The Challenges of Parametric Design in Architecture Today: Mapping the Design Practice

A thesis submitted to The University of Manchester
for the degree of
Master of Philosophy (MPhil)
in the Faculty of Humanities

2012

Yasser Zarei

School of Environment and Development

Table of Contents

[Word Count: 45974]

List of Tables

List of Figures

Abstract

The thesis entitled 'The Challenges of Parametric Design in Architecture Today: Mapping the Design Practice', submitted by Yasser Zarei in 2012 to the University of Manchester for the Degree of Master of Philosophy in Architecture.

Parametric design is a new approach to architectural design based on the concept of parameters. It utilises parameters to set relations between design elements in order to define a range of formal alternatives. In this sense, parametric design provides great opportunities for architects to engineer the design process more efficiently; yet its novelty generates some challenges for architectural practitioners. The aim of this research is to explore the position of parametric design in contemporary architectural practices, identifying its advantages and disadvantages in comparison with traditional computer-aided design (CAD). Specifically, the research will compare the theoretical knowledge and the statements made by theorists and scholars of parametric design to the statements of practicing architects benefitting from the parametric approach.

This aim is achieved through three thematic parts. The first part investigates the design process through two points of view, focusing on the notions of 'role' and 'driver'. The second part identifies the position of parametric design in practice, specifically focusing on whether it is a style or just a set of techniques. Finally, the third part explores the advantages and disadvantages of parametric design and its distinctions in comparison to traditional CAD and Building Information Modelling (BIM). The research uses the qualitative method based on semi-structured interviews with practicing architects. It also benefits from a two-page questionnaire, which is carried out to get some detailed and specific data regarding the parametric realm.

The outcome of this research shows that, in spite of theoretical statements suggesting otherwise, parametric design can be undertaken without the use of computer programs. Software packages facilitate the process of design. Furthermore, the interviews conducted with architects in practice show that they believe in parametric design as a set of techniques. They do not recognise the new style 'parametricism'.

Declaration

No portion of the work referred to in this thesis has been submitted in support of an application for another degree of qualification of this or any other university or other institute of learning.

Copyright Statement

The author of this thesis (including any appendices and/or schedules to this thesis) owns certain copyright or related rights in it (the "Copyright") and he has given The University of Manchester certain rights to use such Copyright, including for administrative purposes.

Copies of this thesis, either in full or in extracts and whether in hard or electronic copy, may be made only in accordance with the Copyright, Designs and Patents Act 1988 (as amended) and regulations issued under it or, where appropriate, in accordance with licensing agreements which the University has from time to time. This page must form part of any such copies made.

The ownership of certain Copyright, patents, designs, trade marks and other intellectual property (the "Intellectual Property") and any reproductions of copyright works in the thesis, for example graphs and tables ("Reproductions"), which may be described in this thesis, may not be owned by the author and may be owned by third parties. Such Intellectual Property and Reproductions cannot and must not be made available for use without the prior written permission of the owner(s) of the relevant Intellectual Property and/or Reproductions.

Further information on the conditions under which disclosure, publication and commercialisation of this thesis, the Copyright and any Intellectual Property and/or Reproductions described in it may take place is available in the University IP Policy (see http://www.campus.manchester.ac.uk/medialibrary/policies/intellectual-property.pdf), in any relevant Thesis restriction declarations deposited in the University Library, The University Library's regulations (see http://www.manchester.ac.uk/library/aboutus/regulations) and in The University's policy on presentation of Theses

Acknowledgements

Chapter 1

Introduction

Introduction to the Research

Four years ago, Patrik Schumacher used the term 'Parametricism' to refer to a number of new trends of design in architecture based on parametrics. Before that time, parametric design somewhat resided in the margin of architectural underpinning. It was regarded as a way of tackling problems in architectural construction rather than an efficient conceptual method for form generation and testing design alternatives. Calibrating all facets of parametric design in the frame of a new style named 'parametricism' has raised more arguments; although it has also resolved many ambiguities which often stem from the absence of a framework. Yet, among the mainstream architectural practices, parametric design can hardly be recognised as a new style. Still, many practicing architects talk about the challenges of this type of design approach. Even some ambiguities emerge from those architects who benefit greatly from parametric methods, although they may not be aware of the differences of this approach to design in comparison with a traditional computer-aided approach. Similarly, inside academia parametric tools are quite popular among students of architecture. Some students like to apply parametric methods to their design projects using software packages such as Grasshopper. However, they may not always be aware of the challenges of these methods because student projects are not real design situations as experienced by practicing architects.

In the light of these facts, this dissertation investigates the context of architectural practice to offer a deeper inquiry into the parametric realm. It is essential to explain two issues here, before introducing the aim and objectives of this research. Firstly, 'parametric design' is a term employed in this research due to its frequent usage in architectural practice. It refers to the use of parameters in creation of form in the design process[1]. Other notions such as 'parametrically-enhanced design' or even shorter terms such as 'parametrics' refer to the same concept, although they are not much in use in architectural practice in comparison to 'parametric design' – the term 'parametrics' in particular refers to a similar state of meaning vis-à-vis the notion of 'parametric design'. Secondly, it is worth referring to two acronyms, CAD and BIM, since they are used in this research. Computer-aided design or CAD is simply the deployment of computer to assist the designer in design. The semantic domain of CAD is considerably broad and in one sense, it even embraces parametric characteristics. However, in this research 'traditional' CAD is actually the point of reference. It echoes the concepts of

---

[1] 'Parametric design' can also be described as the set of parameters included within design in a way that manipulation of some of parameters would result in automatic alterations of the others. To put it simply, in traditional CAD a room is designed by drawing four lines or a rectangle in which there is no defined relation among these entities. However, in parametric design it is possible to define a relation – for instance between length and width of the room, a simple equation such as L=3*W can be considered.

computer as a drawing tool and a device for representation rather than computation. In contrast to CAD, which is long-established, BIM is a new buzzword in architecture and the construction industry. It stands for 'Building Information Modelling', which literally refers to creating a comprehensive database for a building. BIM aims to facilitate the collaboration among a design team along with their engagement with other construction sectors. While the original premise of traditional CAD was merely to aid the task of drawing – for instance, walls were embodied as parallel lines – BIM platforms are object-oriented which means that all of the elements of design such as walls, doors, and windows are intelligent building objects. BIM provides a single and consistent base for all information associated with the building.

Having mentioned the above issues, the aim of this research is to explore the position of parametric design in contemporary architectural practices, identifying its advantages and disadvantages in comparison with traditional computer-aided design (CAD). The agenda of exploration includes three sets of fundamental questions which have been considered throughout the data-collection process, namely the questions of roles and drivers, the questions of style, and the questions of benefits and challenges. The first set of questions includes these sub-arguments: what is the status of primary drivers, such as the context and the project brief, in parametric design? Does the parametric approach change the role of the designer within the design process? Do the designers still use sketching and physical modelling in parametric design? Is the design process completely reliant on parametric programs? The main question in the second category inquires into the position of parametric design – whether it is a style or just a set of techniques. The last set of questions includes normative issues such as: what are the main advantages and disadvantages of parametric design? Which elements make parametric design essentially different from traditional CAD?

As a response to these questions, three objectives have been set for this research. The first objective, arising from the first set of questions, is to explore the status of primary drivers, the role of designer, the role of sketching, and the role of computer programs in the parametric design process. The second objective investigates the position of parametric design – if it is a style or a set of techniques. Finally, the last objective aims to identify the benefits and drawbacks of parametric design in today's architectural practice.

The division of this research into chapters utilises the three thematic parts emerging from research objectives in order to come up with an organised structure. Chapter 2 acts as an overture to the whole body of investigation. Providing a background to the investigation on parametrics, it attempts to show the ambivalent nature of parametric design on several layers.

Chapter 3 briefly discusses the research methodology, the participants of the research and the method of data analysis. In Chapter 4 and 5, the aim is to look at the parametric realm from two opposite directions; the first one, which may be called the 'outside to inside', focuses on the notion of 'primary driver'. Chapter 4 utilises this concept and tries to investigate it in relation to the parametric approach. In contrast, Chapter 5 employs a reverse viewpoint; the investigation looks from 'the inside to the outside' and here the concept of 'role' instead of driver is explored within the design process. Having considered architectural design as a complex and uncertain process that needs an insider to spell it out, the most important roles are explained and mapped based on interviews with architects and supporting statements drawn from secondary sources. These two chapters are also important in terms of explaining possible changes in the parametric realm. Chapter 6 seeks the position of parametric design in architectural practice to explore to what extent architects believe that this novel approach to design can be considered a style. In this respect, the words and statements of Patrik Schumacher are further explored and analysed. Chapter 7 attempts to take a normative position on parametrics by discussing its advantages and disadvantages, and its distinction in comparison to traditional CAD and Building Information Modelling (BIM). Finally, Chapter 8 concludes the research and recommends further areas for research into the realm of parametric design.

Chapter 2

The Position of
Parametrics

This chapter investigates some major arguments and statements in the literature addressing parametric design. In this sense, it tries first to contextualise the parametric approach to design by providing a brief background on parametrics. Following this, it explains the ambivalent nature of parametric design and attempts to argue where, when, and how this ambivalence happens. These ambivalent positions are significant because they cast doubt on investigation so far in the domain of parametrics and as a result furnish the realm of inquiry with a set of questions worth exploring.

2.1. The State of Knowledge on Parametrics

Investigation into the position of parametrics often includes going into two bodies of research; one related to the digital domain arising from technological advances and the other exploring the methodological issues regarding design activity. At the general level of inquiry, one may ask how to adopt digital technology in the design methodology. However, with a more specific view one may question to what extent parametric design offers a methodological standpoint. Both of these questions are still too broad and, as a result, many 'why' and 'how' questions might emerge from them that betray the lack of clarity in the parametric realm.

For Scholars like Antoine Picon, digital technology is defined as a culture, since it has changed lifestyles all over the world. Digital culture is an ever-present influential phenomenon at various levels, from identification of individuals to the location of a current activity. According to Picon, the question now is no longer whether digital technology is good or bad; it is more about the direction architecture is taking under its influence[1]. Many researchers, including Mario Carpo, assume the digital as a paradigm[2] which considerably affects domains such as architecture not only in terms of design thinking and the process of creation, but also in the production and manufacture of buildings. Monitoring the world of design specifically in recent years demonstrates new trends arising from symbiosis with digital artefacts. With the aid of the digital, architects now foster their ideas and, more importantly, create new spaces. As Peter Zellner writes, 'architecture is becoming like 'firmware', the digital building of software space inscribed in the hardware of construction'[3].

Although the origins of parametric design go back to no more than fifteen years ago, thinking towards such approaches in architecture has a longer history back to when the first attempts for design simulation by computers began. Computers provide two grounds of investigation like any other digital artefact: theoretical underpinnings, and practical implications and ramifications. In the practical domain, efforts were more focused on the capabilities of a computer program as a benefit to the design process. As a result, the concern here was more on enhancement and evolution of such systems. According to Robert Woodbury, the first computer-aided design system was parametric[4], programmed by Ivan Sutherland for his PhD thesis on Sketchpad[5] in 1963. Sutherland's platform was one of the first

---

[1] Antoine Picon, *Digital Culture in Architecture : An Introduction for the Design Professions* (Boston, MA: Birkhaeuser, 2010).
[2] Mario Carpo, *The Alphabet and the Algorithm* (Cambridge, Mass. ; London: MIT Press, 2011), p. 169.
[3] Peter Zellner, *Hybrid Space : New Forms in Digital Architecture* (London: Thames & Hudson, 1999), p. 13.
[4] Robert Woodbury, *Elements of Parametric Design* (London: Routledge, 2010), p. 11.
[5] Sutherland's Sketchpad, as the first CAD system at that time, was innovative. Yet the problem was, instead of lines of codes or commands written by users on the computer screen, Sketchpad utilised several buttons to apply commands such as move, copy, and paste. Obviously for many design projects more buttons were needed. Nevertheless, it could be said that most of the CAD

attempts at implementing a concept that became central to many parametric packages – that of the concept of 'constraint'. In general, a constraint explains a relation between two or more objects; for example, it restrains a group of lines to be parallel or perpendicular, even it can define a relation inside an object between features such as the size of the diameter or the area. Normally, two families of constraints are set within the design process: geometric constraints and physical constraints. The evolution of these two types of constraints specifically enhanced many computer platforms.

On the theoretical side, the most important question revolved around the possibility of having a system that was able to design without the supervision of a human. In this context, some thinkers, such as Rittel, stressed that design in the sense of forming judgments can never be simulated by a computer, because the designer has to imagine all possible solutions before the computer program runs[6]. Still, as a theory-oriented concept, researchers such as Christopher Alexander tried to introduce the concept of 'pattern' and 'a pattern language'[7] in design in order to provide a common standpoint in both the design and the computer realm. Nevertheless, the limitations in the definition of patterns themselves were a serious challenge. Some of the researchers tried to go deeper into the design methodology, proposing cognitive models of design. For instance, Peter Rowe, in his book *Design Thinking*, introduced design as an activity occurring between problem-space and solution-space. He tried to use concepts such as 'decision trees' in order to come up with an epistemic model of design. A 'decision tree' is an abstract structure which depicts a problem-space by the aid of 'nodes' representing decision points and 'branches' representing courses of action[8]. Rowe's rationalistic model of design is interesting in the sense that it can picture the difference between parametric design and conventional design using CAD packages or pen and paper. Whereas in conventional design decisions have a static aspect, in parametric design all of the actions are parametrically related. As a result, the decision tree would have a dynamic feature that makes the whole of the design controllable and the ultimate form malleable.

---

developers pursued Sutherland's concept in creating and developing CAD platforms. The following link explains how Sketchpad works: Bigkif, 'Ivan Sutherland : Sketchpad Demo ', <http://www.youtube.com/watch?v=USyoT_Ha_bA&list=PL23D79E83F8F52102&index=9&feature=plpp_video>, accessed 10 April 2012.

[6] H.W.J Rittel, 'Second-Generation Design Methods', in Nigel Cross (ed.), *Developments in Design Methodology* (Chichester: Wiley, 1984), p. 323.

[7] Christopher Alexander, Sara Ishikawa, and Murray Silverstein, *A Pattern Language : Towns, Buildings, Construction* (New York: Oxford University Press, 1977).

[8] Peter G. Rowe, *Design Thinking* (Cambridge, Mass.: MIT Press, 1987), p. 65. The phrase 'backtracking' is used to refer to the feature of reversibility within the design process. According to Rowe, 'backtracking is most clearly illustrated in relation to a decision tree. The initial pass through the problem space is given by the link-node sequence a1, b1, c2, d3, e1. When a difficulty is encountered at node e1, backtracking occurs to the point c2 before a forward pass is made through the next nodes.

In a broader sense epistemic models such as Rowe's emerged from two[9] paradigmatic stances concerning design activity. The first considers design as a process of problem-solving and the second defines it as a process of reflection-in-action. Kees Dorst and Judith Dijkhuis explain these two paradigms by describing the position of designer, the type of design problem and design process, and the domain of design knowledge. In the first paradigm, design is a process of rational problem-solving and the designer is an 'information processor' looking at ill-defined and unstructured problems with a lens of positivism. In the second paradigm, by contrast, every problem is 'essentially unique', a 'universe of one', which is viewed through a constructionist position by a person constructing his or her own 'reality', facing it with a process of reflection-in-action. While in the first outlook designers take 'classical sciences like physics as the model for a science of design', in the second view designers use their artistry to shape the design in reality[10]. The first paradigm entails the acceptance of positivism as the unique doctrine that leads to the problem-solving position[11]. In contrast, the second paradigm encourages researchers to build up a new tenet by borrowing principles from philosophies such as pragmatism. For thinkers like Donald Schön, who was also the main theorist of 'reflection-in-action'[12], design is not just about problem-solving; it is often about how to find and set the problem as well.

In addition to those paradigmatic taxonomies in the design methodology, for some researchers the crucial concern was how to map different factors in the design process rather than proposing an overall paradigmatic picture. Interestingly, scholars like Bryan Lawson used the concept of 'constraint' in order to come up with a cognitive model for the design process. Lawson, in his book *How Designers Think*[13], discusses the role of client, user and legislator, who respectively have a major influence on the final scheme. He explains that constraints can be established internally or externally. Some of them are dictated to architects from the environment and the location of the design project. Some also exist in the process implicitly. Finally, Lawson illustrates four categories of constraints that connect every design to its social

---

[9] Daniel Fallman (see: Daniel Fallman, 'Design-Oriented Human-Computer Interaction', *Proceedings of the SIGCHI conference on Human factors in computing systems* (Ft. Lauderdale, Florida, USA: ACM, 2003), 225-32.) introduces three categories of paradigmatic stances which are the conservative, the romantic, and the pragmatic. He explains the rational problem-solving or the systematic method of designing as a conservative view to design in which the designer is seen as 'glass box'. In comparison, he introduces the second paradigm as a pragmatic account which is a reflective conversation with the materials of the design situation. In this view, the designer is a 'bricoleur', 'someone who makes with what is available or encountered in a specific situation' (p. 227). Fallman presents the romantic account in which the character of the designer and the features of the design process are completely mystical. Here, the designer is a 'black box', 'someone who is able to generate creative designs but not able to, or at least not interested in, explaining how they came about'. Nevertheless, the description of Fallman for the third paradigm does not offer a new theme for investigation. Presenting the aspects of the third account as mystical is equal to nothing philosophically.

[10] Kees Dorst and Judith Dijkhuis, 'Comparing Paradigms for Describing Design Activity', *Design Studies,* 16/2 (1995), 261-74.

[11] Herbert A. Simon, *The Sciences of the Artificial* (3rd ed. edn.; Cambridge, Mass. ; London: MIT Press, 1996).

[12] Donald A. Schön, *The Reflective Practitioner : How Professionals Think in Action* (Aldershot: Avebury, 1991), p. 374. Schön uses the term 'technical rationality' to explain that a positivist epistemology will be resulted in a process of problem-solving.

[13] Bryan Lawson, *How Designers Think : The Design Process Demystified* (4th ed. edn.; Amsterdam ; London: Architectural, 2006).

context to propose his cubic constraint-based model. These items are symbolic, formal, practical, and radical; they shape the ultimate product and create limitations in each stage of the design activity. It could be said that Lawson's model points to a considerably larger domain of ideas on constraint than is often acknowledged in parametric design. Constraints in the parametric realm often have a mathematical nature and characterise a set of functions rather than displaying a postulated pivot. Maybe this is the reason why a parametric methodology is still overwhelmingly expected among architects. However, it is worth mentioning that the experiences of working with constraints and the evolution of ideas within this axial concept have resulted in many changes. As Robert Aish, a quintessential thinker in computer-aided design interestingly explains, now the parametric packages are being rediscovered based on constraint systems just like the rediscovery of 'concrete several centuries after the pantheon' was constructed[14].

Another significant parametric aspect, especially in relation to constraint, is the concept of 'relation', which is seen as a promising theme for many researchers. According to Woodbury, the difference between parametric design and conventional design is the capability to make relations among the design objects. Defining relationships has not previously been considered as part of design thinking, since the conventional defined activities in design were 'add and erase'. However, in parametric packages, designers benefit from two extra capabilities, namely 'relate and repair'[15]. Through this perspective, the idea of constraint is epitomised in the idea of relation.

More promising themes are found in the works of Patrik Schumacher, the theorist of parametricism[16], specifically his recent book *The Autopoiesis of Architecture*[17]. Schumacher aims to extend the theoretical positions of parametric design in the frame of the new style parametricism. For him, parametricism is 'the' unique style after modernism; the other styles have been just a metamorphosis of modernism through time. Schumacher describes parametricism as:

a new style rather than merely a new set of techniques. The techniques in question – the employment of animation, simulation and form-finding tools, as well as parametric modelling and scripting – have inspired a new collective movement with radically new ambitions and values[18].

---

[14] Robert Aish, 'Designing at T + N', *Architectural Design,* 81/6 (2011), 20-27, p. 25.

[15] Robert Woodbury, *Elements of Parametric Design* (London: Routledge, 2010), p.24.

[16] Patrik Schumacher, 'Parametricism: A New Global Style for Architecture and Urban Design', *Architectural Design,* 79/4 (2009a), 14-23

[17] Patrik Schumacher, *The Autopoiesis of Architecture, Volume Ii: A New Agenda for Architecture* (John Wiley & Sons, 2012). And also the first volume: Patrik Schumacher, *The Autopoiesis of Architecture: A New Framework for Architecture* (Wiley, 2011b).

[18] Patrik Schumacher, 'Parametricism: A New Global Style for Architecture and Urban Design', *Architectural Design,* 79/4 (2009a), 14-23, p. 15.

Schumacher's position will be discussed later on in this research in more detail. However, it is worth mentioning the final promising area of inquiry, which arises from a broader domain of computation. If design is described as the act of computation, and not representation with pen and paper or on computer screen, then design methodology has to be reframed and executed in a new way. For some researchers, such as Carpo, this is reminiscent of the 'cybernetic architecturology'[19] of the seventies, which is now regenerating in another format. Therefore, the current question is 'how' to design by employing algorithms instead of sketching and making physical models. The promising themes in parametrics, as well as concerns such as definition or clarification of a parametric methodology, provide evidence that there are still many unclear aspects worth exploring. In this sense, the next section of this chapter discusses the ambivalent positions that parametric design reflects.

---

[19] Mario Carpo, *The Alphabet and the Algorithm* (Cambridge, Mass. ; London: MIT Press, 2011), p.35.

2.2. The Ambivalent Nature of Parametric Design

The nature of parametric design is ambivalent, and I believe this ambivalence can be indentified in four important areas. The aim of this section is to explain and clarify these areas. The first emerges from an analysis of the term 'parametric design' itself. Etymologically, parametric design is a paradoxical term. The adjective 'parametric' refers to 'parameter', which originates from the Greek 'para', meaning a subsidiary or assistant, and the word 'metron', which means 'measure'[20]. This in turn opens two windows on its meaning – one which is particularly mathematical, reflecting a measurable factor that defines a system or sets the conditions of its operation, and the other which is more general, describing the boundary and the scope of a specific process or activity. Briefly, 'parametric' can be defined as something precise.

In contrast, the meaning of the word 'design' is diametrically opposed to this concept of precision. While its origin comes from the Latin word 'designare', which means 'to designate'[21], it often implies an unpredictable or uncertain activity dealing with quite 'ill-defined' problems. The nature of design stresses the impossibility of definitive formulations. Hence, some researchers such as Peter Rowe argue that, in order to develop a general portrait of design thinking, designers should be watched at work where they have to 'move back and forth between the problem as given and the tentative proposals they have in mind'[22]. Nigel Cross uses the term 'designerly way of knowing'[23] to emphasise this unique feature of design. In addition, the nature of design often forces designers to think more about how to find and 'set the problem'[24] instead of focusing on how to solve it. Yet, for the skilful practitioner, the act of setting often greatly benefits from an extensive repertoire of experiences. Therefore, on the one hand, we can observe the irregular nature of design and on the other hand, we see the precision and rigor of the term 'parametric'. This demonstrates that, by exploring the semantic layers of the words, parametric design describes an amphibological state; a phrase that contradicts itself.

The lexical contradiction between the two words 'parametric' and 'design' provides the prelude to a more important domain, which can be termed the second ambivalent area of parametrics: the relationship between mathematics and architecture. On this ground, a

---

[20] Oxforduniversity, *Oxford Dictionary of English* (Oxford: Oxford University Press, 2010).
[21] Ibid
[22] Peter G. Rowe, *Design Thinking* (Cambridge, Mass.: MIT Press, 1987), p. 2.
[23] Nigel Cross, 'Designerly Ways of Knowing: Design Discipline Versus Design Science (the 1920s and the 1960s, Two Important Periods in the Modern History of Design)', *Design Issues,* 17/3 (Sum 2001), 49-55.
[24] Donald A. Schön, *The Design Studio : An Exploration of Its Traditions and Potentials* (London: RIBA Publications for RIBA Building Industry Trust, 1985).

historical perspective appears in the concept of 'parameter', recalling the revitalisation of mathematics in architecture, since for many architects mathematics was previously seen as an isolated pure science. This revival demands postulation of a roadmap on which a harmony of engineered precision can be plotted. In one sense, the architectural concern is often about the creation of space, employing mathematics as a tool for its definition and illustration. However, the ambivalence we have identified arises from precisely this position. Architecture has always been a realm with rather clear objectives but normally without guiding rules. Mathematics, by contrast, is largely described as a 'form of knowledge best understood as a game with lots of rules, but no clear objective'[25]. Therefore, scholars including Antoine Picon suggest that mathematics should be reduced to two positions: mathematics as a theoretical foundation, and mathematics as a tool[26]. Perhaps the use of mathematical tools in design by the current regime of architecture shows that the focus is now more on practical implications rather than theoretical underpinnings. Yet, seeing mathematics as an ideological impulse for design has also been significant in recent years. Nonetheless, the root of the mathematical ideology can perhaps be traced to three decades ago, when a dramatic rise in deploying digital tools coincided with a transition from Derrida to Deleuze in philosophy[27]. This was not only the outset of parametric design and other approaches to architecture like folding; it also ushered in a new era in which monadic thinking was overwhelmingly on the spot. It is surprising that the creator of 'monad', Gottfried Leibniz, was also the inventor of calculus along with Isaac Newton. Both of these concepts, monad and calculus, have been the axial characters of the new digital approaches, including parametrics.

The concept of monad has remained at the very heart of architectural theory. For instance, the term 'monadology' is frequently used in domains such as genetic architecture, referring to the idea of 'extension and transformation of some of the propositions, especially those that define attributes and properties of relationships among monads'[28]. Referring to Mies van der Rohe's quote that architecture is the 'art of putting two bricks together', Karl Chu interestingly employs the term 'bit', which in one sense can replicate the concept of monad in computation. In Chu's view, architecture now is the 'art of putting two bits together, at least

---

[25] Brett Steele, 'Weapons of the Gods', in Jane Burry and Mark Burry (eds.), *The New Mathematics of Architecture* (London: Thames & Hudson, 2010), p. 6.

[26] Antoine Picon, 'Architecture and Mathematics: Between Hubris and Restraint', *Architectural Design,* 81/4 (2011), 28-35.

[27] Anonymous, 'A Conversation between Sanford Kwinter and Jason Payne', in T. Sakamoto and A. Ferré (eds.), *From Control to Design: Parametric/Algorithmic Architecture*
(Actar-D, 2008), p. 226.

[28] Karl Chu, 'Metaphysics of Genetic Architecture and Computation', *Architectural Design,* 76/4 (2006), 38-45, p. 42. Chu explains that 'Leibniz's Monadology is arguably the earliest endeavour to propose what is now known as an open-source architecture based on the principles of philosophical genetics: the principle of generative condensation, the principle of combinatorial expansion, and the principle of the conservation of information'.

bits that are programmed to self-replicate, self-organise and self-synthesise into evermore new constellations of emergent relations and ensembles'[29]. Whereas monad plays a role in the theoretical levels, the term 'calculus' appears in more practical grounds inside the structure of computer tools to generate architectural form. For some architects, the creation of form in architecture now totally depends upon calculus. For example, Greg Lynn explains his experience of calculus-based form:

> I realised that contemporary software could be animated because the geometric engines such packages were using were calculus based. You could move an object and then interpolate a whole collection of variables into infinitesimally smaller steps. The points in space were fluid due to the calculus relationships of variables, and this was giving form and shape to the models. I initially, in Animate FORM, focused my thinking on the revolution in motion, and only later realised that the real revolution was in the use of a 300-year-old invention: calculus[30].

While for Lynn calculus is assumed as an instrumental vehicle in the process of form generation, for Picon it is a fact which has estranged architecture from mathematics and endures to the present day[31]. Calculus marginalised the use of arithmetic and geometry in design as well. More broadly, in Picon's term, the computer is a 'veil' that hides the presence of mathematics. This sounds noticeable in contemporary architectural practice and I would argue that this is where the third area of ambivalence is found. The presence of computers in the design process has always been accompanied with a big question mark for those who ask about its status. The main category of the question being asked falls within the ambit of the influentiality of computer programs in general and parametric packages in particular. This discussion has probably emerged from the fundamental question of 'what is the role of the computer in the design process?' Is it a useful tool in the toolbox, which significantly facilitates the workflow? Or, is it firmly linked to the design process in a way that means its absence would paralyse the whole design activity? Many researchers still discuss this hotly disputed topic. For some thinkers, accepting the current parametric software and scripting techniques as design tools is a profound error because these tools are just for realisation or resolution[32]. In contrast, for theorists such as Patrik Schumacher, parametric programs define a new perspective on design, which can only be characterised as a new architectural style called parametricism[33]. According to Schumacher, every style is composed of methodological rules, dictating what paths of research to avoid (negative heuristics) and what paths to pursue (positive heuristics). Negative heuristics or 'taboos' in parametricism are summarised as

---

[29] Ibid

[30] Ingeborg M Rocker, 'Calculus-Based Form: An Interview with Greg Lynn', *Architectural Design,* 76/4 (2006), 88-95, p. 90.

[31] Antoine Picon, 'Architecture and Mathematics: Between Hubris and Restraint', Ibid.81 (2011), 28-35, p. 30.

[32] Anonymous, 'A Conversation between Sanford Kwinter and Jason Payne', in T. Sakamoto and A. Ferré (eds.), *From Control to Design: Parametric/Algorithmic Architecture* (Actar-D, 2008),p. 233. Kwinter also explains that his feeling is not that parametrics represent a false path; it is that the new generation of architects have the wrong tool. (Ibid, p. 237)

[33] Patrik Schumacher, 'Parametricism: A New Global Style for Architecture and Urban Design', *Architectural Design,* 79/4 (2009a), 14-23.

avoiding rigid geometric primitives such as squares, triangles and circles, simple repetition of elements or juxtaposition of unrelated elements or systems. On the other hand, positive heuristics or 'dogmas' consider all forms to be parametrically malleable, to be able to inflect and correlate systematically, and to have the capability of gradual differentiation[34]. For Schumacher, negative heuristics articulate 'strictures' that prevent going back to what is not fully consistent with the parametric core, while positive heuristics furnish a rubric with principles or techniques for working in a particular direction.

In one sense, it seems that parametricism can resolve the paradoxical phrase 'parametric design' in terms of the meaning, especially as its techniques and methods may affect the nature of design and, as a result, revolutionise design thinking. Yet, treating parametricism as a new style is controversial and must be proceeded by a broad study of architectural practice. However, going back to the third ambivalent position, certain methods or terms often increase this ambivalence. Computer programs are normally working through a series of algorithms. Contrary to design problems, which have an ill-defined nature, an algorithm must be well- defined. Sometimes, the outcomes of the algorithm can be probabilistic, but it is not because the algorithm is not set and programmed properly; it is because the employed rules or functions within the procedure of programming have a stochastic feature. This is the point where, for researchers such as Kostas Terzidis, an algorithm can be illustrated as, 'a theoretical construct with deep philosophical, social, design, and artistic repercussions' rather than only a computer implementation or a series of lines of code[35]. The importance of algorithms in the parametric realm may recall the term 'Objectile', another reference to the Deleuzian philosophy[36]. It seems that the kernel of this term still resides in the theoretical ground, even though somewhat different interpretations appear in academic books and papers. In order to explain objectile, Deleuze uses a geometrical idea; a series of curves, the related parameters and variables, and the reduction of variables to a 'single and unique variability' or 'the fold'[37]. One of the clearest messages of objectile is the idea of variation. Objectile differs from but is sometimes confused with object. According to

---

[34] Ibid, p. 16

[35] Kostas Terzidis, *Algorithmic Architecture* (1st ed. edn.; Amsterdam ; London: Architectural Press, 2006), p. xii.

[36] Gilles Deleuze, *The Fold : Leibniz and the Baroque* (Athlone P., 1993).

