

Integrated BIM and Parametric Modeling: Course Samples with Multiple Methods and Multiple Phases

WEI YAN

Texas A&M University

INTRODUCTION AND PROBLEM STATEMENT

The course introduces Building Information Modeling (BIM) in architecture and BIM principles, methods, and applications in the design process and the building lifecycle. The course objectives include: (1) teaching basic and advanced BIM technologies to model and retrieve building information in the building lifecycle from design to facility management; and (2) enabling students to gain knowledge of architectural computing methods that they can apply in design studios, thesis work, research, and professional practice. The objectives of this course are pursued by means of theoretical lectures, readings, practical exercises that address the following components:

- Computer-Aided Design (CAD)
- Building Information Modeling
- Parametric modeling
- Information retrieval using databases, Web, and Application Programming Interface
- Applications in simulation, visualization, and construction scheduling

Each semester there are about 15 students in the programs of Masters of Science, Masters of Architecture, and Ph.D. in Architecture attending the class. The background of the students varies, but most of them have limited CAD experience, and few of them have 3D modeling experience and even fewer have computer programming experience.

There are two phases of the course:

1. BIM Basics - Modeling and Parametric
2. Design
3. BIM Advanced - Accessing BIM through
4. Web, Database, and Programming

Instruction goals include the understanding of implicit and explicit parametric modeling, understanding the integration of BIM and parametric modeling, experiencing API programming as a power method for BIM and parametric modeling.

BIM authoring tools normally employ parametric methods for structuring the building models and enable changes of one building component to be propagated to other building components. However, existing BIM curriculum and parametric modeling curriculum are separated, with BIM focusing on the documentation of building design and its applications in building construction and operation, while parametric modeling focusing on creative geometry modeling (using tools like Rhino). It is the intent of this course, that the integrating of the two powerful design and modeling methods will foster critical design thinking process, which are enhanced by the understanding of the following major advantages of learning and using the two modeling methods – 3C's (coined by the author):

1. Creativity (form generation)
2. Constructability (converting geometry
3. to building components)
4. Computability (of models)

We present three course samples in the following sections to address the 3C's collectively at the introduction level: Sample 1 creating parametric

shapes that may lead to creative forms; Sample 2 creating parametric solid models and building components for constructability; and Sample 3 retrieving BIM data and calculating for computability. The samples are focused on integrated modeling with BIM and parametric methods, whereas the other course components of databases, Web applications, simulation, visualization, and construction scheduling are not addressed in this paper.

We have chosen Autodesk Revit Architecture® as the modeling tool for its capabilities of both BIM and parametric modeling. Extensive Autodesk Revit tutorials are given in the first three weeks of each semester, followed by specifically designed homework and projects exercising methods introduced with the samples.

For Sample 1 and Sample 2, multiple methods are used to resolve the problems and the methods are compared for students to learn the advantages and limitations of each method. For Sample 3, multiple phases are introduced, with a new functionality added into each new phase. This way, students gradually learn the challenging computer programming method and understand the possibility to extend the power of the method step by step.

SAMPLE1: PARAMETRIC CURVE MODELING

The first sample of parametric geometry is a helix curve. This sample is inspired by *Elements of Parametric Design* (Woodbury, 2010), which is originally programmed in Bentley®'s *GenerativeComponent*®, a parametric geometry modeling tool. It is also inspired by Kron's (2010a) implementation of the same model using Autodesk Project Vasari® – a stripped down version of Revit for conceptual design.

We started from introducing parametric equations. The parametric equations of a helix curve are known as:

$$\begin{aligned}x &= a \cos (t) \\y &= a \sin (t) \\z &= bt\end{aligned}$$

(1)

We then introduced four difference methods: two implicit and two explicit methods, depending on

whether explicit parametric equations (1) are used to model the helix curve.

Method 1: Utilizing 3D geometry's surface

The first method uses a cylinder as an auxiliary shape for constructing the helix (Figure 1 Top-Left) (Light, 2010). Using this method, a designer can create the cylinder geometry first, divide the surface of the cylinder into a grid, and then trace the points on the grid to create the helix curve. This is a simple method that students can learn in minutes. The method requires that the designer has the knowledge about the relationship between a cylinder and a helix curve. The parameters of the helix curve are the radius and height of the cylinder. They are indirect parameters to the curve and the method is an implicit parametric method in which the designer doesn't need to know or manipulate the parametric coordinate equations of the helix curve explicitly. However, this method is limited in that knowledge about the relations of different geometries (a cylinder and a helix curve) is required and also only curves that can be traced from primitive geometry can be created using this method.

