Manuel Shvartzberg

and

Matthew Poole

Edited by

in Architecture

Digital Technologies

Parametricism

The Politics of...
Contents

Acknowledgements

List of contributors

List of illustrations
**Parameter Value**

Chapter 13

...
A first wave of digital integration began with the translation of some techniques from hand methods to their computerized counterparts. Slide rules give way to calculators, drafting tables to copies of AutoCAD, and steel manuals to early structural computation programs. While the results of these processes were the same—even more drawings produced with digital accuracy—two important representational shifts that presaged so-called parametricism emerge. First, even though methodological concepts like "paper space" and "layers" enabled efficient drawing production, the underlying representation of the building in the form of electronic lines, arcs, and circles was made at full scale and full precision, creating the earliest version of a geometric "model" (in some sense) of the building. That model was created by direct manipulation of mathematical vectors, and the resulting electronic drawing was then scaled and projected by plot onto paper. Second, and more importantly, the digital vectors that comprised the scale drawings of the building also created an explicit mathematical and virtual definition of the building, an analyzable model of sorts that could be manipulated by both mouse and equation.

The manipulation of the lines, arcs, and circle of CAD evolved, over time, to the concept of blocks: integrated, connected bunches of geometry that could be repeated, reflected, or scaled and that made the management of large swaths of digital geometric data more palatable and (allegedly) drafting more efficient. As computational power and storage capacity of desktop computers increased via Moore's Law, blocks could give way to more robust chunks of digital geometry called objects that carried with them certain parametric and other data allowing them to be somewhat self-aware of their role in a drawing and therefore contextualized by those parameters to fit a certain set of circumstances. For example, the graphic collection of information representing a "door" would include a scalable plan drawing of the door and its swing, insert itself into the similar representation of its wall, and report its characteristics to the chart that was the door schedule; it could be counted and coordinated more rapidly into the overall drawing set. But because much of the information in that drawing set—things like floor patterns, details, fixtures and furniture, or ceiling systems—was still entirely graphic, the parametric intelligence of object-oriented CAD was limited at best and largely deployed in the service of better technical drafting. It was easy to copy, block, and scale these smart drafting elements. Ironically, the age of CAD brought with it an explosion of additional drawing in architecture, particularly in production drawings, and why not? The incremental cost of digital drafting space was essentially zero, and output mechanized via plotters.

Eventually twc-dimensional drafting platforms like AutoCAD would evolve to allow three-dimensional representation, but still aimed at the ends of drafting.

In the parallel universes of manufacturing and animation, however, three-dimensional geometric modeling had accelerated into the mainstream, and with it tools like Rhino, Pro Engineer, and Maya, each of which brought with them the ability to create geometry and access it by both direct manipulation and by formula; the equation that generated the geometry was bi-directionally accessible to a clever programmer who could "extract" the length of a line or the radius of an arc, manipulate it by script, and regenerate the resulting geometry. Parametric form generation, or the use of scripted mathematics that accessed the now completely available "analytical model" of the geometry, was born, and with it a generation of what Dean Robert Stern of Yale University has endearingly called "The Blobmeisters." Scripted manipulation of geometry parameters blossoms to this day when designers create algorithms that access the analytical geometric model (the now explicit math) and generate forms based on those routines. As Dean Stern suggested with his gentle rib, a great deal of that work was inspired by the non-rectilinear, sinuous forms and shapes that would result. And, since building component providers could access that same geometry to power computer-controlled fabrication, the blobs that appeared on screen soon appeared on the construction site. When it is easy to draw a curve, how much harder can it be to build it? Apparently, not much. Even in mainstream design and construction, AutoCAD made drafting rounded shapes so easy that it influenced the expressive language of institutional architecture of the 1990s; drawing those curves and cutting the resulting extruded shapes were much less of a challenge.