[37] Ibid, p. 19-20. The goal of the objectile as Deleuze explains by his example is no longer defined by 'an essential form, but reaches a pure functionality, as if declining a family of curves, framed by parameters, inseparable from a series of possible declensions or from a surface of variable curvature that it is itself describing… this is a very modern conception of the technological object. it refers neither to the beginnings of the industrial era nor to the idea of the standard, but to our current state of things, where fluctuation of the norm replaces the permanence of a law; where the object assumes a place in a continuum by variation; where industrial automation or serial machineries replace stamp forms. The new status of the object no longer refers its condition to a spatial mould – in other words, to a relation of form-matter – but to a temporal modulation that implies as much the beginnings of a continuous variation of matter as a continuous development of form'.

Mario Carpo, objectile is a machine or 'an algorithm'[38]. For Picon, objectile is the designator of the capacity of calculus to 'generate an infinite number of objects as elements of a continuous series'[39]. It seems that Picon imagines objectile as a bridge, linking the theoretical concepts of monad to the practical ideas of calculus. Objectile is quite relevant to the parametric realm in two aspects. First, it is not an object, but a set of rules that generates objects. Thus, it can define several alternatives for a problem and, in this sense, it resonates with the task of parametric design. In addition, these sets of rules must be linked together. Therefore, the concept of 'relation'[40] comes forward, which is central to the domain of parametrics.

The ambivalence in the usage of computers, as mentioned, is not limited to terms or theoretical positions. According to Picon, it is evident in the entire body of digital architecture. For Picon, the vagueness of the term 'digital architecture' has been increased further by the series of offices that have pioneered the use of computer-aided design where the senior partners in fact have little familiarity with computer programs. Picon extends his argument to the entire domain of the digital, using the term 'digital culture' in order to explain the enigmatic character of this domain[41].

In addition, the bifurcation in the nature of the mathematics-architecture relationship as discussed before is also largely present here in the domain of the digital. Researchers therefore look at this realm of inquiry from two dominant viewpoints: digital technology as a context with cultural or sociological issues in liaison with architecture, and digital technology as a collection of different tools being used in the design process, playing an epistemic or ontic role. Many of the researchers in the first category have sought the impacts of the digital realm on architectural design by looking at architecture through an interdisciplinary lens or a macro level of inquiry[42]. Conversely, the other group of thinkers has focused on the role of digital tools when they are added to a part or whole of the process of design to supply an action or enhance a capability of practice. This approach is a microscopic view[43] that investigates the

---

[38] Mario Carpo, *The Alphabet and the Algorithm* (Cambridge, Mass. ; London: MIT Press, 2011), p. 40. 'The objectile is not an object but an algorithm – a parametric function which may determine an infinite variety of objects, all different (one for each set of parameters) yet all similar (as the underlying function is the same for all)'.

[39] Antoine Picon, *Digital Culture in Architecture : An Introduction for the Design Professions* (Boston, MA: Birkhaeuser, 2010), p. 74.

[40] Javier Monedero, 'Parametric Design: A Review and Some Experiences', *Automation in Construction,* 9/4 (2000), 369-77, p. 371. parametric design 'implies the use of parameters to define a form when what is actually in play is the use of relations'.

[41] Antoine Picon, *Digital Culture in Architecture : An Introduction for the Design Professions* (Boston, MA: Birkhaeuser, 2010), p. 60. According to Picon, Frank Gehry's firm begins every project with traditional techniques of design such as sketching and cardboard modelling. However, we should notice that Gehry' office has been one of the first practices utilised a CAD program (CATIA) in design. They employed both traditional model-making and computer modelling in many projects. Concerning CATIA, Gehry explains that 'we have got powers that architects do not normally have'. See: 3dscatia, 'Franck Gehry Testimony | Key Role of Catia for His Architecture Creations ', <http://www.youtube.com/watch?v=C0WnjbsTT9Q>, accessed 06 August 2012.

[42] See for instance: Carpo 2001, 2008; Brey 2005; Yaneva 2005; Janssen 2006; Picon 2006, 2010; Latour 2007; Houdart 2008; Latour & Yaneva 2008; Guy & Karvonen 2011.

[43] See for instance: Suwa & Tversky 1997; Koutamanis 2005; Knight et al 2005; Darses et al 2008; Dokonal & Knight 2008.

details of the design process critically to find its shortcomings by considering devices such as digital pen and paper, digital gloves and tactile screens or apparatus used in Augmented Reality (AR). Therefore, the ambivalence in the use of computer programs, which contributes to the digital culture, may be further expanded and defined as a new paradigm. For thinkers like Charles Jencks, this paradigm 'stems from a longer cultural shift, a change in worldwide, in religion, perhaps politics and certainly science'[44]. Carpo uses the word 'revolution' and explains that projects such as Frank Gehry's buildings (particularly the Guggenheim Bilbao) brought the digital turn to the architectural forefront. He concludes that digital technologies did trigger an architectural revolution. Yet, for Carpo, Gehry's case is misleading, since Gehry's design at the time of Bilbao was based largely on handmade, sculptural models[45]. However, it seems that such a turn was sufficient for Jencks to coin the term 'Bilbao Effect'[46], although Jencks's argument was completely different from Carpo's reference to Bilbao. The digital turn also helped architectural practitioners to think about how to begin designing from the very start by using digital tools. Therefore, in this sense, parametric or algorithmic approaches became a serious ground for investigation. The attempts here have been considerably focused on non-deterministic design processes and the 'concept of formation'[47] rather than the concept of form. The concept of formation provides the avenue to the last ambivalent area: materiality and parametrics.

Materiality for many scholars is now another state-of-the-art topic, both on the engineering and technical side, and a theoretical base for contemplation. Some philosophers, such as Manuel De Landa, argue that the root of this approach to materials and materiality originates from Deleuze and his philosophical impact on other disciplines, including architecture[48]. Moving the current trend of architecture towards performative design proves De Landa's statement. Materiality is not just about the aspects of materials. It is about how

---

[44] Charles Jencks, *The New Paradigm in Architecture : The Language of Post-Modernism* (7th ed. edn.; New Haven ; London: Yale University Press, 2002), p. 1.

[45] Mario Carpo, *The Alphabet and the Algorithm* (Cambridge, Mass. ; London: MIT Press, 2011), p. 37.

[46] Charles Jencks, *The Iconic Building : The Power of Enigma* (London: Frances Lincoln, 2005), p. 199.

[47] Rivka Oxman, 'Theory and Design in the First Digital Age', *Design Studies,* 27/3 (2006), 229-65, p. 249.

[48] Columbiauniversity, 'Manuel De Landa on Deleuze and the Use of Genetic Algorithm in Architecture', <http://www.youtube.com/watch?v=50-d_J0hKz0>, accessed 20 May 2012. For De Landa, the novelty of Deleuze is creating a new materialism which gets rid of essences; because 'Essentialism says that matter and energy are inert. They cannot give rise to new forms on their own. Idealism is a form of essentialism'. According to De Landa, for establishing this new materialism, Deleuze borrows three concepts from science: 1. Population thinking from evolutionary biology: this is seen in genetic algorithm which is based on 'searches'. De Landa explains that 'collectivity searches blindly the spaces of possibility'. 2. Intensive thinking from thermodynamics based on two concepts of intensive and extensive properties of the materials: extensive properties like length and volume can be divided into spaces, while intensive properties like pressure, speed, and temperature cannot. For example, when we have a 90°C gallon of water, dividing the gallon into two smaller gallons does not reduce the temperature of water to 45°C. This is the physical definition, but Deleuze delves into these definitions more. Deleuze says: 'the intensive differences drive the processes'. De Landa explains that 'genetic information depends on the existence of intensive differences, in order to keep the process going'. 3. Topological thinking from mathematics. Deleuze refers to the differential calculus and the rapidity or slowness of a curve in changing throughout its path.

those materials are perceived by human senses. Hence, it is positioned at the intersection of two diametrically opposed categories: one that is 'totally abstract, based on signals and codes', and another that is 'ultra-concrete, involving an acute and almost pathological perception of material phenomena and properties'[49]. The dilemma occurs when the world of codes tries to depict the world of materials. In the real world, materials, internal constraints and external forces are connected. However, in the virtual space of programming, forms, materials and forces are seen as separate entities. Material here is an aspect that should be assigned to the form. It is not an intrinsic character, but an augmented value. Some researchers argue that design computation provides 'the possibilities of integrating physical properties and material behaviour as generative drivers in the design process'[50]. Although the idea of integration is being seriously followed by the programming industry – we can see Building Information Modelling (BIM) as one of the recent outcomes – the fact that the majority of computer packages still follow the traditional framework of material definition casts doubt on such concepts. In addition, another reason for the inappropriate picture of materiality in computer packages is the fact that many architects discuss the attribute of tool-making in this context. It seems that tool-making is a critical skill for today's practice to enhance or even introduce the material properties which are not programmed in the body of software by developers.

Materiality also brings another issue, stemming from how the architectural form is generated. It reminds architects to be cautious of their own obsessions on material. The important thing to remember is that, when architects try to translate these concerns into the language of parametrics, they should take into account materiality as the main argument, employing parametric methods to clarify it. Any other attempt such as shoehorning materials between other features of parametric programs (which is seen more or less in most of the parametric packages now) can be a diversion from the material origin, weakening both the material aspects and the parametric framework. This is the state that Sanford Kwinter refers to as the 'parametric blanket'[51], which I would argue it is the correct way to describe it; indeed, I believe that parametric ambivalence culminates in this featureless veneer.

---

[49] Antoine Picon, *Digital Culture in Architecture : An Introduction for the Design Professions* (Boston, MA: Birkhaeuser, 2010), p. 157.

[50] Moritz Fleischmann et al., 'Material Behaviour: Embedding Physical Properties in Computational Design Processes', *Architectural Design,* 82/2 (2012), 44-51, p. 45. 'Material behaviour computes form. In the physical world, material form is always inseparably connected to internal constraints and external forces; in the virtual space of digital design, though, form and force are usually treated as separate entities…' (p. 44).

[51] Anonymous, 'A Conversation between Sanford Kwinter and Jason Payne', in T. Sakamoto and A. Ferré (eds.), *From Control to Design: Parametric/Algorithmic Architecture*
(Actar-D, 2008), p. 235. 'Parametric software and the script-generated continuums of contemporary digital work have a material quality of which something physical as well as metaphysical could be said. What I used to call the 'parametric blanket' (largely because these works resemble a featureless blanket thrown over a highly articulated traditional workshop model) has nonetheless

2.3. Parametric Design and the Ambiguity of Taxonomy

Aside from the ambivalent features discussed in the previous section, there is an ambiguity in the position of parametrics in relation to computational positions. It perhaps comes from the fundamental question of what the aim of computation is. To answer it, one must first define what computation is. There are some rather naive definitions for 'computation', which just make its meaning more complicated rather than clarifying it. However, some researchers often define computation by differentiating it from other similar topics, notably computerisation[52]. Methodologically, computation has an exploratory characteristic, often about ill-defined processes, while computerisation involves working with material that has already been preset and as a result is well defined. Computation adds a sort of discovery to design, which may be quite novel for many designers – this is the place where it can also be problematic, because designers often hold 'an ethos of rationalistic determinism'[53]. Nevertheless, computation can bring a new approach to design in which the objects are calculated. It is different from traditional design, in which architects usually represent what they have imagined and shaped in their minds. Architects like Antoni Gaudi or Frei Otto were using a sort of natural computation to produce architectural form and, in this sense, their methods were not tied to computer programs. However, computation in the way that is being discussed today is often equivalent to the use of a computer. Computation often requires the knowledge of programming, which in turn demands knowing about the 'lexicon of allowable words' and 'a way of combining the elements legally'[54], also known as syntax. However, apart from this primary discussion over the aim of computation, one might ask: 'what is the state of parametrics here?' So far, few researchers have tried to crystallise the parametric position among computational approaches. We can suggest two possible reasons for this: either the domain is still under investigation, so there is nothing to say clearly, or computation is quite broad and, as a result, not classifiable. Both of these reasons seem logical. For more clarification, it is important to refer to two taxonomies in this regard.

Toni Kotnik interestingly explains the structure of computational approach by borrowing the term 'Turing machine'[55]. He explains that every computer-aided design follows

---

the materiality that could sustain discussion, a history that is at the very least interesting (dating, no doubt, from Kipnis's and Shirdel's work at the AA in the early 1990s)'.

[52] Kostas Terzidis, *Algorithmic Architecture* (1st ed. edn.; Amsterdam ; London: Architectural Press, 2006).

[53] Kostas Terzidis, *Expressive Form : A Conceptual Approach to Computational Design* (London: Spon, 2003), p. 6.

[54] Paul Coates, *Programming.Architecture* (Abingdon: Routledge, 2010), p. 2.

[55] 'A Turing machine is a device that manipulates symbols on a strip of tape according to a table of rules. Despite its simplicity, a Turing machine can be adapted to simulate the logic of any computer algorithm, and is particularly useful in explaining the functions of a CPU inside a computer'. It was coined by Alan Turing in 1936. Turing machines help computer scientists understand the limits of mechanical computation. Source: Wikipedia, 'Turing Machine ', <http://en.wikipedia.org/wiki/Turing_machine>, accessed 20 March 2012.

a finite collection of Turing machines $\{T_1^{cad}, \ldots, T_m^{cad}\}$, each one 'mediating between possible user input and consequential output'. These are typically in graphical form, and displayed on the screen with every available tool of the software. Furthermore, the concatenation of these machines in turn generates new Turing machines. Kotnik argues that this theoretical model also 'constitutes the basis of the latest generation of CAD software, like Bentley's Generative Component or the Grasshopper plug-in for Rhinoceros'[56]. Exploring the framework of Grasshopper and listening to the words of its creator, David Rutten, somewhat clarifies Kotnik's argument[57]. Furthermore, by the aid of Turing machines, Kotnik classifies three levels of design computability (the representational, the parametric, and the algorithmic) by employing five key terms: the input 'in', the function 'f', the parameter space 'para', the output 'out', and the variation set 'var'. The following figure is an ideogram borrowed and drawn according to Kotnik's description:
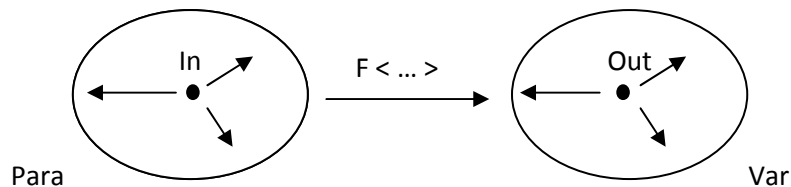
Figure 1: An abstract model of computation (drawn based on Kotnik's model)

In the representational category, the design process is implemented by the visual reasoning of a conventional paper-based design approach and through the computational lens, a relationship 'f' between the specific input 'in' and unique output 'out'. A parametric level deploys such a given relationship 'f' as a 'spectra of possibilities' between the input 'in' and output 'out', which is feasible by continuous variation throughout the parameter space 'para'. At the algorithmic level, the relationship is characterised by 'the utilisation of the formal description of 'f' and its application as a design strategy'[58]. Although employing a Turing machine concept in Kotnik's model looks architecturally surprising, his classification is just an extended format of dichotomous verifications between process-driven and product-driven architecture. Whereas, at the representational level, the outcome of design is prioritised, in parametric and algorithmic categories the focus is more on the process.

[56]Toni Kotnik, 'Digital Architectural Design as Exploration of Computable Functions', *International Journal of Architectural Computing,* 8/1 (2010), 17, pp. 6-7.
[57] Architecturalassociation, 'David Rutten/Grasshopper, Computing Architectural Concepts ', <http://www.aaschool.ac.uk//VIDEO/lecture.php?ID=1212>, accessed 9 October 2011.
[58] Toni Kotnik, 'Digital Architectural Design as Exploration of Computable Functions', *International Journal of Architectural Computing,* 8/1 (2010), 17, pp. 7-9.

Another classification, as described by Rivka Oxman, is relevant to this discussion, although she uses the phrase 'digital design' instead of computational design. Her taxonomy embraces five paradigmatic models for digital design, defined by mapping various relationships between the designer, the conceptual content, the applied design processes, and the design object itself. These categories are CAD models, formation models, generative models, performance models, and integrated compound models. In Oxman's model, traditional CAD is set up on a posterior automation of design drawings and visual models, which shapes the interaction with 2D and 3D formal representations. Conversely, in digital formation models, formal qualities are not predefined. Hence, it produces three sub-classes: topological design, which tries to depart from the more typologically deterministic logic in order to accommodate the new complexity of non-linearity; associative design, which is based on parametric design techniques and associative geometry, establishing inter-dependencies between the various objects, and dynamic design, which is based on animation, morphing and time-based modelling techniques to create a form. The third paradigmatic model, the generative model, is characterised by the application of computational mechanisms for formalised generation processes. Two distinct sub-approaches are introduced here: 'shape grammars' and 'evolutionary models'. While shape grammars employ mathematical expressions to produce transformational shape-generative rules, evolutionary models explain the natural generation of forms. The forth category pictures performance-based design, in which the object is generated by simulating its performance. This approach also implies two sub-classes: performance-based formation and performance-based generation models of design. Finally, the fifth model is an integration of all of the previous models including formation, generation, evaluation and performance[59].

It seems that Oxman's taxonomy adds more ambiguity rather than clarification to the digital domain. By referring to compound models, perhaps she is literally trying to explain that what happens in reality needs an integrated base and these paradigmatic models just exist in the layers of theory. In addition, although in Oxman's framework, formation models include the parametric domain; some parametric approaches are actually more performance-based or evolutionary-based rather than following the formation category. These two taxonomic examples seemingly show that, even though drawing a sharp distinction among different categories in the computational domain may be analytically valuable, in reality and in a design situation a sort of integration may occur which cannot be monitored clearly due to the complexity and fuzziness of the practice. Therefore, we can conclude that the nature of

---

[59] Rivka Oxman, 'Theory and Design in the First Digital Age', *Design Studies,* 27/3 (2006), 229-65, pp. 245-260.

parametric design here appears as a multiplicity of characters, disobeying any cataloguing or constraining through a framework.

Chapter 3
The Research Methodology

This chapter firstly provides a brief explanation of the methodology used in this research. It also introduces the participants of the study, describing how and when they took part. It then focuses on strategies for data analysis, specifically the bottom-up approach and the coding system employed in this research.

3.1. The Research Methodology

The overall scheme of the methodology in this research is qualitative[1], based on semi-structured interviews. However, a simple questionnaire also supports every interview because some details cannot be framed in an interview question. Employing a qualitative strategy appears here as a necessity, since the nature of the research is a comparative analysis between statements stemming from secondary sources and different explanations or interpretations expressed in primary sources. Semi-structured interviews were selected as the method of qualitative data collection because they are close to everyday conversations. They are 'conducted according to an interview guide that focuses on certain themes and that may include suggested questions'[2]. Therefore, it is possible to define the framework of every interview by a set of bottom-line questions, extracted and categorised based on the literature review. Hence, the literature here acts as the generator of the research agenda, addressing the gaps of knowledge in the parametric realm. The locus and the context of the study can be anywhere, because the aim is to investigate the arguments regarding parametric design. Therefore, the sampling scheme of this research is based on a purposive selection of practicing architects and those university lecturers who also have a background in practice. The practicing architects were selected by direct observation of their projects on the web, which its process will be explained here. It is important for this research to cover a variety of voices and especially to have some samples from education, owing to the fact that education is the context that often absorbs and tests new approaches to design.

Following the above rationale, Manchester School of Architecture, along with some practicing architects who are working in Manchester, were the focuses of this study. In order to identify and contact those practicing architects, the Manchester Directory for architecture companies was investigated thoroughly at the outset of the research. The directory is accessible its web site,
http://www.manchesterdirectory.co.uk/local/Building_and_Construction/Architects/, and includes names, contact numbers and company address. Attempts were made to identify which companies may work in parametric design. This was achieved by direct observation of the projects designed by the companies through visiting the website of each company, seeking

---

[1] Matthew B. Miles and A. Michael Huberman, *Qualitative Data Analysis : An Expanded Sourcebook* (2nd ed. edn.; Thousand Oaks, Calif. ; London: Sage, 1994), p. 10. 'Qualitative data embodies in words which in turn based on observation, interviews, or documents, helping the researcher to get a good knowledge on what the real life is. Factors such as richness and holism (providing thick description), local groundedness, flexibility, an efficient tool for testing and developing hypotheses, and reinterpreting quantitative data can be assumed as the strength of qualitative data'.

[2] Steinar Kvale and Svend Brinkmann, *Interviews : Learning the Craft of Qualitative Research Interviewing* (2nd ed. edn.; Thousand Oaks ; London.: Sage Publications, 2009), p. 27.

the images of a project, speculating about the likely usage of parametric software, and coming up with a hypothesis that the company in question probably benefitted from parametric software or designed some of its projects parametrically. The next step was to contact those selected companies.

	Name	Name of the company	Type of contact for setting the interview	Result
1	Matthew Smith	Ian Simpson Architects	Emails, visiting in person	Successful – interviewed twice
2	Simon Reid	Reid Architects	Visiting in person	Unsuccessful – no answer to the inquiry
3	Dave Smith	3DReid	Emails, phone calls, visiting in person	Unsuccessful – no answer to the inquiry
4	Alex Solk	Sheppard Robson	Emails, phone calls, visiting in person	Successful – interviewed
5	Amy Hanley	MSA (worked previously in Sheppard Robson)	Emails	Successful – interviewed
6	John Lee	Arca, MSA	Emails, visiting in person	Successful – interviewed
7	David Wood	Chapman Taylor	Emails, visiting in person	Successful – interviewed
8	Trevor M. Cousins	FDG (Fairhurst Design Group)	Emails, phone calls, visiting in person	Successful – interviewed
9	Rick Sharp	FDG (Fairhurst Design Group)	Emails, phone calls, visiting in person	Successful – interviewed
10	Andrew Wallace	Andrew Wallace Architects	Emails	Unsuccessful – No answer to the inquiry
11	David Kelly	Aedas	Emails, phone calls, visiting in person	Unsuccessful – No answer to the inquiry
12	Irwin Lopez	Denton Corker Marshall	Emails, visiting in person	Unsuccessful – the company was not interested in participation
13	Griff Evans	Ombler Iwanovski Architects , MSA	Emails, visiting in person	Successful – interviewed
14	Phil Osborne	AFL Architects	Phone calls, visiting in person	Successful – interviewed
15	Tony Skipper	John McAslan & Partners	Emails, phone calls, visiting in person	Unsuccessful – the company was not interested in participation
16	Christian Derix*	Aedas	Emails	Unsuccessful - No answer to the inquiry.
17	N/A (general inquiry)	JM Architects	Phone calls, visiting in person	Unsuccessful - No answer to the inquiry.
18	N/A (general inquiry)	Austin - Smith – Lord	Emails, phone calls,	Unsuccessful - No answer to the inquiry.
19	Vikram Kaushal	MSA, Logan and Wilcox	Emails	Successful – interviewed
20	Daniel Richards	MSA	Emails, visiting in person	Successful – interviewed

Table 1: The companies and people who were contacted to request interviews

(* Contact with Christian Derix was an exception. He previously worked in the London office of Aedas and has not worked in Manchester. However, due to his knowledge in computational design, this contact was made, although it did not result in an interview.)

Table 1 on the previous page shows the name of the companies and people with whom contact has been attempted. The percentage of having a successful interview as a result was more than fifty percent, which is a quite acceptable result. Failed interview attempts were largely the result of a complete absence of response on the part of the company. Nevertheless, two companies showed no interest in participation in the research despite initial acknowledgment. We may assume that those who did not answer the inquiry are more traditional offices, despite their web profiles, which give the impression of a parametric approach. Perhaps they were not familiar with parametric design and were therefore reluctant to speak about it.

The study was not required to proceed through the ethical procedure of the University of Manchester. With regard to the interviews, the consent form and participant information sheet were approved by the School of Environment and Development (SED). These forms were given to the participants in advance. At the beginning of every interview, the aims, objectives, the nature of the research, its duration, and the identity and contact details of the researcher were explained. Participants were also told about the way the data would be collected, stored, analysed, used, and protected in the process of the research, as well as how the outcome of the research would be used. As mentioned previously, all of the interviews carried out in this research were of a semi-structured format. In every interview, a series of five major questions (please see appendix 3), starting from general concerns about the way of form generation in the company and ending with more specific questions about parametric design, were followed. The nature of a semi-structured interview also made it easier for some minor questions to be added to the body of the main questions based on the response of the interviewees. Furthermore, the open-endedness of the questions gave the participants more freedom for discussion, extension, and clarification of their responses. Every interview was recorded using a dictation machine. The time for every interview normally varied between 30 and 45 minutes. However, in one case, it took more than an hour due to the tendency of the interviewee to give more elaboration and explanation regarding the practice of that company. At the end of every interview, interviewees were also asked to fill in a paper questionnaire (please see appendix 4).

The next section briefly introduces those who contributed to this research. In addition to these participants, the following people also contributed to this research via email. All of them are working on the development of parametric approach in architecture. They were asked different types and limited numbers of questions, since it was not feasible to apply the

same agenda due to the lack of time and accessibility. The email interviewees were: Dr. Roland Hudson, Assistant Professor of Faculty of Architecture and Planning in Dalhousie University in Canada; David Rutten, the creator of the Grasshopper plug-in (the main tool in parametric design) and employee at Robert McNeel & Associates software development company, USA and Daniel Davis, PhD student in Spatial Information Architecture Laboratory (SIAL) at the RMIT University in Australia and the blogger of the weblog Digital Morphogenesis (http://www.nzarchitecture.com/blog/).

Brief introduction to the interviewees

Matthew Smith, an architect and specialist in working with parametric software at Ian Simpson Architects. This office is a famous name in Manchester, offering services to clients from 1987. The company has been actively involved in designing various projects, from residential buildings and private mixed-use projects to recreational and cultural centres, not just in the UK, but also around Europe. Matthew has been with this practice for 6 to 7 years, but he has been in design for more than 12 years. He has benefitted from different CAD programs and especially has many years of experience in Generative Components, Grasshopper and other parametric packages. He participated in this research twice, on 17 October 2011 and 3 November 2011.

Alex Solk is an associate partner in Sheppard Robson Company in Manchester and has been working for the practice for more than a decade. The company is one of the well-known practices in the UK and has branches in the cities of Manchester, London and Glasgow. The company also has an office in Abu Dhabi in the UAE. Their design includes a broad range of typologies, from residential and educational buildings to transport and infrastructure projects and master-planning. Considering environmental issues to produce sustainable architecture, they have employed parametric programs for a number of projects, specifically in their recent in order to explore new ideas in design. Alex was interviewed on 22 November 2011 in his office.

David Wood is an architect with significant experience in architectural practice and currently works for Chapman Taylor, one of the oldest architecture companies in the UK. They established their practice about fifty years ago and, since that time, have grown to become a diverse, international practice. They have two offices in England (Manchester and London) and

fourteen offices in other cities all over the world such as Paris, Moscow, Shanghai, and Abu Dhabi. The interview with David was conducted on 1 November 2011 in his office.

Amy Hanley is a member of ARB, RIBA, and FHEA. She worked at Sheppard Robson Company in Manchester before joining the Manchester School of Architecture. The focus of her research now is 'freedom and architecture', the development of applied pedagogical principles and the communication of architectural language. However, her previous employer is one of the major centres of deploying parametric design in architecture, which gave her valuable experiences in working with this design approach. She was interviewed on 25 October 2011.

John Lee is an architect and the principal of the ARCA office in Manchester. He studied at the Bartlett School in London and graduated with a First Class BSc (Hons). He completed his study with a Masters degree in Advanced Architectural Studies, working with what is now the Space Syntax Laboratory. This experience led to a concern for the link between building technology, texture and materiality, and spatial configuration; in other words, the physical expression of architectural form. He also spent nine years with Grimshaw Architects. Mixing architectural practice with education, he has taught at MSA since 2004. The interview with John was conducted on 28 October 2011.

Phil Osborne is the creative director of the AFL office in Manchester and has a Bachelor's degree in Architecture (Hons). Having been involved in a number of projects, Phil has experience in a wide range of major commercial, residential and urban design projects, as well as ongoing involvement in the design of major sporting venues. AFL has worked in architecture for nearly fifty years now. It is most famous for designing stadiums all over the world and collaborating with great names in architecture, such as Norman Foster. One of the company's projects was the master plan for the redevelopment of the Old Trafford Stadium in Manchester. Phil was interviewed in his office on 7 November 2011.

Trevor M. Cousin is the architect and associate director of the Fairhursts Design Group (FDG). He has been in architectural practice for more than a decade. The company has been involved in many architectural projects as well as master-planning and landscape design. They offer their services through two offices in the UK, in Manchester and Southampton. They also have four branches outside the UK, in Paris and Lyon in France, and Dubai and Abu Dhabi in the UAE. One of their projects is the Arthur Lewis Building, designed for the University of Manchester.

Rick Sharp is an architect and executive associate at the Fairhursts Design Group. Like Trevor, he has many years of experience working in architectural practice in a broad range of design

projects. Rick and his colleague, Trevor, were both interviewed on 4 November 2011 at the FDG company office in Manchester.

Griff Evans is an architect working at Ombler Iwanowski Architects in Manchester. He joined the company in 2007 and has experience in designing and delivering a variety of design and refurbishment projects, including the award winning Croma Restaurant in Prestwich. He is also a part-time lecturer at the Manchester School of Architecture.

Vikram Kaushal is a graduate of the Manchester School of Architecture. He established Logan & Wilcox in 1999. Kaushal is currently teaching at the MSA. He is also investigating the opportunities and dilemmas of a world of networked "I.A" (Intelligent Artefacts) posing for architecture and urbanism. He is also interested in research on intelligent artefacts, information transformation, ubiquitous computing and data-driven design. Vikram participated in this research on 31 October 2011.

Daniel Richard is a PhD student in Architecture at MIRIAD (Manchester Institute for Research and Innovation in Art and Design) and is a member of the Novel Computation Group at MMU. His research benefits from bio-inspired computation to explore the production of complex performance-driven assemblies that can provide new territories for sustainable architecture. By having a good knowledge of computer programming (especially Python), he has contributed to several workshops and training sessions on the subject of the computational approach to design. The interview with Daniel was conducted on 8 November 2011.

3.2. The Strategies of Data Analysis

In the qualitative realm, the most important task for researchers is how to analyse their data and turn it into a meaningful structure. Luckily, the low number of participants in this research considerably alleviated some of the major problems that are experienced in every qualitative data analysis, such as 'frequent data overload, and the time demands of processing and coding data'[3]. Nonetheless, some of the problems inherent in qualitative research were experienced, such as difficulties in transcribing the interviews. This research employed three stages of data analysis, namely data reduction, data display and conclusion drawing. Data reduction was a process whereby the volume of the qualitative data was reduced into manageable chunks of information, which then were displayed, categorised and re-structured. These chunks may be assumed as themes of inquiry, supporting the main arguments of the research, and they often emerge from a series of codes. Codes 'are tags or labels for assigning units of meaning to the descriptive or inferential information compiled during a study'[4]. Codes can also contain other families of codes inside themselves. Naming a code is also important and it must be as close as possible to the concept behind the referred text. The coding system in this research followed a bottom-up strategy, moving from codes to themes. In this research, after reading thoroughly through the transcripts of the interviews, the data was reduced to three thematic parts. These parts were representative of the main arguments that were pursued within the interviews. In order to arrive at a point that fuses primary and secondary sources on parametrics en bloc, the same coding system was applied to the secondary literature too. This in turn provided a logical comparative analysis, erecting a scaffold to support the analysis of parametrics from these sources.

Having clarified similarities and contradictions, drawing conclusions commenced, which provided the next five chapters of this research. Table 2 shows themes arising from the coding system used for the analysis.

---

[3] Matthew B. Miles and A. Michael Huberman, *Qualitative Data Analysis : An Expanded Sourcebook* (2nd ed. edn.; Thousand Oaks, Calif. ; London: Sage, 1994), p. 2. Also, according to Miles and Huberman, a chronic problem with the qualitative method is that it is carried out chiefly with words rather than numbers. Most words are meaningless unless the researcher looks backward or forward to other words. In addition, some predicaments might emerge from the way of transferring the verbal statements into the written text. In terms of covering all of the words as well as participants' gestures, transcripts can be 'thick' or 'thin'. They can offer a different range, from 'pauses, word emphases, mispronunciations, and incomplete sentences of an apparently incoherent speaker, to a smooth, apparently straightforward summary of the main ideas presented by the speaker'. (p. 51-56)

[4] Ibid, p. 56. Miles and Huberman explain that 'the coding is a way of forcing you to understand what is still unclear, by putting names on incidents and events, trying to cluster them, communicating with others around some commonly held ideas, and trying out enveloping concepts against another wave of observations and conversations' (p. 62).