Method 2: Using point parameters

In Kron (2010a) (Figure 1 Top-Right), the author uses a similar method assuming that points on the helix moving along multiple circles at different levels. The parameters that drive a control point of the curve include: the distance between a control point to the center, the angle of the point-to-center line, and the height of the point. The designer needs to know that a helix curve can be created and controlled using these parameters. This method directly controls the point parameters but does not use the explicit parametric coordinate equations (1) either. Compared to Method 1, this method is more flexible, for example, a conic helix curve can be easily created using the radius parameter of the points. However, it requires the specific knowledge about the control points' relative location and orientation to the center point.

Method 3: Using point coordinates

Method 3 uses the explicit coordinate equations to create the control points. Control points are created using x , y , z coordinate equations (1) implemented in Revit Family Editor (Figure 1 Middle-Left). The designer needs to know the trigonometric function and

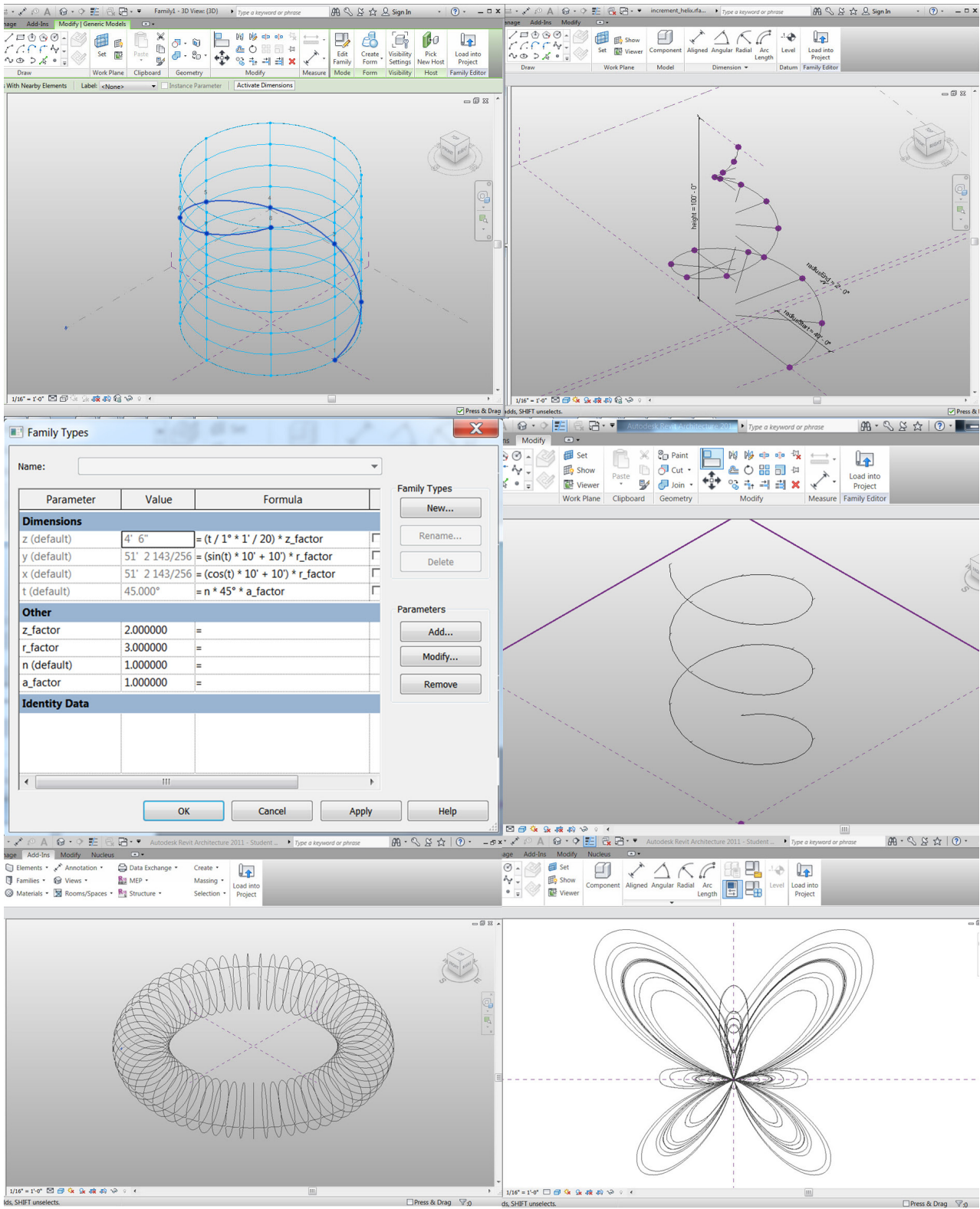


Figure 1. Top: Helix curves created using Method 1 (left) (Light, 2010) and Method 2 (right) (Source: Kron, 2010a). Middle: Parametric equations of Method 3 and the resulting helix curve.