Object-oriented CAD and parametric geometry engines presaged the advent of a disruptive technology in the building industry, namely building information modeling (BIM). BIM took its cues from the technology of manufacturing design that had for years created detailed "solid models" of airplanes and cars that including geometry, materiality, behavior and parametric characteristics. The polarity of data in such models was reversed, in the sense that drawings were derived from the solid models themselves rather than being authored directly by the software.

Architects, engineers, and builders had inherited much of their digital toolbox from other disciplines, and manufacturing modelers in particular were in the main unable to handle the large scale, complexity, and degrees of abstraction necessary to represent a building, and particularly the arcane language of architectural drawing. Thus, various vendors created a generation of purpose-built tools known as "building information modelers." BIM platforms, which began to mature in the mid-2000s, provided the first real full "simulative" representations of buildings based on data structures and an underlying digital epistemology.
of architecture. Drawings, renderings, schedules, and projections could be generated as reports from a database that included geometry, annotations, material and behavioral data, and parameters to control various aspects of the represented element. As important, the elements exist in parametric relationship to one another (allowing systems of elements to be manipulated as relational groups). The data can be accessed for such manipulation by increasingly mature software "ports" called application programming interfaces. With the advent of BIM, the building industry obtained its first tools for building behavioral simulation, complete with parameters and access, representing both the intent and the result of design acts and the ability to manipulate those elements by algorithmic routine in ways more robust and provocative than merely bending flat planes into curves.

In broad strokes, it is possible to see the evolution of digital technology in building representation in three successive frames of reference. In the first "era," manual drafting techniques were supplanted by electronic drawings, but drawings were still the primary means of creating the representation of the design itself, and documentation remained the main focus of the tools. It has been argued, in the first installment of a recent exhibition tracing the history of computation in architecture, that parametricism as a design strategy was "invented" by Peter Eisenman in concert with Chris Yessios in their work on the competition entry for the Bicentrum at Goethe University. In that project, primitive scripted computation was combined with computer graphics as scheme generator, creating what curator Greg Lynn characterized as the first parametric project at the dawn of computerized drafting.

A second disruption came with BIM, where drawings were no longer the object of the representational instruments, but rather an artifact extracted when necessary from the parametric model of the building. That parametric model becomes a prototype for the finished building and can be visualized and measured in detail. The emergence of algorithmic techniques, seen combined with both BIM and the computational power of the cloud, brings us to the beginning of a third era of digital contextualization, where various systems of relationships, physical components, large data structures, and other digitized information connect and combine to virtualize all of the systems of building. The implications of the era of context will depend as much on the power and efficacy of digital tools as the designer's will to change the nature of the design process itself. Operating in the context of a wide array of parameters made accessible by integrated digital representation, architects can refactor, reprioritize, and restructure both the fundamental heuristics and the results of the design process itself.

In his 1970s treatise The Architecture Machine, architecturally trained computer scientist Nicolas Negroponte posited that digital technologies first mimic the processes that they are designed to replace, then extend them, and eventually disrupt them completely. This trajectory, understood by Negroponte while he studied how architects might someday use computers, is legible in the transition from CAD to BIM, although the current use of BIM is hardly disruptive, yet. While adoption of BIM is widespread in western building economies it continues to be used in the service of traditional project methodologies, including drawing-based design delineation and typical delivery methods and its full capabilities in the parametric are only now beginning to emerge as a potentially disruptive influence, enabling new capabilities in project collaboration, simulation, digital fabrication, and facilities management.

But BIM as both tool and methodology is far from the "standard" that CAD has become. The technology and accompanying design process changes are still evolving rapidly in parallel with changes in practice protocols necessary to fully deploy its benefits. In the near term, use of BIM is easier as computers have become faster, storage cheaper and more broadly accessible, and interconnectivity between machines deeper. Social networking methods are rapidly replacing traditional "store and manage" web sites as the infrastructure of collaboration. The last generation of CAD drafters is fading from the scene to be replaced by their more model-savvy successors, the architecture and engineering graduates of the last decade. As challenging as members of both generations have found the transition to BIM to be, the real disruption predicted by Negroponte is the advent of parametrics in combination with the accelerating capabilities of BIM empowered by cloud computation.

