

Generative modelling of an Albertian capital

Modelação generativa de um capitel Albertiano

Eduardo Castro e Costa, José Pinto Duarte

castroecosta@fa.utl.pt, jduarte@fa.utl.pt

Universidade Técnica de Lisboa.

Filipe Coutinho, Mário Krüger

filipecoutinho@darq.uc.pt, kruger@ci.uc.pt

Faculdade de Ciências e Tecnologia da Universidade de Coimbra

ABSTRACT – The main objective of the ongoing investigation described in this paper is to produce a Corinthian capital through generative modelling and digital prototyping, and according to Leon Battista Alberti’s treatise on architecture, *De re aedificatoria*. This investigation is part of the Digital Alberti research project, which aims to decode Alberti’s treatise through the use of new technologies. This treatise can be interpreted as a set of instructions regarding the art of building. The instructions related to Alberti’s column system were translated into computational models, capable of generating digital instances of column elements according to the classical canons prescribed in the treatise. These instances were then digitally produced, materializing Alberti’s theory. Despite Alberti’s thoroughness, some detailing information is missing, namely for the Corinthian capital; observation and computational geometry were important for filling in the gaps. Such investigation allows determining the suitability of different modelling strategies, as well as the potential of different prototyping technologies.

Key words: Alberti, *De re aedificatoria*, Corinthian capital, generative modelling, digital fabrication.

RESUMO – O objectivo principal da investigação em curso descrita neste artigo é produzir um capitel Coríntio através de modelação generativa e prototipagem digital, de acordo com *De Re Aedificatoria*, o tratado de arquitectura de Leon Battista Alberti. Esta investigação é parte integrante do projecto de investigação Alberti Digital, que procura descodificar o tratado de Alberti mediante o uso de novas tecnologias. O tratado pode ser interpretado como um conjunto de instruções para a arte de bem construir. As instruções relativas ao sistema de coluna de Alberti foram traduzidas em modelos computacionais, capazes de gerar instâncias digitais dos elementos da coluna segundo os cânones clássicos prescritos no tratado. Estas instâncias foram depois produzidas digitalmente, materializando a teoria de Alberti. Apesar do rigor de Alberti, falta alguma informação de pormenorização, nomeadamente para o capitel Coríntio; observação e geometria computacional foram importantes para preencher os espaços vazios. Esta investigação permite determinar a adequação de diferentes estratégias de modelação, bem como o potencial de diferentes técnicas de prototipagem.

Palavras-chave: Alberti, De Re Aedificatoria, capitel Coríntio, modelação generativa, fabricação digital.

Context

The activities described below are but a small part of a research project named Digital Alberti. The project’s main objective is to determine the influence of Alberti’s theory on Portuguese architecture, by decoding his treatise on architecture, *De re aedificatoria*, through the use of digital computational technologies. In particular, it aims to determine such an influence by converting the treatise into shape grammars and then determining the changes to those shape grammars required to account for the generation of Portuguese classical buildings. Currently, two shape grammars are being developed, one for the column system and another for churches, describing the different levels of detail found in Alberti’s treatise. Eventually the two grammars will converge, serving as a base for a compositional analysis tool.

Among the many tasks necessary to achieve the main objective is the production of models that materialize, both virtually and physically, the treatise’s content. The work described below is part of an effort to model and produce all the column system elements, so as to support the development of the corresponding shape grammars.

Alberti’s treatise contains his considerations on the art of building. This document was first published in 1485, without illustrations, although in many later editions illustrations were added by other authors (Alberti, 2011). Therefore, besides aiding in the construction of the shape grammar, the task’s outputs will serve as the treatise’s missing illustrations, in the form of drawings, digital three-dimensional models and interactive computational and virtual models. These elements will be featured in an exhibition to be held at the end of the Digital Alberti project, as well as in a published book and in educational software on the subject.

Methodology

Prototyping the Corinthian capital implied two main processes: a) translating the part of the treatise describing the capital into a computational model, and b) generating the physical models. The translation of the treatise comprises four phases: i) understanding the treatise, by decoding the instructions prescribed by Alberti; ii) building the computational model, by implementing those instructions into a computer program; iii) filling in the gaps, by completing the algorithms where instructions are missing, and iv) optimizing the computational model, by implementing subroutines and classes (Table 1).

Understanding the treatise

The first step in understanding Alberti's treatise was its analytical reading, focusing on Books Six and Seven, which comprise chapters describing the column system. In these chapters, several definitions can be found regarding the system's elements, definitions which will henceforth be called "rules". In order to better understand these rules, a thorough analysis was undertaken of one of the elements, the Doric base, described in Chapter 7 of

Book Seven. The same process was then applied to all other elements.

Reading and annotation

Consider the following passage from the treatise, regarding the Doric base: "*The height of the base was then divided into three parts, one of which was taken up by the thickness of the die*" (Alberti, 1988, p. 207). This rule defining the height of the die can be mathematically described as $H_{die} = 1/3 \cdot H_{base}$ in which "the height of the die" corresponds to H_{die} , and the rest of the phrase corresponds to the latter part of the equation. Therefore, while reading the text, the rules were highlighted and translated into equations.

Essential to the treatise's analysis was the annotation effort, composed of sketching the column elements, so as to clarify the proportional relations prescribed by the rules, and scribbling their mathematical translations, allowing them to be put in perspective and seen as a whole (Figure 1).

In the development of all other elements, such as the Corinthian capital, the same methodology was followed (Castro e Costa *et al.*, 2011). Alberti's rules for

Table 1. Methodology.