Theme	Code Description	Code
The Questions of 'Role' and 'Driver'	Factors of Generating the Form	FGF
	Type of Approach to Design	TAD
	Role of Computer Programs in Parametric Design	RCPPD
	Dependency of Designed Forms on Software	DDFS
	Role of Sketching or physical Modelling in Design	RSMD
	Role of Designer	RD
The Question of Style and the Position of Parametric Design	Roots of Parametric Design	RPD
	A Style or a Set of Techniques	SST
The Up and Down Sides of Parametric Design	Advantages of Parametric Design	APD
	Disadvantages of Parametric Design	DPD
	Distinctions in Comparison with traditional CAD	DCCAD
	Distinctions in Comparison with BIM	DCBIM

Table 2: Research themes and coding system for data analysis

Table 3 shows how a code can be extracted from an interview transcript. The code in question – a Style or a Set of Techniques (SST) – investigates the question of style in parametric design and the arguments arising from this realm of inquiry.

Quotations	Code	interviewee	Date
It is a technique to help. Well, a lot of people use it a as a new architecture style, plenty of architects, we can think of big international architects… It is not just what I use… I prefer to think of those technologies as techniques to help; but I appreciate many architects use it as a style.	SST	Alex Solk	22.11.2011
I think calling it style is slightly restrictive… who is a parametric designer? One who is just using these kinds of particular programs, I think it is possibly restrictive I guess, because you are going to say that you have to use these programs or this methodology.	SST	Matthew Smith	17.10.2011

Table 3: Coding an interview transcript

In comparison to that table, Table 4 exemplifies applying the same coding system to a secondary source. Through this way, contradictions and possible inconsistencies come into view clearly. Additionally, drawing conclusions becomes simpler by vividly recalling those contrasts. In this case, for example, it will be discussed later in the next chapters that almost none of the interviewees recognised parametric design as a new style. They believed that it is just a set of techniques arising from technological advances and this is in sharp contrast to what is available in the secondary sources. The comparative strategy in all of this research follows this pattern, which is possible by many tables.

Quotations	Code	Author	Date
Parametricism can only exist via sophisticated parametric techniques… Today it is impossible to compete within the contemporary avant-garde scene without mastering these techniques.	SST	Patrik Schumacher	2008
Parametricism is a new style rather than merely a new set of techniques. The techniques in question – the employment of animation, simulation and form-finding tools, as well as parametric modelling and scripting – have inspired a new collective movement with radically new ambitions and values.	SST	Patrik Schumacher	2009

Table 4: Coding a secondary source

To conclude this chapter, it is important to point out the inductive strategy used in getting results from the collected data. Although induction may yield approval or disapproval of a claim or a hypothesis, it can also exacerbate some research problems, including fallacious generalisations. The nature of the questions in this research entailed an inductive 'system of inquiry'[5] , especially because parametric design is a new approach to architectural practice. In this sense, the variety of background across participants and the type of arguments they would say were more important than the context of the study, whether it was in Manchester, London, or even another country.

---

[5] The term 'system of inquiry' or 'paradigm' used by some researchers such as Groat and Wang often coveys the notion of a worldview or ' the ultimate truthfulness of which cannot be established' (Linda Groat and David Wang, *Architectural Research Methods* (New York ; [Chichester]: Wiley, 2002), p. 21).

Chapter 4

Parametric Design and
The Status of Primary Drivers

Focusing on the notion of 'driver', this chapter investigates the status of primary drivers to parametric design. The idea of a primary driver is not a new topic in architecture. Yet, if someone wants to look into the blackbox of design, from outside to inside, drivers may seem a major argument. Signposting the design process, they lead architects to the conclusion point. In one sense, they are related to the pivotal concept of parametrics, the notion of 'constraint'. To explain the concept of 'primary driver', the main question of this chapter is, 'how do primary drivers appear in parametric design, and does the parametric approach change the nature of primary drivers?'

The Question of Drivers (Outside to inside)

Different factors come forward to show their importance when a designer decides to tackle a design problem. Central to all design professions is the concept of problem-solving, because essentially every design project is created as a response to a need. The act of designing is a vague and indefinite process in which the starting point, the origin or the foundation and even the stages within the process are often uncertain. This is the reason why some thinkers refer to it as an art rather than a science[1]. Indeed, some researchers such as Paul Coates claim that what is now used as the design methodology was not actually invented by architects, but by engineers and operational research scientists. The goal of this invention was to 'optimise between maximising capabilities and minimising cost'[2]. Again, this has always been a problem for architectural design, where the incommensurability of many criteria prevents obtaining an optimum. While such criteria sometimes lose their influence within the ambit of theory, they pragmatically compete with each other as drivers of architectural design.

Here, I believe the concept of 'driver' deserves more elaboration, since architects frequently refer to its foundational role. Semantically, 'driver' can refer to two standpoints, stemming from a subjective or objective position. On the subjective side, driver can be defined as motivation; a driving force which leads architects to the conclusion point. Objectively, (and I would argue this is closer to the typical view of today's architects) it implies an instrumental role in guiding architects. While in this view, it is an aid, it acts as a constraint or a determinant too. A good metaphor is to think of a 'driver' as a taxi driver who is obliged to pick up the architect and drive him or her by considering a condition: the destination is not entirely recognised, but it is located in the city centre. Such a condition essentially makes both the driver and the passenger overwhelmed. Even though the goal may look rather clarified, many contingencies may occur in reaching it, which shows the problematic nature of the activity. The role of drivers within the design situation is similar to this metaphor and, in this sense, they deserve more attention and elaboration. This is more significant for primary drivers, which are considered the source or the starting point of design.

---

[1] Many architectural theorists referred to this point; for instance, see Kostas Terzidis, *Algorithmic Architecture* (1st ed. edn.; Amsterdam ; London: Architectural Press, 2006), p. 42. He argues that 'Design has no formula; it is an iterative, trail-and-error process; so it is an art rather than a science'. This stance is a bit ambiguous, since he sees an artistic taste throughout an engineered procedure and perhaps it contradicts the debate in his book, *Algorithmic Architecture*, in which he explains the computational methods of form generation. He also argues in his earlier book, *Expressive Form*, about a third generation of architecture arising from formal consideration in which form is achieved through a hybridisation of creativity and determinism by computational methods: 'For the first time perhaps, form might be aligned with neither arbitrary creativity nor computational determinism but with creative computation and computational creativity'. (Kostas Terzidis, *Expressive Form : A Conceptual Approach to Computational Design* (London: Spon, 2003), p. 6.)

[2] Paul Coates, *Programming.Architecture* (Abingdon: Routledge, 2010), p. 160.

One of the most influential studies on this topic was carried out by Jane Darke. Her study tries to offer a theoretical framework for the design process based on the axial idea of 'generator', a concept similar to the state of driver. In search of an answer to the question, 'do architects have an image in mind during the design process about the expectations of the users?'[3], she challenges the analysis-synthesis[4] model which was quite common in the past. Her model is extended based on the concept of the primary generator followed by a conjecture. 'Further understanding of the problem is gained by testing this conjectured solution'[5]. In fact, 'primary generator' can be a group of related concepts rather than a single idea. These objectives form a starting point for the architect – a way into the problem. Once the initial concept has been generated, it is tested against the various requirements and modified if necessary[6]. In Darke's view, 'design is seen as a process of 'variety reduction', with the very large number of potential solutions reduced by external constraints and by the designer's own cognitive structures'[7].

The aim of this research is not to come up with a new epistemic model for the design process. However, it is essential to understand the nature of these starting points and to investigate their status in parametric design. According to Kostas Terzidis, the starting point acts as a pivot, which is important for at least two reasons: first, a focal point of referencing and second, an origin belongs to a distant past[8]. However, he claims that this pivot is out of 'style, fashion, or mannerism', which seems disputable, since for many architects the pivotal point is essentially engendered by applying stylistic principles or personal ethos, stemming from deductive reasoning. Nevertheless, the notion of the starting point furnishes parametric design with this question: is the primary driver the same in parametric design or not? Does the approach towards design, whether parametric or not, affect the drivers?

Intuitively, it seems that the primary driver is not dependent on the approaches or tools helping architects in design, and it continues to be a primary driver whether an architect

---

[3] Jane Darke, 'The Primary Generator and the Design Process', *Design Studies,* 1/1 (1979), 36-44, p. 36.

[4] Analysis-synthesis model in design or A/S has been a hot topic for a long time. Questioning its feasibility, many researchers have tried to present a new model. Once Colin Row explained that this sort of duality between researchers arises from two mental orientations: '1. an act of analysis will automatically result in an act of synthesis. 2. The other is the inversion of the first point of view'.

[5] Jane Darke, 'The Primary Generator and the Design Process', *Design Studies,* 1/1 (1979), 36-44, p. 38. The primary generator is defined as 'a broad initial objective or small set of objectives, self-imposed by the architect, a value judgement rather than the product of rationality'.

[6] Ibid. Darke asked her interviewees to describe the 'evolution of the design, the existence or otherwise of an image of the future user, and the source(s) of any image of this kind'. She tried to indentify the starting point of design through this way. For example, in her paper, she quotes from one of her interviewees: '…the site is spontaneously described as 'the mare starting off point'. The second factor, the soil strength, was not a designer-imposed constraint but was obviously a major determinant of the solution'. (Ibid, p.39)

[7] Ibid, p. 38

[8] Kostas Terzidis, *Algorithmic Architecture* (1st ed. edn.; Amsterdam ; London: Architectural Press, 2006), p. 8.

designs using parametric methods or not. Through speaking to architects, they refer to two main drivers, which are quite independent of the design approach; the project context and the client's brief. Traditionally, these overarching factors have always been the pinnacle of a design project. Context includes socio-economic and environmental aspects of the design problem, while brief demonstrates the client's demands and the necessities of the project. However, the question is, 'what criteria can give precedence to one of these factors in the design domain?' Are they constant over time?

Primarily, context has always been of paramount importance to architects due to its direct impact on design. It may engender different categories itself, notably physical concepts and socio-economic ideas. Similarly, the interpretations of the same context could be different among architects. While some of them might underline the environmental and physical issues, others may seek the significance of the socio-economic concerns more than other matters. We see the problem of pinning down context – it is a catchall term which is easy to express but difficult to flesh out. Thus, it is more practical to refer to it not as a sole factor but as a family or a class of elements interwoven with each other. In this view, 'the project context is probably the biggest driver'[9]. Architects often start with a simple site-analysis to work out the design context, particularly to investigate an important class of factors inside it which act as physical constraints. If the context is imagined as a site of interacting forces, then these forces can drive the form to harmonise with the site and its landscape. According to the famous architect and theorist of architecture Greg Lynn, form can be shaped by the collaboration between an 'envelope' and the active context in which it is situated. Lynn describes that the specific form of a hull stores multiple vectors of motion and flow from the space in which it was designed. Although this form is designed to anticipate motion, there is no expectation that the shape of the hull will change. 'An ethics of motion neither implies nor precludes literal motion'[10]. For Lynn, dynamics mean supposing forces over matters, while statics means eliminating forces from matters. Therefore, he stresses that, by considering the interactive space of force and matter, architectural form can shift from 'autonomous purity to contextual specificity'[11].

Lynn's description of 'contextual specificity' seems to be quite relevant to parametric design. It could even be said that it is one of the distinctions of parametric design. While older approaches tried to allocate some sort of specific form, they often ignored the context in which the design is shaped. The parametric outlook considers all of the dynamic properties or

---

[9] Interview with John Lee, 28.10.2011

[10] Greg Lynn, *Animate Form* (New York: Princeton Architectural Press, 1999), p. 10.

[11] Ibid, p. 11

physical forces of a site in the frame of parameters. Hence, it attempts to define a framework, helping architects to think of the context in a designerly way. In this sense, it is unique in comparison to previous design approaches. While the analysis of site forces was high on the agenda to drive the architectural form for many architects, such as Le Corbusier, this orientation never had a mathematically parameterised tendency. It was more based on the architect's intuition rather than the computer's precision. Hence, it seems that deploying contextual issues is one of the principles of parametric architecture.

Considering design as both a source of requirements and a solution to a problem, the client's brief also plays an important role. In the light of different interpretations, the notion of 'brief' implies different courses of actions. For instance, Lynn argues about a fixed relationship that he posits can only be seen in static ideas, naming it 'functional fixity'[12]. Although he refers to functional fixity as a fixed relationship to the design program, it is not clear why he thinks this is just applicable to static concepts; presumably, he has tried to compare it with his characterised theory of 'animate form'. I would therefore be sensible to examine to what extent such views are feasible in architectural practice. Yet there are many architects working and presenting their architecture who believe that form results from a thorough understanding of the functions that are requested by the clients. Alex Solk, a senior architect at the Sheppard Robson office, defines the generation of form in the following terms:

I think it is the close interplay between the functions of the building… I think the client brief is to understand how people are working and that will generate a use for the building and that will have some sorts of impacts on form. I think you have a site response, which would be an environmental response initially. [13]

Having gained the experience of designing a broad range of different buildings, Solk emphasises the relation between form and functions, which is not surprising. It could be assumed as a review of the traditional motto described by Louis Sullivan, 'form follows function'[14], which still seems to be extant among architects. In contrast to what Solk says, Patrik Schumacher, the theorist of parametricism (arguments about parametricism and the questions of style will be explained in Chapter 6), in his recent book *the Autopoiesis of Architecture*, argues a different view. He believes that parametric design in the frame of parametricism uses a circular formula which oscillates between form and function:

Two principal ways of controlling the double contingency of the design research situation leading to two formulae…: the formula 'form follows function' and the formula 'function follows form'. The first formula is the formula of

---

[12] Ibid, p. 13. 'Functional fixity: buildings are often assumed to have a particular and fixed relationship to their programs, whether they are intersected, combined or even flexibly programmed'.

[13] Interview with Alex Solk, 22.11.2011

[14] Form follows function: one of the first ideas on architectural functionalism described by Louis Sullivan, the American architect. This concept was used and extended by some of his followers such as Frank Lloyd Wright.

Modernism and the second formula is the formula of Postmodernism, Deconstructivism and Folding. Both are self-simplifying formulae, because the underlying double contingency of the design situation is only tamed (structured), not eradicated. Another way to control the double contingency of the design research situation is to operate via an integrative oscillation between formal and functional advances. This leads to the circular/oscillating formula 'form follows function follows form'. The theory of architectural autopoiesis identifies and promotes this formula with respect to the maturing design research program of Parametricism[15].

Solk is an experienced architect and perhaps his words prove the dominance of functional concerns in comparison to formal issues associated with the clients' needs. On the other hand, it can also be construed that, because clients usually do not have a clear idea about the form of a building, they come with their own obsession, although 'they do not know how to articulate it'[16]. If they do so, this might be just some rather vague, general ideas about the physical appearance of the designed project. In addition, it is essential to differentiate between function and functionality in terms of the meaning, since it will be a key for understanding the drivers of the form. Function is quite a tangible term, illustrating the aspects of a place, whereas functionality is rather a rational image in the designer's mind, acting as 'an inner logic'[17] to drive the design process. It is also imperative to point out the difference between performance and function, since they are frequently confused with each other. Performance is normally used to refer to an immense range of characteristics of a designed object, including function and materiality. In other words, when function is reinvented in the new frame provided by themes such as sustainability, it changes its name to performance. Some thinkers argue that the notion of 'performance' is underlying, since it is against the dialectical conflict between functionalism and formalism[18], and as such can embrace both of them in one body. In all likelihood, Solk's stance of function is closer to performance rather than function itself, because he later refers to considering environmental responses. Nevertheless, the majority of architects still affirm that the brief does not really depend on the approach or methods, whether parametric or not, which a designer selects to begin with. More likely, apart from functional requirements, other issues can be found inside the brief. Phil Osborne, an experienced architect in AFL office who has been in practice for more than twenty years, describes that the brief is still the primary driver of design:

Well, the primary driver is the brief. The brief from the client and previous experience and research into the building type… The brief needs to include the spatial requirements, the aspirations of the client, whatever the project might

---

[15] Patrik Schumacher, *The Autopoiesis of Architecture: A New Framework for Architecture* (Wiley, 2011b), p. 268.

[16] Riceuniversity, 'Facticity by Neil Denari', <http://www.youtube.com/watch?v=XKQyW053Ois>, accessed 26 Jan 2012.

[17] Toni Kotnik, 'Digital Architectural Design as Exploration of Computable Functions', *International Journal of Architectural Computing,* 8/1 (2010), 17, p. 10. Kotnik also argues that 'Functionality in an architectural sense is, in general, not stated explicitly as a casual relationship that can be computed, but is instead used more vaguely on a conceptual level as a link between an often non-quantifiable architectural purpose and its expression through form and materiality'. Probably, the reason why he argues in this way is to explain different levels of computation in design.

[18] Michael Meredith, 'Never Enough', in Tomoko Sakamoto and Albert Ferre (eds.), *From Control to Design : Parametric/Algorithmic Architecture* (Barcelona ; New York: Actar-D, 2008), p. 8. 'Constructing use as the performance of architecture, however, is not about reconstructing a neo-neo-functionalism or a post-post-functionalism; it is against the dialectical opposition (functionalism / formalism) of form follows function'.

be and the cost parameters… We do not have standard responses to questions. Brief is the critical element and if you get the brief wrong, you generally end up doing a lot more work than you should have done [19].

There are two interesting points in Osborne's description of the brief. First, his company does not have 'standard responses' to the clients and pursues every design project as a unique one which needs a thorough, yet critical, understanding of the brief. Second, he refers to the cost parameter, which means that he believes it is quite crystallised inside the brief. Conventionally, cost is a leading factor coming out from the heart of the brief. The amount of money that clients are prepared to invest in the project is of the utmost importance. It in turn gives the architect the freedom of going towards more free-form design. Recognising cost as one of the key parameters in the form generation is one of the reasons why parametric design has been much more effective in picturing the actuality of the project in comparison with traditional CAD and other traditional methods. This concern is clear in the words of Rick Sharp, an architect and executive associate at FDG (Fairhurst Design Group):

Obviously, it depends on what clients are looking for, first of all, and so the project brief is going to be a key towards whatever we do in terms of form generation from that. Then cost obviously has a big part as to whether we have freedom in design or whether you are looking at more rigid grid and something that is more traditional cost-based. If we have the opportunity to do something differently, then obviously it depends on the site…[20]

Sharp's explanation proves that cost is still considered a big issue for large-sized companies, which can dictate the degree of freedom in going towards free-form architecture. However, what else could be the primary driver? Apart from intrinsic drivers such as the context and the brief, can one assume the architect's approach towards design or the type of the project a design driver? To what extent is this sensible? As explained, some factors, specifically the cost of the project, result in alteration of the design process and the approach towards form generation. Looking at the iconic projects in the world appears to validate this hypothesis. Most of these projects have been designed by famous companies or star architects whose names and portfolios alone are often sufficient to convince the clients that the proposed design is worthwhile and will be cost-effective. Here, cost is not the main factor that orientates the architect to design in a certain way. On the contrary, it is the approach of the architect that dictates the financial issues. In this regard, John Lee, the principle of Arca, explains two outlooks on the generation of form in architectural design, arising from the client's brief:

There are two ways to talk about. One is to say as an ordinary architect – if you like practicing in an average European city with an average economic condition, for the most part, form generation is a question of economics. So, for the ordinary architect working in the ordinary condition in any European city, generation of form is to do with how to create the simplest place consistent with meeting the requirements of the brief that clients have set, but obviously the nature of form generation is one aspect of it. The other of course, is a kind of form generation

---

[19] Interview with Phil Osborne, 7.11.2011
[20] Interview with Rick Sharp, 4.11.2011

that makes the architectural journals and it's more of an iconic or landmark specific building and obviously something like Frank Gehry's or Peter Eisenman's or alike…[21]

Lee makes a clear distinction between two types of practice: star architects and mainstream designers. He may have made this point because his office, Arca, is a small company which has not been engaged in the projects that those great names in practice are often offered. Therefore, it is not surprising that Lee speaks about economy here. Hence, he generalises his view to argue about two types of buildings; ordinary and iconic. Although the essence of this duality could be true, the border between them does not seem to be clear. For example, is it enough to see a name of an architect in a glossy architectural journal to call him or her a superstar? Which authority nominates a building as an iconic project? However, in mainstream architectural practice, economy seems to be the main concern. It has the potential to convert an initial premise into a very different conclusion. Trevor Cousin, the architect and associate director of the FDG who designed the Arthur Lewis Building for the University of Manchester, explains:

It is of my interest that you are in the Arthur Lewis Building. We worked on that one and we had, I think, several design types on that building... We were dictated by the cost, budget and obviously a rectangle is the most efficient form and it was also dictated by how the building is needed for use in terms of the solid office arrangement; so we did look at other ways of creating a building, how it may happen... It was an iconic form on a particular site that suited that site. But quite often that does get through it... Absolutely, what appeared may not be what we were looking for[22].

Apparently, having an iconic project in their portfolios for many architects is ideal, since it can give a positive reputation. Comparing the majority of landmark projects, such as the Imperial War Museum in Manchester, Cousin's colleague, Rick Sharp, describes the differences in budgeting depending on the type of project.

if we were given an Imperial War Museum to do, then… it would also depend on the budget as to what we produced and unfortunately most architects are very much restricted by the budget they have to work with, but that does not mean that they do not want to design something with a bit more flair which has a bit more individual identity[23].

Beside these realistic senses of economy, a rather unique view on this hotly disputed topic is described by William Mitchell. In his introduction to the book *Expressive Form*, Mitchell explains that architects are not only constrained by time and limited expenditure of resources, but are also confined by the current 'shape economy'[24], the state that is bestowed on them by computer programs. Mitchell's explanation opens a new window, which is quite relevant to parametric design in particular. It is a serious place for discussion on whether computers can

---

[21] Interview with John Lee, 28.10.2011

[22] Interview with Trevor M. Cousin, 4.11.2011

[23] Interview with Rick Sharp, 4.11.2011

[24] Mitchell's introduction to the book *Expressive Form*. See: Terzidis, Kostas (2003), Expressive form: a conceptual approach to computational design (London: Spon).

dictate their frameworks to designers or not. In this sense, software developers can be considered meta-designers, since they design a design platform for the real designers. Later in the next chapter, the role of computer programs will be discussed; however, it is first worth mentioning several issues vis-à-vis the primary drivers to conclude this chapter.

The primary driver was the key concept investigated in this chapter. It seems that its nature has remained the same in parametric design. Architects perceive the driver independent of the approach, since it is dictated by external factors such as clients. Obviously, architects cannot change the brief, even if they have the latest technology of form generation[25]. In addition, drawing a consensus about the primary drivers in design appears to be nearly impossible because there is a plethora of quite different factors affecting the process, and this makes the task of classification problematic. On the other hand, for many architects all of the factors have the same share in design, and the relations and interactions among them are much more significant and influential than their positions per se. Still, some architects suggest that thinking purely about form without thinking about the practicalities of realisation is impossible. Therefore, they consider architectural form as 'the relation between the context, the client, the design team, budget, and time. All of these factors come in to design form'[26]. Furthermore, it is essential to note that, although an architect's approach can sometimes be assumed as a driver, it has a more intuitive origin in comparison with factors such as the project context. Sometimes, this origin is quite clear in the iconic projects around the world designed by famous architects. They use these individual approaches to sign their design project. Yet, in the majority of buildings, these approaches appear as peripheral figures.

---

[25] Interview with Phil Osborne, 7.11.2011: 'You do not go to the client and say: 'well, our latest design tool generated this. Will you change your brief?''.
[26] Interview with Amy Hanley, 25.10.2011

Chapter 5

Mapping the Roles In the Process of Parametric Design

This chapter employs a different strategy to discuss parametrics. It draws a vantage point from inside to outside by focusing on the notion of 'roles' in the design process. The aim is to investigate the footsteps of the parametric approach within the design process through mapping the roles of designers, their viewpoints toward computation in general and parametrics in particular, and, more importantly, the role of sketching and computer programs.

5.1. The Question of Roles (Inside to Outside)

5.1.1. The Role of Designer

Design activity is an uncertain and irregular task, and the question of drivers (discussed in Chapter 4) can only cover one area of this broad realm. For this reason, researchers such as Albena Yaneva argue that a more comprehensive view of the design process is obtained when architects are seen 'from the inside out'[1]. A design project is more complex than it may seem from the outside. Many contingencies and unforeseen events can happen, sometimes beyond the control of architects, and may change the direction of the project as a result. Thus, in order to look at the parametric approach in relation to the design process, I will investigate the possible changes arising from the idea of 'role'. The notion of 'role' here presents a deeper inquiry into the design process by focusing on the role of designers and their individual approach to parametric design along with the role of non-human actors such as sketching, physical modelling, and computer programs. Such a notion may recall the Actor-Network Theory (ANT) and the way all actors are pictured through a network. Yet it is essential to say that the mapping in this research is not illustrative and visual; it is textual, containing different arguments which may arise from every role[2].

To begin with, the question of roles focuses on designers. Does parametric design change the role of designer? This question is highly important, especially in terms of design pedagogy. More importantly, is it valid to talk about 'parametric designers'? What are the criteria for this designation? Is a parametric designer someone who just uses parametric programs? Or, is it someone who deploys the parametric approach in any software packages? The role of designer brings two levels of inquiry in mind. On the first level, it urges us to investigate the role of architects as individuals working with tools – in other words, the objective side of the architect. The second view explores the subjective side of the architect, or his/her approach towards design and the idiosyncrasies coming along with it.

The answer to the objective inquiry is probably situated in the connection between designers and their computer programs. Kostas Terzidis argues that two antithetical thought camps based on two ideologies in this regard can be imagined. First, there is a view that says designers are only tool-users. They simply use tools and then they seek to connect humanistic

---

[1] Albena Yaneva, *Made by the Office for Metropolitan Architecture : An Ethnography of Design* (Rotterdam: 010 Publishers, 2009): 'To understand the societies produced by architects, we need to look at them from the inside out' (p. 100).
[2] The concept of 'mapping' here is employed to show the range of possible options attached to the domain of roles. Mapping is 'an operation that associates each element of a given set (the domain) with one or more elements of a second set (the range)'. Oxforduniversity, *Oxford Dictionary of English* (Oxford: Oxford University Press, 2010).

philosophies with digital phenomena by searching humanistic ideas or principles, such as Greg Lynn's explanation about the connection between the plasticity of computer-generated form and Gilles Deleuze's descriptions of smoothness. For Terzidis, such an explanation, while it may have phenomenal value, does not certainly reflect a truth at a mathematical level. The second ideology stresses that designers must be tool-makers as well. This approach, rooted in computation, offers the means for design explorations using computers as vehicles. Its protagonists are software developers, computer scientists and mathematicians[3].

Both of these ideologies may appear feasible at first. Yet, assessing the state of architectural practices today shows that the majority of architects are still tool-users. This group posit that within the parametric process the role of designer is unchanged, since again the architects need to spell out the brief.[4] They still use the tools to respond to client demands. In contrast to Terzidis, some practicing architects explain that having a pragmatic approach is more important, and that the aim is to arrive at a point that enables the architects to employ different methods to reach more alternatives.[5]

Although the pragmatic view has its own place among practicing architects, the duality between tool-making and tool-using is also seen in practice. In particular, some believe that architects must promote their position from being a mere consumer to an innovative producer. Thus, the other side of this dichotomy becomes apparent, and it belongs to the tool-makers – those architects who consider themselves as digerati. They think of going beyond the limitations of conventional design, underscoring that architecture programming is gradually moving towards programming architecture. These are notably young graduates of architecture schools, who state that software can never meet designers' needs. Yet this statement is also seen among experienced architects as well, albeit possibly for a different reason. These older architects are probably diehard fans of traditional pen and paper methods, preferring to act in the same way as they have for decades past and ignoring entirely the inexorable progress of technology. Griff Evans, a practicing architect at Ombler Iwanowski Architects in Manchester, believes that 'much new software is inappropriately applied to architectural design because it comes from other disciplines like manufacturing and engineering'[6]. Perhaps this is rooted in the complex nature of design, which cannot be thoroughly framed in the body of software. In

---

[3] Kostas Terzidis, *Algorithmic Architecture* (1st ed. edn.; Amsterdam ; London: Architectural Press, 2006), p. 56.

[4] Interview with Phil Osborne, 7.11.2011

[5] Interview with Matthew Smith, 17.10.2011: his view is interesting, since he is an expert at working with parametric programs.

[6] Email from Griff Evans, 17.11.2011. My meeting with Griff was quite informal, so I did not use my dictation machine to record the dialogue. We had a chat over the subject (about 25 minutes). I received this response by email four weeks after the interview on 17.11.2011. His view here is rather surprising, because he mentioned that 'we are too small and do not have a need for using parametric programs'.

fact, most software developers only use the latest technology to present new platforms of software packages rather than considering design methodology and its possible alterations within time. Following what is prepared by software companies and being mere users of such platforms puts the designer's work at risk of being dictated by the structure of software, a phenomenon called 'the Whorfian effect'[7]. This is another reason that the second camp of designers (tool-makers) are wrangling with software developers. However, it seems that the solution is not subtle. In order to break the barriers of the software, architects should be able to write in the language of computers and increase their own skills in communication with digital environments, instead of being just passive users. Dr. Roland Hudson, a lecturer and a specialist in working with parametric programs, explains:

> The designer needs a tool-kit of parts that can be customised and extended according to the needs of a project. No piece of software can anticipate these requirements. The designer needs to be a tool-maker and the software needs to support this… The designer must not sit and wait for the software to be available. It is merely a question of why and how to apply.[8]

Tool-making not only changes architectural viewpoints toward practice, but also furnishes design education with the idea of the cycle of 'homo faber homo fabricatus'[9]. However, this approach to tools desperately requires time to be tested and used globally, even though in some of the schools now, especially at the postgraduate level, there are courses in which students are taught specific computer programming for architectural purposes[10].

Robert Aish, a quintessential thinker in architectural computation, brings an intriguing question forward: do program developers make design tools more intuitive so that the designers do not have to be confronted with underlying abstractions (algorithm or design principles), or do developers make design tools more intuitive so that the underlying abstractions are more easily understood by the designers?[11] In explanation to that paradoxical

---

[7] Kostas Terzidis, *Algorithmic Architecture* (1st ed. edn.; Amsterdam ; London: Architectural Press, 2006), p. 59.

[8] Email from Roland Hudson, 19.10.2011: my acquaintance with Roland was quite accidental. I was looking for some sources of information about parametric design and I found his thesis and his articles in this regard. He helped me to make a contact with Ian Simpson Architects in Manchester and also agreed to answer my questions via email. Roland has valuable experience both in the education and in practice of parametric design. He managed several workshops in the UK within the years of 2007-2010 while he was working on his PhD thesis, 'Parametric Strategies in Architecture', at the University of Bath.

[9] 'We make a tool and the tool makes us... design tools that lead towards the production of novel concepts, ideas, or forms, which, in turn, have an effect in the way designers think thereafter. That way of thinking is incorporated in the next generation of tools that will, in turn, affect the next generation of designers, and so on'. (See: Kostas Terzidis, *Algorithmic Architecture* (1st ed. edn.; Amsterdam ; London: Architectural Press, 2006), p. 22.)

[10] There are courses in some British schools of architecture such as Bath University and AA School in London which specifically investigate programming approaches in architecture. Patrik Schumacher who has been both the leader of the Architectural Association Design Research Lab (AADRL) and the partner of Zaha Hadid Architects (ZHA) for several years interestingly explains in his recent books, *the Autopoiesis of Architecture*, that the schools of architecture should become 'labs' in two distinct ways: first, 'to scan society to find architectural problems and define briefs even if no client has yet articulated them. This updates the agenda of architecture and thus helps architecture to anticipate challenges rather than waiting to be pushed by a client. Secondly, to chart potentials that might inspire the search for problems based on discovered 'solutions''. (See: Patrik Schumacher, *The Autopoiesis of Architecture: A New Framework for Architecture* (Wiley, 2011b), p. 138.)