the parameters (e.g. period and magnitude) controlling the functions in the equations. The result is shown in Figure 1 Middle-Right.

Method 4: Using point coordinates with API

Method 4 extends Method 3 by using Revit API programming (in Microsoft C# language) to implement the calculation of the same coordinate equations (1). It is easier to create a large amount of control points automatically to generate a long and smooth curve. In addition, more mathematical functions can be used in API programming, which enables more complex forms to be generated.

Each method has its focus of learning and limitation of use. The implicit methods are easy to use once the students have a good understanding of the relationship among solids, surfaces, and curves. The explicit methods use the parametric equations and are more general, which enables the change of the equations easily to create other curves like torus and butterfly curves (Figure 1 Bottom). The parametric equations of a torus curves are known as:

$$\begin{aligned}x &= \cos(t)(R+r \cos(u)) \\y &= \sin(t)(R+r \cos(u)) \\z &= r \sin(u)\end{aligned}$$

where the two parameters t and u both vary between 0 and 2π , and R and r are the major radius and minor radius, respectively. The parametric equations of a butterfly curve are given by Fay (1989):

$$\begin{aligned}x &= \sin(t)(e^{\cos(t)} - 2\cos(4t) - \sin^5(t/12)) \\y &= \cos(t)(e^{\cos(t)} - 2\cos(4t) - \sin^5(t/12))\end{aligned}$$

By simply substituting the equations but keeping the same Revit family structure or API program structure, we can create the new curves (Figure 1 Bottom). It is clear that the butterfly curve cannot be created using Method 1 and 2 because we don't have reference geometry that can be used to trace the control points to form the curve.

To a certain extent, these sample methods demonstrate an analogy of the transition from Euclid's geometry constructions (with the use of auxiliary lines) to Descartes' coordinate geometry. The class was given related readings and was a little surprised that "The Greeks could not do arithmetic" (Page, 2009; Turchin, 1977).

SAMPLE 2: PARAMETRIC SOLID AND BUILDING COMPONENT MODELING

This sample is recursive geometry and its corresponding BIM component. This sample was originally inspired by the physical model of a project from a design studio (Figure 2 Top), and later by (Woodbury, 2010) and its implementation in Revit (Kron 2010b).

The algorithm applied to the design of the structure is: at every new level, the square element with a fixed setback value is rotated so that its corner points touch exactly the corresponding edges of the element of the previous level.

The parametric methods of this sample will demonstrate: (1) how mathematics is used in design. Quadratic formula solutions and trigonometry functions are used to resolve the mathematical problem of calculating the rotation angle for each level (the angle increment is different at each level); (2) how algorithms (in this case, recursive algorithm) are used in design. The apparent differences between the computer models are the results of different parameters following the same algorithm - this helps student comprehend the simple essence of the seemingly complex forms; (3) parametric modeling fosters design changes to be made quickly; (4) the precision of the computer model will enable the physical model to be precisely created with CNC machines (though the physical model was done by using another CAD model of the design in this case); and (5) physical materials can be applied to the geometry, indicating that the structure can potentially be built, which is a significant advantage of BIM-based parametric modeling over geometry-based parametric modeling methods.

Method 1 – Creatively Using Modeling Tool's Functionalities

Register (2010) developed a method to create recursive forms in Revit. Using the method, Kron (2010b) created the recursive forms shown in Figure 2 Bottom. The method is to manually insert two similar geometry groups to each other recursively (Register 2010). Specifically for the recursive square geometry, the method draws a square on the edges of the previous square recursively by using the Snap capability of Revit (Kron, 2010b). There is no complex angle calculation in this method. Using this simple