Treatise		→	Computational model		→	Physical model	
Translating the treatise							
Understanding the treatise		Building the computational model		Filling in the gaps		Optimizing the computational model	

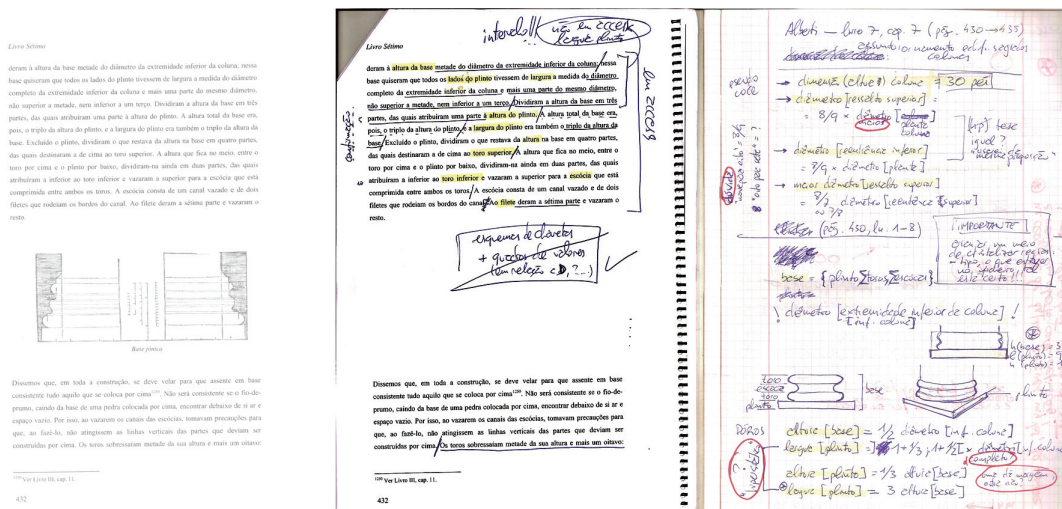


Figure 1. Reading and annotation.

this capital, found in Chapter 8 of Book Seven of the treatise, were systematized. Tables 2a and 2b describe the systematized rules, internally numbered, along with their mathematical translation and their location in the English edition of the treatise (Alberti, 1988). Throughout the tables, three main types of operations can be identified: proportion, subdivision, and detailing.

Structuring the rules

As the complexity of the system grew, writing down the rules on paper or through word processing proved insufficient. A relational database, however, provided for a more efficient record of the rules, surpassing

the two dimensions of the notebook. This upgrading allowed for eliminating redundant variables and for a better understanding of the links between rules. Being able to see the rules as part of a whole led to the identification of a structure, in which those rules are organized in a hierarchical tree. Also a pattern was identified, in which rules follow a general form $A = k \cdot parent(A)$, where A represents a parameter of the column element (typically a dimension), $parent(A)$ represents the element hierarchically superior to A (its reference), and k represents a constant value, prescribed by Alberti, usually in the form of a fraction (multiplier). In the previous example, A would be the height of the die, $parent(A)$ would be the height of the base, and k would be one third.

Table 2a. Rules for modelling the Corinthian capital.

pg.208	In.06	capital	height of the Corinthian capital is equal to the diameter at the base of the column	#01	
			$H_{capital} = Dimoscape$ (H = height; D = diameter, initial variable)		
	In.07	capital	and is divided into seven modules.	#02	
			$M = 1/7 \cdot H_{capital}$ (M = module, auxiliar variable)		
	In.08	abacus	The abacus takes up one module	$H_{abacus} = 1 \cdot M$	#03
		vase	and the remainder is occupied by the vase,	$H_{vase} = 6 \cdot M$	#04
		vase	whose base has the same width as the top of the column,	#05	
		$lowerW_{vase} = D_{sumoscape}$ (W = width*)			
	09	vase	and whose upper rim has the same width as the bottom of the column.	#06	
			$upperW_{vase} = Dimoscape$		

* width is used instead of diameter so as to simplify the system

pg. 208	In. 11	abacus	The abacus is ten modules wide	$W_{abacus} = 10 \cdot M$	#07
		abacus	but half a module is clipped off at each corner.	(detailing rule: see scheme)	#08
	In. 12	abacus	The abaci of all other capitals consist of straight lines; those of the Corinthian curve inward until the distance between them is the same as the width at the	#09	
			bottom of the vase.	(detailing rule: see scheme)	
	14	border	The border of the abacus takes up a third of its height,	$H_{border} = 1/3 \cdot H_{abacus}$	#10
	In. 15	border	and its lineaments are identical to those at the top of the shaft of the column.	#11	
			(detailing rule: see scheme) (lineaments top column shaft → pg. 188, Book Six, Chapter 13)		

Table 2b. Rules for modelling the Corinthian capital (continuation).

pg. 208	In. 17	vase	The vase is girt with a fillet and an astragal, (astragal from sumoscape lineaments, fillet, addressed further...)	#12	
		leaves	which cover it with two interlapping rows of leaves standing out in relief; each row contains eight leaves. $aWleaf = 1/8 \cdot Pleafrow$ (aW = arc length; P = perimeter)	#13	
	In. 19	leaves	The first row is two modules high, as is the second.	$Hleafrow = 2 \cdot M$	#14
		stalks	The remaining space is taken up by the stalks sprouting out from the leaves to the full height of the vase.	$Hstalkrow = 2 \cdot M$	#15
pg. 208	In. 21	stalks	These stalks are sixteen in number;	$aWstalk = 1/16 \cdot Pstalkrow$	#16
		stalks	four of them unfold on each face of the capital, two from the same knot on the right, and two from the same knot on the left; (detailing rule: see scheme and ‘Filling in the Gaps’)	#17	
pg. 209	In. 01	stalks	the two end ones hang below the corners of the abacus in a form of spiral, (detailing rule: see scheme and ‘Filling in the Gaps’) (spiral → Ionic capital, pg. 207, Book Seven, Chapter 8)	#18	
		stalks	while the middle ones also curl, so that their ends meet in the center. (detailing rule: see scheme and ‘Filling in the Gaps’)	#19	
pg. 209	03	flower	Between these middle two a flower sprouts prominently from a vase, (detailing rule: see scheme and ‘Filling in the Gaps’)	#20	
		flower	as far as the top of the abacus.	$Wflower = 1 \cdot Habacus$	#21
	In. 04	vase	The rim of the vase, where it is visible and not covered by stalks, is one module thick.	$Hrim = 1 \cdot M$	#22
		leaves	Each leaf should be articulated into five or, possibly, seven lobes. $aWlobe \in \{1/5 \cdot Pleaf, 1/7 \cdot Pleaf\}$	#23	
	06	leaves	The tips of the leaves hang forward half a module.	$Zleaf = 1/2 \cdot M$ (Z = depth)	#24
	In. 07	leaves	As with all carving, deeply incised lineaments will add great charm to the leaves of the capital. (detailing rule: see scheme and ‘Filling in the Gaps’)	#25	