[11] Robert Aish, 'Designing at T + N', *Architectural Design,* 81/6 (2011), 20-27, p. 25.

position, Aish argues that software developers design an abstract artefact: a design tool. Tools transmit advantage from the tool-maker to the tool-user. These advantages become a convenience, and the convenience can simply become a convention: 'some tools, or the way some tools are used, make certain forms or processes 'convenient' and this can have a deeply conservative influence on design'[12]. Additionally, this sort of convention can produce a paradox in tool-making, which arises from attention to creativity. Aish posits that 'a truly creative tool is one that when used by a perceptive designer creates results beyond those envisaged by the original software developer'[13]. This is a statement that appears logical if one considers the meaning of the design creativity. Therefore, Aish clearly proposes that designers must think beyond their tools and not limit themselves by the current programs available to them. In his view, this is a mutual challenge and the solution is that designers must become their own tool-builders.

Although Aish's stance sounds quite reasonable here, the biggest challenge is to make such a position feasible, since there are still problems on the educational side. Perhaps Aish's tool-making idea will work better with future generations of architects. Practically, a rather middle-ground standpoint regarding tool-making and tool-using (although it is closer to the tool-making outlook) would suggest that architects must learn to use a coding language to connect the current software packages in a more efficient way. In this approach, the designer is looking in from outside[14] of these packages while trying to join these programs. Since most of the platforms have scripting capacity, designers can write 'a list of services' that allows them to join several programs.[15] By this method, they can generate a form in a program which is progressive in form-creation and then connect it with another package, which is fitted in another design task such as rendering. Through the computational lens, this method can be assumed as an algorithmic procedure. It creates 'pipelines between various software platforms', letting architects build a 'non-linear form generation process'[16]. The explained middle-ground approach is also quite feasible in terms of the educational concerns, since it is not practical for current architects and practitioners to spend a considerable amount of their time educating themselves and learning computer programming. Hence, this outlook tends to be more popular among practicing architects, especially those who have used scripting techniques via programming languages such as AutoLISP inside software such as AutoCAD.

---

[12] Ibid, p. 27

[13] Ibid

[14] Interview with Daniel Richard, 8.11.2011: 'What I am doing is rather than creating from inside a package. I am operating from the outside in an open source programming manager and then I am sort of hacking into simulation software, hacking into form generation things like 3DS MAX'.

[15] Ibid

[16] Email from Daniel Richard, 31.10.2011

Interviewing practicing architects shows that parametric design follows this last stance and tries to situate itself in-between the two aforementioned camps (tool-making and tool-using), since in this way the design activity will run more efficiently.

Having investigated the objective side of the designer, attention will now be concentrated on the subjective ground of inquiry, the role of a designer's idiosyncratic manner and intuition. This way of seeing design activity is high on the agenda when designers talk about their projects. In one sense, the challenge for architects has always been to 'try to be creative'[17], although they are restricted by their clients. Therefore, architects normally tend to accustom themselves to a thinking framework. This is clear in the words of John Lee, an experienced architect, when he explains his mentality and the way he considers parameters in design:

I am persuaded by Aalto's description of his working methods where he absorbs all of the facets and aspects of the brief. Then he forgets them and draws and because he does not forget them in reality. When I am designing something, I imagine a sieve in my head with the ideas that are building it and the parameters are on top of this… deciding which one is going to drive the form and drive the disposition, the organisation, and the brief[18].

Lee's statement presents his individual approach to parameters in design. In this sense, it is unique. Yet, in terms of the generation of architectural form, two distinct approaches among architects can be recognised: form-making and form-finding. Based on parametric literature, form-making refers to the process of invention which arises from a top-down approach to design and deductive reasoning, while form-finding expresses a discovery using the bottom-up strategy in the production of architectural form, benefitting from inductive reasoning. According to Coates, standard problem-solving is mostly top-down. It is assumed that 'we know what the problem is as a whole; we just need to find out what the components are'[19]. On the contrary, approaches such as the evolutionary strategies are mainly bottom-up, considering that 'we can define the low-level components of a problem; we just do not know how they can be combined to form a solution'[20]. The parametric approach can therefore be situated in the form-finding outlook. However, the problem with Coates's view is the lack of clarity in what he means by 'standard problem solving', since the nature of design is 'problem solving', but what makes it different among architects is the approach that they employ to solve the design problem. Similarly to Coates, Toni Kotnik argues about qualitative and quantitative approaches, which are characterised by movement from the general to the more

---

[17] Interview with Rick Sharp, 4.11.2011

[18] Interview with John Lee, 28.10.2011

[19] Paul Coates, *Programming.Architecture* (Abingdon: Routledge, 2010), p. 167.

[20] Ibid

specific and vice versa[21]. For Spuybroek, these two approaches are named 'grid' and 'network'. Where 'grid' is the reminiscences of the traditional top–down outlook, 'network' presents the bottom-up view in which material interactions play a significant role in the generation of form[22]. Additionally, Neil Leach describes how this second approach is one of the features of current attempts in design, seeking an approach that can 'challenge the hegemony of top-down processes of form-making, and replace it with a bottom-up logic of form-finding'[23]. However, the following question then emerges: to what extent are these positions really implemented in the messy land of architectural practice?

Arguably, it is difficult to say if today's architect is making a form or finding a form. Every architect has different interpretations about these two approaches to design. What is widely experienced in reality is aesthetics, as well as detailing and articulation of the final finished form. For some architects, this is the place 'where the power of the architectural expression lies'[24]. Hence, prioritising form over function, or thinking in the frame of making or finding, would result in restriction. Therefore, it is quite difficult to validate such a dichotomy by separating form-making from form-finding, since the idea of 'finding' or 'making' can bring different layers of meaning. When architects explain the process through which they normally design, they use both terms. They start with making the architectural alternatives for the form of the building, while they end the project by finding a form to suit the functions of the project.[25] Thus, most designers cannot make a distinction between these two points. Similarly, Greg Lynn explains that the aim of parametric tools is to give the architect the power of blending the hierarchy of the parts and the whole. Therefore, talking about a top-down or a bottom-up approach may look rather native:

Unfortunately, the initial response to parametrics was an abdication of the problem of the design of the whole in favour of the programming of the component. The use of parametric software is all about the design task shifting from either top-down or bottom-up to the territory of parts-to whole fusion. I shy away from words like 'feedback'

---

[21] 'A qualitative design approach is characterised by a tendency from the general to the more specific, with the final form a result of a search process guided by constant reduction of vagueness. In contrast, a quantitative design approach is characterised by a tendency from the specific to the more general, with the final form a result of a controlled process of constant enhancement of primary shapes'. See Toni Kotnik, 'Algorithmic Design. Structuralism Reloaded?',
<http://www.schwartz.arch.ethz.ch/Publikationen/Dokumente/Structuralismus.pdf>, accessed 6 January 2012.
[22] Lars Spuybroek, 'Wetgrid: The Soft Machine of Vision', in Neil Leach (ed.), *Designing for a Digital World*
 (London: Academy Editions, 2002), p. 96. Spuybroek explains that grid is one of the oldest tools in architecture. 'If an architect wants to create order… he jumps into his metaphorical helicopter. Flies up, and drops a grid onto the situation. It is a military action… The grid is directly connected to the top-down view itself and to the top-down view of order and coherence at the same time… it is quite different from a network. The network is of course the contemporary view of emerging order, of self-organisation, of bottom-up coherence. So, the first view is where order is forced upon matter, from above, while in the second view order emerges out of material interactions; pattern, stability, order on the edge of chaos'.
[23] Neil Leach, 'Digital Morphogenesis', *Architectural Design,* 79/1 (2009), 32-37, p. 34.
[24] Email from Griff Evans, 17.11.2011
[25] Interview with Trevor M. Cousin, 4.11.2011: 'I think quite often, we find the form to suit the function of the building… going back to the Arthur Lewis Building again, I mean, we started that with form-making, ended project with a form-finding to suit the project'.

and 'synergy' between parts and whole because so far the experimental architects have just jumped from top-down determination of parts to bottom-up determination of wholes … I find this theoretically naive and it avoids the most interesting thing about parametrics, and that is the ability to fuse the hierarchy of parts and whole to produce a deeply modulated whole as well as infinitesimal variation among parts[26].

It seems that these two approaches go together to help the designer come up with a finalised scheme. At one point, designers suppose that they are making something and, at the same time, they are finding a solution for the design problem.[27] Therefore, even though it is frequently argued in academic literature that there are two rather distinct sides towards form, this distinction looks not to be valid in the voices of practicing architects. Even though architects have always been publically identified as form-makers, they undertake much form-finding in their design:

I would say it is an inventive process. I mean you are thinking ahead to the questions about parametrics… the angle of the sun and the optimum floor size produce a given value punching those parameters in the form; you will find the form that ideally suits that and I think for the most part, it is the other way round… Well, I am just finding the form that arises from me and my situation and my response to the city. So maybe this distinction between form-finding and form-making is an artificial one…[28]

Nonetheless, some believe that today's architectural practice in the form of parametric design is closer to the form-finding approach due to the emphasis on material performance and the process-driven tendency, instead of the appearance aspect and representational viewpoints. As Branko Kolarevic explains, this shift from the 'making of form' to the 'finding of form' becomes feasible through replacing 'the stable by the variable' and 'singularity by multiplicity'[29]. Taking a neutral perspective on form-making or form-finding, we can observe the concept of synthesis, which focuses on having a performative design instead of characterising the top-down or bottom-up approaches. This rather distinct yet widespread vantage point tries to focus on performance and the process in which a form is executed rather than its invention. Proponents of this approach declare that 'architecture should perform rather than simply form'[30]. Thus, instead of focusing on formal approaches like form-finding or form-making, architects should seek a synthesis of ideas about materials, their assembly or performance, and use all of these to drive the form. A good example in this regard is Lars Spuybroek's reference to Deleuze's example of 'sword-making'. Spuybroek argues that a designed object is a 'history of different states of material' that architects need to negotiate

---

[26] Ingeborg M Rocker, 'Calculus-Based Form: An Interview with Greg Lynn', *Architectural Design,* 76/4 (2006), 88-95, p. 92.

[27] Interview with Matthew Smith, 3.11.2011: 'I cannot make a distinction. I do not need… I suppose it uses making but it sounds like you are finding'.

[28] Interview with John Lee, 28.10.2011

[29] Branko Kolarevic, *Architecture in the Digital Age : Design and Manufacturing* (New York, NY: Spon Press, 2005), p. 13.

[30] Michael Meredith, 'Never Enough', in Tomoko Sakamoto and Albert Ferre (eds.), *From Control to Design : Parametric/Algorithmic Architecture* (Barcelona ; New York: Actar-D, 2008), p. 9.'Architecture should perform rather than simply form; structurally, environmentally, economically, programmatically, contextually, or in multiple formal arenas'.

with in order to tease out the 'best sword' possible[31]. Here, the capacity of materials and their performance defines a framework for the generation of form. As a result, the ultimate form will be highly manageable and flexible in terms of architectural alterations, and closer to the reality of construction in terms of its practicality.

The 'performative turn' as Leach explains, as well as Spuybroek's example stated above, can now be explored through new digital techniques[32]. To do this, architectural form must be 'informed by considerations of performative principles to subscribe to a logic of material formation'[33]. Performance must be assumed as a determinant for the creation of architectural form. In order to make it achievable, the translation of material characteristics into the language of the digital must be signified. It entails an acquisition of a thorough knowledge of such materials and a recognition of possible paradigmatic taxonomies. For instance, Greg Lynn argues that two possible paradigms can be imagined about materials: the first one, which is traditionally established as the tectonic paradigm, deals with how to put things together and make larger parts. It requires the arrangement of components, a hierarchy of systems and assembly, discrete layering and superposition. The second is the plastic paradigm, based on fusion of materials in a matrix, layering without distinction, fibres over members, and a woven orientation[34]. However, Lynn's explanation is just one example of how paradigmatic taxonomies can feed architectural creation and how the design process can be reviewed in accordance with the materials.

Nevertheless, it has been said that 'algorithmic control of the processes of parametric variations is one of the methodological cornerstones of future performance-based systems'[35]. Fortunately, many parametric programs have these synthesised features. They are quite adapted to these facets of design, giving designers the ability of controlling the parameterised form through the manipulation of its parameters. Again, this question arises: if the synthesis of form is considered the overarching goal of design, to what extent does it depend upon tools like computer programs? Broadly speaking, to what extent are the designed forms dependent on the software? The next sections will try to answer such questions by investigating the role of sketching and computer programs.

---

[31] Lars Spuybroek, 'Wetgrid: The Soft Machine of Vision', in Neil Leach (ed.), *Designing for a Digital World*
(London: Academy Editions, 2002), p. 100.

[32] Neil Leach, 'Digital Morphogenesis', *Architectural Design,* 79/1 (2009), 32-37, p. 35. Also in p. 34, he explains that this is 'a move away from an architecture based on purely visual concerns towards an architecture justified by its performance. Structural, constructional, economic, environmental and other parameters that were once secondary concerns have become primary'.

[33] Ibid, p. 34

[34] Umtaubmancollege, 'Greg Lynn: University of Michigan Taubman College Future of Design', *the Future of Design Conference* <http://www.youtube.com/watch?v=AqmN0-zVAK8>, accessed 18 February 2012.

[35] Rivka Oxman, 'Performance Based Design: Current Practices and Research Issues ', *IJAC International Journal of Architectural Computing* 6/1 (2008), 1-17, p. 6.

5.1.2. The Role of Sketching and Physical Modelling

As the archetypal design medium is pencil and paper – more precisely, pencil for 'adding', eraser for 'subtracting'[36], and paper for recording – to what extent is sketching still key in parametric design? Does the parametric designer still need to sketch or make physical models? Traditionally, these two, especially sketching, have been emblematic of externalisation. For many architects, sketching means 'how to think visually'. However, a study on the process of sketching shows that even though sketches and externalisations in general are claimed to be central to the design task, they are not essential activities for expert architects in the early phases of conceptual designing[37]. Therefore, we can argue that, if sketching is still seen in parametric design, is it possible to map other roles to it? The answer is important, since if sketching provides multifarious characteristics it cannot be simply marginalised by the software.

In addition, sometimes a form is too difficult to draw by hand. What would be the solution for such a situation? As Mario Carpo discusses, the last resort of the designer may be to abandon the modern design process altogether and return to the traditions: 'If you can't draw what you have in mind in order to have others make it for you, you can still try to make it yourself'[38]. In fact, Carpo addresses two issues here: first, a sort of design activity which is independent of sketching, perhaps a kind of form-finding that was carried out in the past by architects such as Antoni Gaudi; second, when 2D sketching is not useful for the embodiment of ideas, architects can shift to 3D thinking by models. Although this might reduce the importance of sketching as a tool for visual thinking, it cannot ignore the rest of its advantages. For instance, thinkers like Spuybroek argue that sketching is not just a simple externalisation. It is a process in which an architect reduces the complexities of design[39].

In contrast to what is seen in literature and researches on sketching, speaking with architects demonstrates that they still prefer to initially sketch or make models in spite of the existence of very sophisticated programs. Even though CAD packages such as Sketchup are a continuation of sketching techniques, giving the architects a high capacity of manipulation during the early stages of design, shortcomings such as the lack of presenting a tangible environment for drawing pose a barrier to adoption. As a result, most architects appear to contradict the belief that they should start designing on the computer screen. They see

---

[36] Robert Woodbury, *Elements of Parametric Design* (London: Routledge, 2010), p. 11.
[37] Zafer Bilda, John S. Gero, and Terry Purcell, 'To Sketch or Not to Sketch? That Is the Question', *Design Studies,* 27/5 (2006), 587-613
[38] Mario Carpo, *The Alphabet and the Algorithm* (Cambridge, Mass. ; London: MIT Press, 2011), p. 32.
[39] Lars Spuybroek, 'Wetgrid: The Soft Machine of Vision', in Neil Leach (ed.), *Designing for a Digital World* (London: Academy Editions, 2002), p. 98.

sketching as an inseparable component of the design process which, while its use in design has been reduced with the sharp rise of employing computer programs, is not yet possible to leave it out.[40] Furthermore, some architects believe that sketching helps them to crystallise their concepts and arrange them in an organised format. Therefore, hand sketching is for them a 'place of clarity'.[41] Matthew Smith, an architect and expert in parametric software, explains that, despite the great usage of parametric software in his practice, he and his colleagues need to draw to make sense of what is being designed:

In most processes it starts with sketches… in this practice we do a lot of sketching and we do a lot of models… that's going all the time and then, you know, even if you have got something in CAD or parametric models or script it parametrically... still draw a detail or things like that to make sense of the whole[42].

Although, as Smith explains, sketching can help architects make sense of what is designed parametrically in a computer program, on some occasions it is used for other reasons. Sometimes, the need to do traditional things such as sketching or modelling arises from a client's request and sometimes it is done to reassure the client in the case of any false preconceptions or misunderstanding of the design. The clients 'can be blind by an image sometimes rather than actually being able to look at something physical and to understand it'.[43] Therefore, they ask for a sort of guideline, and the best idea is to show them a real thing such as a maquette to make sense of the entire project. However, in some cases this need has another cause. Especially in large-scale projects, or those which have a commercial base, the project demands showcasing through what can be termed a 'chic' or 'stylish' model. Therefore, model making becomes a serious stage not only for architects, but also for the clients to take advantage of by putting it in 'their reception or the exhibition place'.[44]

Sketching is sometimes assumed as a social actor that brings all of the design team together for a discussion on alternatives or possible decisions[45]. In this case, all of the principals, associates and architects in a company sit around to 'design things together'[46];

---

[40] Interview with Trevor M. Cousin: 4.11.2011: 'personally, I always do sketch on paper to get some form, graphic writings and then try... But I still find it very difficult to just go straight to software and start drawing something with no need to a piece of paper...'

[41] Interview with Amy Hanley, 25.10.2011: 'I have always found the hand-sketching my place of clarity…To clarify my thoughts and ideas. Because I do not have to think about what the pen is doing while the pen is. I am drawing…Yeah, I find it much easier to sort of clarifying the ideas by quickly sketching that focuses purely on the ideas'.

[42] Interview with Matthew Smith, 17.10.2011

[43] Interview with Rick Sharp, 4.11.2011

[44] Interview with Trevor M. Cousin, 4.11.2011: 'we also do physical models as well… because clients still quite often need to 3D… something that clients all request once the design is finalised… they just want physical models to put in their reception or the exhibition place'.

[45] Interview with Phil Osborne, 7.11.2011: 'usually our design responses start with sketches, hand-drawn sketches and a discussion in a design group'.

[46] Interview with Matthew Smith, 17.10.2011

designing through discussion or 'talkitecture'[47]. In addition, sketching can be considered an attempt to preserve the power of public decision-making and to go beyond the computer screen and software packages which have alienated practitioners. In this view, the architecture office per se, as Meredith argues, is parametric, constructed of opinions and persons. Everything in an office can be internalised as a parameter[48]. Therefore, the raison d'être of sketching in an architecture office is sometimes motivated by more than the need for a tool for individual thinking. It establishes connections amongst the thoughts of the design team. Consequently, it cannot be dependent upon the approaches to design, whether parametric or not.

In order to conclude this discussion on the role of sketching and physical modelling, it is worth mentioning that there are many voices underscoring the role of software in parametric design; some like Schumacher even talk about sketching in Maya or Rhino to explore 'radical design ambitions'[49]. Yet the role of sketching and modelling is still valid and recognisable in today's architectural practice. However, if its fundamental role in the past was essentially summarised as thinking visually, in the parametric approach it is shifted to other domains such as clarification of ideas (since sketching is free from software constraints), or collaborative thinking in a design team.

---

[47] Some researchers believe that design can only be learned by doing. Woodbury explains that 'Talkitecture' is a derogatory term, reserved for those who discuss but not draw'. (Robert Woodbury, *Elements of Parametric Design* (London: Routledge, 2010), p. 9.)
[48] Michael Meredith, 'Never Enough', in Tomoko Sakamoto and Albert Ferre (eds.), *From Control to Design : Parametric/Algorithmic Architecture* (Barcelona ; New York: Actar-D, 2008), p. 9.
[49] Patrik Schumacher, 'Interview: Patrik Schumacher by Feng Xu', <http://www.patrikschumacher.com/Texts/Interview_WA_May%2009_english.htm>, accessed 26 January 2012.'At the current relatively early stage of parametric design it is more fertile to start sketching in Maya or Rhino – including Mel-script, Rhino-script and Grasshopper – to explore the radical design ambitions of parametricism. This way of working requires a later remodelling/rationalization in more precise parametric design systems - like Digital Projects – that are better capable to take fabrication constraints into account'.

5.1.3. The Role of Computer Programs

The position of the computer, as discussed in Chapter 2, reflects an ambiguity in the realm of architectural design. What is discussed as the computer is not usually the object itself, but a reference to the idea of software packages and their functionality in the design process. In addition, the exact scope of the computer's involvement in architectural design is still unclear for many researchers. For instance, Antoine Picon believes that the debate over the role of the computer in design has been present since the beginning of computer-aided architecture[50]. The second issue arising from that debate is about the character of the computer per se beside an architect, and the interactions between these two actors. For instance, one of the surprising, yet interesting, stances is Lynn's description of the computer as a pet whose wildness must be controlled and domesticated[51]. Having given it an 'anthropomorphic'[52] delineation, Lynn also posits that the computer is neither a brain nor nature, since it does not think critically and is unable to create organic shapes[53]. In response to Lynn's position, it is clear that the increasing sophistication of computer technology will disprove his statements. Deploying genetic algorithms in packages such as Grasshopper or Kangaroo[54] is one piece of evidence that organic shapes can be created, even though they are not yet as flawless as those of true natural origin. Having said this, the roots of statements like Lynn's are more important than their validity. These roots should be sought by exploring what the computer brings to the design process; in other words, its functionality.

Through this outlook, functionality can authorise two distinct usages of the computer in the design process: computerisation and computation. While the former often includes the use of the computer as a representational tool in which architectural objects are designed, the latter stresses the position of the computer as a tool for discovery in design in which

---

[50] 'Algorithmic architecture or the computer as a double': an introduction by Antoine Picon to the book *Algorithmic Architecture*. Picon starts his debate by questioning the role of computer.

[51] Greg Lynn, *Animate Form* (New York: Princeton Architectural Press, 1999), p. 20: 'the computer might be considered as a pet. Like a pet, the computer has already been domesticated and pedigreed, yet it does not behave with human intelligence… Its wildness must be controlled'.

[52] Kostas Terzidis in his book *Algorithmic Architecture* uses this term to clarify Lynn's idea about the role of computer in design.

[53] Ibid, p. 19: 'Computer is not a brain… Machine intelligence might best be described as that of mindless connections. When connecting multiple variables, the computer simply connects them, it does not think critically about how it connects' -'The computer is not nature. Although it makes shapes that are temporally and formally open to deformation and inflection, those shapes are not organic'. However, Lynn in an interview explains that 'computer simply is another design medium'. He describes that 'I really wanted to explore the software medium at a basic as well as a deeper level, and I have always started with the first principles of the software and its geometry engines. Too many people understand the computer in terms of facilitating expression or in terms of a pseudo-scientific system when it is more simply just another design medium.' See: Ingeborg M Rocker, 'Calculus-Based Form: An Interview with Greg Lynn', *Architectural Design,* 76/4 (2006), 88-95, p. 90.

[54] Grasshopper is a plug-in for Rhino programmed by David Rutten. Galapagos is another applet written by Rutten based on genetic algorithm. Kangaroo is similar to these two, but programmed by Daniel Piker. For more information, see Architecturalassociation, 'David Rutten/Grasshopper, Computing Architectural Concepts ', <http://www.aaschool.ac.uk//VIDEO/lecture.php?ID=1212>, accessed 9 October 2011.

architectural objects are calculated. Having epitomised such classification, the question as to which category embraces parametric design. One the one hand, the specificity of parametrics is nearly equal to the use of computation in architectural design. On the other hand, some practicing architects point out that this is not necessary, since the notion of parametric design is much greater than its attachment to computer programs[55]. Perhaps part of the answer should be sought in the origins of parametric design, for which we need to consider whether it was technological growth that led to the flourishing of the parametric approach, or an effort inside the domain of architecture in order to systematise the design process. In parallel to the discussed duality (computation versus computerisation), two positions can also be understood in architectural practice. For the majority group which thinks that the computer is just an advanced tool providing facilities for design, the parametric approach is still a sort of computerisation. Hence, it is not necessary for one to go deeper into the details of complicated tasks such as computer programming. Yet computation falls within the ambit of the second group's ideas: those who believe that it is impossible to be creative with computers without knowing a programming language. For this group, programming is the key that can open the black box of design more effectively and, as a result, not only parametric design, but also all other approaches to design, have to put it high on the agenda. Pondering these two discernible outlooks gives rise to another key concern: why do these two groups have a different view on computers?

While most of the practicing architects in the first group declare the primacy of computer software throughout design, it is still a tool in the toolbox helping them to engineer the design process. As a result, architectural form 'rarely arises directly from computer-aided design'[56]. Yet the software is all grist for the architect's mill, making the manipulation of the form 'easier, more fashionable, and more mainstream'[57]. In other words, it helps architects to explore formal possibilities[58] as well as to 'develop the right response to the climate, the environment and the brief'[59]. Accordingly, computer packages here become facilitators rather than design tools. They alleviate the difficulties experienced within the design activity; however, they are still dependent on being used correctly by architects[60] and on the successful

---

[55] Interview with John Lee, 28.10.2011: 'I am a parametric designer, but I do not use the computer. Does that make sense?!'

[56] Email from Griff Evans, 17.11.2011

[57] Interview with David Wood, 1.11.2011. He also mentioned: 'I do not know to what extent the software you choose is the best you end up with. I would like to think that your design is in the form that you like, and then [you] make the software produce the form you like'.

[58] Interview with Phil Osborne, 7.11.2011: 'well, all the production of information is done with computers and research is done on computers. Exploring form is aided by computer; but done by the thought process of the individuals'.

[59] Interview with Alex Solk, 22.11.2011

[60] Interview with Rick Sharp, 4.11.2011: 'it helps you to develop things. But it only helps in terms of how you use it… it is much more about what is going on in our heads and how we use the software'.

transfer and embodiment of ideas on the computer screen. As Kostas Terzidis expresses, in large international firms such as Skidmore, Owings and Merrill (SOM), the situation is the same. They utilise the computer as an efficiency tool while developing the entire design through traditional manual means[61]. The reason why computers are used as tools for organisation, productivity, and presentation can be different; perhaps this group of architects are unaware of the inner logic of computers. Terzidis adds that even famous architects, such as Neil Denari, Greg Lynn, or Peter Eisenman, call computers 'tools' to describe computational processes, yet none of them have any formal education in computer science[62]. Notwithstanding these beliefs, Terzidis considers himself in the second camp – those who think that the computer is more than a tool.

In comparison with the above practice-based opinions, researchers on the scholarly side try to accommodate themselves in-between by referring to different interpretations of the term 'tool'. Although implicit, employment of the term leads them to be proponents of the first bloc. For instance, Mario Carpo argues that, despite all of the unlimited versatility that a computer offers, it is still a tool, 'a technical mediator that in this instance is interposed between a designer and an object of design'[63]. Striking a sterner tone, Antoine Picon states that although this tool has changed many things, 'it has left the core of architectural thinking still totally dependent upon the designer's intuition'[64]. Thus, it is not a true partner in the process of design conception.

Similarly, for most of those who contributed to this research as interviewees, computer programs speed up the process of drawing or presenting architectural concepts. In this sense, they are beneficial. The case becomes even clearer if the type of programs being used in design is investigated. As mentioned in Chapter 3, in this research a questionnaire was used at the end of every interview to cover some important issues through statistics. According to the questionnaire data, it seems that the majority of companies that contributed to this research still use those computer programs which offer simple and user-friendly interfaces yet conform to the latest technology, rather than considering a specific approach in the selection of a package. Hence, platforms like Sketchup are quite popular. Figure 2 is the result of the analysis of the diversity of CAD programs being used by the interviewees and their companies. The graph clearly depicts that most of the interviewees (80%) benefit from Sketchup, which is quite surprising considering the common belief that AutoCAD is 'trendy'

[61] Kostas Terzidis, *Algorithmic Architecture* (1st ed. edn.; Amsterdam ; London: Architectural Press, 2006), p. 40.
[62] Ibid, p. 36
[63] Mario Carpo, *The Alphabet and the Algorithm* (Cambridge, Mass. ; London: MIT Press, 2011), p. 34.
[64] Antoine Picon, *Digital Culture in Architecture : An Introduction for the Design Professions* (Boston, MA: Birkhaeuser, 2010), p. 94.

among architects. In addition, programs with a parametric character, such as 3DSMAX, are also frequently used in the generation of architectural form.
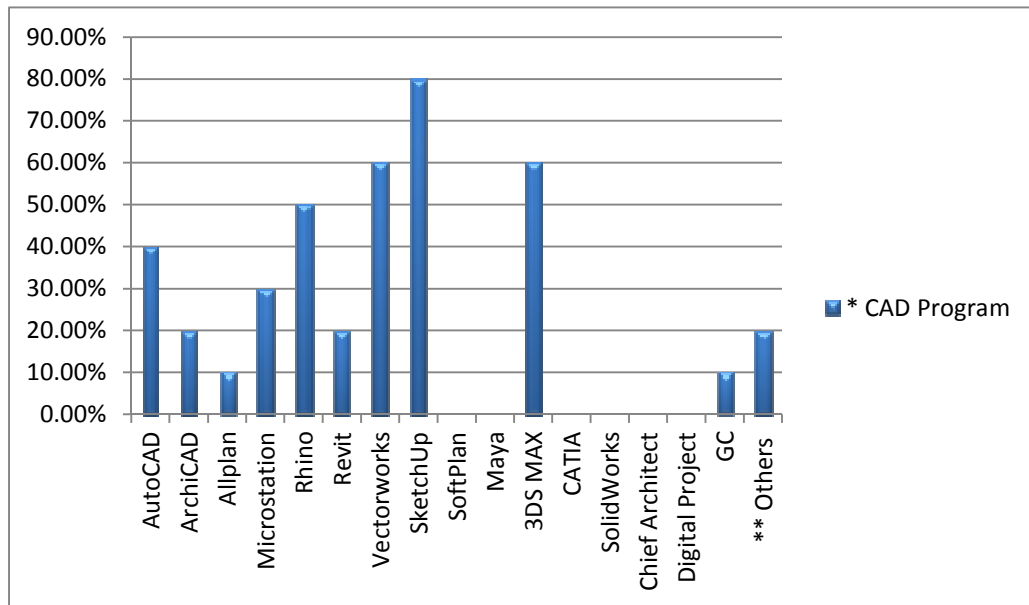

Figure 2: Diversity of the use of CAD programs among the interviewees

* Most of the recent versions of these CAD programs have adopted parametric features, so making a distinction between a parametric and non-parametric program is not possible here. Yet it could be said that some of them, such as GC, Digital Project, CATIA or Rhino, are primarily parametric rather than simply incorporating some parametric features.
**The category 'others' shows programs which do not support parametric features.

The above graph shows that Vectorworks has the same rate of usage as 3DSMAX, although it is not really known as a parametric package. Some programs like Rhino benefit from parametric plug-ins such as Grasshopper, which especially is popular with students of architecture. It is also important to note that new architectural approaches like parametrics often compel software developers to enliven their lucrative industry by adding new features to their software, justifying the release and purchase of newer versions. Therefore, it is difficult to make a distinction between a parametric and non-parametric program. The other issue to consider is the small number of participants of this study, which limits the rate of responses to several options.

The final point to make is related to the possible conflict between computer programs and sketching. It was revealed in the previous sections that, even though architects design parametrically, they still use sketching in the design process. However, to what extent are these two methods of designing consistent with each other? Is there a tension in the transition from hand-sketched ideas to computer programs? Some practicing architects find it intriguing

that an architect can go back and forth between sketching and designing on computer[65].