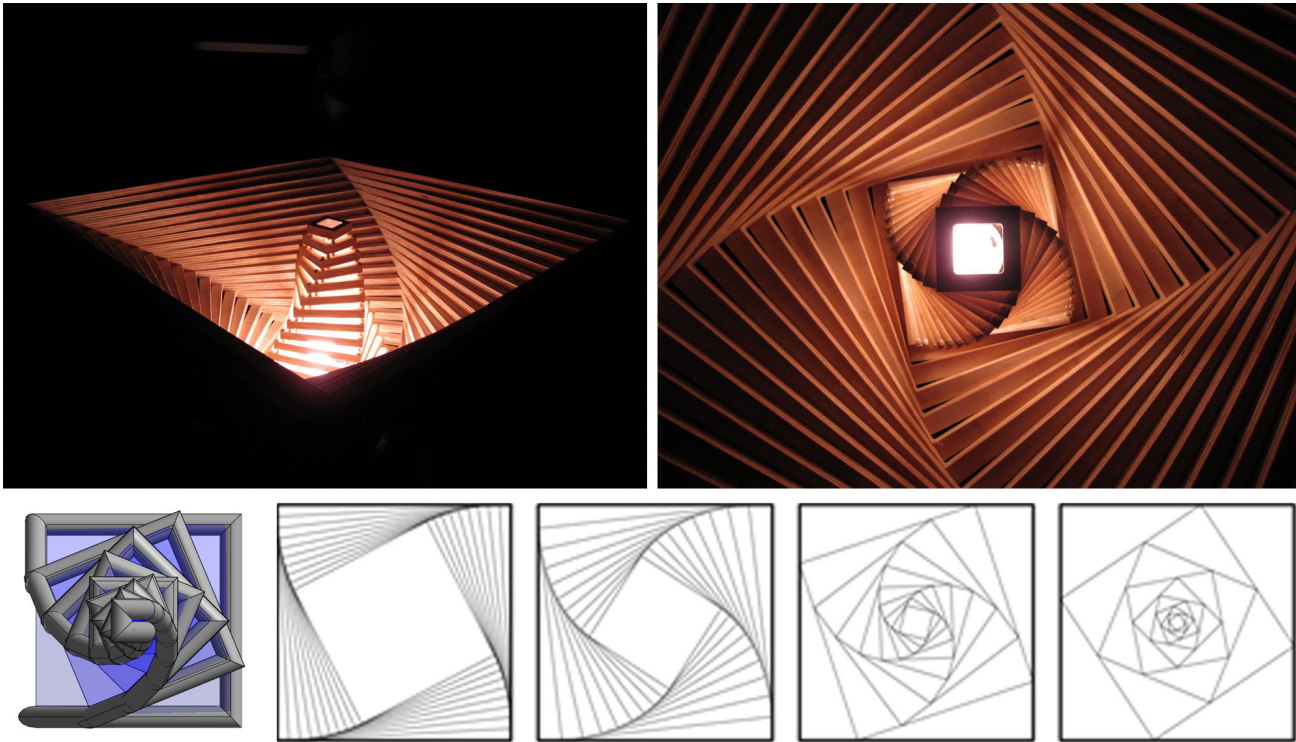


Figure 2. Top: Physical model of a lighting appliance of recursive geometry, designed, built, and photographed by Benjamin Wilde for his studio project at Texas A&M University. Bottom: Recursive geometry created using Method 1 (Image source: Kron, 2010b).

yet creative method, students can develop an intuitive understanding about the idea of recursive geometry by playing the manual process.

Method 2 – Using External Database and Scripting

Method 2 uses explicit parametric formulas by creating the quadratic equations and resolving them to calculate the angle between each square and its previous square (i.e. angle α in Figure 3 Top). Angle α is different for each square. An accumulated angle, which is the sum of a given square's α angle and all the previous squares' α angles, is needed to finally orient the given square. However, Revit lacks the ability to store previous α angles in its graphical user interface of the parametric modeling editor, therefore another solution using external tools was sought. The parametric design can be implemented partially in Revit Family Editor for calculating each α angle (Figure 3 Middle) and partially in Microsoft Access® with VBA scripting for calculating accumulated orientation angle. The process is: in Revit, create copies of the squares, export the project to MS Access database that contains the individual α

values, sum a square's α value with all its previous squares α values, and assign the sum to the final orientation of the square. Repeat the same steps to calculate all squares' summed angles. Finally import back the database to update the model. The results are shown in Figure 4 Top and Middle. We also demonstrated that the geometry can be converted into BIM components with different materials to enable constructability study. The materials include: default mass material, brick material, and glass material, as seen on Figure 4 Top-Right, Middle-Left, and Middle-Right, respectively.