Figure 2 represents the hierarchical structure of the Corinthian capital's components and the corresponding rules, explaining the steps in the modelling of the Corinthian capital.

Building the computational model

In order to test the models described in the previous chapter, they should be implemented into a computational algorithm. The selected tool for this task was Grasshopper (GH), a visual programming plugin for the geometric modelling software Rhinoceros (Fig-

ure 3). As a visual programming interface, GH made it possible to develop generative computational models of the columns system's elements. These generative models are able to generate instances of column elements – and ultimately of the system itself –, in real time and according to Alberti's rules, which had been properly translated into the programming language. These models are also parametric, so that an element's parameters, typically related to proportion and subdivision operations, can be adjusted, allowing for the generation of several plausible variations of the corresponding element.

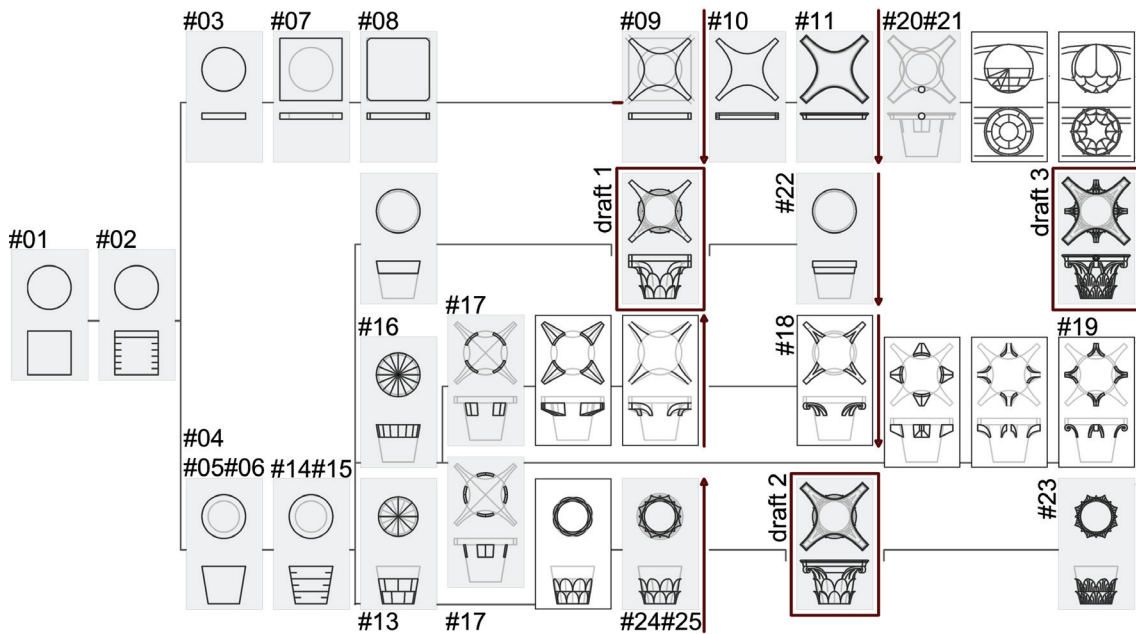


Figure 2. Modelling process for the Corinthian capital.

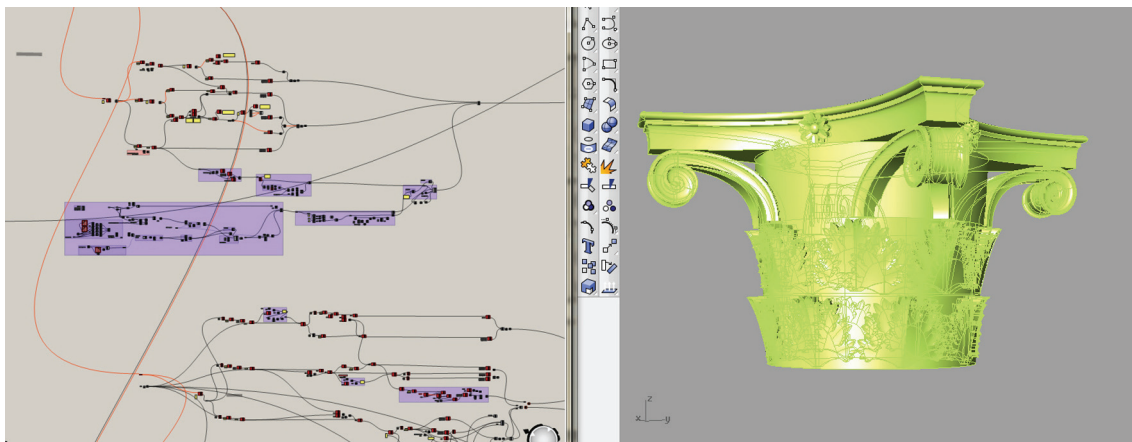


Figure 3. Visual programming in Grasshopper and Rhino.