Perhaps architects do this because it enables them to combine the sense of freedom given by

sketching with the potential of parameterised precision in a sophisticated program. However,

the problem is the amount of data exchanged in these two states. When an architect changes

the mood of design from paper-based to digital[66], it is most probable that part of the data

produced in the first stage cannot be translated into the second. This data loss can heighten

tension among the design team. Specifically, it is seen that the first ideas are usually provided

by the chief architects or associates of the studio through sketching, while the extension and

development of those ideas is carried out by the other members of the design group, normally

architectural practitioners who are more computer literate. Therefore, it could be said that the

traditional conflict can still be seen, although at a slighter level.

To summarise, despite all of the assertions and claims articulating or even

exaggerating the role of computer programs in design, they are little more than a tool for the

majority of designers. Some factors might make architects use computers in the design process,

which will be discussed shortly. However, the extant use of sketching in design also proves that

there is still a long way of reaching the point that computers are assumed mere explorers. The

aim of all computational approaches is to promote architectural design to a level where, as

Terzidis explains, computers are acknowledged not only as machines for imitating what is

understood, but as vehicles for exploring what is not understood[67]. If this goal is reached, the

following momentous question will have to be asked: 'who designs?'[68]

The Dependency of Designed Forms on Software

Focusing on the relation between software and generated forms in the parametric design

process, two categories of questions that need exploration appear. The first is about the

possibility of existence of these forms without software and the second, which in essence

emerges from the first, considers what factors would make this dependency. Answering both

of them, especially the first one, is highly important not only for practicing architects, but also

---

[65] Interview with David Wood, 1.11.2011: 'we start normally with sketching I would say… I think it is intriguing where you go between the sketch, the cardboard model and then move to parametric design where it sets a grid model that you can alter and change the form'.

[66] Interview with Amy Hanley, 25.10.2011: 'I guess it is problematic when you are transferring between your drawing and your computer, potentially about where the information is stored and how it gets integrated into design…'

[67] Kostas Terzidis, *Algorithmic Architecture* (1st ed. edn.; Amsterdam ; London: Architectural Press, 2006), p. 59.

[68] Ibid. referring to the evolutionary process of computer in design, Terzidis explains that first, the role of computers was to take the place of humans in the design process. Later, the role was creating systems as intelligent assistants to designers which are capable of augmenting the decision-making abilities. Today, the roles vary from 'drafting and modelling to form-based processing of architectural information'. The future? It is worth exploring the role in the context of the question 'who designs?'

for students of architecture and perhaps software programmers and developers too. According to Terzidis, a common theme among contemporary practicing architects is 'the use of computer as an exploratory formal tool and the increasing dependency of their work on computational methods'[69]. Yet the question for many researchers is the possibility of design without parametric programs.

Going into the realm of architectural practice, it could be said that architects were once creating and generating architectural form in a so called free-form[70] or organic style whose history dates back to a long time ago when there was not a digital tool to help architects even carry out simple calculations. Having looked at Antoni Gaudi's architecture, such as the Sagrada Familia Church in Barcelona in Spain (Figure 3 and 4), the design of which was started in 1883, it can be clearly seen how problematic and ambitious it was to design such a massive picturesque masterpiece at that time without the aid of computer programs[71]. The project in question is one of the exemplars of a complex structure, which has enticed some designers to simulate it in a computer environment within the last decade.[72]

Two terms, 'biomimicry' and 'biomorphic', might be confused with the way that the parametric approach works. Biomimicry is about how to emulate functional aspects of an organism in nature. Likewise, the term 'biomorphic' refers to the same process, yet the focus is on formal aspects[73]. Biomorphic methods of form generation in particular are close to concepts such as Gaudi's hanging chains[74]. Perhaps the only difference is that biomorphic methods often emerge from organisms rather than natural objects. If biomorphic is defined as

---

[69] Ibid, p. 58

[70] Interview with David Wood, 1.11.2011: 'obviously, people were creating organic buildings for a long time before the tools getting presented'.

[71] Interview with Vikram Kaushal, 31.10.2011: 'I think… it is very much helpful, isn't it? Looking at Frei Otto or Gaudi you know, these guys did not need computer system…Yeah, it was probably very difficult, wasn't it… without the tools'.

[72] Mario Carpo, *The Alphabet and the Algorithm* (Cambridge, Mass. ; London: MIT Press, 2011), p. 32. Carpo explains that Gaudi built some parts of the Sagrada Familia without drawings, but supervising all and everything in person, 'as an artisan/author who explains viva voce or shapes with his hands what he has in mind. It is not by chance that Gaudi is a famous case study among contemporary designers.' One of these designers is Mark Burry (1996), a theorist in parametric design at Royal Melbourne Institute of Technology (RMIT); Burry explains the parametric aspects of Gaudi's architecture in his article 'Parametric Design and the Sagrada Familia'. Also in the book, Jane Burry and Mark Burry, *The New Mathematics of Architecture* (London: Thames & Hudson, 2010), the writers describe that 'Gaudi tackled the complexity of spatial subdivision strategies without the aid of digital computation, but computers can now be useful tools for interpreting the church's naturalistic geometrical complexity'. (p. 35)

[73] See: Thersaorg, 'Biomimicry in Architecture', <http://www.youtube.com/watch?v=ijIfdswvh5U>, accessed 06 August 2012.

[74] In this respect, Elke Krasny's book about tools in architecture seems interesting. The following notes extracted from her book briefly explain Gaudi's approach:

'Gaudi questioned classic methods and historical model forms and used nature with its wide variety of curved, structurally optimised forms as his most important source of inspiration… Gaudi was distrustful of drawing, even though he had financed his studies by working as a technical draughtsman. Despite his talent in this area, he drew as little as possible, attempting instead to work out the form of the building by means of collaboration with the skilled workers on the building site and through the use of models… If the line taken by a flexible hanging chain is inverted, it describes an upward-pointing arch… By changing the chords and weights the model arrived at the structurally optimal form by itself, so to speak – it was in a certain sense the precursor of modern-day parametric design methods…The sculptor Vilarrubias photographed the hanging model. For the photos pieces of cloth were inserted at places in the model to represent the areas of solid wall. Gaudi used these photographs, turning them upside down to paint over them so that he could examine the form of the building both inside and outside'. Please see: Elke Krasny, *The Force Is in the Mind : The Making of Architecture* (Basel: Birkhäuser, 2008), pp. 59-62.

the emulation of every natural object, then Gaudi's architecture should be considered biomorphic[75]. However, I do not intend to explain the features of Gaudian architecture or his form-finding methods to generate a form. The reference to Gaudi just clarifies answering to the question of dependency on computer programs. This is especially because the heuristic methods of natural computation used by Gaudi, and later by Frei Otto, the German architect, bought huge impacts not only in terms of theoretical positions in the scholarly realm, but also among practicing architects. For instance, Lars Spuybroek introduced his method called 'Wet Gird' by referring to Gaudi and Otto. In the description of his approach, Spuybroek explains that these two architects used 'material computers to calculate shape and structure'[76]. Computers for them were any tool helping an architect to create a structure from material properties. In a deeper sense, it could be said that for many researchers Gaudi and Otto have been pioneers of form-finding in architectural design; perhaps the first parametric designers, albeit without any computer tools. Consequently, although it is a herculean task to design without sophisticated parametric packages, it is not impossible. Therefore, to pose the questions of feasibility and dependency, why do architects tend to work with parametric software?

Figure 3: Sagrada Familia Church

Figure 4: Sagrada Familia Church (Entrance)

---

[75] It is important to mention that there is a difference between natural objects and organisms. Natural objects such as bubbles embrace a broader domain. In particular, bubbles have been a source of inspiration for architects including Frei Otto.

[76] Lars Spuybroek, 'Wetgrid: The Soft Machine of Vision', in Neil Leach (ed.), *Designing for a Digital World* (London: Academy Editions, 2002), p. 98. About his method, Spuybroek describes that: The 'wet grid' is then an in-between situation'. Like a 'liquid crystal'… it is neither lines nor surfaces. Wet grid is similar to 'optimised path system' by Frei Otto. It is also very close to Gaudi's suspended chain modelling technique. (ibid, p. 97)

This question offers two ways to respond. If it is answered through the framework of style, akin to what Schumacher believes ('Parametricism can only exist via sophisticated parametric techniques'[77]), then the dependency will be a necessity, since it is impossible to compete with contemporary practices without mastering these techniques. This framework is defined by computers, and the discussion would be utter nonsense without it. However, if parametric design is considered outside of any paradigmatic or stylistic approach, then the question of dependency will be valid. The next section investigates the influential factors of this dependency.

What Features can be given to Architects by using Computer Programs?

As discussed, this question arises that what factors might make architects dependent upon software? In search of the hallmark of this dependency, these main issues can be discovered by speaking with practicing architects.

Among all of the influential benchmarks, the size of the project is an overarching indicator, arising directly from the client's demand and consequently the brief. It is said that, for small or even modest buildings, architects can work out many issues such as environmental considerations manually.[78] Therefore, the question of dependency greatly loses its importance. Even in medium-sized projects, many formal explorations also can be achieved in traditional CAD. However, in large-scale designs or commercially-based projects, the use of parametric programs appears to be a necessity.

Time is another concern which is usually relevant to the workflow within the process. Parametric programs considerably reduce the time required for form exploration. In addition, they help architects to search and find certain spaces[79], whereas without them these spaces perhaps will remain in the architects' minds forever.

On some occasions, architects need to create a more realistic model, not only in order to understand the problems of the proposed design and its structural pitfalls, but also to

---

[77] Patrik Schumacher, 'Design Research within the Parametric Paradigm', (updated 6 July 2011) <http://www.patrikschumacher.com/Texts/Design%20Research%20at%20Zaha%20Hadid%20Architects.htm>, accessed 10 January 2012.

[78] Interview with Alex Solk, 22.11.2011

[79] Interview with Matthew Smith, 3.11.2011: 'Definitely, it is possible to design without them… in terms of using parametric things something like that; it just makes a lot easier, unless you need time… unless maybe to find certain spaces'.

present a down-to-earth sensible alternative to clients. This is the place where architects would find parametric software an invaluable tool. This is where the software 'works well'.[80]

5.2. Concluding the Roles in the Parametric Realm

By bringing two definitions of design activity into the parametric domain (design as a representation and design as a computation) in this section, I tried to answer the set of questions relating to the parametric design process. The aim of these questions on roles in the design process is to investigate to what extent parametric design brings changes to the role of designers or computer programs. Having focused on such concerns, two camps of designers in relation to tools were explored. Both of these groups may have their own definition of design. However, it is not surprising that the tool-making camp has its own proponents among computational designers – those who see design as a sort of calculation. On the other hand, tool-users are those who stress that design is still the act of representation of ideas embodied in architects' minds. Like most of the scientific spectrums, there is a grey area of possibility, an in-between voice which perhaps might be considered more pragmatic, a view which refutes both the narrowness and passivity of tool-using, and the complexity and boldness of tool-making. It seems that, inside the architectural practice, this third view is gradually becoming a promising one for attention and elaboration.

An investigation into the role of non-human actors does not make any major distinction between the parametric approach and traditional design. The role of sketching and physical modelling is still fundamental in those offices using parametric programs; however, some minor shifts are seen in the recognition of the status of sketching. Whereas in the past sketching was assumed as the main activity for developing concepts, now this capacity is slightly weakened by the use of robust computer programs. Yet sketching still acts as a collaborative tool for thinking and discussion over a design project.

Lastly, statements by theorists such as Patrik Schumacher illustrate the tight interrelationship of computer programs and parametric design, underlining the unattainability of parametric forms without software. However, there is evidence disproving such assertions. Looking to the past and considering the works of architects such as Frei Otto or Antonio Gaudi is concrete proof. These architects did not have sophisticated computer programs. Instead, they used a sort of 'natural computation' to produce forms which are now assumed as parametric. More importantly, architects are not able to change the brief, although they can

---

[80] Interview with Amy Hanley, 25.10.2011

produce eye-catching forms by the aid of software[81]. The design activity is still based on how architects interpret the client's demands and extend that to attain a design solution. Some factors, such as time or the size of the project, force architects to use parametric programs in order to tackle the design problems. However, it is still possible, yet quite difficult, to design without software. Due to this reasoning, parametric programs cannot change the design process and the nature of design is still the same, even though parametric design provides a new vantage point to the generation and manipulation of architectural forms.

---

[81] Interview with Phil Osborne, 7.11.2011: 'our process... is still focused on interpreting the brief and developing the solution, the response to the clients' needs. The software we use will all be geared to serving that mood... You do not go to the client and say: 'well, our latest design tool generated this. Will you change your brief?''

Chapter 6

Parametric Design: Is It a Style or a Set of Techniques?

Having focussed on the parametric concerns of the design process, this following chapter take a broader perspective to investigate the roots and the position of parametric design. The inquiry focuses on the second thematic part of the research, which investigates the parametric realm among architects to understand if it can be recognised as a new architectural style or just a set of techniques. Some ambiguities about the assumed dichotomy between the two will be described and, lastly, a possible solution will be explained.

The Question of Style

As discussed earlier (Chapter 2), the introduction of the computer was quite influential in design. It ushered in systemic views of the design process. Toni Kotnik argues that the diffusion of systemic notions and concepts from science into the architectural discourse is still being explored for design purposes[1]. Plainly, design in the parametric approach is the tautological image of quantity, depicted in parameters and variables. Therefore, it could be argued that architects have strived to go towards the quantitative approach whereas they were previously inclined to contemplate qualitatively, emphasising the role of intuition. For Michael Meredith, parametric design fits within 'an evolution of so-called postmodernism'[2]. If design can be considered an evolutionary process rather than being stagnated in an absolute end which cannot tolerate metamorphosis, I believe Meredith's view would sound quite logical. Yet to consider this we must make a believable connection between postmodernism and parametric design. To put it another way, does the origin of parametric design come back to postmodernism? How do architects think about the roots of parametrics?

The general answer to the question of origins would be in regular technological advances that make architecture upgrade itself with a selection of brand new capabilities. These novel ways help architects to facilitate design more efficiently. Still, this permanently in-progress paradigm is lurking in the margins of design, since this cannot 'replace the original thought'[3] in the process of thinking. Aside from technological advances, which are too broad to be assumed as a root, Patrik Schumacher believes the origins of parametrics lie in the transformation from 'societies of Fordist mass production to post-Fordist network societies'[4]. This stance, although it is narrower than the first, still sounds equivocal. Similarly, the movement from standardisation to customisation, as stated by scholars such as Mario Carpo[5], does not offer a clarified position regarding the case of origins, since the broadness of the rationale eliminates any possible deductive chain resulting in the current state of parametrics.

Probably, one of the roots can be seen in the need for architectural integrity. Moving to larger practices or working at an international level turned the unification of the entire

---

[1] Toni Kotnik, 'Digital Architectural Design as Exploration of Computable Functions', *International Journal of Architectural Computing,* 8/1 (2010), 17, p. 3.

[2] Michael Meredith, 'Never Enough', in Tomoko Sakamoto and Albert Ferre (eds.), *From Control to Design : Parametric/Algorithmic Architecture* (Barcelona ; New York: Actar-D, 2008), p. 6.

[3] Interview with Phil Osborne, 7.11.2011: 'well, it is the increasing sophistication of the computer technology, isn't it? I am just cautious of… I do not believe it could replace original thoughts in the person's brain… but I am fifty, you know, not fifteen!'

[4] Patrik Schumacher, 'Interview: Patrik Schumacher by Feng Xu', <http://www.patrikschumacher.com/Texts/Interview_WA_May%2009_english.htm>, accessed 26 January 2012.

[5] Mario Carpo, *The Alphabet and the Algorithm* (Cambridge, Mass. ; London: MIT Press, 2011).

design into a necessity. It had an impact on manufacturing as well. Although this was a step towards improvement, some architects might argue that it revitalised a form of standardisation which was pushed back by post-modernism. Specifically, some architects believe that this is a sort of pattern which happens whenever a society is struggling with an economic recession. Having referred to the Latham Report[6], Griff Evans explains that:

> There is always a strong force trying to push architects and building design towards standardisation, and improve the integration of production information. This is often cyclical and arises during recessions when the construction industry is deemed to be inefficient. Standardisation is necessary on large projects involving a lot of repetition, but buildings are not really prototypes like cars or aeroplanes; there needs to be a creative and complex response to individual sites and different user requirements. So most buildings will always be 'one-offs' in some respect, and let us hope it stays that way.[7]

Evans's outlook is interesting, since it is diametrically opposed to the views held by many, including Carpo, who believe that customisation is the current trend of many approaches in design. However, Evans refers to another significant point: the integration of information for production. This need resulted in an exploratory strategy in the construction industry to seek new approaches to design which could bridge the gap between design and manufacturing. Working in close collaboration, practitioners along with researchers succeeded in giving birth to approaches such as parametric design. In accordance with this reasoning, integrity brought two other consequences. First, it diverted companies' attitudes toward their clients and, as a result, it raised the level of clients' expectations of architects. Now, there is no doubt that the world of design is turning into a much more competitive environment[8] than the past, and those architects who want to win a project need to act at a different level of design generation and presentation in comparison to their rivals. Secondly, integrity created an educational issue: the necessity of adoption of these 'newfangled' technologies in architecture schools. After the introduction of early computer programs in architecture, many architects tried to harness software capabilities within the design process. Yet for several years the use of these programs remained pedagogically questionable and only in the 90s they infiltrated into architectural education, which was around the same time as the introduction of the first parametric programs. Perhaps architecture schools were 'design research laboratories'[9] which not only

---

[6] 'Latham Report' is an influential document written by Sir Michael Latham, published in 1994, entitled 'Constructing the Team'. 'Latham was commissioned by the United Kingdom government and industry organisations to review procurement and contractual arrangements in the UK construction industry, aiming to tackle controversial issues facing the industry during a period of lapse in growth as a whole'. (Wikipedia, 'Latham Report', <http://en.wikipedia.org/wiki/Latham_Report>, accessed 8 January 2012.)

[7] Email from Griff Evans, 17.11.2011

[8] Interview with Rick Sharp, 4.11.2011

[9] Patrik Schumacher, 'Architecture Schools as Design Research Laboratories', <http://www.patrikschumacher.com/Texts/Architecture%20Schools%20as%20Design%20Research%20Laboratories.htm>, accessed 8 January 2012.

trained a different generation of architects but also established the foundation for creating and developing new computer platforms for design.

This trend in education is continuing now; however, it seems that there is another shift in education which simply argues that architecture must amalgamate itself with new capabilities such as scripting and open-source programming. The recent generation of trained architects, 'who are able to operate code, do not need to be limited to existing software'[10]. Thus, they are thinking of a new approach which embraces all of the design facets, giving them this opportunity to explore more architectural concerns by computer. This can still be considered a parametric approach. Yet there is a tendency among researchers like Kotnik or Terzidis to call it new names: a post-parametric or an algorithmic approach.

Is it a Style or a Set of Techniques?

If the general roots of parametric design can be found in the digital turn in architecture, then to what extent is this new phenomenon broad and widespread enough to be called a style? Even the digital paradigm is still a matter for debate among researchers, who ask 'which paradigm is shifting?'[11], if such a shift indeed exists. Coining parametric design as a style seems controversial too, since neither its current position nor its roots are clear enough to be accepted by the architectural community. The notion of 'parametricism' as stated by Patrik Schumacher is largely seen in architectural education and academia, and this is probably the reason why recent graduates of architecture know more about it in comparison to their practicing counterparts. However, Schumacher's usage of parametricism can propagate a set of questions worth considering, and demanding more clarification.

The first question comes from the confusion over the use of the phrases 'style', 'paradigm' and 'techniques' in relevant literature. 'Style' comes either from the Old French 'estile', which means 'a stake, pale' or from the Latin 'stilus', meaning 'stake, instrument for writing, manner of writing, mode of expression'[12]. It is perhaps the oldest term in comparison with 'paradigm' and 'techniques'. The root of 'paradigm' dates back to the late 15th century. It

---

[10] Email from Daniel Richard, 31.10.2011: 'I would suggest there is a younger generation coming through who are simply more computer literate. They are able to operate code, and do not need to be limited to existing software – we can build our own. This is not just happening in architecture. This is happing in many creative disciplines and is being progressed by open-source initiatives and blog culture'.

[11] Mario Carpo explains in the preface of his book, *The Alphabet and the Algorithm* (2011): 'it may be too soon to tell if the digital is a revolution in architecture, but it is not too soon to ask what may be upended if it is. If the digital is a 'paradigm shift', which paradigm is shifting? If architecture has seen a 'digital turn', what course has turned?'

[12] Douglas Harper, 'Online Etymology Dictionary', <http://www.etymonline.com/index.php?term=style&allowed_in_frame=0>, accessed 8 May 2012.

derives from the Greek 'paradeigma' and in turn from 'paradeiknunai', which is composed of 'para' (meaning 'aside') and 'deiknunai' (meaning 'show'). Finally, the history of the word 'technique' goes back to the nearly 19th century from the French or Latin 'technicus'. However, 'technic' as an adjective goes back to the early 17th century, and in turn is derived from the Greek 'tekhnikos' or from 'tekhnē', meaning 'art'[13]. Semantically, paradigm[14] can reflect a bigger domain of meaning; it includes something beside the act of showing and presenting, while style refers to the act itself.

The second question enquires into the position of parametric design as a new style. Schumacher's ambition to radicalise the doctrine of parametricism in some positions seems surprisingly strange. For instance, he explains that 'what matters in architecture are not great buildings but great styles'[15]. This sort of looking at the body of architecture through the lens of style allows Schumacher to grasp two principal concepts of design, form and function, but with different names: the 'code of beauty' and the 'code of utility'. Therefore, in his view, architectural styles are those specific programmes that adjust the application of architecture's code of values[16]. Schumacher argues that style is virtually the only category outside architectural circles through which architecture is recognised. 'A named style needs to be put forward in order to stake its claim to act in the name of architecture'[17]. In fact, what he mentions here is attaching an external source to the domain of parametricism. However, through this perspective, styles can be interpreted as fashions, 'a matter of appearance' which is perhaps ephemeral and superficial. In order to discard such connotations, Schumacher defines styles as 'design-research programmes'[18]; in this definition they cannot be reduced to mere matters of appearance. Accordingly, he believes that a style can make up such programmes, just like the way a paradigm frames scientific research programmes: 'a new style in architecture and design is akin to a new paradigm in science'[19]. Similarly, the shifts from one

---

[13] Oxforduniversity, *Oxford Dictionary of English* (Oxford: Oxford University Press, 2010).

[14] One of the frequently-cited concepts of paradigm is Thomas Kuhn's definition. He uses two different senses of meaning: On the one hand, paradigm 'stands for the entire constellation of beliefs, values, techniques, and so on shared by the members of a given community. On the other, it denotes one sort of element in that constellation. The concrete puzzle-solutions which employed as models or examples, can replace explicit rules as a basis for the solution of the remaining puzzles of normal science'. See: Thomas.S. Kuhn, *The Structure of Scientific Revolutions* (University of Chicago Press, 1996), p. 175.

[15] Patrik Schumacher, *The Autopoiesis of Architecture: A New Framework for Architecture* (Wiley, 2011b), p. 274.

[16] Ibid, p. 257. In addition, Schumacher describes that 'styles provide the guidelines and criteria that help us to identify the beautiful and the useful… each style answers the questions what is useful? And what is beautiful? In its own way, with divergent results... Aesthetics values programme the code of beauty. The code of utility is programmed by performance values. Substantial styles provide both of these values'. (pp. 257-258)

[17] Patrik Schumacher, 'The Parametricist Epoch: Let the Style Wars Begin ', <http://www.patrikschumacher.com/Texts/The%20Parametricist%20Epoch_Lets%20the%20Style%20Wars%20Begin.htm>, accessed 8 January 2012.

[18] Ibid

[19] Ibid

dominant style to another might be compared to paradigm shifts in science[20]. Paradigm shifts are usually started by a state of crisis inside the old paradigm, which is often brought about by new empirical evidence that 'overburdens the explanatory capacity of the old paradigm'[21]. However, Schumacher borrows Imre Lakatos's[22] terminologies of the 'hard core', the 'protective belt' and the 'research programme' to explain that shifting between styles entails coming up with two new sets of heuristic principles that define every architectural style: formal heuristics, which programme the code of beauty, and functional heuristics, which programme the code of utility[23].

Schumacher's statements become even more controversial as he seeks innovation in architecture via the progression of styles, likening the competition of styles to 'style wars'[24]. Within this battlefield, three types of styles can be considered: passive style, active style, and active-reflective style. Interestingly, the only passive style in Schumacher's taxonomy is Gothic – the other styles have always been active. I would argue that the notion of a 'passive' style is meaningless. Every style is essentially active when it is introduced to society. An active style, according to Schumacher, is knowingly selected, and its promoted design principles guide its evaluation[25], while an active-reflective style is 'a discursive phenomenon', a communication structure within the autopoiesis of architecture. Furthermore, Schumacher's description of the active-reflective style does not offer any major distinctions. Only when he defines another level of subdivision inside this classification, namely epochal, subsidiary, and transitional styles[26], does the case become slightly clearer. Whereas modernism was an epochal style,

---

[20] Schumacher uses the term 'style' in a way that scientists use the term 'paradigm'. Therefore, it may be construed that the two terms, 'paradigm' and 'style', are interchanged here. As explained before, there is an ambiguity in the notions of 'style', 'paradigm', and 'techniques' which is clearly seen in the parametric literature, specifically, in Schumacher's papers and books. In particular, 'style' and 'paradigm' may reflect the same connotations, even though they have different denotations. I believe we can term the realm of the digital 'a paradigm', borrowing Thomas Kuhn's definition of paradigm. The digital realm is a paradigm, because it is a new collective way of thinking. Yet, this 'new way' only makes sense, when we consider the old or the obsolete way. Therefore, the description of a paradigm entails a 'shift'. This is the reason why we often see the term 'paradigm shift' and not 'paradigm'. In addition, this holistic approach in using the digital realm as a 'paradigm' can avoid confusion over two other terms 'style' and 'techniques'. While the digital paradigm has a longer history (dating back to 50s and 60s), we can observe styles such as postmodernism, deconstructivism, and parametricism (if we accept that it is a style) inside this paradigm.

[21] Patrik Schumacher, *The Autopoiesis of Architecture: A New Framework for Architecture* (Wiley, 2011b), p. 274. Schumacher refers to the Popper's falsification principle and explains that scientific theories are often thick-skinned to be falsified. According to Schumacher, the case is the same for the architectural styles: 'the history of science shows that scientific theories are much more resilient than Popper's falsificationist methodology suggests. They usually have a 'thick skin', stubbornly ignoring initial failures and criticisms on their way to eventual success. The same goes for emerging styles. They are not necessarily discouraged by the initial public rejection, nor by the initial failure to meet basic targets like functional, technical or economic efficiency'. (p. 280)

[22] Imre Lakatos, the Hungarian philosopher of mathematics and science, known for introducing the concept of the 'research programme' in his methodology of scientific research programmes. Wikipedia, 'Imre Lakatos ', <http://en.wikipedia.org/wiki/Imre_Lakatos>, accessed 8 May 2012.

[23] Patrik Schumacher, *The Autopoiesis of Architecture: A New Framework for Architecture* (Wiley, 2011b), p. 286.

[24] Patrik Schumacher, 'The Parametricist Epoch: Let the Style Wars Begin ', <http://www.patrikschumacher.com/Texts/The%20Parametricist%20Epoch_Lets%20the%20Style%20Wars%20Begin.htm>, accessed 8 January 2012.

[25] Patrik Schumacher, *The Autopoiesis of Architecture: A New Framework for Architecture* (Wiley, 2011b), p. 241: 'Modernism, Postmodernism, Deconstructivism, and Parametricism are clear examples of active-reflective styles'.

[26] According to Schumacher, Subsidiary styles build up when 'the initially wide open space of exploration of the style starts to be exhausted, loses its innovative dynamic and experiences a sense of stagnation' (ibid, p. 246), while 'Epochal styles are those that

post-modernism and deconstructivism are mere transitional episodes. However, Schumacher's explanation still seems unclear here. For instance, in his view, one of the features of the transitional period is that it can bring 'a plurality of simultaneous, competing styles' inside the epochal style. Furthermore, Schumacher takes into account another class of style, named 'subsidiary'[27], which sounds confusingly similar to the previous category, the transitional style. In this explanation, he offers a viewpoint close to the concept of fashion, a point that he criticised beforehand. Table 3 summarises Schumacher's classifications:

Styles	Epochal Style	Subsidiary Style	Transitional Style
Passive Style	Gothic		
Active Style	Renaissance	Mannerism	
	Baroque	Rococo	
Active-Reflective Style	Neo-Classicism		
	Historicism	Neo-Gothic	
		Neo-Renaissance	
		Neo-Baroque	
		Eclecticism	
			Art Nouveau
			Expressionism
	Modernism	Neue Sachlichkeit	
		Organicism	
		Rationalism	
		Brutalism	
		Metabolism	
		High-Tech	
			Postmodernism
			Deconstructivism
	Parametricism		

Table 5: Classification of styles according to Patrik Schumacher (redrawn based on Schumacher, *The Autopoiesis of Architecture: A New Framework for Architecture*, p. 253)

One of the problems with Schumacher's position is the context in which he discusses it. For more than a decade, he has worked in a quite well-known office, Zaha Hadid Architects (ZHA), where the atmosphere of design and presentation is quite different from small or middle-scale offices. According to Kostas Terzidis, avant-garde firms including ZHA stand in sharp contrast to corporate architectural practices such as Skidmore, Owings, and Merrill (SOM), where design

---

demonstrate long-term viability because they offer a systematic solution to the essential problems and challenges of the respective epoch' (ibid, p. 253).

[27] Patrik Schumacher, 'The Parametricist Epoch: Let the Style Wars Begin ', <http://www.patrikschumacher.com/Texts/The%20Parametricist%20Epoch_Lets%20the%20Style%20Wars%20Begin.htm>, accessed 8 January 2012. 'Subsidiary styles' emerge under the umbrella of epochal styles. 'These subsidiary styles represent either parallel variations or historical sequences that enrich and progress the respective epochal style…Within modernism we can distinguish Functionalism, Rationalism, Organicism, Brutalism, Metabolism and High-Tech'.

development is still achieved through traditional manual means[28]. Therefore, it is not surprising that the majority of architects do not accept parametric design as a style. Through a hermeneutical point of view, the idea of style in people's mind is different from one person to the next and, in this way[29], the origins of a style require a 'psychoanalysis'[30] of the creator. Moderate[31] voices in this debate suggest that the definition of something as a style or set of techniques is quite dependent on the individual. Maybe the popularity of the term 'style' among star architects or international companies is much greater than it is for other architects or firms. Furthermore, some believe that it is dangerous to call parametric design a style due to the traditional tension between education and practice. They emphasise that 'it becomes a style within academic, but within practice, the view is quite opposite'[32]. This is diametrically opposed to Schumacher's definition of parametricism[33], because he sees the new style beyond the borderline between education and practice.

An interesting viewpoint on the parametricism debate comes from Michael Meredith. He formulates parametricism as 'positivism + expressionism', arguing that having a unifying singular style is an 'ontological impossibility'. He compares the representational aspects of the 90s (1990-1999) with the realism of the 00s (2000-2009). In Meredith's view, architecture in the nineties was based more on geometrical, formal systems with no attention to materiality, whereas after 2000 the approach turned to a physics-based, performance-driven outlook on design, considering materiality within the design process[34]. Yet some of the categories in

---

[28] Kostas Terzidis, *Algorithmic Architecture* (1st ed. edn.; Amsterdam ; London: Architectural Press, 2006), p. 40.

[29] In Schumacher's view, styles are defined based on a micro to macro range. He believes in individual styles as well. Schumacher explains that style 'can be applied with different degrees or levels of aggregation: we might talk about the individual style of an individual architect, about national/regional styles, or finally about epochal styles. An epochal style is the dominant style of a particular civilisation within a particular historical era'. Patrik Schumacher, *The Autopoiesis of Architecture: A New Framework for Architecture* (Wiley, 2011b), p. 242.