This method provides an opportunity to introduce the use of external database and scripting functions to facilitate the understanding of interoperability between BIM and external software tools.

Method 3 – API Programming

Alternatively, the creation of the recursive geometry can be done from scratch in Revit API, which is our Method 3. This method uses the same parametric equations as in Method 2 or shown in Figure 3 Top. The method explicitly uses the recursive al-

gorithm that can create fractal geometry, as shown in Figure 3 Bottom.

This recursive algorithm in the form of programming source code provides the same results but a very different view to the process of Method 1, which was more intuitive to designers (other than programmers). Method 3 is more automatic in terms of creation process than Method 1 and 2. Method 2 and 3 enables more mathematical functions be used in the process as long as the scripting and programming languages contain them.

The API program also assigns a material (glass) to the geometry, to demonstrate the power of computer programming to create both geometry and materials as BIM components. The result is shown in Figure 4 Bottom.

Method 4 – Combining Manual Creation and API Programming

We have also created Method 4 that combines manual creation of initial geometry (the square with its parameters and formulas) and API programming for implementing complex calculations or algorithms (i.e. the recursive algorithm). This method can be most effective to designers because of balanced modeling and coding. The result models of this method are similar to those shown in Figure 4 Top and Middle.

Method 5 – API Integrated with Equation-based Modeling

To take it even further, we also demonstrated a new method that uses equation-based modeling, in which the quadratic equations (Figure 3 Top) do not need to be resolved manually, but automatically by the equationbased modeling language – Modelica - using Newton’s numerical analysis method. Through communication between Revit API and Modelica, the equations’ solutions (angles) could be passed to Revit from Modelica, and used to create the recursive geometry.

Each method has its focus of learning and limitation of application. Method 1 requires a manual process of creating parametric models but it can help students develop an intuitive understanding about the process of creating complex geometry. Method 2 links BIM and an external database for an under-

standing of software tools’ interoperability. Method 3 and 4 challenge students with computer programming, but students can have more indepth control and better understanding about parametric equations and the algorithms for creative form generation. Finally, Method 5 demonstrates the power of computing technology, and that architects should learn mathematics but only half of it – they need to know how to do mathematical modeling or creating the equations representing the relations among parameters, but do not need to know how to resolve the equations as this task can be done by computers. This way, architects can focus their efforts on creative design thinking, instead of tedious, routine tasks that computers can do better than human.

This is a comprehensive sample demonstrating how to create a BIM-based parametric model for resolving a real-world design problem (Figure 2 Top). With this sample, we can exercise: conceptual massing, parameters, formulas, trigonometry (tangent and arctangent functions), algebra (polynomial/quadratic equations), database exporting/importing, scripting, and programming.

SAMPLE3: PARAMETRIC BIM PROJECT MODELING

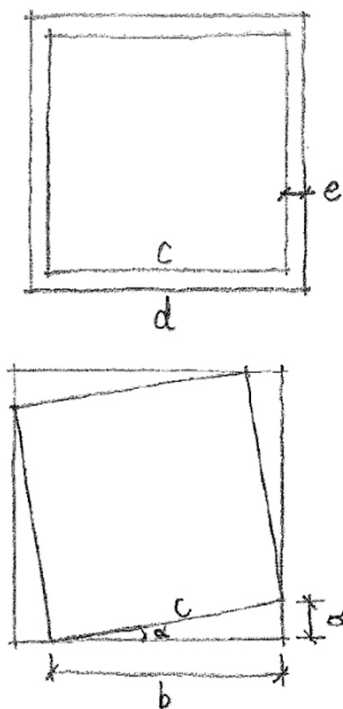
Sample 3 demonstrates BIM and parametric modeling at the building project level. Computability of BIM can be very useful for energy simulation, cost estimation, etc. To get student started to understand using BIM data for computing, in Sample 3, a simple calculation of a room’s dimension drives the design of the furniture. Three phases (instead of methods) of modeling are introduced to help students learn the method step by step.

Phase 1. Parametric Component

The task of Phase 1 is to create a well-defined Table-Chair component family, in which the table’s width and depth drive the layout of the chairs and the number of chairs on each side of the table (Figure 5 Left). Simple formulas calculating the number of chairs based on the table’s dimensions are exercised.