GH algorithms are developed by connecting components. These connections represent relationships, bearing some correspondence with the relationships in the modelling schemes presented in the previous chapter. Likewise, the steps, or states, in the same schemes sometimes correspond to the interconnected components which perform tasks. However, it rarely happens that one single GH component can perform the task represented in the scheme states. Usually, the tasks have to be carried out by groups of (much) more than one component. These groups have eventually been structured into subroutines, which will be addressed below.

As stated before, three predominant types of operations are performed while modelling the column elements: proportion, subdivision, and detailing. Proportion operations typically assign values to dimension parameters of given elements, based on some other elements. Subdivision operations establish levels in the hierarchical structure. Detailing operations are responsible for molding the elements into their final form.

Of all three types, detailing operations stand out for two reasons. First, their implementation is more complex, implying the use of more GH components, when compared with proportion and subdivision operations. And second, because Alberti prescribes only a few formal features for the detailing of some elements, which hardly is enough to determine their exact shape. This is the case of the sprouting stalks, the acanthus leaves and the sprouting flower. In order to achieve a satisfactory level of detail, some gaps had to be filled in, and therefore many detailing operations are based on something other than the treatise alone.

Filling in the gaps

As the treatise was insufficient for generating the missing pieces (Castro e Costa *et al.*, 2011), it was necessary to look into other sources, such as illustrations of later editions of *De re aedificatoria*, and observation of built examples. Illustrations of other treatises were also con-

sulted, but only aspects not conflicting with Alberti's rules were taken into account. Then experimentation began by modelling the missing pieces as close to those sources as possible, trying to infer the rules that could generate them.

Sprouting stalks

The Corinthian capital features two types of stalks (Table 3, #18, #19). For the sake of coherence and simplicity, both types were modelled in the same way, the differences between them being solely in terms of proportions and the inscribing solid, or envelope. This implied the development of a fairly versatile algorithm. For the sake of better understanding, the larger stalk modelling process will be described. The generic stalk was modelled as an extrusion of a profile, or section, along a generating curve, or path as explained in following.

Path curve

Regarding the path curve, it is possible to extract two features from the rules above: its endpoints (starting point on leaves: #15, ending point on abacus: #18) and its shape at the end (a spiral: #18). The rest had to be filled in. The path was defined as a three-dimensional curve, resulting from a combination of two 2D-curves contained in perpendicular surfaces.

In the few rules prescribed for the stalks, Alberti points to a relationship between the larger ones and the abacus (#18). Therefore, in the model, the horizontal projection of the path curve was designed to conform with the abacus' contour (Figure 4, left).

The stalks are also said to be "sprouting out from the leaves" (#15), hinting at a vertical direction for the path curve's tangent at its starting point. They should "hang in the form of a spiral" or "curl". Therefore, this spiral was designed according to Alberti's definition for the Ionic capital (Alberti, 1988, p. 207), which generates a two-point involute, to be exact.

In order to complete the vertical projection, connecting the spirals to the leaves, a curve was needed that

Table 3. Rules for modelling the sprouting stalks.

#15	The remaining space is taken up by the stalks sprouting out from the leaves to the full height of the vase.
#17	four of them unfold on each face of the capital, two from the same knot on the right, and two from the same knot on the left;
#18	the two end ones hang below the corners of the abacus in a form of spiral,
#19	while the middle ones also curl, so that their ends meet in the center.

had the following features: vertical tangent at starting point; inward tangent at endpoint, so as to maintain the continuity with the spiral curve (principle of continuity); and least possible number of control points (principle of simplicity). A curve that meets these requirements is a NURBS curve of the 2nd degree, with 3 control points (Figure 4, right).

Profile curve

No explicit rules can be found in Alberti's treatise regarding the stalk's profile. Therefore, its shape was adapted from the lineaments prescribed in the treatise for other parts. In the first implementations of the Corinthian capital, the bottom of the column shaft (Alberti, 1988, p. 186-188; Figure 5) was used and later replaced by the lineaments for the volute of the Ionic capital. Both geometries benefit from the material qualities of the fillet, preventing the occurrence of acute angles, which are prone to breaking (Figure 5).

The profile algorithm was meant to be used later in the detailing of the Ionic capital. However, later in the

optimization phase, Alberti's lineaments were generalized, rendering the profile easier to generate and more coherent with the whole system.

Acanthus leaves

Like the Corinthian capital, the algorithm that generates the acanthus leaf by itself is still a work in progress, as it is constantly being upgraded, aiming at two main objectives: similarity and versatility.

The generated acanthus leaf should be indistinguishable from the numerous sculpted examples found in classical architecture. This implies an ongoing two-fold investigation, by extracting the rules both from a thorough observation of sculpted examples, preferably in Alberti's religious buildings, and by trying to understand some of nature's geometrical rules. Versatility is present in the algorithm on two levels. On the one hand, it is developed so that its transformations can be applied on any surface, thus allowing it to be used later in the detailing of the sprouting flower and, in the future, in the detailing of the scroll in the Ionic capital, for instance (Figure 6, left).