[30] Interview with John Lee, 28.10.2011: his view to style is interesting. Lee describes that 'during my masters, I looked at the work of Peter Eisenman. I've been interested in the relationship between process and the physical form or the building and Eisenman famously adopted a number of different process-driven or process techniques… produce forms and my difficulty with it is the thing that applies to parametricism too. In the end what determines is what is built. It seems to me that in the work I did with Eisenman, that was a big point and for him the process is everything and I am just reckoning that on Schumacher. In the end, it did not generate Eisenman's style. Eisenman'style came from him or form psychoanalysis… That is what the style comes from. It does not come from numbers or processes. The process follows his ideas about how building should look'.

[31] Interview with Alex Solk, 22.11.201: 'It is a technique to help. Well, a lot of people use it a as a new architecture style, plenty of architects, we can think of big international architects… It is not just what I use… I prefer to think of those technologies as techniques to help; but I appreciate many architects use it as a style'.

[32] Interview with Vikram Kaushal, 31.10.2011

[33] Patrik Schumacher, 'Parametricism: A New Global Style for Architecture and Urban Design', *Architectural Design,* 79/4 (2009a), 14-23. Schumacher defines parametricism as 'a new style rather than merely a new set of techniques. The techniques in question – the employment of animation, simulation and form-finding tools, as well as parametric modelling and scripting – have inspired a new collective movement with radically new ambitions and values' (p. 15). Parametricism also used by other critics of architecture such as Mario Carpo in his book *The Alphabet and the Algorithm* and his lecture in MARC (Manchester Architecture Research Centre) on 24 October 2011.

[34] Theharvardgsd, 'The Eclipse of Beauty: Parametric Beauty (Mario Carpo, Michael Meredith, Ingeborg Rocker)', <http://www.youtube.com/watch?v=OxN4LWPlwX8&list=PL23D79E83F8F52102&index=5&feature=plpp_video>, accessed 10 May 2012. In Meredith's comparison, the representation of 90s (1990-1999) has these aspects: *1. End of history 2. Process-driven methodologies to avoid post-modern semiotics 3. Internal and disciplinary 4. Avant-garde 5. Critical theory 6. Geometry 7. Complexity 8. Formal (systems) 9. Non-materiality (white) 10. Alias (Maya),3DMAX, Rhino 11. Plans, Sections, Process drawings 12. Aesthetics (Art) 13. Advanced architects were not licensed/academic 14. Magazines which explain ideas, assemblages.* In contrast,

Meredith's comparison look ambiguous. Particularly, when he compares the type of software used in these two periods, it is not clear why he refers to 3DMAX or Rhino as aspects of the 90s and not the 00s.

Statements such as Meredith's and Schumacher's are frequently argued in architectural theory. Yet, as mentioned earlier, opposite statements often emerge from practice. It seems that one of the reasons why practicing architects believe that parametric design is just a set of techniques rather than a style is that the essence of designing buildings, which is responding to the brief's requirements, is still the same[35]. However, this is a position to question, because in all of the styles which have been monitored throughout the last century, this major part has also been the same. The brief of every architectural project is the fundamental part of the design process. It is quite independent of everything including style. Thus, it cannot be assumed as a conditional state for a style. On the other hand, some architects claim that the domain of parametric design is wider than can be seen, especially in terms of technical possibilities. They therefore argue that 'calling it style is slightly restrictive'[36], specifically because parametric design does not bring a distinct perspective to the role of the designer in this realm.

A Solution to the Dichotomy

In this chapter, I have investigated the roots of parametric design as well as its position to determine if it is a style or just a set of techniques. Although the roots of parametric design are not entirely clear, there are many discussions over its nomination as a new style. Holistically, it is possible to argue about it by setting two standpoints: parametric design under the title of parametricism and parametric design as an extended family of techniques. When the first position is taken into account, such a style must bring a methodology in which the principles and the framework are fully acknowledged. Based on what architects state in practice, such a framework is still not provided. In addition, that framework can experience what is called in

---

the realism of 00s (2000-2009) presents these features: *1. End of representation 2. Built better than unbuilt, construction logic, performance 3. External, engaged in the world, opportunistic 4. Rear-garde 5. Practice, post-critical 6. Physics 7. vernacular, diagrammatic 8. Functional, performance, sustainable systems 9. Materiality (rough) 10. BIM, Revit, Catia, Ecotect 11. Diagrams (big text), arrows 12. Urban (life) 13. Advanced architects were licensed/professional 14. Magazines to get work, Dwell, Wallpaper, Blogs.*

[35] Interview with Phil Osborne, 7.11.2011

[36] Interview with Matthew Smith, 17.10.2011: 'I think calling it style is slightly restrictive… who is a parametric designer? One who is just using these kinds of particular programs, I think it is possibly restrictive I guess, because you are going to say that you have to use these programs or this methodology'.

science the principle of 'incompleteness'[37]. If someone wants to calibrate all of the aspects and facets in the frame of a new style, he/she may possibly lose some of this information, just like the act of reduction of architecture into a piece of writing. The advantage of liberating parametric design from any stylistic taxonomy is that it cannot be limited to a series of specific arguments. Hence, architects can be assumed as parametric designers even when they do not use parametric tools or parametric modelling.

Focusing on the tools used is another way of addressing the position of parametrics, although the discussion here ends up being very vague. This ambiguity emerges from how these tools are introduced to the domain of parametrics. For instance, when Schumacher defines parametricism, he refers to form-finding tools as well as scripting and animation-making devices as the base of this new style. However, in another paper, he argues that the parametric design tools by themselves cannot create the new style or the stylistic shift from modernism to parametricism because many 'late modernist architects like Norman Foster are employing parametric tools in ways which result in the maintenance of a modernist aesthetic'[38].

None of these two positions, whether holistic or atomistic, can resolve the problem. I would argue that only by considering two categories[39] of parametric design in practice can this hypothetical dichotomy between style and technique be clarified. The first category assumes parametric design as a method for conceptual modelling. In this way, parametric design requires the knowledge of programming and scripting. Since it provides a rather methodological framework that architects should learn and use, it is accurate to call it a style. The second category furnishes parametric design with the idea of architectural construction. It employs data-processing techniques to manage the process of design, which is tightly linked to manufacturing issues. Viewed through this lens, parametric design is just a set of techniques rather than a stylistic approach.

---

[37] Gödel's incompleteness theorems: these theorems proved by Kurt Gödel, the German mathematician, in 1931. See: Globalbeehive, 'Kurt Godel: The World's Most Incredible Mind', <http://www.youtube.com/watch?v=i2KP1vWkQ6Y&feature=related>, accessed 10 April 2012.

[38] Patrik Schumacher, 'Design Research within the Parametric Paradigm', (updated 6 July 2011) <http://www.patrikschumacher.com/Texts/Design%20Research%20at%20Zaha%20Hadid%20Architects.htm>, accessed 10 January 2012.

[39] This classification is borrowed from the paper by M. Stavric and O. Marina, 'Parametric Modelling for Advanced Architecture', *International Journal of Applied Mathematics and Informatics* 5/1 (2011), 9-16. However, their argument is about modelling, not answering the question of style in parametric design.

Chapter 7

The Up and Down Sides
of Parametric Design

This chapter discusses parametric design by setting a normative position. Firstly, an overview obtained from the analysis of the questionnaire is described. This is followed by a more detailed discussion on the advantages and disadvantages of parametric design and its distinction in comparison to traditional CAD and Building Information Modelling (BIM).

7.1. The Up and Down Sides of Parametric Design

Before hearing from architects about the up and down sides of parametric design, a short analysis based on the questionnaire responses is presented. Figure 5 provides information about five items, namely finance, data management, learning time, usage problems and changes in approaches towards design. From observation, these aspects appear more important than the other aspects of parametric design. The answers to these five categories are ranked in a five-point Likert Scale, clarifying how architects think about the above five issues.
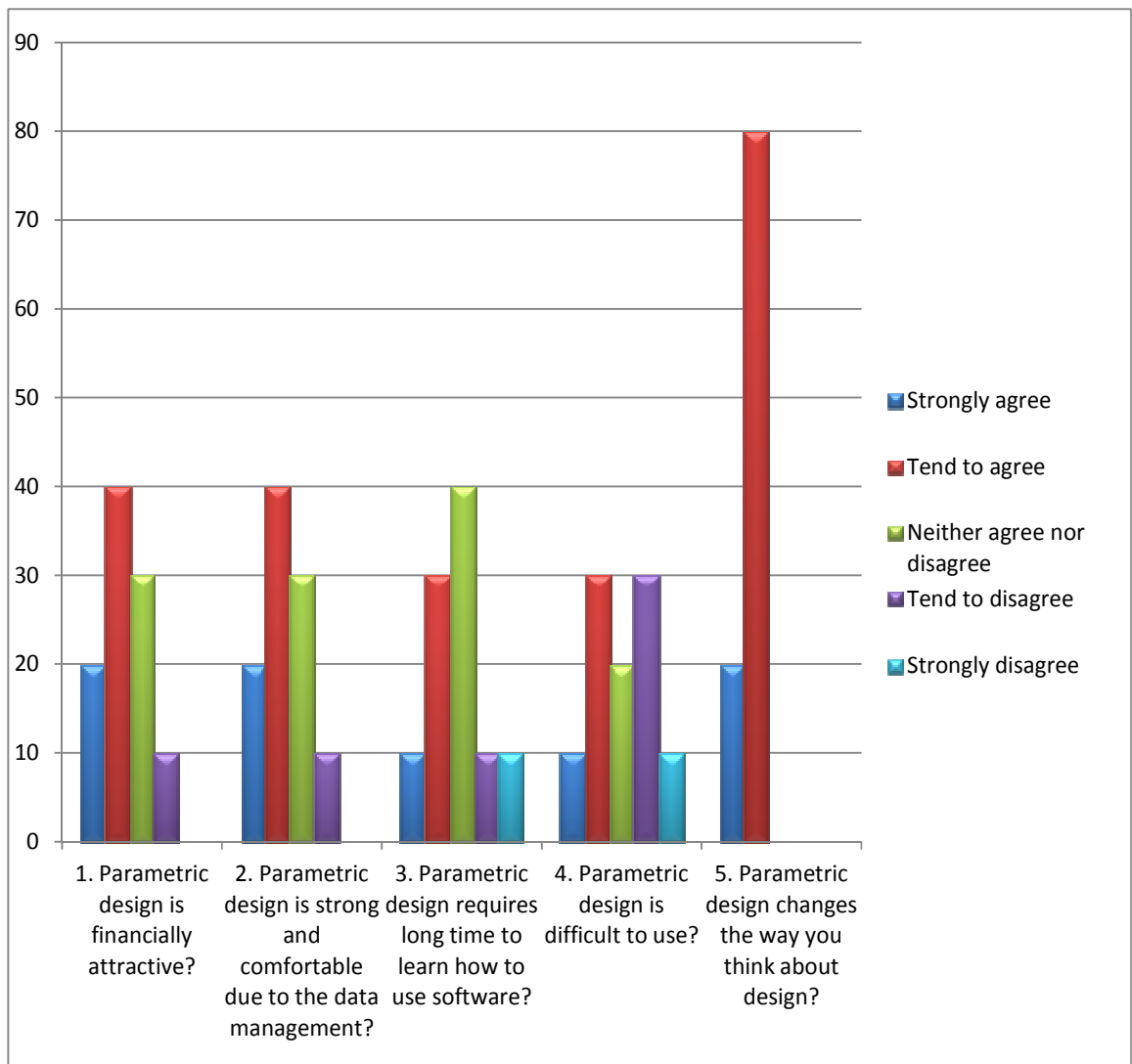
Figure 5: The assessment of parametric design based on five categories: finance, data management, learning time, usage problems and changes in approaches towards design. The comparison is established on a five-point Likert Scale.

As the graph clearly illustrates for question five, almost all of the interviewees believe that parametric design changes the way they think about design. This aspect clearly stands out

from the other items. In terms of the difficulties of using parametric programs (question four), the responses of participants are firmly equal. While one-third of them disagree that parametric design is difficult to use, one-third of them agree with this stance. The rest (20%) do not make a distinct position. In terms of the finance, data management and learning time, the Likert Scale is more focused on the middle items (tend to agree, and neither agree nor disagree) rather than the two extremes, which probably shows that architects are somewhat ambivalent in recognising the item in question as a parametric merit.

This short overview of respondents' opinions furnishes the third set of questions introduced in Chapter 1 with the idea that parametric design offers a distinct vista on design by focusing on some fundamental aspects of the design process and handling them in a different way. The next sections will establish a deeper understanding of the advantages, disadvantages and distinctions of parametric design.

7.1.1. The Advantages of Parametric Design

Some recognisable facets of parametrics turn it into a more sophisticated mode of design. The reasons why a typical architectural practice employs a parametric approach can differ – while some firms follow a competitive strategy by simply keeping up with the latest software, such as parametric packages, the majority tend to see parametrics through the lens of effective functionality; for example, as an improvement of design opportunities.

In general, the ability to rigorously explore more design alternatives and to therefore see better solutions emerging from design problems is pointed out as the main benefit of parametric design. Furthermore, unlike traditional CAD, which still depends highly on sketching or physical modelling, some of the advantages of parametric design would be seen early on in the exploration of design possibilities. When architects specifically think about free-form structures, parametric design gives them 'a great opportunity for exploring more exciting forms'[1].

Yet, great benefits also exist in the later stages of the design process for the automation of construction documentation and higher levels of architectural control in production. In terms of financial issues, there are also benefits in reducing the person-hours spent on exploring design, and the tedious activity of drawing details that can be extracted

---

[1] Interview with Amy Hanley, 25.10.2011: 'I think… it is a great opportunity for exploring more exciting forms and shapes. But when I think that is not just about the whole form of the building, it is about the components, the elements of constructing those forms; even every sort of standard primary structures for offices, for instance, could have a modulated modified form'.

from architectural models. Concisely, parametric design can bridge the gap between the design and manufacture of the building. Roland Hudson, an architect and a researcher of parametric design describes the opportunities that parametric design can offer:

I see specifically opportunities that parametric design presents in assisting and developing knowledge of a design problem, formalising or capturing that knowledge in an external manner, developing strategies for construction, conducting design investigation and, lastly, implementing a construction documentation method. All can and should be approached simultaneously and cyclically throughout the design phase until the process converges on an approach which embraces all[2].

Through probing into the surrounding literature, one of the fundamental issues that is frequently cited is the ability to make relationships between objects, using equations to define the associative geometry[3]. Robert Woodbury indicates that defining relationships has not previously been considered as part of design thinking[4], since the conventional defined activities in design were 'add and erase'. Now designers have two extra capabilities, namely 'relate and repair'. For Woodbury, 'relating' demands explicit thinking about the type of relation, and 'repairing' happens after an erasure, 'when the parts that depend on an erased part are related again to the parts that remain'[5]. Hence, these two acts imposed pivotal changes on past systems. It is reasonable to consider them as benefits of parametric design. Besides these issues, architects recognise several other differentiators about parametric design when compared with traditional ways of designing. These issues can be divided into three classes, namely optimisation of the design process, the capability of making a range of solutions for the design problem, and engineering the design process more efficiently.

Optimisation of the Design Process

Optimisation is a term derived from the Latin 'optimus', which means 'best'[6]. It literally means a desirable state that satisfies the conditions of the problem among a number of options. According to Mark and Jane Burry, optimisation methods can be divided into stochastic and deterministic approaches. While 'stochastic' (which is derived from the Greek 'stokhastikos' and in turn from 'stokhazesthai' and 'stokhos' (aim)[7], and means to aim or to shoot with a bow at a target) offers a random characteristic, in the deterministic methods of optimisation there is no randomness. Values are assumed accurate and each action determines precisely the next

---

[2] Email from Roland Hudson, 19.10.2011

[3] Branko Kolarevic, *Architecture in the Digital Age : Design and Manufacturing* (New York, NY: Spon Press, 2005), p. 17.

[4] Robert Woodbury, *Elements of Parametric Design* (London: Routledge, 2010), p. 24.

[5] Ibid, p. 11

[6] Oxforduniversity, *Oxford Dictionary of English* (Oxford: Oxford University Press, 2010).

[7] Ibid

state. Thus, 'the same optimisation routine repeated with the same starting conditions for the same number of iterations will result in the same outcome each time'[8].

To a certain extent, parametric design utilises both of these methods (although it is often more deterministic than stochastic) in the optimisation of the design state, which includes design solutions and the paths leading to them. It helps architects to set up constraints and to measure things in a different way. Therefore, for those architects who are looking for an optimum design, parametric design is ideal. Vikram Kaushal, a practicing architect and a lecturer at the Manchester School of Architecture (MSA), explains that, in commercial architecture, reaching this optimal point is significant; parametric design thus becomes highly beneficial in this case:

It allows us to visualise and to understand things in a different way because it highlights things that are very difficult at first to measure... if you want to have optimum design, then parametric is ideal… But architecture has not always been an optimum, optimising everything, you know. But we can save a lot and learn a lot. So there are some disadvantages and advantages and, I think, in terms of the commercial use, the advantages are greater than the disadvantages[9].

Because of its considering of parameters which can control the entire design, parametric design is financially well-defined, supplying architects with a high level of control over the cost of their project. In addition, it can bring benefits to the clients as well by monitoring the economy of design and consequently it can help the designer to minimise the risk of the development by 'integrating as much information and parameters as possible into one platform that can span the entire workflow'[10].

A Range of Solutions for the Design Problem

This benefit is perhaps the most clear out of the characteristics of parametric design. As discussed previously, by setting dynamic links among all of the decisions made throughout the process and supplying an associative logic, architects can simply change one design option to another and produce a range of solutions instead of coming up with just a sole response to the design problem. In fact, this aspect of parametrics originates from the concept of 'relation' which has been previously discussed and, although it is a clear benefit at first glance, it is also a

---

[8] Jane Burry and Mark Burry, *The New Mathematics of Architecture* (London: Thames & Hudson, 2010), p. 119. 'The process of optimisation describes the synthetic search for this best state within a model, whether of a biological system or architectural or structural system, usually under a set of restrictions, implied or expressed… no matter how deterministic the procedure or the algorithm of optimisation may be, the optimal state is always relative to the details of the system in which it is sought… Before the introduction of electronic computation in architecture, some architects and engineers used dynamic analogue (physical) models to compute optimal shapes and structural scenarios'. (pp. 117-118)
[9] Interview with Vikram Kaushal, 31.10.2011
[10] Christian Derix, 'In-between Architecture Computation', *International Journal of Architectural Computing,* 7/4 (2009), 565-86, p. 568.

challenge, because it demands clarification of the interdependencies created by different requirements. Matthew Smith explains that, in parametric design, the process of decision-making is much easier because of the range of solutions available:

I think one of the good things is that it makes you think about the relationships between the different elements of design, the relation in the context with various parameters… you do not think about a fixed solution, you are thinking about 'what are the factors and influences and things that can shape this building or this project?'… You can make lots of different decisions. Well obviously, if you do this in a manual way, it takes your time… you eliminate a lot of monotonous tasks… As an architect, you are not adding the values just like a manual worker; you are adding value in designing and thinking about the problem, not manually changing every line… that is an advantage of it very much. You think about the problem and guess a more holistic way. You think about the problem and try to get a solution to that… You make a range of solutions and different outcomes, because you can vary things and do not do it manually[11].

Although such capabilities sound like a great advantage, designers must be cautious, since a mistake in change of one parameter can have a ripple effect on the design. This can be even more daunting when someone else rather than the original designer tweaks a parameter in ignorance. It would be time-consuming to detect such minor changes and, as a result, it can reduce work efficiency. Therefore, applying parametric methods in this sense requires a superior managerial insight into the abilities of practitioners in an architectural office – a level of management that can probably only be achieved in large-sized practices.

Engineering the Design Process

One of the considerable advantages of parametric design is its ability to enable effective management of the design process. In other words, it gives the architect the chance to engineer the design process by offering the following advantages: speeding up the entire design activity, eliminating tedious actions and making many different decisions[12]. It also helps architects accelerate negotiations with the client. It is probably true that clients are much more interested in the outcome of a design than its process[13]. However, at the later stages of design when the alternatives come along to compete with each other, the need to have an effective conversation with the client is inevitable in order to finalise what has been designed so far. Architects believe that parametric design can enhance negotiations with the client because it can make the design environment flexible and ready to accommodate any subsequent changes. It enables architects to respond quickly to the client in different situations.

---

[11] Interview with Matthew Smith, 17.10.2011

[12] Ibid

[13] Ibid: 'In terms of this practice in my experience, I think clients are mainly interested in the outcomes and all presentations of the outcomes and how they reflect a kind of agenda. I do not think you might find differently, if you would talk to the people. I think in terms of the clients, they are probably not very interested in the process'.

7.1.2. The Disadvantages of Parametric Design

In the light of bringing a different outlook to design, architects argue that parametric design also brings disadvantages to their practice. Some shortcomings come from the novelty of the computer programs which are used in parametric design. However, some arise from the parametric approach itself, which not only changes the way architects design but also considerably alters the role of an architect in a design office. In general, Malcolm McCullough explains that 'parametrics work better in domains whose subject matter is engineered form itself'[14]. This sort of engineering is quite achievable in areas such as product design, where parameterisation implies considering only a few factors. However, in architecture, it turns into a serious challenge.

A problematic issue which might be related to the parametric approach in general is that it never resolves what parameters are necessary for design. In other words, to a certain extent it does not bring a methodological framework for design and, as a result, architects still need to elaborate most parts of the design in their minds. Computational strategies such as the algorithmic method try to cover this shortcoming, although in those approaches the role of the architect is often being supplanted by the software.

Another problem arises from the fact that most parametric programs have been designed and attached with 'a traditional workflow alignment in mind but allow for more process thinking'[15]. Additionally, operators of these systems have to anticipate all project directions beforehand in order to create the geometries and to build the interrelationships. Parametric programs should be designed with a parametric approach, and the developers of the parametric packages need a true understanding of this approach, since they effectively 'design a design'.

The other general shortcoming – not only in parametric design itself, but in all of the software packages – is the need to have additional form-analysis software. Although parametric packages have tried to fill this gap by designing a structure for form generation, the challenge is how to connect form-generation software to form-analysis software[16].

---

[14] Malcolm Mccullough, '20 Years of Scripted Space', *Architectural Design,* 76/4 (2006), 12-15:'… especially in mechanical components for complex assemblies such as vehicles. Parametric design works less well, where physical configuration and performance are just the means, and a more emergent usage pattern is the end… Parameterisation breaks down when the design problems are wickedly under- or over-constrained, or where the design variables are less obvious'. (pp. 14-15)

[15] Christian Derix, 'In-between Architecture Computation', *International Journal of Architectural Computing,* 7/4 (2009), 565-86, p. 568.

[16] Interview with Daniel Richard, 8.11.2011: 'We have heard for a while that the biggest problem with the form in design is the ability to connect form-generation software with form-analysis software. You know packages like BIM and like Grasshopper are

Apart from these fundamental problems, the disadvantages of parametric design can be classified into four main categories, depicting perhaps why parametric design is still on the margin or less well-known among architects in spite of its discussed pre-eminence: unnecessary complexity with too much information, the problem of authorship, constraining creativity with a reactive structure, and learning and training difficulties.

Unnecessary Complexity with too much Information

One of the underlying problems, which at first glance seems to be a serious challenge for parametric design, is the complexity of parametric packages. According to Aish and Woodbury, parametric modelling 'may require additional effort, may increase complexity of local design decisions and increases the number of items to which attention must be paid in task completion'[17]. In addition, architects argue that there is no need to have such a mazy structure for a design problem, as this only makes the design activity more complicated. It is just like 'using a sort of sledgehammer to crack a nut'[18] – they do not need all of that power for designing.

Despite these sorts of opinions, which sound plausible, some architects believe that there is a 'Catch-22'[19] situation involving parametric design. Especially on the commercial side, architects are asked to provide more information with more high-level graphics in a short period than is possible by merely using the latest parametric programs. On the other hand, architects must think about which software is suitable, considering this fact that parametric packages are costly. In addition, architects may provide a level of information for their clients that seems useless not only to the clients, but also to the builders and manufacturers:

> I think, in terms of disadvantages, there is an expectation from clients that they get a lot more information that they never used to. So they will expect much more detailed images and graphics than they are able to display… I think that the disadvantage is that you need to provide so much more information for people to then go to the next stage… With a building which is of more traditional construction and also the way the building industry still often works with prefabrication, then… it doesn't necessarily justify going into that level of detail within software.[20]

Confronting such complexities within complicated software interfaces become a more serious challenge when the lack of time makes collaborative design a necessity. It is largely ubiquitous

---

certainly allowing us much greater connection between these two, but in terms of being able to use these, you know in the way that I am talking about, we still have problems'.

[17] Robert Aish and Robert Woodbury, 'Multi-Level Interaction in Parametric Design', in Andreas Butz et al. (eds.), *Smart Graphics* (Lecture Notes in Computer Science, 3638: Springer Berlin / Heidelberg, 2005), p. 151.

[18] Interview with Daniel Richard, 8.11.2011

[19] Interview with Trevor M. Cousin, 4.11.2011: 'yeah, it is again in the commercial side; because you could say to the client that you can provide more information, more high-level graphics and it costs or if the client is pretty aware of the software, then you did send what expected [from them]. So let me say it's a Catch-22 situation'.

[20] Interview with Rick Sharp, 4.11.2011

among principals of architectural practices to sketch and work on tracing papers and then give that piece of drawing to other architects and practitioners, asking them for more work and generating more alternatives. As in every design project, there is a hierarchy of people working together, and the parametric designer is obliged to work within that team. The tension very often emerges when the rest of the team are working much more traditionally by hand, and increases when some of the stages done on paper cannot be exactly transferred into parametric programs, and so the parametric designer is deemed ignorant of what is being designed. This is probably one of the reasons why the number of parametric experts in companies is unusually low, as well as the fact that many architects are not 'numerate'[21]. Architects are often trained to think visually in three dimensions. Hence, they draw things in peculiar interfaces such as Grasshopper without really understanding where they are in the virtual space.

This complexity also brings about another issue: demand for more powerful computers. This need is particularly seen in small and modest companies, since most of the parametric packages are quite bulky in size. In addition, due to the limited capacity of stand-alone systems, architects need to share their information with each other through an internal network which again highly depends on the computers used in the design process to support these characteristics. Architects thus ask for upgraded computer systems that are able to meet their current requirements. Yet some of the problems even cannot be resolved by having more advanced systems, because they emerge from a structural defect, such as the problem of termination of the algorithm. 'Parametric modellers like Grasshopper do not allow loops in their models'[22] and this perhaps shows again the weak points in the body of the current parametric packages.

The Problem of Authorship

Most of the time architects are defined as the authors of a project. Mario Carpo[23] believes that authorship entails the idea that buildings should be identical copies of the designs. As he

---

[21] Interview with David Wood, 1.11.2011

[22] Fabian Scheurer and Hanno Stehling, 'Lost in Parameter Space?', *Architectural Design,* 81/4 (2011), 70-79, p. 76.

[23] See Carpo's lecture in Harvard University: Theharvardgsd, 'The Eclipse of Beauty: Parametric Beauty (Mario Carpo, Michael Meredith, Ingeborg Rocker)', <http://www.youtube.com/watch?v=OxN4LWPlwX8&list=PL23D79E83F8F52102&index=5&feature=plpp_video>, accessed 10 May 2012. Carpo specifically explains that 'In a digital design environment, modern authorship is replaced by the new format of hybrid participatory agency. This hybridisation of agency is an essential aspect of parametricism because it is embedded in very technical nature… an open-ended parametric notation can be fixed or finalised by the same person who wrote it, but also by somebody else or something else and/or it may evolve… in parametric design, script is genus and the individual object is the species'.

explains, in a parametric design process, some parameters are variables by definition. This variability may be automated and controlled by the system; for example, a program may be instructed to generate any number of variations, randomly or as a function of some external factors. 'Some parameters may be chosen, at some point, by someone other than the 'original' author, and possibly without his or her consent'[24]. This problem culminates in the algorithmic approach where scripting is the hallmark of design, and confusion over the ownership of algorithmic forms and the issue of intellectual property may arise. Again, the problem occurs when the author of the original script may not be the only author of the final product, and consequently may not determine all of its final features. The only difference between the parametric and algorithmic view in this sense is that in the algorithmic approach a 'design decision may be made by an algorithmic process not intended by the designer'[25]. Nevertheless, it could be said that authorship is a serious concern that should be added to the challenges of parametric design.

Constraining Creativity with a Reactive Structure

One of the clear-cut boundaries between traditional CAD and parametric design is the idea of setting constraints in design. Even though this can be assumed as an advantage of parametric design, it can play a negative role at the same time. Architects argue that constraints sometimes confine the creativity of designers.[26] Moreover, due to the reactive structure developed by the software's engineers, architects again find themselves restrained by a preset of parameters. While some of these parameters are flexible, they still offer limited conditions. The ideal is achieved when the designer is able to have an interactive conversation with the parametric program:

> It is not interactive, it is reactive. It is reactive towards a set of parameters that have been set up by developers, the software engineers or the software architects. So there is only a certain amount of outputs that might be in their millions, but they are set… Interactivity is something where the system engages back with you, so it can have a conversation; like Pask's idea of conversation theory, it becomes about a dialogue. I do not know if you have seen the [film] Space Odyssey and how the machines talk to you… that kind of idea where… that is the next step, you know![27]

---

[24] Mario Carpo, *The Alphabet and the Algorithm* (Cambridge, Mass. ; London: MIT Press, 2011), p. 42.

[25] Kostas Terzidis, *Algorithmic Architecture* (1st ed. edn.; Amsterdam ; London: Architectural Press, 2006), p. 26.

[26] Interview with Alex Solk, 22.11.2011: 'I think the disadvantages are quite small. But you can criticise it for stumping your creativity and it just says: no, you cannot do this… you know it stops you and that is either the limit on the software application or the person using it and I imagine that people actually just do the pure programming to make whatever they want, but I have no experience of that'.

[27] Interview with Vikram Kaushal, 31.10.2011: he clearly stresses that interactivity is an idea worth exploring. He interestingly refers to the American Film, the Space Odyssey, in which a computer speaks with a human.

Learning and Training Difficulties

Underscoring the complexity of parametric packages, a further issue related to the problem of education and training lies[28]. While there are great benefits to using parametric programs, their 'integration into a practice'[29] is quite a big challenge. Practitioners should learn how to use them properly and how to understand the overall management of them. More likely, this is the reason why the number of those who know parametric software is remarkably lower than the other practitioners in a design group. Interestingly, this is even the case in large companies and international offices. For instance, Greg Lynn refers to this training problem and, in particular, the difficulties of scripting in parametric programs:

> We do some scripting and programming in Microstation Generative Components, but this involves sending people in the office to training sessions with Robert Aish as well as emailing him back and forth for specific tasks and having him come to the office every six to nine months. We started using Gehry Technologies software and have found the parametrics very robust, and we are programming and writing custom design tools with this software more and more. For discrete tasks, we use this software all the time now[30].

According to Robert Woodbury, mastering parametrics requires architects to be part designer, part computer scientist and part mathematician, which is more commonly seen among young designers[31]. Designers are not alone in facing increasing complexity in their tasks and tools. Woodbury explains that 'many disciplines face a fundamental need and opportunity to do more with their computing tools'[32]. He refers to six skills, all of which are required for parametric mastery: conceiving data flow, divide-and-conquer strategies, naming, thinking with abstraction, thinking mathematically, and thinking algorithmically[33]. Although parametric pedagogy seems to be a pitfall that perhaps can be gradually resolved within time, it is currently a challenge for many practicing architects. This is the reason why in many projects, notably residential buildings, traditional CAD is frequently used. The usage of parametric

---

[28] Interview with Alex Solk, 22.11.2011:' I think things like Grasshopper or Rhino are complicated enough and you need to train people in it, but ultimately they become less complicated in time as well; because they are used by relatively more people'.

[29] Interview with Rick Sharp, 4.11.2011

[30] Ingeborg M Rocker, 'Calculus-Based Form: An Interview with Greg Lynn', *Architectural Design,* 76/4 (2006), 88-95, p. 95.

[31] Robert Woodbury, *Elements of Parametric Design* (London: Routledge, 2010), p. 8.