Phase 2. Parametric Project

Phase 2 of the sample asks students to access parameters of building components, do calculation utilizing API programming’s math functions, and



knowns: d, e (and $c = d - 2e$)
 unknowns: α ($\alpha = \arctan(a/b)$)

$$a + b = d$$

$$a^2 + b^2 = c^2$$

$$a^2 + (d - a)^2 = c^2$$

$$a^2 + d^2 - 2da + a^2 = c^2$$

$$2a^2 - 2da + (d^2 - c^2) = 0$$

using quadratic formula solutions:

$$x = \frac{-B \pm \sqrt{B^2 - 4AC}}{2A}$$

$$a = \frac{2d \pm \sqrt{4d^2 - 4 \cdot 2 \cdot (d^2 - c^2)}}{4}$$

$$= \frac{d}{2} \pm \frac{1}{2} \sqrt{d^2 - 8de + 8e^2}$$

(sign will be determined by the actual geometry: ("−" is valid for a) ("+" is valid for b))

$$b = d - a$$

$$= \frac{d}{2} \mp \frac{1}{2} \sqrt{d^2 - 8de + 8e^2}$$

$$\alpha = \arctan(a/b)$$

Parameter	Value	Formula
Constraints		
Default Elevation	4' 0"	=
Materials and Finishes		
Material	<By Catego	=
Dimensions		
Width1 (default)	12' 0"	= Width - (Linear Change Per Level * 2 * LevelNo)
Width (default)	12' 0"	=
Linear Change Per Level (default)	1' 0"	=
LevelNo (default)	0	=
Height (default)	6' 0"	=
Base2 (default)	6' 0"	= Base1 + Height
Base1 (default)	0' 0"	= LevelNo * Height * (1 + 1 / 3)
AngularChangePerLevel (default)	10.584°	= atan((d / 2' - sqrt(8 * e ^ 2 - 8 * d * e + d ^ 2) / 2) / (d / 2' + sqrt(8 * e ^ 2 - 8 * d * e + d ^ 2) / 2))
A1 (default)	0.000°	=
Other		
n (default)	0.186860	= tan(AngularChangePerLevel)
m (default)	0.186860	= (d / 2' - sqrt(8 * e ^ 2 - 8 * d * e + d ^ 2) / 2) / (d / 2' + sqrt(8 * e ^ 2 - 8 * d * e + d ^ 2) / 2)
e (default)	1' 0"	= Linear Change Per Level
d (default)	14' 0"	= Width1 + (Linear Change Per Level * 2)
Thickness (default)	2' 0"	= Linear Change Per Level * 2
Identity Data		

```
private void CreateSquare(int depth, double angle)
{
    if (depth > 2)
    {
        angle += Math.Atan((d / 2 - Math.Sqrt(8 * e * e - 8 * d * e + d * d) / 2) / (d / 2 +
            Math.Sqrt(8 * e * e - 8 * d * e + d * d) / 2));
        CreateSquare(depth - 1, angle); //Recursively create squares.
    }
}
```

Figure 3. Top: Parametric modeling diagram, parameters, equations, and solutions. Middle: Parametric formulas of Method 2, implemented in Revit. Bottom: Recursive algorithm in C#.