Today's generation of designers take the availability and utility of modeling for granted. The widespread use of scripted geometry modelers—primarily Rhino driven by Grasshopper—has spawned a set of young designers with different skills and sensibilities from their analog (or even CAD-enabled) predecessors. Design representation today is strongly rooted in digital models and scripts, and the generation of form/idea comes in equal parts from direct manipulation of geometry as the output of algorithms. A script represents both a strategy for a solution and the imbedded logic of its intent. When that set of sensibilities and methods is applied to the representational potency of cloud-enabled BIM, Negroponte's anticipated disruption is inevitable. The "things that architects make" will be different, the ways we create them new, and the reasons that we make them will change. The results must inevitably be different as well.
For these purposes, we will posit the rapid and eventually complete evolution of two technologic components: BIM as a platform for ideation, representation, and simulation; and parametrics as a method for accessing, interrogating, and manipulating the characteristics of that BIM-based representation. The modeler instantiates the representation of the building in digital form, and access to its characteristics (and their subsequent manipulation by scripts) provides the parametric opportunity. We will further stipulate that the computational infrastructure that makes both possible is shifting from desktop computers to cloud computing, unbinding parametricism from the limits of computational power (CPU resources) or data storage and access. You can create and save as much information as you want, move it virtually anywhere, access any version on multiple devices from studio to job site, and apply essentially infinite processing cycles to that information.

Irrespective of the ends to which these powerful capabilities may be applied (which is the subject of much of the balance of this book), it is clear that much broader portions of the design enterprise will be robustly represented, simulated with accuracy, and analyzed extensively. Drawings and charts are replaced by digital models and analysis software. Models are more detailed, and evaluation of those models by algorithms will run in the cloud simultaneously with design manipulation. Decision-making based on intuition, rules of thumb, or outsourced engineering talent will be replaced with immediate feedback on the technical implications of a given design strategy. Where a design was once represented almost exclusively through the proxy of geometry projected on a paper drawing, higher resolution representation will include not just more detail, but performance, materiality, even imbedded construction logic. Imbuing representation with parametric formulation extends that representation with both intent and result.

This is a critically important if largely underestimated change in design methodology that is bubbling up from the workstations of today's digitally empowered designers, with implications for the nature of the design process and the role of architects themselves. The traditional patriarchal relationship between architect and engineer (and, to a lesser extent, architect and builder) is potentially toppled by the emergence of computational analysis because work that was once demanded of engineers during the development of a design ("make this stand up" or "keep this heated or cooled" or "operate this within the following energy load") can be done, at least in part, by analytical software running in parallel with modeling software. Provisional answers to questions about what may or may not work are immediately available to the architect running a model and, for example, energy analysis or code simulation tools simultaneously. More rapid generation of alternatives, culling of options that fail to meet certain performance criteria, and a general assignment of technical evaluation to computation should result in better design, at least in theory.

Computers have long been used for analytical purposes in architecture, through purpose-built software assigned to tasks like energy analysis or structural evaluation, or more generic strategies empowered by the ubiquity of spreadsheets. Parameters, by definition, bound the solution space of analytical results. But the shift described here is in the relationship of these tools to the designs themselves when directly connected via those parameters to the underlying digital design model, and the potential to use optimization strategies like scripting to manipulate those models based on the results of analysis. A parametric strategy could be used in such a way to precisely locate a building on its site relative to its openings, glazing materials, and movement of the sun. Automating the process to answer that question without direct human intervention frees that human to spend her time solving more ineffable questions of composition—or drinking coffee.

Techniques already in use in manufacturing such as multivariable design optimization use computational means to "pre-select" ranges of design options based on the parameters of performance and outcome defined by the designer at the outset of the design exercise. These methods create the "exploration space" within which a range of solutions can be found and selected, largely because the analytical models that underlie the design are explicit, accessible, and available for iterative evaluation. When computation is infinitely available and largely free, analysis is no longer a burden to be outsourced to engineers but a design resource in and of itself.

