the afore-mentioned construction rules and derived the mathematical relationships that define the coordinates of key points. Space constraints prevent us from including the complete derivation process and the GPL program listing. Instead, we include a sample run and a view of the resulting three-dimensional model (Figure 10).

We have discovered that the greater the width of the church the wider the dome becomes as its apex becomes lower. In general, however, Moutsopoulos's theory yields a higher dome than usually found in the Byzantine periods.

Aside from the ability to vary the dimensions and discover the limiting cases beyond which Moutsopoulos's theory does not hold true, we discovered a relationship between the width of the side aisles and the interior width of the church. Assuming that the main vault of the church is semi-circular and B is equal to C (as drawn by Moutsopoulos), we found that a quadratic equation exists (Figure 11) that provides a solution for D in terms of B (the other being negative and thus impossible) that closely approximates the Hilandar monastery church.

Comparing the two theories it becomes clear that Moutsopoulos is concerned with the transverse section while Kalligas refers mainly to the longitudinal section. It is interesting to note, however, that both theories describe methods for determining the apex of the dome. Kalligas constructs a line, at eye height, from the "beautiful gate" towards the rim of the base of the dome while Moutsopoulos draws the line from the ground plane and in the transverse section. For both theories to hold true at the same time the dome should be nearer to the "beautiful gate" than it is to the sides. The precise relationship in order to view the apex of the dome (Figure 12, 13) in both cases is: Gb should be less than or equal to D(1-H/C) where:

Gb is the perpendicular distance from the edge of the dome in plan to the "beautiful gate" (see Figure 9).

D is the width of the side aisle (see Figure 8).

H is the height of the viewing eye (see Figure 8).

C is the distance from the floor to the apex of the cross vaults (see Figure 8).

```
GED> csg
Enter CSG file name: Church.csg

Creating "pc_temp"
Enter A: 20
Enter B: 18
Enter C or (0) if unknown: 0
Deriving C...
 Enter D: 2.5
 C=       53.30
Enter Eye Height: 5
I recommend GB <= D*(1-H/C)  2.27
Enter GB: 2.27
Enter column height: 26.65
Enter height to column ratio: 2
Enter Wall thickness: 1
Enter L: 16
Enter K: 16
Condition Satisfied
C =       53.30
(Z4+(0.5*B)-D) =       53.30

DONE
```

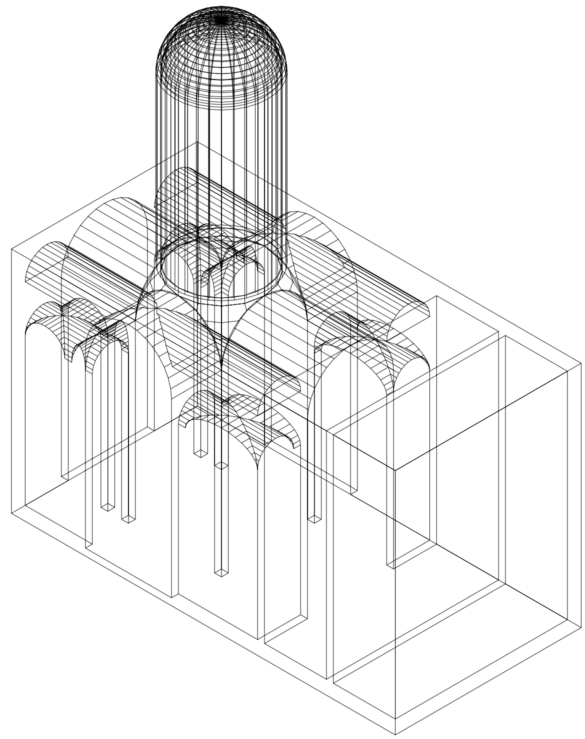

Figure 10. The GPL implementation of the parametric church.

$$ax^2 + bx + c = 0 \Rightarrow x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

$$D^2 + \left(\frac{3}{2}B\right)D - \frac{B^2}{2} = 0 \Rightarrow \begin{array}{l} D = -1.7808B \\ D = +0.2808B \end{array}$$

Figure 11. The quadratic equation defining the relationship between B and D.

4. Conclusion

GPL was created to teach CAD students about parametric modeling, and provide to the general CAD student a language by which one can create variational shapes. However, we found that parametric modeling system can greatly benefit research in the history of architecture, our effort representing only a small example toward this direction. But we believe that this tool offers unprecedented opportunities in design education and the design professions as well allowing expedient visualization of various alternatives by a simple manipulation of key design aspects. Parametric modeling can allow rapid exploration of design alternatives that are guaranteed to follow a pre-determined set of relationships similar to “What-If” analysis capabilities found in spreadsheet software.

Currently, we are concentrating on transforming the GPL programming language into a stand-alone compiler of three-dimensional parametric scene descriptions. In addition, the software is being ported to various hardware platforms. Future work will also include fine tuning the parametric model and testing it against real churches built during the Byzantine period.