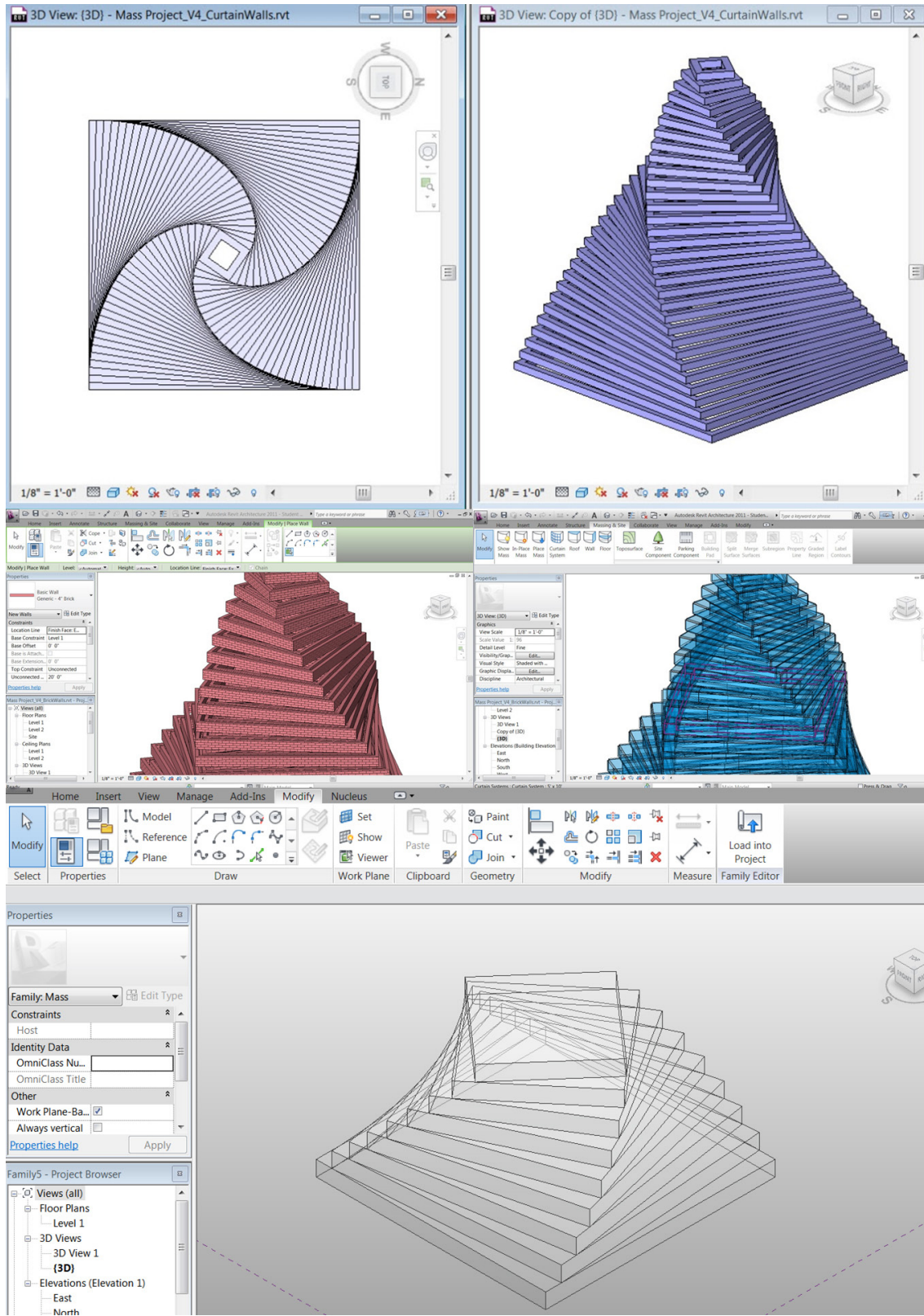


Figure 4. Top and Middle: Parametric BIM models created using Method 2. Bottom: Parametric BIM model created using Method 3.

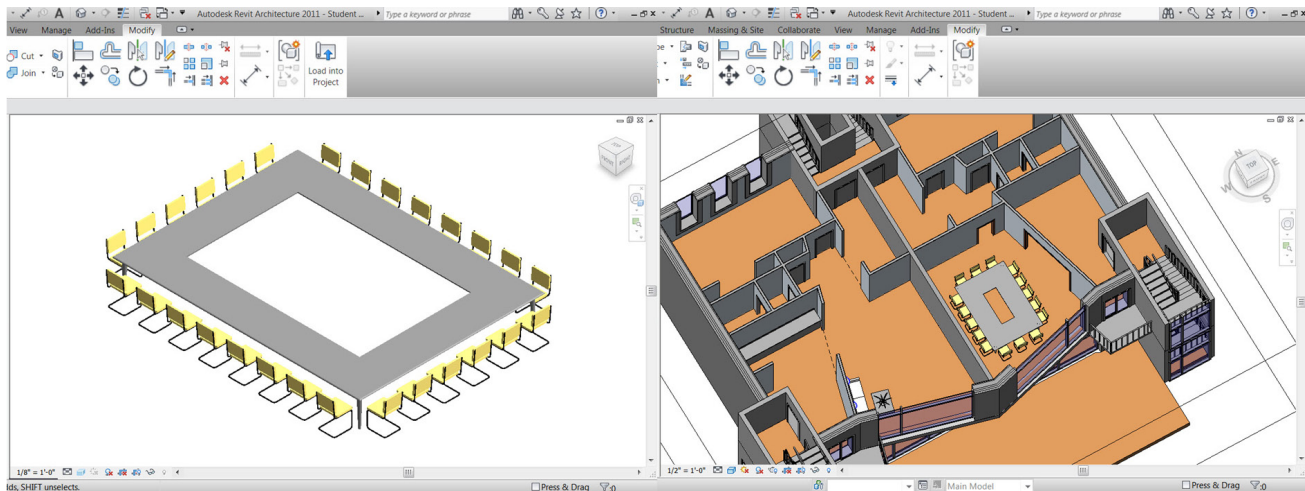


Figure 5. Parametric table-chair model (left) and parametric project including a sample building and the table-chair model. (Building model source: Autodesk, Inc.)