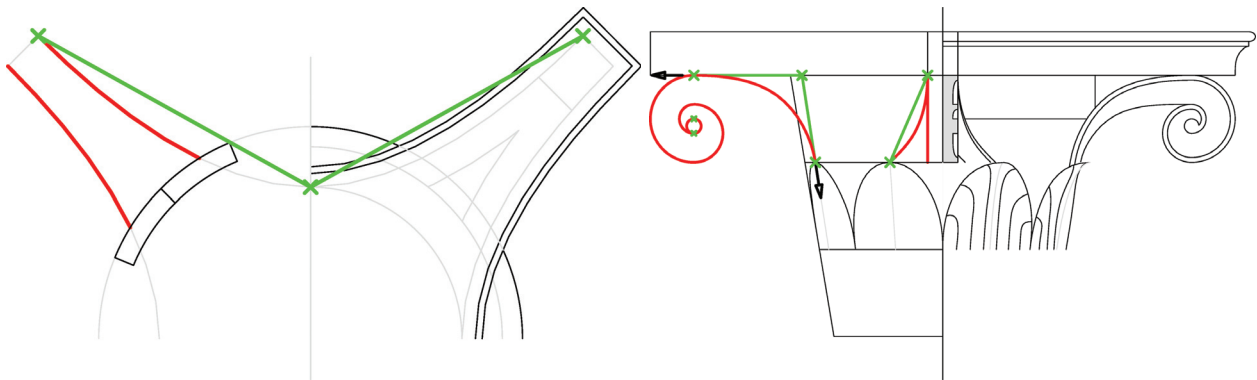


Figure 4. Stalk's path curve. Left: Horizontal projection. Right: Vertical projection.

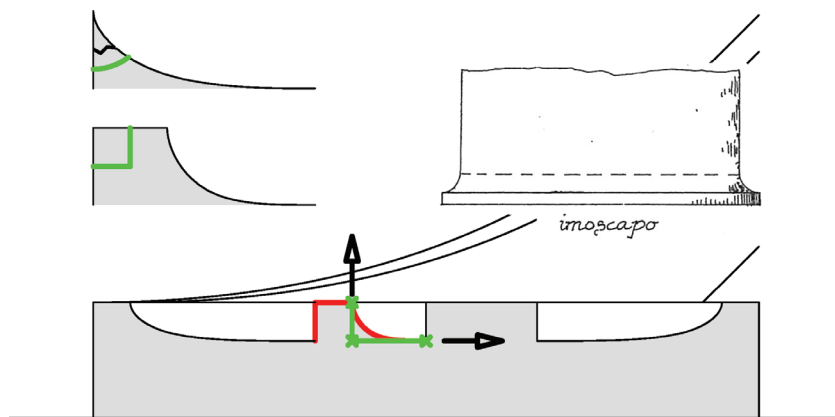
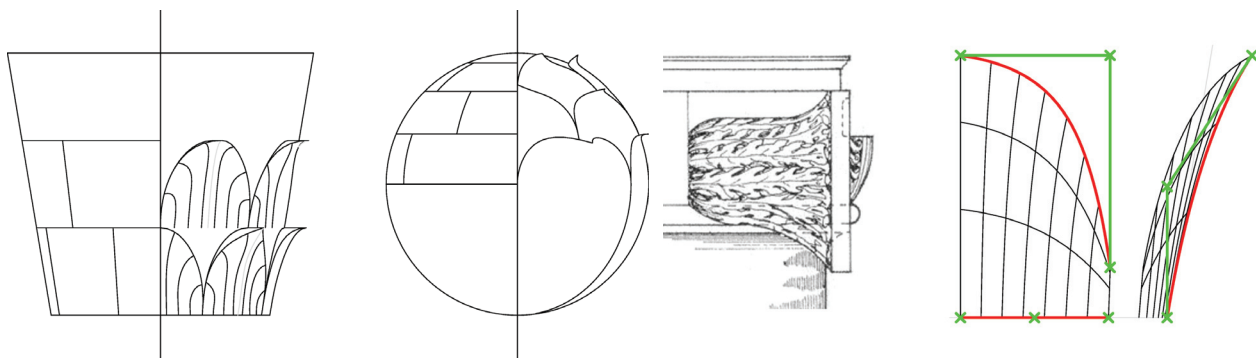


Figure 5. Profile lineaments; *imoscapo* illustration in Morolli e Guzzon, 1994.

Table 4. Rules for modelling the acanthus leaves.

#13	which cover it with two interlapping rows of leaves standing out in relief; each row contains eight leaves.
#14	The first row is two modules high, as is the second.
#23	Each leaf should be articulated into five or, possibly, seven lobes.
#24	The tips of the leaves hang forward half a module.
#25	As with all carving, deeply incised lineaments will add great charm to the leaves of the capital.

**Figure 6.** Left: Leaf algorithm applied on vase and sprouting flower, and example of the Ionic capital (*in* Morolli e Guzzon, 1994). Right: geometry of stylized leaf.

On the other hand, the algorithm outputs a surface whose complexity is kept to a minimum, allowing for a wider range of subsequent geometric operations, such as the subdivision of the leaf into its lobes (Table 4, #23).

As previously stated, the algorithm carries out a series of geometric operations on a given base surface, “transforming” it into a leaf – actually the base surface is not itself transformed; it merely acts as a container of geometric parameters used by the algorithm to generate a new shape. The leaves on the Corinthian capital derive from the sub-surfaces resulting from previous successive subdivision operations in its vase, culminating in rule #13.

The acanthus leaf was modelled as a NURBS surface, allowing it to keep a reduced number of control points and thus conforming to the principle of simplicity. It is the result of a lofting operation between two curves, the lower one being derived directly from the vase’s horizontal sections, and the upper one being a transformation of the base surface (Figure 6, right). The loft was selected for its coherent and thus predictable behavior concerning UV coordinates. UV control is crucial for guaranteeing further geometric operations on the output geometry, the leaf, such as its subdivision into lobes, and all the sub-

sequent detailing necessary to shape it according to the sculpted references.

Sprouting flower

As previously stated, the sprouting flower petals were modelled using the same algorithm that generates the stylized acanthus leaves. The base geometry for the flower was a simple sphere conforming to rule #21 (Table 5).