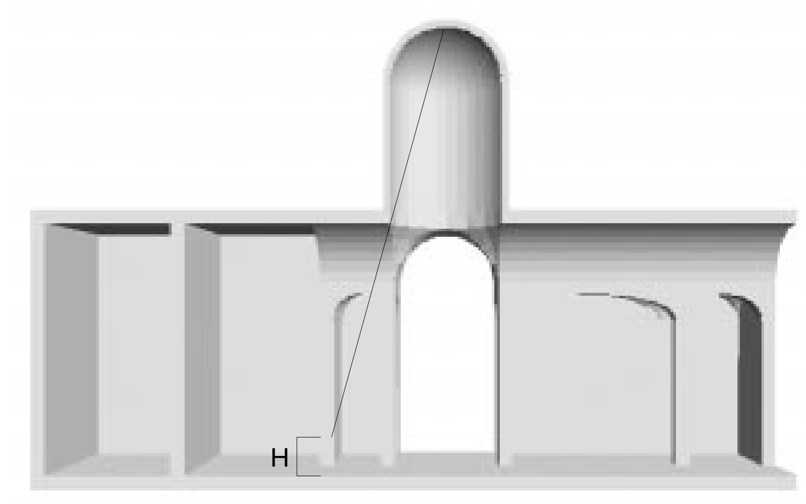

Figure 12. A longitudinal section of the parametric model indicating the line of sight from the entrance of the nave (at eye height) to the apex of the dome.

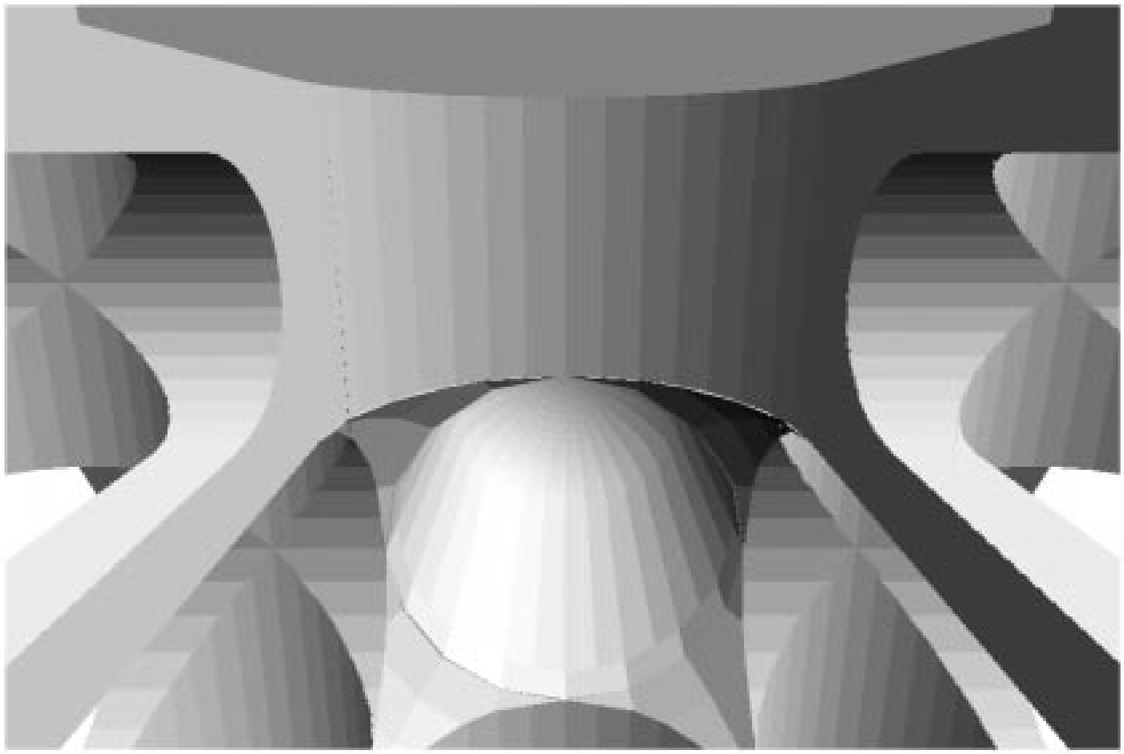

Figure 13. A perspective view towards the apex of the dome at the entrance to the nave.

References

Eastman, C.M. “Conceptual Modeling in Design,” in Fundamental Developments of Computer-Aided Geometric Modeling, L. Piegl (ed.), (San Diego: Academic Press, 1993).

Eastman, C.M. and M. Henrion. “GLIDE: a language for a design information system,” (ACM SIGGRAPH Proceedings, 1977).

Kalligas, M. E Aisthetiki tou Xorou tis Ellinikis Ekklisias sto Mesaiona, (Athens, Greece: 1946).

Moutsopoulos, N. Hronika Aisthtikes, (Thessaloniki, Greece: The University of Thessaloniki, 1963).

Paoluzzi, A., Sansoni, C. “Solid Modeling of Architectural Design with PLASM Language,” in CAAD Futures ‘93. (New York: North-Holland, 1991).

Requicha, A.A.G. 1980. “Representations for Rigid Solids: Theory, Methods and Systems,” ACM Computing Surveys 12, no. 4. (Baltimore: ACM, 1980); 437-464.

Turner, J.A., Hall, T.W., Borkin, H.J. “GEDIT User’s Manual,” (Ann Arbor, Michigan: The University of Michigan, 1993).

Iakovos Potamianos is a Ph.D. candidate (History and Theories of Architecture) in the Doctoral Program in Architecture at the University of Michigan.

James Turner is a Professor of Architecture at the College of Architecture and Urban Planning, The University of Michigan.

Wassim Jabi is a Ph.D. candidate (Computer-Aided Design) in the Doctoral Program in Architecture at the University of Michigan.