When design heuristics and analysis become closely bound together, the fundamental methodologies of design are radically altered. The definition of the problem space—the range of possible viable solutions—can be generated from underlying intent that has been defined by the selection and manipulation of key project parameters. The generation of possible solutions is unconstrained by computation cycles, computerized or human, and thus broad iteration of answers is possible. Those answers can be evaluated and measured by simulation algorithms—computer programs that operate alongside those that generate answers—for informed selection. The mathematics of multiple analytical models that are the basis of ideation and iteration is explicit, accessible, and usable in support of finding reliable and predictable results. Analytical models now join their physical and digital counterparts in the architect's toolkit. And, "parametricism" per se is redefined to include performative optimization toward results.
impact on the environment is large, is an opportunity for architecture itself to expand its breadth, reach, and importance.

As sensor technology and reality-capture techniques like laser scanning become more embedded in the building operations process, designers can further access huge amounts of data about how other buildings are performing after design. Comparing those results, and evaluating design options based on that evidence, creates another “big data” stream of input into the ideation process, a basis of reference for precedent and outcomes, and assurance that a promise of a result is backed up with real-world experience, all potentially supported by parametric analysis.

The evolution of design computation suggests that the technologic capability to work in this manner will soon be possible, constrained only by the willingness of architects to embrace a new paradigm of obligation and service. But the track record of the profession is not promising in this regard, in that architects have spent most of the twenty-first century disabusing themselves of responsibility for many aspects of buildings, and particularly the operating results of their efforts. The oft-heard cry of “loss of influence and control” has been matched by a disinterest in taking responsibility for the measurable outcomes of buildings, particularly building performance, construction cost, and schedule. A class of professionals willing to do so has emerged accordingly.

Other important parts of the modern economy are embracing outcome-based methods in relentless efforts of self-improvement: data mining for drug outcomes, large-scale climate modeling and analysis for crop yields, digital simulation as the basis for automobile design, manufacturing and marketing, and virtualization of film and gaming. The medical industry is experimenting with compensation models tied to patient outcomes rather than number of procedures performed. Those trends set expectations for the building industry and circumscribe its potential to reform its objectives, processes, and result. Architects can and must position themselves in the design and construction process anew.

Modern construction economic systems—that define the ways architects are involved and paid, and the means of procuring and executing buildings—derive from a basic exchange of financial value between the participants and the underlying assumptions about the related value of each player’s contribution. These systems today are based largely on simplistic, reductive, and mostly ineffective strategies designed to limit short-term exposure to cost and risk, and
are rarely deeply connected in any way to the underlying intent of the building projects they deliver. In fact, the most common performance metric in most delivery models is "lowest first cost"—choices are almost always made based on the lowest proposed cost of any component in the delivery system, be it architectural fees, contractor bids, or material prices. The contributions of the players are thereby undervalued by definition even when projects begin with the highest social aspirations. While the reasons for this situation are complex, there is increasing dissatisfaction with results among building clients who regularly experience cost overruns, schedule misses, and technical failures. As recent innovations in project delivery suggest, they are looking for another way.

The introduction of BIM to the mainstream building market spurred what may now be seen as the first move away from commoditized building economics toward outcome-based models. Rapidly evolving systems known generally as "integrated project delivery" create teams of co-dependent designers, builders, and owners who share the risk and reward of building projects that are measured by mutually agreed outcomes that benefit the project rather than individual contributors to that project. The informational transparency of BIM created the opportunity from which integrated projects now sprout, but the ability of a project team to understand outcomes based on a deeper understanding of a digitally simulated parametric project could be the most important shift. Broader more powerful parametric simulation means deeper more important outcomes can be predicted. The underlying means of production will shift accordingly and the value created and delivered by architects dramatically improved. And as analysis algorithms and approaches move from the more proximate technical problems of building performance to broader sociological or behavioral simulation, the architect's ability to project outcomes, promise results, and create real value improve significantly.