Optimizing the computational model

Implementation of the first computational models, namely for the Doric base, the column, the Corinthian capital and, especially, for the Doric entablature, suggested the need for optimizing the system’s design. On the one hand, it became clear that several elements, like the tori and the scotiae, and moldings, like the ovolo and the gullet, share some common properties, making it possible to interpret them as topologically similar entities. That pointed to their possible implementation as computational classes. On the other hand, it often happened that the same groups of Grasshopper (GH) commands were used

repeatedly, especially while modelling the Doric entablature. In computer programming, it is considered to be good practice to create blocks of code, called subroutines, which contain instruction sequences used repeatedly. Therefore, the subroutine is defined only once, and then called for throughout the program whenever needed. This strategy is useful for at least two reasons: 1) the code gets lighter, since the instructions contained in the subroutine are no longer repeated, and 2) if those instructions are to be changed, they are updated only once, thus reducing the probability of error (Scott, 2006). Since GH is not prepared for implementing subroutines, the need arose to find a scripting language that would do so.

The subroutines and the classes were implemented using VB.NET, due to previous experience with similar languages. VB.NET is an object oriented programming language, which makes it suitable for implementing classes. In this language, subroutines are associated with classes. Therefore, implementing the classes proved twice as useful, solving the two identified problems. Work began on the definition of a class that would represent the geometries that shape the various column elements, both at design and programming levels. The class was named *coxel*, an acronym of COLUMN ELEMENT, such as *pixel* for picture element, or *voxel* for volume element.

The concept of coxel

The coxel concept was born right at the beginning of the investigation process, with the perception that

Alberti's rules can be expressed by a general form. Like the rules, the column elements themselves show a set of similar properties, or parameters, such as height and width, determined by Alberti's rules.

In terms of geometric modelling, each element of the column system comprises one or more surfaces. Each one of these surfaces is associated with a coxel. A coxel is an abstract entity, which can nevertheless be interpreted as an envelope containing the associated surface. Therefore, the coxel does not represent the surface; rather, it generates it.

Let us analyze the Corinthian capital (Figure 7). As stated, it comprises several elements, such as the abacus and the vase. The abacus is subdivided into three different elements, collar, fillet and recess. The vase, in turn, can be subdivided into the leaves and the stalks. For each of these elements, an instance of the coxel class is created, containing the element's characteristics, thus making it possible to generate the corresponding surface.

One coxel can generate many different surfaces. The same envelope can relate to a torus or a scotia, for instance. Figure 8 shows the sections of all possible surfaces, generated by a coxel the height of which equals its width. By translating these sections, be it around an axis or along a line, or curve, the elements of a capital or of an entablature can be obtained. By combining the surfaces, it is possible to generate more complex shapes, such as the Albertian moldings (Figure 8). Finally, it is also possible to apply certain algorithms to the surfaces, like the one that generates the acanthus leaves. Figure 7 shows the

Table 5. Rules for modelling the sprouting flower.

#20	Between these middle two a flower sprouts prominently from a vase,
#21	as far as the top of the abacus.

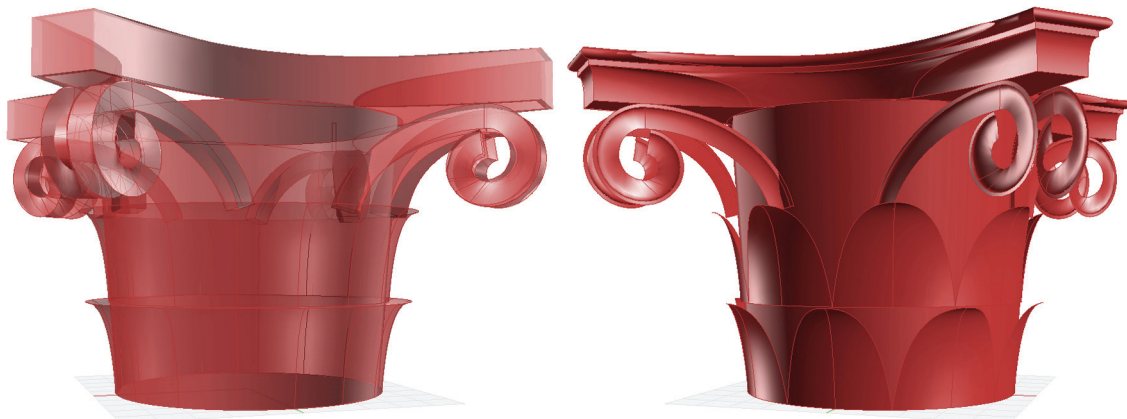


Figure 7. Left: Corinthian capital modelled with coxels. Right: Corresponding surfaces.

Corinthian capital's elements translated into coxels and the surfaces generated by them.

Implementing the coxel class made it possible to enhance the systematization of the computational models and thus to optimize them. Through the use of abstraction, it brings the modelling process closer to the thinking process of the architect (Scott, 2006). Comparing the two computational models for the Corinthian capital, before and after implementing the coxel, there is a significant reduction in the program size and complexity, and a smaller processing time is noticeable when changing the model's parameters.

Producing physical models

The modelling process described previously made way for the digital prototyping of physical models, to test both the development of the computational models and their materialization. The models were prototyped using

Fused Deposition Modelling (FDM) and 3D Printing (3DP). While both prototyping technologies are considered additive, they differ in terms of the state of the material used (Pupo *et al.*, 2009).

The testing of different technologies had two main goals: a) to determine the physical qualities of the modelled geometry; and b) to assess the suitability of each technology for producing the different elements of the column system, eventually extrapolating the conclusions for similar shapes, other than classical elements (Castro e Costa *et al.*, 2011). Since the scope of this paper is not the techniques themselves, we will not go into detail about the functioning or the procedures needed by each technology, except for what is relevant for the goals.