pass the calculation results to parameters of other building components. The goal of instruction is to let students learn the Object-Oriented Programming and modeling concepts, and learn that parameters that define the building geometry and behaviors are the critical entities that need to be identified carefully and accessed freely to carry design intent. The process to establish a parametric relation between any pair of objects is emphasized:

1. Get the two objects
2. Get parameters of the objects
3. Get values of the parameters
4. Calculate new values using the obtained parameter values from one object
5. Set parameters of another object using the new values

In Step 4 above, designers can utilize the programming language (C#)'s math functions (which are much more powerful than the Revit's built-in formula functions) to do a calculation that may be based on research of building codes (fire egress, energy, etc.).

Specifically, this sample allows users to change the room dimensions, which will result in the change of the table dimensions, propagated to the change of the chairs (Figure 5 Right).

Phase 3. Adding External Data through API

In Phase 3, external databases are added into the project and linked to the project model through

API. Specifically, MS Excel® and CSV (Comma Separated Values) datasheets are used to store the data. Students will understand that external data can be used to update the model or to enable other analyses for building design.

CONCLUSIONS AND FUTURE WORK

The pedagogic objective of the course is to help students develop BIM and parametric modeling skills and an understanding of the enabling technologies that enhance Creativity, Constructability, and Computability (3C's). Creativity is enhanced by form generation using parametric methods. More extensive study of geometry modeling using different parametric methods and tools is suggested for further study. Constructability is investigated in the class through the use of building semantics, materials, and 3D modeling. 4D (with schedule) and 5D (with schedule and cost information) software applications are demonstrated in class too. Computability is exercised through a simple sample in class, but simulations of building energy and users' behaviors are briefly demonstrated in class. More extensive study and research in each of the areas after the course are encouraged for students.

Through exercise of the samples using multiple methods, students can compare the methods and develop essential ideas and skills for BIM and parametric modeling. Identifying critical parameters and their associated methods (parametric formulas and algorithms) are important for both BIM and

parametric design. Multiple phases of an exercise enable learning of challenging subjects gradually.

The course outcome (projects) and student evaluation (which has been high in the department) show that the teaching method with course samples enables effective learning of the complex subject of integrated BIM and parametric modeling. However, more quantitative evaluation of the course needs to be conducted in the future.

This paper has been focused on the selected samples that the students study in the class. More complete course content including the syllabus, schedule, homework, project assignments, readings, and student works can be found in Yan (2011).

REFERENCES

- Fay, T. H. (1989). The Butterfly Curve. *Amer. Math. Monthly* 96 (5): 442-443.
- Kron, Z. (2010a). Parametric Patterns VI: Increment. URL: <http://buildz.blogspot.com/2011/01/parametric-patterns-vi-increment.html>. Last accessed on April 8, 2011.
- Kron, Z. (2010b). Parametric Patterns X.1: Recursion, encore. URL: <http://buildz.blogspot.com/2011/02/parametric-patterns-x1-recursion-encore.html>. Last accessed on April 8, 2011.
- Light, D. (2010). How to create in Helix in Revit 2011 using adaptive component. URL: <http://www.youtube.com/watch?v=8XNnlx1yhbg>. Last accessed on April 8, 2011.
- Woodbury, R. (2010). *Elements of Parametric Design*, 1st ed. Routledge. URL: <http://www.designpatterns.ca/>. Last accessed on April 8, 2011.
- Page, J. (2009). Introduction to constructions. URL: <http://www.mathopenref.com/constructions.html>. Last accessed on April 8, 2011.
- Turchin, V. (1977). The Phenomenon of Science. A cybernetic approach to human evolution. New York NY: Columbia University. URL: <http://pespmc1.vub.ac.be/pos/Turchap11.html>. Last accessed on April 8, 2011.
- Register, S. (2010). Parametric Patterns X: Recursion. URL: <http://buildz.blogspot.com/2011/02/parametric-patterns-x-recursion.html>. Last accessed on April 8, 2011.
- Yan, W. (2011). ARCH 653, Texas A&M University. URL: <http://faculty.arch.tamu.edu/wyan/ARCH653/>.