Further, the fundamental legal standard by which architectural competence is measured, the standard of professional care, by its very definition inhibits innovation and deep process change. Because it defines proper professional judgment in terms of what another competent practitioner might have done in the same circumstance, pushing methods and design techniques forward in the manners that are described here is not just difficult, it is professionally dangerous. And, a generally accepted principle of risk assumption that is vigorously enforced by providers of architect's professional liability insurance is the avoidance of guarantees or warranties; provision of professional services means never promising actual results. So the standards of risk, and with them the architect's willingness to embrace it, must be disrupted; speculations of similar competence replaced with measurable results based on commitments made early by the designer. Those commitments could be based on the results of computational, parametric simulation of the building considered and eliminate what is possibly the single greatest barrier between architects and their potential to create measurable results.

Larry Ellison's success with his catamaran was, in some ways, predictable. He was confident that the right combination of technological prowess—design simulation that leveraged operational parametric telemetry—would change the odds of success in his favor. With formidable technology (and a lot of cash) he disrupted the age-old sport of sailing. That success was measured by the speed and agility of a sailboat on the open water against a formidable opponent and made possible by the control of the physical by the digital. As architects we can set the definition of success in design and building as we see fit, if we are willing to deploy the available tools and provocative methodologies that result. It is an opportunity not to be missed.

Notes

2. As further described by Carpo in The Alphabet and the Algorithm.
3. Moore's Law, invented by Intel's Andrew Moore early in the age of personal computers, suggested that the overall processing power of computers doubles every two years.
7. Wikipedia defines multidisciplinary optimization as follows: "Multi-disciplinary design optimization (MDO) is a field of engineering that uses optimization methods to solve design problems incorporating a number of disciplines. It is also known as multi-disciplinary optimization and multidisciplinary system design optimization (MSDO). MDO allows designers to incorporate all relevant disciplines simultaneously. The optimum of the simultaneous problem is superior to the design found by optimizing each discipline sequentially, since it can exploit the interactions between the disciplines. However, including all disciplines simultaneously significantly increases the complexity of the problem." See http://en.wikipedia.org/wiki/Multidisciplinary_design_optimization [accessed July 21, 2014].


11. Professional competence (or its lack, negligence) is defined by positing "what would a competent professional in similar circumstances have done?" This standard is largely a matter of judgment, and the definition of "competence" and "similar circumstances" broad and undefined.

Chapter 14

Spinoza's geometric and ecological ratios

Peg Rawes

This chapter questions the relationship between the latest evolution of digital architecture, parametricism, and the claim that these computationally driven design methods also establish new environmental conceptualizations for architecture. Continuously increasing rates in climate change accentuate the need for the profession to responsibly develop sustainable design approaches for tackling the attendant social and environmental issues of resource-depletion and pollution. Within this context, the digital design community has developed a vocal set of agendas which, it argues, align its manipulation of computational software with wider notions of organic and biological life and, by implication, "environmental" awareness. However, substantial climate change evidence also shows that economic and social action is required, rather than another design "style" driven by modernist principles of autonomy. I therefore ask if these computational software geometries can be considered ecological in any viable way; and, if there is to be any conversation about process-based, materialist architectures, just who benefits from these cultures? Is parametricism of any real value for aiding the design of non-pathological, ecologically responsible buildings, which require ratios of ethical, economic and material intelligence, rather than myths of emergence?

I question the seductive rhetoric of parametric discourses, asking if their digital geometric techniques enable diversity on a number of fronts: first, can they

*An earlier shorter version of this chapter was given as a keynote address at "Interstices Under Construction Symposium: Technics, Memory and the Architecture of History," at the University of Tasmania, Launceston, Tasmania, in November 2011, and published as "Spinoza's Geometric Ecologies" in Interstices 13: Journal of Architecture and the Related Arts (Auckland: Enigma; He Aupiki, 2012), pp. 60-69. We would like to acknowledge and thank the publisher Enigma; He Aupiki, Auckland, New Zealand, for their generous permission in allowing us to publish a significantly revised and updated version of Peg Rawes' essay.