Fused Deposition Modelling

The FDM model of the Corinthian capital was produced at an early stage of its computational model. Some

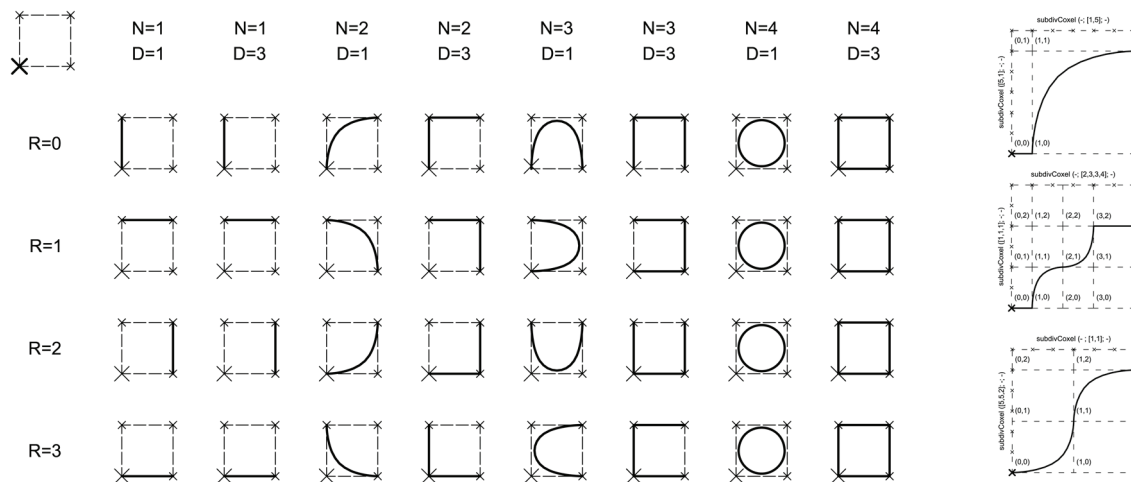


Figure 8. Left: Possible coxel geometries. Right: Moldings as combinations of coxels.



Figure 9. First iteration of the computational model and corresponding FDM model.

elements had yet to be defined, namely the smaller stalks and the sprouting flower. Nevertheless, a prototype would be useful to assess the suitability of some design decisions. The available FDM equipment allowed for the production of models that could fit inside a 20x20x30 cm envelope. In order to decrease the amount of building material, the model was produced at half scale (10x10 cm) and made hollow. Production took about 12 hours.

FDM proved to be a suitable technology for prototyping the complex geometry of the Corinthian capital, producing a fairly detailed model, and apparently resistant to shock. However, FDM prototypes can be considered anisotropic, in the sense that results differ according to the direction along which they are produced. The capital was produced along its axis' direction, which could be perceived by the layers perpendicular to it. This property is not necessarily a disadvantage. In the capital's case, however, it caused the tips of the acanthus leaves to break (Figure 10), because its upper layers were too small and, thus, very fragile. This suggested two solutions, one for the modelling phase and one for the prototyping phase.

The modelling approach comprised preventing the occurrence of acute angles on the leaf edges by extruding the outer surface along an inward vector. This solution was actually implemented on the computational model. The prototyping approach required positioning the model for production in a way that made it possible for the shape to be decomposed in larger layers. In the capital's case, however, due to its multiple predominant directions, the only option would be to prototype each leaf separately, which could be an option for larger models but redundant for this one.

3D Printing

The second iteration of the Corinthian model was produced using a different prototyping technology, 3D Printing (3DP). Compared with the previous iteration, this one was larger and geometrically more detailed; because of the higher resolution of 3DP compared to FDM, potential mistakes were expected to become more obvious.

The size of the physical model was limited to a 20x20x25 cm envelope, allowing for a 13 cm high and

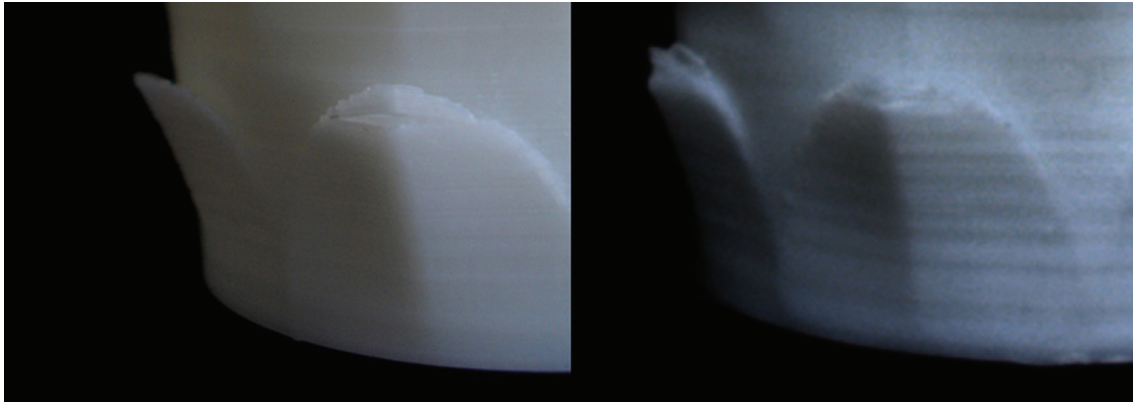


Figure 10. Broken leaf tips.



Figure 11. Second iteration of the computational model and corresponding 3DP model.

approximately 19 cm wide capital, the production of which took 8 hours.

In terms of material characteristics, the model produced in 3DP has some similarities with a ceramic object, both in the production process and in its fragility. In fact, one of the smaller stalks was broken right after the capital's production.