[32] Ibid, p. 65

[33] The six skills are described as follows: '1. *Conceiving data flow:* parametric approaches to design aim to provide designers with tools to capture design decisions in an explicit, auditable, editable and re-executable form. 2. *Divide-and-conquer design strategy*: divide the design into parts, design the parts and combine the parts into an entire design, all the while managing the interactions among the parts. 3. *Naming*: parts have names. This is designerly practice, not physical law. But there is a good reason for this – names facilitate communication. 4. *Thinking with abstraction*: to abstract a parametric model is to make it applicable in new situations, to make it depend only on essential inputs and to remove reference to and use of overly specific terms. It is particularly important because much modelling work is similar, and time is always in short supply. 5. *Thinking mathematically:* designers 'do' mathematics. Practically though, designers use mathematics more than they do mathematics. To use mathematics is to begin with established mathematical fact and to rely on it to make a construction or, even more loosely, as a metaphor for a design move. To do mathematics is to derive theorems (new mathematical facts) by inference from prior known statements. 6. Thinking algorithmically: long practice in using, programming and teaching parametric systems shows that, sooner or later, designers will need to write algorithms to make their intended designs... almost all current systems have a so-called scripting language'. (Ibid, pp. 26-34)

design is perhaps seen more in cultural, educational, and commercial projects, or those buildings which are first presented to an architectural competition.

7.1.3. The Distinctions of Parametric Design

The final section of this chapter seeks to clarify the distinctions between parametric packages and traditional CAD and Building Information Modelling (BIM). Before proceeding to what architects say about this theme of exploration, it is worth comparing these two categories (parametric design and traditional CAD) in terms of the aspects that are usually discussed as major differentiations. Figure 6 illustrates a statistical comparison based on these seven items: enhancing the workflow in the design process, offering a more rigorous approach, offering more alternatives, facilitating unexpected solutions, demanding less time for exploration within the design process, offering a higher level of control in production, and bringing competitive advantages.
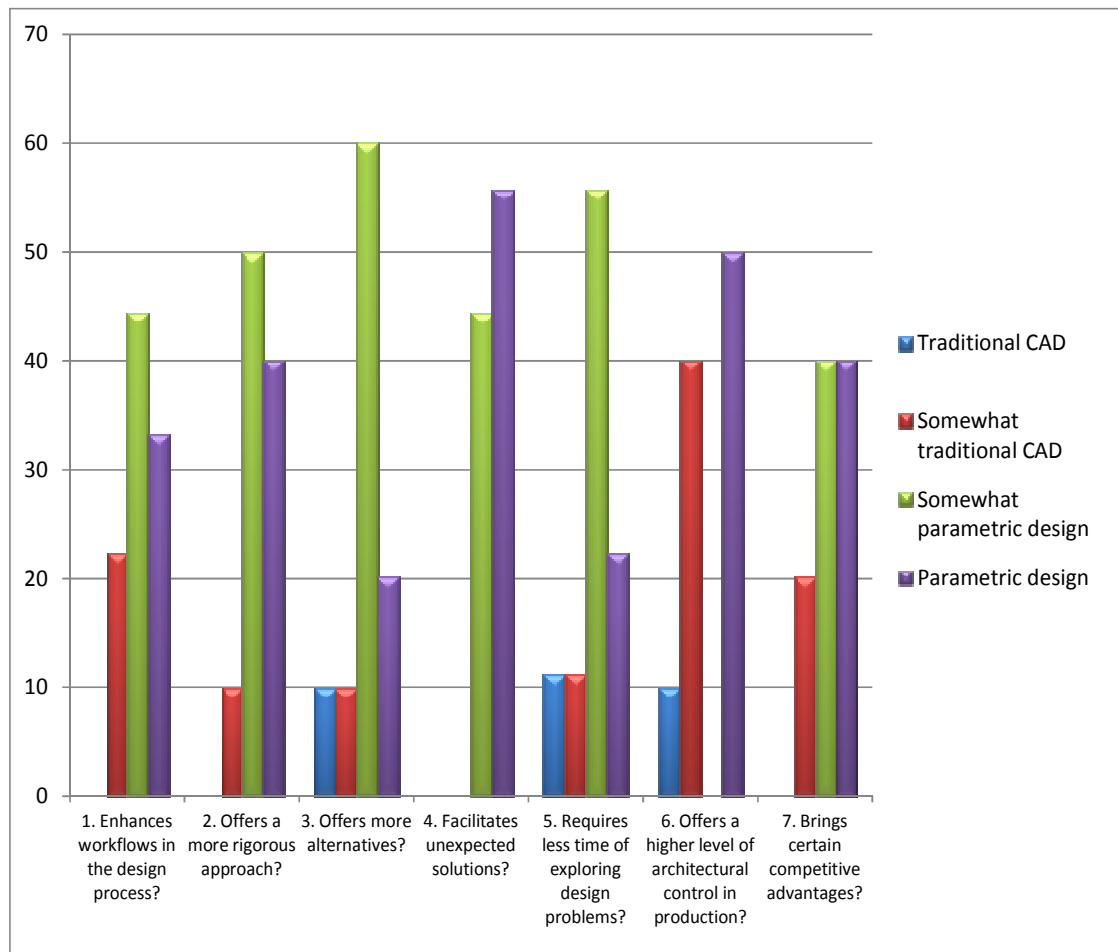
Figure 6: A statistical comparison between traditional CAD and parametric design.

The interviewees were requested to select one of the options from a four-point scale of answers, covering a range from parametric design to traditional CAD. In terms of the categories that these seven questions refer to, the questions can be divided into three classes. Questions one to five reflect concerns about the design process. Question six inquires about production and manufacturing. Finally, question seven asks whether there are competitive advantages in parametric design in comparison with CAD.

As the graph depicts, in almost all of the questions, interviewees see the superiority of parametric design over traditional CAD. In terms of facilitating unexpected solutions (question four), parametric design has a very distinctive role and all of the interviewed architects recognised it clearly. In question six, however, despite the common belief that says parametric design offers higher levels of architectural control, interviewees roughly gave the same level of importance to both parametric design and traditional CAD. Yet, apart from that question, the graph clearly shows that architects can distinguish between traditional CAD and parametric design. However, in order to express the distinctions more specifically, some of the axial aspects will be argued here by referring both to the secondary sources and to the statements of the interviewees.

Firstly, the concept of setting constraints, which is the intrinsic distinction of parametrics from CAD, should be argued. In the past, architects had to think about parameters in their mind and then transfer them onto the computer screen. However, most of those parameters are now set inside the package and, as a result, the architects do not need to focus on them beforehand. Some constraints can also be defined and grouped into themes, such as zoning or circulation, by the user of the parametric package. Alex Solk believes that parametric platforms make constraints externalised in design, whereas traditional CAD meant they were in architects' minds:

> You put your parameters in and then it will generate your form which is constrained by these parameters. I think when you are doing stuff in CAD, you probably have more of those parameters in your head like 'don't go more than six meters towards the next building'[34].

Occasionally, in some CAD packages like AutoCAD, architects have had the chance of automating a part of the design process, although in a quite limited way, by using programming languages such as AutoLISP. However, scripting is an explicit capability in most parametric programs, which enables the designers to go beyond the mouse-based actions. It liberates them from the factory-set limitations of the software. According to William Mitchell, who wrote an introduction to the book *Expressive Form* by Terzidis, a typical early CAD system

---

[34] Interview with Alex Solk, 22.11.2011

was 'a shape factory' with the aim of enhancing productivity like other factories. 'It provided the traditional vocabulary of straight lines, circles, and arcs, together with the standard repertoire of Euclidean constructions, and it was faster'[35].

Traditional CAD packages did not have the capacity to support the relations between the elements of design. Parametric programs enable the designer to apply 'chains of dependency and associativity'[36] to the objects of design whenever requested. Moreover, they provide a higher level of management of all of the drawings in a project, such as plans, sections, and elevations, whereas in traditional CAD everything was separated. Focusing on the associative property, Woodbury offers a clear-cut explanation. He states that previously, in conventional design tools, the creation of a model was easy, but making changes was difficult and tedious. Hence, tools could hinder design exploration. Deleting a part was also simple, since parts were independent. In comparison, parametric design requires the designer to focus on the logic that binds the design components together. Although this can take time, the result is that 'the system takes care of keeping the design consistent with the relationships and thus increases the designer ability to explore ideas by reducing the tedium of rework'[37].

A negative characteristic of parametric design in comparison to CAD is in transforming the position of an architect into a spreadsheet manager. This shift is not really understood by 'the rest of the industry or by clients as a whole'[38]. Having investigated a number of architectural offices, it could be claimed that different duties are usually attached to parametric designers in today's practice that did not exist when working with traditional CAD tools just ten years ago. Therefore, architects see themselves with a metamorphosed identity, which consecutively leads to complex issues such as authorship of the projects.

Finally, parametric design is closer to the reality of a project and its construction. It was argued that the evolutionary process of CAD towards a more realistic approach has resulted in the appearance of parametric programs. Going back to the first releases of CAD packages such as Microstation, most of the architects viewed that generation of software as 'an extension of the drawing technique on a piece of program'[39]. Now, parametric programs offer a different way of conceptualising, which in essence is much closer to the reality of the building. Setting geometrical and physical constraints within design as well as considering

---

[35] From introduction to the book *Expressive Form* by Terzidis. See: Kostas Terzidis, *Expressive Form : A Conceptual Approach to Computational Design* (London: Spon, 2003).

[36] Email from Roland Hudson, 19.10.2011.

[37] Robert Woodbury, *Elements of Parametric Design* (London: Routledge, 2010), p. 24.

[38] Interview with Rick Sharp, 4.11.2011

[39] Interview with Phil Osborne, 7.11.2011

issues such as cost parameters helps architects to become closer and closer to what a building really needs for designing.

Distinctions in Comparison to BIM

Most architects think that both parametric design and BIM are the same, or at least come from the same origins. Though both of them work with parameters, two main factors separate them from each other.

The first item is related to the scripting ability that all parametric programs offer, which enables the architect to have a sort of dialogue with the computer program. Similar to CAD, BIM structure is rigid and sometimes too elaborate for thinking about architectural form. It does not allow architects to enjoy the kind of freedom that they are given while working with parametric packages. BIM packages 'do not allow scripting functionality'[40], which again restrains designers in choosing standard procedural commands.

Emphasising innovative approaches to design, most of the parametric packages have been developed as a result of 'aspirations towards new form', while BIM is 'more about facilitating more efficient design'[41]. BIM is a very good tool on the commercial side, because it allows architects to share and model the whole design. Everything is updated when designers change something, which in this way is quite similar to parametric programs like Grasshopper[42]. However, programs such as Grasshopper are normally used for form-finding or form-generating rather than making a database for the elements of design.

In one sense, BIM is a 'building tool' rather than 'an architectural tool'[43]. It brings all of the stakeholders of a project together to facilitate collaboration throughout the design process. In other words, BIM acts as a link among different actors, whereas parametric packages are more about how to facilitate the process of achieving form. However, a recent piece of research on BIM shows that thinking through problems together, the fundamental purpose of BIM, is often not realised in practice. While BIM tries to link project participants more tightly

---

[40] Interview with Daniel Richard, 8.11.2011
[41] Interview with Amy Hanley, 25.10.2011
[42] Interview with Vikram Kaushal, 31.10.2011
[43] Interview with Alex Solk, 22.11.2011

together technologically, they remain organisationally separated, often suffering from a lack of timely access to crucial information[44].

In comparison with parametric programs, BIM is quite young and perhaps this is the reason why it may seem mysterious to those who are not closely related to the world of computer programs. Nevertheless, it seems that it is more comprehensible to those architects who have background knowledge in CAD, whereas for these groups parametric programs sound extremely peculiar and problematic to learn and work with.

---

[44] G. Neff, B. Fiore-Silfvast, and C. S. Dossick, 'A Case Study of the Failure of Digital Communication to Cross Knowledge Boundaries in Virtual Construction', *Information Communication & Society,* 13/4 (2010), 556-73, p. 556. Their research is based upon ethnographic fieldwork on three construction projects and interviews with 65 architects, engineers, and builders across the USA; a qualitative study on BIM. They also 'argue that people sometimes have a difficult time overcoming the lack of interpretive flexibility in digital coordinating tools, even when those tools are built to encourage interdisciplinary collaboration'.

Chapter 8

Conclusions and Recommendations

This final chapter provides research conclusions and makes recommendations for further research. The first section is composed of three concluding arguments about parametric design. The second part provides a roadmap for future research not only in parametric design but also in the broader domain of computational design.

8.1. Conclusions of the Research

In order to present this conclusion and prepare the ground for further research, I follow the three-part structure of the dissertation described in the introduction. The first part essentially revolves around the parametric issues relating to the design process. Having mentioned the notion of 'role' and 'driver', I explained that the process in which an architectural form is created largely depends upon how the architects define and interpret the notion of 'generation'. Whether the focus is on the process or the final product, the design is the act of computation or representation. Represented forms are normally result-driven and architects design by the aid of a variety of tools including computer programs. However, computers here are just facilitators rather than design tools. In contrast, defining design as a process of computation entails different principles. For the most part, design here is process-driven. Computed forms are designed through a series of algorithmic programming. Algorithms in this sense are design tools, because they produce forms and geometries. Nevertheless, the nature of the algorithm is not dependent on computers. It was mentioned that for some architects like Antoni Gaudi or Frei Otto, computation was meant to be a form-finding process by the aid of concrete materials rather than software. However, computation among today's architects commonly stems from the idea of computer programs. Looking from outside to inside of the design process using the concept of 'driver' showed that, even though parametric design for many architects is assumed as a novel approach to design, it cannot change the primary drivers of every design project, such as the context and the client's brief, because these notions have independent natures.

Concentrating on the concept of 'role' in the design process confirmed two dichotomous positions among parametric designers, namely tool-users and tool-makers. Both of these groups may define design activity in their own way. The tool-making camp has its own proponents among computational designers, while tool-users are generally those who see design as the act of representation. In parallel with the type of designer, their individual tendencies and ideas towards form can be different. Again, another dichotomy arises from the selection of a top-down or bottom-up strategy in the generation of form. For some architects, form has always been created within an inventive process, while for the other groups it can be construed as a discovery. Some also discuss a synthesis of these two views, in which form is performed. For this group, having a pragmatic approach is more important than the duality of form-making or form-finding. In addition to the role of designers and their approach towards form, probing into the role of non-human actors in the parametric realm clarifies some shifts in the importance of sketching from one aspect to another. Although the role of sketching is still

significant in parametric design, its efficacy is now more focused on being a tool for collaboration rather than conceptualisation. Finally, the role of computer programs, as discussed, are becoming increasingly important in today's architectural practice, since many sophisticated software packages like Grasshopper are commonly recognised as parametric tools. In addition, some factors such as time or size of the project might push architects to use parametric programs frequently. However, it is still possible, yet quite difficult, to perform parametric design without software.

Investigating the concept of 'driver' as well as mapping the roles of the design process provided the ground for the second argument: the question of style. If the roles in the parametric realm are ushering in a new state, can it be called a new style? As we have discovered, the term 'style' can be variously interpreted among architects. The quintessential theorist of parametricism, Patrik Schumacher, argues the question of style very extensively by using terms such as epochal style and transitional style. Yet, speaking with architects showed that they are unwilling to call the new trend a style. They argue that doing this can restrict them to a set of methods, or result in imprisonment in a limited framework. Furthermore, the status of parametric tools also makes the position of parametrics as a style dubious. The fact that many late-modernist architects including Norman Foster utilise parametric packages in their design may weaken the argument of parametricism as a style. In Chapter 6, it was mentioned that the question of style would probably be solvable when it was reduced to two positions: parametrics as a conceptual modelling tool and parametrics as a tool for modelling construction. If someone takes into account the parametric conceptualisation, to a certain extent it can be assumed as a stylistic shift. It demands specific knowledge of computational techniques that each architect should learn to employ scripting and computer programming. Thus, in this view, it sounds logical to call parametric design a style. On the other hand, deploying parametric tools in the way seen, especially in the construction and building industry, has a different nature. It is just an extension of what was being undertaken in the past by the aid of traditional CAD. In this sense, these tools are only techniques to help rather than appearing in the frame of a new style.

Finally, the benefits and challenges of parametric design were discussed in Chapter 7. For some practicing architects, the reason why parametric design is viewed as an exclusively distinct approach is because of the certain advantages bestowed to them by this novel way of design. Parametric design furnishes architecture with the ideas of designating constraints to optimise the problem-solution space, churning out a range of alternatives. It lays upon

architects the ability to engineer the design process. However, it leaves many issues unanswered. The main problem with this approach is the lack of a comprehensive methodological framework for design. This is especially true in parametric platforms. These programs should be designed with a parametric approach, and this requires a true understanding of this outlook by the developers of the package. In addition, although parametric programs offer a robust and sophisticated platform, at the same time they bring too much complexity, which in turn brings problems such as learning difficulties. The problem of authorship, which is tangible among computational methods, seems to be a serious concern here too. Thus, notwithstanding many advantages of parametric design, these investigated obstacles act against its promotion to a popular stance among architects.

8.2. Recommendations for Further Research

Despite some voices who insist that parametric design is struggling to determine its current problematic issues, I think it will stand the test of time due to its beneficial engagement with contemporary architecture. However, I believe that architects should always bear in mind the challenges of parametric modelling. At the beginning of each project, it is crucial for the principals of a large practice to make sure that they have competent designers knowledgeable about parametrics in order to cope with the challenges of parametric design. It is also reasonable for a small-sized company or even a freelance architect to ask 'does this project really demand parametric techniques?'

In addition to hands-on questions like the above, I believe parametric design and, in general, computational design requires a long-term plan that should begin within academia. In a similar view to Patrik Schumacher, I think parametric design must be considered a 'design research programme'. Yet I tend to disagree with Schumacher's choice to call parametric design 'parametricism'. My rationale emerges from this research and what I perceived from architectural practice during this study. Nevertheless, I believe without a doubt that, if parametricism is construed as an overall computational picture of the current ethos of architecture, it is definitely worth exploration both in education and in practice as a dominant paradigm. One outcome of this investigation specifically on the educational side would be the introduction of a new framework in architectural pedagogy to guide students to legitimate interdisciplinary knowledge, and to encourage them to go beyond the borders of conventional design.

In my opinion, architecture schools can be a reliable starting point for two reasons. Firstly, most of them contain students with a variety of backgrounds. This variation is more vivid in the countries like the UK where, besides the fame of the schools, the richness of the urban texture in terms of architecture attracts many students from all around the world. Therefore, British schools of architecture are remarkable labs for the examination of the discussed framework. Secondly, architectural schools can be a potential place for the assessment of current computational tools, because, as Donald Schön once mentioned, they are the 'supermarket of alternatives'[1]. A multifarious range of idiosyncrasies can be found among students along with their supervisors in schools, which makes the realm of inquiry more intriguing. In addition, architecture schools can also help the researcher to investigate the cycle of 'homo faber, homo fabricatus', to see the effects of the previous and current tools on the education of architects, and to understand the future requirements both for the architects and the tools.

Some reasons that emerge from architectural practice entail digging even more into architectural education to mark out the significance of the computational approach. In practice, it is often seen that architects tend to react more to technical trends in manufacturing instead of discussing the theory of computation. What is needed more in this realm is a pragmatic approach towards the nature of design, rather than philosophising. Architects may not be keen on reading Deleuze. If someone simply investigates the Deleuzian concepts in the messy land of practice, one of the outcomes would possibly be finding that quite a few architects know about, for instance, the term 'objectile'. However, if they are asked to talk about the strategies of dealing with the client's brief, it is likely to provide a fertile ground of different arguments. As a result, I believe one of the potential topics of investigation in the schools of architecture is to seek the extent of alignment of computational modules with practical needs.

The design activity has always been a one-off process. In a sense just as a symphony, it is conducted in a specific time and place. Although the notes of that symphony have been written once by its own composer, different performances may create different feelings in the audience. I think the entity of practice is similar to that symphony. However, this symphony demands a different performance in which the cycle of education-practice should be seen as practice-education. In this regard, the clear message of practice for education is to deploy

---

[1] Donald A. Schön, *The Design Studio : An Exploration of Its Traditions and Potentials* (London: RIBA Publications for RIBA Building Industry Trust, 1985)

computation in a way that jettisons superficial knowledge of working with a variety of software, and turns the designer into a mere tool-user. Despite the efforts made by software developers to present up-to-date platforms which are aware of design requirements, these packages still have many shortcomings. Those designers who are mere tool-users often feel themselves restrained by the limitations of the software. As a result, creation – the kernel of design – has been drastically threatened. I believe the dynamic nature of practice clearly dictates that students of architecture should not just learn to work with numerous software packages in design. They should instead try to focus on coming out of academia with a specific tool-kit, a palette of techniques composed of the knowledge of programming and scripting, as well as having familiarity with the main computer programs. Only in this way will they be able to go beyond the limitations of the software and act as a meta-designer who is able to bridge the gap between artistic sense and computational techniques. This is my premise and it obviously needs a further research initiative inside academia to be approved and clarified.

Bibliography

3dsCATIA (2012), 'Franck Gehry Testimony | Key role of CATIA for his architecture creations ', <http://www.youtube.com/watch?v=C0WnjbsTT9Q>, accessed 06 August.

Aish, Robert (2011), 'Designing at t + n', *Architectural Design,* 81 (6), 20-27.

Aish, Robert (2012), 'DesignScript: Origins, Explanation, Illustration
Computational Design Modelling', in Christoph Gengnagel, et al. (eds.), (Springer Berlin Heidelberg), 1-8.

Aish, Robert and Woodbury, Robert (2005), 'Multi-level Interaction in Parametric Design', in Andreas Butz, et al. (eds.), *Smart Graphics* (Lecture Notes in Computer Science, 3638: Springer Berlin / Heidelberg), 924-24.

Alexander, Christopher, Ishikawa, Sara, and Silverstein, Murray (1977), *A pattern language : towns, buildings, construction* (New York: Oxford University Press).

Anonymous (2008), 'A conversation between Sanford Kwinter and Jason Payne', in T. Sakamoto and A. Ferré (eds.), *From control to design: parametric/algorithmic architecture*
(Actar-D).

Antoniades, Anthony C. (1990), *Poetics of architecture : theory of design* (New York: Van Nostrand Reinhold) xiv, 303 p.

Archer, Bruce (1979), 'Design as a discipline', *Design Studies,* 1 (1), 17-20.

Archer, L.B. (1984a), 'Systematic method for designers', in Nigel Cross (ed.), *Developments in design methodology*
(Chichester: Wiley), 57-82.

Archer, L.B. (1984b), 'Whatever became of design methodology?', in Nigel Cross (ed.), *Developments in design methodology*
(Chichester: Wiley), 347-49.

ArchitecturalAssociation (2011), 'David Rutten/Grasshopper, Computing Architectural Concepts ', <http://www.aaschool.ac.uk//VIDEO/lecture.php?ID=1212>, accessed 9 October

Arnheim, Rudolf (1993), 'Sketching and the Psychology of Design', *Design Issues,* 9 (2), 15-19.

Barrios, Carlos (2005), 'Transformations on Parametric Design Models
Computer Aided Architectural Design Futures 2005', in Bob Martens and Andre Brown (eds.), (Springer Netherlands), 393-400.

Baxter, Ryan J. and Berente, Nicholas 'The process of embedding new information technology artifacts into innovative design practices', *Information and Organization,* 20 (3-4), 133-55.

Bessant, J. R. (1979a), 'The future with microelectronics: Iann Barron and Ray Currow, Frances Pinter Ltd, London (1979) 243 pp, £9.75', *Design Studies,* 1 (2), 125-26.

Bessant, J. R. (1979b), 'Preparing for design studies: ways of watching: Notes towards appropriate methodology for studying functional specialists', *Design Studies,* 1 (2), 77-83.

Bessant, J. R. (1979c), 'The rise and fall of tracked hovercraft: CEI case study: E.R. Laithwaite, Mechanical Engineering Publications, London (1978) 243 pp', *Design Studies,* 1 (2), 126-27.

Bigkif (2012), 'Ivan Sutherland : Sketchpad Demo ', <http://www.youtube.com/watch?v=USyoT_Ha_bA&list=PL23D79E83F8F52102&index=9&feature=plpp_video>, accessed 10 April.

Bilda, Zafer, Gero, John S., and Purcell, Terry (2006), 'To sketch or not to sketch? That is the question', *Design Studies,* 27 (5), 587-613.

Bill, Max and Johnston, Pamela (2010), *Form, function, beauty = gestalt* (London: Architectural Association) 168. [14] p.

Binder, Thomas, et al. (2004), 'Supporting configurability in a mixed-media environment for design students', *Personal and Ubiquitous Computing,* 8 (5), 310-25.

Brause, Rita S. (2000), *Writing your doctoral dissertation : invisible rules for success* (London: Falmer) xvi, 163 p.

Brewer, John D. (2000), *Ethnography* (Buckingham: Open University Press) viii, 211 p.

Brey, P. (2005), 'The Epistemology and Ontology of Human-Computer Interaction', *Computer Science,* 15 (3).

Broadbent, Geoffrey and Ward, Anthony (1969), *Design methods in architecture* (London: Lund Humphries) 204p.

Broadbent, Geoffrey and Brebbia, C. A. (2008), *Eco-architecture II : harmonisation between architecture and nature* (Southampton: WIT) 349 p.

Brooks, Frederick P. (1995), *The mythical man-month : essays on software engineering* (Anniversary ed. edn.; Reading, Mass. ; Wokingham, England: Addison-Wesley Pub. Co).

Brown, Tim and Katz, Barry (2009), *Change by design : how design thinking transforms organizations and inspires innovation* (1st edn.; New York: Harper Business) viii, 264 p.

Burry, Jane and Burry, Mark (2010), *The new mathematics of architecture* (London: Thames & Hudson).

Caldas, Luisa Gama and Norford, Leslie K. (2002), 'A design optimization tool based on a genetic algorithm', *Automation in Construction,* 11 (2), 173-84.

Carlile, Paul R. (2002), 'A Pragmatic View of Knowledge and Boundaries: Boundary Objects in New Product Development', *Organization Science,* 13 (4), 442-55.

Carlos Roberto, Barrios Hernandez (2006), 'Thinking parametric design: introducing parametric Gaudi', *Design Studies,* 27 (3), 309-24.

Carpo, Mario (2001), *Architecture in the age of printing : orality, writing, typography, and printed images in the history of architectural theory* (Cambridge, Mass. ; London: MIT Press) viii, 246 p.

Carpo, Mario (2011), *The alphabet and the algorithm* (Cambridge, Mass. ; London: MIT Press) xi, 169 p.

Carpo, Mario and Lemerle, Fre de rique (2008), *Perspective, projections and design : technologies of architectural representation* (London: Routledge) xi, 196 p.

Carroll, John M., Thomas, John C., and Malhotra, Ashok (1979), 'Clinical-- experimental analysis of design problem solving', *Design Studies,* 1 (2), 84-92.

Casakin, H. P. (2005), 'Design aided by visual displays: a cognitive approach', *Architectural and Planning Research,* 22 (3), 16.

Chasznar, André (2006), *Blurring the lines* (Chichester: Wiley-Academy) 224 p.

Cheng, N. (2005), 'Learning Design with Digital Sketching', in B. & Brown Martens, A. (ed.), *Computer Aided Architectural Design Futures 2005* (pringer Netherlands), 291-300.

Chu, Karl (2006), 'Metaphysics of genetic architecture and computation', *Architectural Design,* 76 (4), 38-45.

Clayton, Mark J., Warden, Robert B., and Parker, Thomas W. (2002), 'Virtual construction of architecture using 3D CAD and simulation', *Automation in Construction,* 11 (2), 227-35.

Coates, Paul (2010), *Programming.Architecture* (Abingdon: Routledge) 187 p.

columbiauniversity (2012), 'Manuel De Landa on Deleuze and the use of genetic algorithm in architecture', <http://www.youtube.com/watch?v=50-d_J0hKz0>, accessed 20 May.

Coyne, Richard (1990), *Knowledge-based design systems* (Addison-Wesley) xi, 567 p.

Coyne, Richard (1995), *Designing information technology in the postmodern age : from method to metaphor* (Cambridge, Mass. ; London: MIT Press) xiii,399p.

Coyne, Richard and Snodgrass, Adrian (1991), 'Is designing mysterious? challenging the dual knowledge thesis', *Design Studies,* 12 (3), 124-31.

Coyne, Richard, Newton, Sidney, and McLaughlin, Sally (1996), *Computers in practice : a survey of computers in architectural practice* (University of Edinburgh, Department of Architecture).

Coyne, Richard Ed (1997), *Design and the net* (EuropIA Productions).

Cross, Nigel (1979a), 'How things don't work: Victor Papanek and James Hennessey, Pantheon Books, New York (1977) 154 pp, £3.45', *Design Studies,* 1 (2), 127-28.

Cross, Nigel (1979b), 'The Utopian design policy of the Shakers', *Design Studies,* 1 (2), 93-100.

Cross, Nigel (1984), *Developments in design methodology* (Chichester: Wiley) x,357p.

Cross, Nigel (1985), *The common writer : life in nineteenth-century Grub Street* (Cambridge: Cambridge University Press) [vi,272]p.

Cross, Nigel (2001), 'Designerly ways of knowing: Design discipline versus design science (The 1920s and the 1960s, two important periods in the modern history of design)', *Design Issues,* 17 (3), 49-55.

Cross, Nigel (2004), 'Expertise in design: an overview', *Design Studies,* 25 (5), 427-41.

Cross, Nigel (2006), *Designerly ways of knowing* (London: Springer) xiii, 114 p.

Cross, Nigel (2007) *Designerly Ways of Knowing* [online text], <http://openurl.man.ac.uk/sfxlcl3?sid=google&auinit=N&aulast=Cross&title=Designerly%20ways%20of%20knowing&genre=book&isbn=3764384840&date=2007>

Cross, Nigel (2008), *Engineering design methods : strategies for product design* (4th ed. edn.; Hoboken, N.J.: Wiley ; Chichester : John Wiley [distributor]) xii, 217 p.

Cross, Nigel (2011), *Design thinking : understanding how designers think and work* (Oxford: Berg Publishers).

Cross, Nigel and Roy, Robin (1974), *Man-made futures. Readings in society, technology and design. Edited by Nigel Cross, David Elliott, and Robin Roy* (London: Hutchinson in association with The Open University Press) pp. 365: plates. 25 cm.

Cross, Nigel and Roy, Robin (1975), *Man-made futures : design and technology* ([S.l.]: Open University Pr.).

Cross, Nigel and Edmonds, Ernest (2003), *Expertise in design : Design Thinking Research Symposium 6* (Sidney: Creativity and Cognition Press, University of Technology Sidney) 1 v.

Cross, Nigel, Christiaans, Henri, and Dorst, Kees (1996), *Analysing design activity* (Chichester: Wiley) xi,463p.

Darke, Jane (1979), 'The primary generator and the design process', *Design Studies,* 1 (1), 36-44.

Darses, F. , et al. (2008), 'Is There Anything to Expect from 3D Views in Sketching Support Tools?', in J. S. & Goel Gero, A. K. (ed.), *Design Computing and Cognition '08* (Springer Netherlands), 283-302.

Deleuze, Gilles (1993), *The fold : Leibniz and the Baroque* (Athlone P.).

Derix, Christian (2009), 'In-Between Architecture Computation', *International Journal of Architectural Computing,* 7 (4), 565-86.

Dokonal, W & Knight, M.W. (2011), 'What is the state of digital architectural design?', <http://cumincades.scix.net/cgi-bin/works/Home>, accessed 25 April.

Dong, Andy, Vande Moere, Andrew, and Gero, John S. (2007), *Computer-aided architectural design futures (CAADFutures) 2007 : proceedings of the 12th international CAADFutures conference* (Dordrecht: Springer) xii, 602 p.

Dorst, Kees and Dijkhuis, Judith (1995), 'Comparing paradigms for describing design activity', *Design Studies,* 16 (2), 261-74.

Dorst, Kees and Cross, Nigel (2001), 'Creativity in the design process: co-evolution of problem-solution', *Design Studies,* 22 (5), 425-37.