Despite the fragility of the material, the stalk's demise could have been prevented through geometrical options in the modelling phase. Figure 11 (on the left) shows how the stalks are sprouting out of the vase. While the larger stalks are supported by the abacus, the smaller ones have no support whatsoever, rendering them vulnerable to mechanical forces. A possible solution would be to increase their thickness as they sprout out of the vase. A more elegant solution, to be tested later is, to model the smaller stalks adjacent to the sprouting flowers.

Analysis of results

3D Printing is not very different from Fused Deposition Modelling. Both decompose the model in horizontal sections, or slices, and process each slice in sequence. However, results differ in at least three characteristics: resolution, rigidity and isotropy (Table 6). In terms of resolution, 3DP is a more precise technology. However, a model produced through FDM is more resistant to shock than the fragile 3DP models. In terms of isotropy, a model produced by 3D Printing can be considered isotropic, whereas the FDM-produced model, due to the lower resolution and to the technology itself, can be very different depending on the orientation of production layers in relation to the geometry.

Regarding the production process, 3DP revealed to be much faster than FDM, generating a larger model in 8 hours, compared with the 12 hours that took FDM to generate a model with half the size of the larger one.

Results and future developments

As shown, the computational model has been constantly upgraded. The optimization phase justifies yet another update, which is still in progress, but will render them more efficient, both conceptually and computationally. Some geometrical algorithms can also be improved.

One objective, for example, will be the perfect detailing of the acanthus leaves. As detailing progresses, gaps will need to be filled in, for which Alberti's built work, as well as his theory on musical consonances, innate correspondences, and the perfect numbers 6 and 10, will be an important and fundamental reference (Alberti, 2011, Book IX, Chapters 5 and 6).



The physical modelling process helped to gain a better understanding of how the geometry influences production quality and vice-versa, which is important when moving from prototyping on to fabrication. In fact, as mentioned above, the last physical model produced, although of satisfactory quality, is still not the final goal, which is to produce a detailed model of the capital in stone in order to emulate the original hand-crafted techniques. As such, future research will address the use of subtracting fabrication techniques, like milling and lathing (Pupo *et al.*, 2009), instead of additive prototyping ones. While the geometrical complexity of the Corinthian capital suggests the use of also complex machining techniques like 5-axis milling, lathing is expected to be suitable for simpler elements like the Doric base. Since subtractive technologies allow working with nobler materials like stone, they can become an adequate alternative to additive prototyping technologies. However, it will be interesting to see to what extent they can match the geometrical versatility and sophistication of FDM or 3DP.

In summary, the data gathered throughout the modelling and prototyping of the Corinthian capital described in this paper will be of great value for the continuation of the project, namely for the development of the computational models and the production of physical models of all the other elements of Alberti's column system.

Acknowledgements

This work is funded by FEDER Grants through COMPETE – Programa Operacional Factores de Competitividade and by National Grants through FCT – Fundação para a Ciência e a Tecnologia, as part of the “Digital Alberti” project (PTDC/AUR-AQI/108274/2008 – FCOMP-01-0124-FEDER-008842). The project is hosted by CES at the University of Coimbra and coordinated by Mário Krüger.

Table 6. Comparative analysis of physical models.

FDM	3DP	tech.	resolution	to shock	isotropy	prod time	size
		FDM	0,25 mm	resistant	anisotropic	12 hours	10x10x7cm
		3DP	0,10 mm	Fragile	Isotropic	8 hours	19x19x13

Reference

- ALBERTI, L.B. 2011. *Da arte edificatória*. Lisboa, Fundação Calouste Gulbenkian, 770 p.
- ALBERTI, L.B. 1988. *On the Art of Building in Ten Books: Trans. Engl. of De re aedificatoria*. Cambridge/Massachusetts, MIT Press, 442 p.
- CASTRO E COSTA, E.; COUTINHO, F.; DUARTE, J.P.; KRUGER, M. 2011. The Corinthian Capital according to Alberti: Generative modelling and digital prototyping of classical architectural elements. *In: Proceedings of the 5th International Conference on Advanced Research in Virtual and Rapid Prototyping, VRAP 2011*, Leiria, Portugal, 2011. *Anais...* Portugal, p. 255-262.
- MOROLLI, G.; GUZZON, M. 1994. *Leon Battista Alberti: I nome e le figure: ordini, templi, fabbriche civili: immagini e architetture dai libri VII e VIII del De re aedificatoria*. Firenze, Alinea Editrice, 247 p.
- PUPPO, R.; CELANI, G.; DUARTE, J.P. 2009. Digital materialization for architecture: Definitions and techniques. *In: E.S. NARDELLI, C.C. VINCENT (Ed.), Proceedings of the 13th Congress of the Iberoamerican Society of Digital Graphics*. São Paulo, Mackenzie Presbyterian University, School of Architecture and Urban Planning, p. 439-442.
- SCOTT, M.L. 2006. *Programming Language Pragmatics*. Amsterdam, Elsevier, 944 p.

Submetido: 27/02/2012

Aceito: 27/08/2012

Eduardo Castro e Costa

Faculdade de Arquitectura. Universidade Técnica de Lisboa.
Rua Sá Nogueira, Pólo Universitário Alto da Ajuda,
1349-055, Lisboa, Portugal.

Filipe Coutinho

Departamento de Arquitectura. Faculdade de Ciências e
Tecnologia da Universidade de Coimbra. Largo D. Dinis,
3000-143, Coimbra, Portugal.

José Pinto Duarte

Faculdade de Arquitectura. Universidade Técnica de Lisboa.
Rua Sá Nogueira, Pólo Universitário Alto da Ajuda,
1349-055, Lisboa, Portugal.

Mário Krüger

Departamento de Arquitectura. Faculdade de Ciências e
Tecnologia da Universidade de Coimbra. Largo D. Dinis,
3000-143, Coimbra, Portugal.