Doucet, I. and Janssens, N. (2011), 'transdisciplinarity, the hybridisation of knowledge production and space-related research', in I. Doucet and N. Janssens (eds.),

transdisciplinary knowledge production in architecture and urbanism: towards hybrid modes of inquiry (11; Dordrecht: Springer), 1-14.

Engeli, Maia (2000), *Digital stories : the poetics of communication* (Basel ; Boston: Birkhäuser) 93 p.

Fallman, Daniel (2003), 'Design-oriented human-computer interaction', *Proceedings of the SIGCHI conference on Human factors in computing systems* (Ft. Lauderdale, Florida, USA: ACM), 225-32.

Fischer, G., et al. (2004), 'Meta-Design: A Manifesto for End user Development', *Communications of the ACM,* 47 (9), 33-37.

Flanagan, Robert H. (2005), 'Generative logic in digital design', *Automation in Construction,* 14 (2), 241-51.

Fleischmann, Moritz, et al. (2012), 'Material Behaviour: Embedding Physical Properties in Computational Design Processes', *Architectural Design,* 82 (2), 44-51.

Garcia, Mark (2010), *The diagrams of architecture* (Chichester: John Wiley) 320 p.

Gasparski, W. W. (1979), 'Praxiological-- systemic approach to design studies', *Design Studies,* 1 (2), 101-06.

Gehry, Frank O. (1996), 'Architectural Projects: Current and Recently Completed Work', *Bulletin of the American Academy of Arts and Sciences,* 49 (5), 36-55.

Gero, John S. (1996), *Advances in formal design methods for CAD* (London: Chapman & Hall, on behalf of the International Federation for Information Processing) vi,298p.

Gero, John S. (1998), 'Concept Formation in Design', *Knowledge-Based Systems,* 11, 429-35.

Gero, John S. (2004), *Design computing and cognition '04* (Dordrecht: Kluwer Academic Publishers) x, 633 p.

Gero, John S. (2006), *Design computing and cognition '06* (Dordrecht, The Netherlands ; [London]: Springer) xii, 714 p.

Gero, John S. (2011), *Design computing and cognition '10* ([Dordrecht?]: Springer) xv, 744 p.

Gero, John S. and Maher, Mary Lou (1993), *Modeling creativity and knowledge-based creative design : edited by John S Gero and Mary Lou Maher* (Hillsdale, NJ: Lawrence Erlbaum).

Gero, John S. and Tyugu, E. (1994a), *Formal design methods for CAD : proceedings of the IFIP TC5/WG5.2 Workshop on Formal Design Methods for CAD, Tallinn, Estonia, 16-19 June 1994* (Amsterdam ; Oxford: Elsevier) viii,362p.

Gero, John S. and Goel, Ashok K. (2008), *Design computing and cognition '08 : Proceedings of the Third International Conference on Design Computing and Cognition* ([Dordrecht ; London]: Springer) xiii, 736 p.

Gero, John S. Ed and Sudweeks, Fay Ed (1992), *Computational models of creative design : 2nd International round table conference : Preprinted papers* (University of Sydney).

Gero, John S. Ed and Tyugu, Enn Ed (1994b), *Formal design methods for CAD : Workshop : Papers and discussions* (Amsterdam ; New York: Elsevier).

Gero, John S. Ed and Maher, Mary Lou Ed (1999), *Computational models of creative design IV* (University of Sydney).

Gero, John S. Ed, Tversky, Barbara Ed, and Purcell, Terry Ed (2001), *Visual and spatial reasoning in design II* (KCDCC).

Globalbeehive (2012), 'Kurt Godel: The World's Most Incredible Mind', <http://www.youtube.com/watch?v=i2KP1vWkQ6Y&feature=related>, accessed 10 April.

Graef, Richard and Coyne, John J. (1963), *[Mit Christus auf Du und Du: Betrachtungen.] Christ my Friend. Meditations for daily use ... Translated ... by John J. Coyne* (pp. 223. Clonmore & Reynolds: Dublin; Burns & Oates: London) 8º.

Gregory, S. A. and Commander, M. W. (1979), 'New materials adoption study: some contributions to design knowledge', *Design Studies,* 1 (2), 107-12.

Groat, Linda and Wang, David (2002), *Architectural research methods* (New York ; [Chichester]: Wiley) xvii, 389 p.

Guy, Simon and Karvonen, Andrew (2011), 'using sociotechnical methods', in A. DALE, & MASON, J. (ed.), *Understanding social research: thinking creatively about method.* (London: SAGE), 120-33.

Hagan, Susannah (2008), *Digitalia : architecture and the digital, the environmental and the avant-garde* (London: Routledge) xvi, 149 p.

Hammersley, Martyn and Atkinson, Paul (2007), *Ethnography : principles in practice* (3rd ed. edn.; London: Routledge) xi, 275 p.

Harper, Douglas (2012), 'Online Etymology Dictionary', <http://www.etymonline.com/index.php?term=style&allowed_in_frame=0>, accessed 8 May.

Hart, Chris (2001), *Doing a literature search : a comprehensive guide for the social sciences* (London: SAGE) 194p.

Henderson, Kathryn (1999), *On line and on paper : visual representations, visual culture, and computer graphics in design engineering* (Inside technology; Cambridge, Mass.: MIT Press) x, 237 p.

Houdart, S. (2008), 'Copying, Cutting and Pasting Social Spheres:Computer Designers' Participation in Architectural Projects', *Science Studies,* 21 (1), 47-63.

Hudson, Roland (2010), 'Strategies for parametric design in architecture', (University of Bath).

Janssen, Patrick HT. (2006), 'The role of preconception in design', in John S. Gero (ed.), *Design computing and cognition '06*
 (Dordrecht, The Netherlands ; [London]: Springer), 365-83.

Jencks, Charles (2002), *The new paradigm in architecture : the language of post-modernism* (7th ed. edn.; New Haven ; London: Yale University Press) 304p.

Jencks, Charles (2005), *The iconic building : the power of enigma* (London: Frances Lincoln).

Jencks, Charles and Kropf, Karl (1997), *Theories and manifestoes of contemporary architecture* (Chichester: Academy Editions) 312p.

Jencks, Charles and Kropf, Karl (2006), *Theories and manifestoes of contemporary architecture* (2nd ed. edn.; Chichester: Wiley-Academy) 378 p.

Jencks, Charles and Jencks, Charles What is post-modernism (2007), *Critical modernism : where is post-modernism going?* ([5th ed.]. edn.; Chichester: Wiley-Academy) 240 p.

Jones, J.C. (1984), 'A method of systematic design ', in Nigel Cross (ed.), *Developments in design methodology*
 (Chichester: Wiley), 9-30.

Kirsh, David and Maglio, Paul (1994), 'On distinguishing epistemic from pragmatic action', *Cognitive Science,* 18 (4), 513-49.

Knight, M., et al. (2005), 'Contemporary digital techniques in the early stages of design - The effect of representation differences in current systems', *Computer Aided Architectural Design Futures 2005, Proceedings*, 165-74.

Kocaturk, T. & Veltkamp, M. (2005), 'Interdisciplinary knowledge modelling for free-form design - an educational experiment', in B. & Brown Martens, A. (ed.), *Computer Aided Architectural Design Futures 2005* (pringer Netherlands), 465-74.

Kolarevic, Branko (2003), *Architecture in the digital age : design and manufacturing* (London: Spon) v, 314 p.

Kolarevic, Branko (2005a), *Architecture in the digital age : design and manufacturing* (New York, NY: Spon Press) v, 314 p.

Kolarevic, Branko (2005b), 'Digital Morphogenesis', in Branko Kolarevic (ed.), *Architecture in the digital age : design and manufacturing* (New York, NY: Spon Press), v, 314 p.

Kotnik, Toni (2012), 'Algorithmic Design. Structuralism Reloaded?',
        <http://www.schwartz.arch.ethz.ch/Publikationen/Dokumente/Structuralismus.pdf>,
        accessed 6 January.
Kotnik, Toni (2010), 'Digital Architectural Design as Exploration of Computable Functions',
        International Journal of Architectural Computing, 8 (1), 17.
Koutamanis, Alexander and Mitossi, Vicky (1993), 'Computer vision in architectural design',
        Design Studies, 14 (1), 40-57.
Koutamanis, Alexander (2005), 'Sketching with Digital Pen and Paper', in B. & Brown Martens,
        A. (ed.), *Computer Aided Architectural Design Futures 2005* (pringer Netherlands), 321-
        30.
Krasny, Elke (2008), *The force is in the mind : the making of architecture* (Basel: Birkhäuser).
Kuhn, Thomas.S. (1996), *The Structure of Scientific Revolutions* (University of Chicago Press).
Kvale, Steinar and Brinkmann, Svend (2009), *InterViews : learning the craft of qualitative
        research interviewing* (2nd ed. edn.; Thousand Oaks ; London.: Sage Publications).
Lally, Sean and Young, Jessica (2006), *Softspace : from a representation of form to a simulation
        of space* (New York: Taylor and Francis).
Langdon, Richard and Cross, Nigel (1984), *Design and society : the proceedings of the Design
        and Society Section of an international conference on design policy held at the Royal
        College of Art, London, 20-23 July 1982 : a conference organised by the Department of
        Design Research at the Royal College of Art in collaboration with the Design Research
        Society and The Design Council* (London: Design Council) 146p.
Langdon, Richard Ed, et al. (1984), *Design policy* (Design Council).
Latour, B. (2007), 'Can we get our materialism back, please?', *Isis,* 98 (1), 138-42.
Latour, B. and Yaneva, A. (2008), 'Give me a Gun and I will Make All Buildings Move: An ANT's
        View of Architecture', in Reto Geiser (ed.), *Explorations in architecture : teaching,
        design, research*
 (Basel: Birkhaeuser), 80-89.
Latour, Bruno (2011), *Drawing Things Together* (The Map Reader: John Wiley & Sons, Ltd) 65-
        72.
Lawson, Bryan (1994), *Design in mind* (Oxford: Butterworth Architecture) 146p.
Lawson, Bryan (2001), *The language of space* (Oxford: Architectural) v, 263 p., [12] p. of col.
        plates.
Lawson, Bryan (2004), *What designers know* (Oxford: Architectural Press) xi, 127 p.
Lawson, Bryan (2006), *How designers think : the design process demystified* (4th ed. edn.;
        Amsterdam ; London: Architectural) xii, 321 p.
Lawson, Bryan and Dorst, Kees (2009), *Design expertise* (Oxford: Architectural Press) 321 p.
Leach, Neil (2002), *Designing for a digital world* (London: Academy Editions) 128p.
Leach, Neil (2009), 'Digital Morphogenesis', *Architectural Design,* 79 (1), 32-37.
Lee, G., et al. (2010), 'Usability principles and best practices for the user interface design of
        complex 3D architectural design and engineering tools', *International Journal of
        Human Computer Studies,* 68 (1-2), 90-104.
Lee, Jae Yeol and Kim, Kwangsoo (1996), 'Geometric reasoning for knowledge-based
        parametric design using graph representation', *Computer-Aided Design,* 28 (10), 831-
        41.
Lewin, Douglas (1979), 'On the place of design in engineering', *Design Studies,* 1 (2), 113-17.
Lindsey, Bruce and Gehry, Frank O. (2001), *Digital Gehry : material resistance, digital
        construction* (Basel ; Boston: Birkha\0308user) 93 p.
Louridas, Panagiotis (1999), 'Design as bricolage: anthropology meets design thinking', *Design
        Studies,* 20 (6), 517-35.
Lynn, Greg (1999), *Animate form* (New York: Princeton Architectural Press) 203p.
Lynn, Greg (2004), *Folding in architecture* (Rev. ed. edn.; Chichester: Wiley-Academy) 111 p.

M Rocker, Ingeborg (2006), 'Calculus-based form: an interview with Greg Lynn', *Architectural Design,* 76 (4), 88-95.

Maher, Mary Lou Ed, Coyne, Richard Ed, and Newton, Sid Ed (1994), *Multimedia and design* (Key Centre of Design Computing).

Martegani, Paolo (2000), *Digital design : new frontiers for the objects* (The IT revolution in architecture; Basel ; Boston: Birkhäuser) 93 p.

McCullough, Malcolm (2006), '20 years of scripted space', *Architectural Design,* 76 (4), 12-15.

Meredith, Michael (2008), 'Never enough', in Tomoko Sakamoto and Albert Ferre (eds.), *From control to design : parametric/algorithmic architecture* (Barcelona ; New York: Actar-D), 239 p.

Miles, Matthew B. and Huberman, A. Michael (1994), *Qualitative data analysis : an expanded sourcebook* (2nd ed. edn.; Thousand Oaks, Calif. ; London: Sage).

Mitchell, William J. (1990), *The logic of architecture : design, computation, and cognition* (Cambridge, Mass.: MIT Press).

Mitchell, William J. (1995), *City of bits : space, place, and the infobahn* (Cambridge, Mass. ; London: MIT Press) 225p.

Monedero, Javier (2000), 'Parametric design: a review and some experiences', *Automation in Construction,* 9 (4), 369-77.

Moussavi, Farshid and Lopez, Daniel (2009), *The function of form* (Barcelona: Actar ; New York : Harvard University, Graduate School of Design) 515 p.

Neff, G., Fiore-Silfvast, B., and Dossick, C. S. (2010), 'A Case Study of the Failure of Digital Communication to Cross Knowledge Boundaries in Virtual Construction', *Information Communication & Society,* 13 (4), 556-73.

Otto, Frei and Songel, Juan María (2010), *A conversation with Frei Otto* (English edn., Conversations: a Princeton Architectural Press series; New York: Princeton Architectural Press) 96 p.

OxfordUniversity (2010), *Oxford Dictionary of English* (Oxford: Oxford University Press).

Oxman, Neri (2010), 'Structuring Materiality: Design Fabrication of Heterogeneous Materials', *Architectural Design,* 80 (4), 78-85.

Oxman, Rivka (2006), 'Theory and design in the first digital age', *Design Studies,* 27 (3), 229-65.

Oxman, Rivka (2008a), 'Digital architecture as a challenge for design pedagogy: theory, knowledge, models and medium', *Design Studies,* 29 (2), 99-120.

Oxman, Rivka (2008b), 'Performance based Design: Current Practices and Research Issues ', *IJAC International Journal of Architectural Computing* 6(1), 1-17.

Özkar, M. (2007), 'Learning by Doing in the Age of Design Computation', in A. Dong, Moere, A. V. , Gero, J. S. (ed.), *Computer-Aided Architectural Design Futures (CAADFutures) 2007* (Springer Netherlands), 99-112.

Peng, Chengzhi (2003), *Design through digital interaction : computing communications and collaboration on design* (Bristol: Intellect) xi, 212 p.

Penttilä, H (2006), 'Describing the changes in architectural information technology to understand design complexity and free form architectural expression', *ITcon* 11 (Special Issue The Effects of CAD on Building Form and Design Quality), 395-408.

Pessant, J. R. and McMahon, B. J. (1979), 'Participant observation of a major design decision in industry', *Design Studies,* 1 (1), 21-26.

Picon, Antoine (2006), 'Architecture and virtuality - Towards a new material condition', *Arq,* (63), 10-15.

Picon, Antoine (2010), *Digital culture in architecture : an introduction for the design professions* (Boston, MA: Birkhaeuser).

Picon, Antoine (2011), 'Architecture and Mathematics: Between Hubris and Restraint', *Architectural Design,* 81 (4), 28-35.

Redström, Johan (2006), 'Towards user design? On the shift from object to user as the subject of design', *Design Studies,* 27 (2), 123-39.

RiceUniversity (2012), 'Facticity by Neil Denari', <http://www.youtube.com/watch?v=XKQyW053Ois>, accessed 26 Jan.

Rickaby, Peter (1979), 'Speculations on the future practice of architecture', *Design Studies,* 1 (2), 118-23.

Rittel, H.W.J (1984), 'Second-generation design methods', in Nigel Cross (ed.), *Developments in design methodology*
(Chichester: Wiley), 317-27.

Rittel, H.W.J and Webber, M.M (1984), 'Planning Problems Are Wicked Problems', in Nigel Cross (ed.), *Developments in design methodology*
(Chichester: Wiley), 135-44.

Rosa, Joseph (2003), *Next generation architecture : folds, blobs and boxes* (New York: Rizzoli) 240 p.

Rowe, Peter G. (1987), *Design thinking* (Cambridge, Mass.: MIT Press).

Rowe, Peter Noel (1967), *Engineering research and undergraduate teaching ... An inaugural lecture, etc* (London: H. K. Lewis & Co.) pp. 15: plates. 26 cm.

Sakamoto, Tomoko and Ferre, Albert (2008), *From control to design : parametric/algorithmic architecture* (Barcelona ; New York: Actar-D) 239 p.

Saunders, William S. and Rowe, Peter G. (1996), *Reflections on architectural practices in the nineties* (New York: Princeton Architectural Press) 258p.

Scheurer, Fabian and Stehling, Hanno (2011), 'Lost in Parameter Space?', *Architectural Design,* 81 (4), 70-79.

Schnabel, Marc Aurel (2007), 'Parametric designing in architecture', in Andy Dong, Andrew Vande Moere, and John S. Gero (eds.), *Computer-aided architectural design futures (CAADFutures) 2007 : proceedings of the 12th international CAADFutures conference*
(Dordrecht: Springer), 237-50.

Schön, Donald A. (1983), *The reflective practitioner : how professionals think in action* (New York : Basic Books, c1983(1991)).

Schön, Donald A. (1985), *The design studio : an exploration of its traditions and potentials* (London: RIBA Publications for RIBA Building Industry Trust).

Schön, Donald A. (1987), *Educating the reflective practitioner* (1st ed. edn.; San Francisco, CA: Jossey-Bass).

Schön, Donald A. (1988), 'Designing: Rules, types and words', *Design Studies,* 9 (3), 181-90.

Schön, Donald A. (1991a), *The reflective practitioner : how professionals think in action* (Aldershot: Avebury) x,374p.

Schön, Donald A. (1991b), *The Reflective turn : case studies in and on educational practice* (New York: Teachers College, Columbia University).

Schön, Donald A. and Wiggins, Glenn (1992), 'Kinds of seeing and their functions in designing', *Design Studies,* 13 (2), 135-56.

Schön, Donald A. and Rein, Martin (1994), *Frame reflection : toward the resolution of intractable policy controversies* (New York: BasicBooks) xx, 247 p.

Schumacher, Patrik (2004), *Digital Hadid : landscapes in motion* (The IT revolution in architecture; Basel ;: Boston : Birkhäuser) 93 p.

Schumacher, Patrik (2012), 'Design Research within the Parametric Paradigm', (updated 6 July 2011)
<http://www.patrikschumacher.com/Texts/Design%20Research%20at%20Zaha%20Hadid%20Architects.htm>, accessed 10 January

Schumacher, Patrik (2009a), 'Parametricism: A New Global Style for Architecture and Urban Design', *Architectural Design,* 79 (4), 14-23.

Schumacher, Patrik (2012), 'The Parametricist Epoch: Let the Style Wars Begin ',
        <http://www.patrikschumacher.com/Texts/The%20Parametricist%20Epoch_Lets%20t
        he%20Style%20Wars%20Begin.htm>, accessed 8 January.

Schumacher, Patrik (2012), 'Architecture Schools as Design Research Laboratories',
        <http://www.patrikschumacher.com/Texts/Architecture%20Schools%20as%20Design
        %20Research%20Laboratories.htm>, accessed 8 January.

Schumacher, Patrik (2011b), *The Autopoiesis of Architecture: A New Framework for
        Architecture* (Wiley).

Schumacher, Patrik (2012), *The Autopoiesis of Architecture, Volume II: A New Agenda for
        Architecture* (John Wiley & Sons).

Schumacher, Patrik (2008b), 'Parametricism as Style - Parametricist Manifesto', 4.
        <http://www.patrikschumacher.com/Texts/Parametricism%20as%20Style.htm>,
        accessed 6 July 2011.

Schumacher, Patrik (2008c), 'Style as Research Programme'.
        <http://www.patrikschumacher.com/Texts/Style%20as%20Research%20Programme.h
        tm>, accessed 6 July 2011.

Schumacher, Patrik (2012), 'Interview: Patrik Schumacher by Feng Xu',
        <http://www.patrikschumacher.com/Texts/Interview_WA_May%2009_english.htm>,
        accessed 26 January

Simon, Herbert A. (1973), 'The structure of ill structured problems', *Artificial Intelligence,* 4 (3-
        4), 181-201.

Simon, Herbert A. (1996), *The sciences of the artificial* (3rd ed. edn.; Cambridge, Mass. ;
        London: MIT Press).

Smith, John (1979), 'Morality and architecture: David Watkins, Clarendon Press, Oxford (1977)
        115 pp, £', *Design Studies,* 1 (2), 128-28.

Smith, Rick (2012), 'Technical Notes from experiences and studies in using Parametric and BIM
        architectural software', <http://www.vbtllc.com/index_rhinobim.html>, accessed 16
        April.

Snodgrass, Adrian and Coyne, Richard (2006), *Interpretation in architecture : design as a way
        of thinking* (London: Routledge) xi, 332 p.

Soo Meng, Jude Chua (2009), 'Donald Schön, Herbert Simon and The Sciences of the Artificial',
        Design Studies, 30 (1), 60-68.

Spuybroek, Lars (2002), 'WetGRID: The soft machine of vision', in Neil Leach (ed.), *Designing
        for a digital world*
 (London: Academy Editions), 128p.

Stavric, M. and Marina, O. (2011), 'Parametric Modelling for Advanced Architecture',
        International Journal of Applied Mathematics and Informatics 5(1), 9-16.

Steele, Brett (2010), 'Weapons of the Gods', in Jane Burry and Mark Burry (eds.), *The new
        mathematics of architecture* (London: Thames & Hudson).

Steele, James (2001), *Architecture and computers : action and reaction in the digital design
        revolution* (London: Laurence King) 240 p.

Suwa, Masaki and Tversky, Barbara (1997), 'What do architects and students perceive in their
        design sketches? A protocol analysis', *Design Studies,* 18 (4), 385-403.

Szalapaj, Peter (2001), *CAD principles for architectural design : analytical approaches to
        computational representation of architectural form* (Oxford: Architectural Press) xiv,
        242 p.

Szalapaj, Peter (2005), *Contemporary architecture and the digital design process* (Amsterdam ;
        London: Elsevier Architectural Press) xii, 275 p.

TED (2012), 'Greg Lynn on calculus in architecture',
        <http://www.ted.com/talks/lang/en/greg_lynn_on_organic_design.html>, accessed 10
        May.

Terzidis, Kostas (2003), *Expressive form : a conceptual approach to computational design* (London: Spon).

Terzidis, Kostas (2006), *Algorithmic architecture* (1st ed. edn.; Amsterdam ; London: Architectural Press).

TheHarvardGSD (2012), 'The Eclipse of Beauty: Parametric Beauty (Mario Carpo, Michael Meredith, Ingeborg Rocker)', <http://www.youtube.com/watch?v=OxN4LWPlwX8&list=PL23D79E83F8F52102&index=5&feature=plpp_video>, accessed 10 May.

TheHarvardGSD (2012), 'Jeffrey Kipnis and Reinhold Martin, 'What Good Can Architecture Do?' ', <http://www.youtube.com/watch?v=HDo40Fr41os&feature=relmfu>, accessed 12 May.

theRSAorg (2012), 'Biomimicry in Architecture', <http://www.youtube.com/watch?v=ijIfdswvh5U>, accessed 06 August.

UMTaubmanCollege (2012), 'Greg Lynn: University of Michigan Taubman College Future of Design', *the Future of Design Conference* <http://www.youtube.com/watch?v=AqmN0-zVAK8>, accessed 18 February

Valkenburg, Rianne and Dorst, Kees (1998), 'The reflective practice of design teams', *Design Studies,* 19 (3), 249-71.

Vries, Marc de, Cross, Nigel, and Grant, Donald P. (1993), *Design methodology and relationships with science* (Dordrecht ; London: Kluwer Academic Publishers) vi, 327 p.

Waters, John K. (2003), *Blobitecture : waveform architecture and digital design* (Gloucester, Mass.: Rockport ; Hove : RotoVision) 183 p.

Weston, Richard (2003), *Materials, form and architecture* (London: Laurence King) 225, [15] p.

Whorf, Benjamin Lee and Carroll, John B. (1991), *Language, thought, and reality : selected writings of Benjamin Lee Whorf* ([S.l.] : MIT Press, 1956).

Wikipedia (2012), 'Imre Lakatos ', <http://en.wikipedia.org/wiki/Imre_Lakatos>, accessed 8 May.

Wikipedia (2012), 'Latham Report', <http://en.wikipedia.org/wiki/Latham_Report>, accessed 8 January

Wikipedia (2012), 'Linguistic relativity', <http://en.wikipedia.org/wiki/Linguistic_relativity>, accessed 20 March

Wikipedia (2012), 'Turing machine ', <http://en.wikipedia.org/wiki/Turing_machine>, accessed 20 March

Woodbury, Robert (2010), *Elements of parametric design* (London: Routledge) xi, 300 p.

Yaneva, Albena (2005), 'Scaling Up and Down', *Social Studies of Science,* 35 (6), 867-94.

Yaneva, Albena (2009), *Made by the Office for Metropolitan Architecture : an ethnography of design* (Rotterdam: 010 Publishers) 111 p., [20] p. of plates.

Yaneva, Albena (2011), 'from reflecting-in-action towards mapping of the real', in I. Doucet and N. Janssens (eds.), *transdisciplinary knowledge production in architecture and urbanism: towards hybrid modes of inquiry* (11; Dordrecht: Springer), 1-14.

Yaneva, Albena (2012), 'Mapping controversies in architecture', (Burlington: Ashgate Pub. Co.).

Zarei, Yasser (2012), 'When Practice Dictates Change: The necessity of a new framework in architectural education', *Emerging Boundaries of Research Principles & Praxis* (Scott Sutherland School of Architecture and Built Environment, Robert Gordon University, Aberdeen).

Zellner, Peter (1999), *Hybrid space : new forms in digital architecture* (London: Thames & Hudson) 191 p.

Appendix 1: Consent Form for the Interview

University of Manchester
School of Environment and Development

The Challenges of Parametric Design in Architecture Today:

Mapping the Design Practice

CONSENT FORM

If you are happy to participate, please read the consent form and initial it:

Please
Initial Box

1. I confirm that I have read the attached information sheet on the above project and have had the opportunity to consider the information and ask questions and had these answered satisfactorily.

2. I understand that my participation in the study is voluntary and that I am free to withdraw at any time without giving a reason and without detriment to any treatment/service.

3. I understand that the interviews will be audio/video-recorded.

4. I agree to the use of anonymous quotes.

I agree to take part in the above project.

_____   _____   _____
Name of participant   Date   Signature

_____   _____   _____
Name of person taking consent   Date   Signature

_____   _____   _____

Appendix 2: Participant Information Sheet

University of Manchester
School of Environment and Development

Participant Information Sheet

1. This research investigates the challenges of parametric design under the title of 'The Challenges of Parametric Design in Architecture Today: Mapping the Design Practice'.

2. The research is conducted by Yasser Zarei who is an MPhil student at the University of Manchester. It is supervised by Dr. Albena Yaneva and Dr. Isabelle Doucet. The duration of the research is two years started from September 2010 and will be finished in September 2012.

3. The aim of the research is to explore the position of parametric design in contemporary architectural practices, identifying its advantages and disadvantages in comparison with traditional computer aided design.

4. You are chosen to this research due to your knowledge in the researched area.

5. You will be asked to answer some questions related to the research topic. Your responses will be used for the fulfilment of the aims and objectives of the research.

6. The collected data will be saved and encrypted on the university server and only the researcher and his supervisors will have access to them.

7. You have the right to withdraw from the participation two weeks after the arranged time for your participation.

8. As this is a student research, you are not paid for your participation in it.

9. The duration of your participation will be normally one hour. However, it can be changed based on your availability.

10. The research will be conducted in an office or a public agreed place.

11. The outcomes of the research will be published as a master thesis in future. It might also be used for presentation purposes.

12. If you have any concerns or further questions or would like to withdraw from the study after the participation has already taken place, please, do not hesitate to contact me. You can find my personal contact details below:

    Mr. Yasser Zarei, Yasser.zarei@postgrad.manchester.ac.uk

    School of Environment and Development, 1st floor Arthur Lewis Building, the University of Manchester, M13 9PL, UK

13. If you would like to make a formal complaint about the conduct of the research, you can contact the Head of the Research Office, Christie Building, University of Manchester, Oxford Road, Manchester, M13 9PL, UK.

Appendix 3: Interview Questions

The Challenges of Parametric Design in Architecture Today:

Mapping the Design Practice

Questions:

1. As a person who has been in architectural practice, what do you think about form-generation in architecture now?

2. How do you create architectural form in your practice?

    a. If you use computer programs, to what extent are you dependent on those programs? Can you design without their help?

3. Focusing on form-related issues, how would you describe today's architect? A form-finder or a form-maker?

4. In your opinion, is there a change in the use of software in architectural design now? Or in the way architects work with parametric programs in the design process?

    If you believe there is a change in the usage of software packages,

    a. Why architects are going toward this mood? How would you describe the roots of this shift?

    b. Can you call it a new style in architectural design or these are just techniques helping architects in the design process?

5. In regard to parametric design,

    a. What are its main advantages and disadvantages?

    b. For which type of projects do you use it?

    c. In your opinion, is there a difference between parametric programs and traditional CAD packages? If yes, could you highlight the main issues?

Appendix 4: Research Questionnaire

1. Which CAD system(s) is used in the company?

☐ AutoCAD	☐ Microstation	☐ Revit	☐ Digital Project
☐ ArchiCAD	☐ SketchUp	☐ Maya	☐ SolidWorks
☐ Allplan	☐ Vectorworks	☐ CATIA	☐ GenerativeComponents (GC)
☐ 3DS MAX	☐ Chief Architect	☐ Rhino	☐ SoftPlan
☐ Other - please specify: ---			

2. How familiar are you with parametric design?

☐ Never heard about it	☐ Know a little bit but have never used it by myself	
☐ Know and tried to use	☐ Use it occasionally	☐ Use it often

3. Which factors define your choice of the design type? (Whether parametric or not)

☐ My existing knowledge and skills
☐ Size and complexity of the project
☐ Availability of software
☐ Availability of training courses
☐ Client's requests
☐ Other (please specify) ---

4. Which type of design would you use for the following projects?

	Parametric design	Traditional CAD	Not applicable
1. Residential – private	☐	☐	☐
2. Residential – local authority	☐	☐	☐
3. Residential – commercial	☐	☐	☐
4. Commercial - office	☐	☐	☐
5. Commercial – industrial	☐	☐	☐
6. Educational	☐	☐	☐
7. Hospital	☐	☐	☐
8. Cultural and Recreational	☐	☐	☐
9. Competition	☐	☐	☐

5. Do you agree that

	Strongly disagree	Tend to disagree	Neither agree nor disagree	Tend to agree	Strongly agree
1. Parametric design is financially attractive?	☐	☐	☐	☐	☐
2. Parametric design is strong and comfortable due to the data management?	☐	☐	☐	☐	☐
3. Parametric design requires long time to learn how to use software?	☐	☐	☐	☐	☐
4. Parametric design is difficult to use?	☐	☐	☐	☐	☐
5. Parametric design changes the way you think about design?	☐	☐	☐	☐	☐

6. In your opinion, which type of design

	Traditional CAD	Somewhat traditional CAD	Somewhat parametric design	Parametric design
1. Brings certain competitive advantages?	☐	☐	☐	☐
2. Offers a more rigorous approach?	☐	☐	☐	☐
3. Offers more alternatives?	☐	☐	☐	☐
4. Facilitates unexpected solutions?	☐	☐	☐	☐
5. Offers a higher level of architectural control in production?	☐	☐	☐	☐
6. Requires less time of exploring design problems?	☐	☐	☐	☐
7. Enhances workflows in the design process?	☐	☐	☐	☐

7. In your opinion, what are the main disadvantages of parametric design? (Please write)

Your name: **Company:**   **Date:**