Structure, Architecture, and Computation: Past and Future

[One must not] fear that the adoption of forms and volumes closely following natural laws must lead to a monotonous and unsupportable uniformity of products.

—Pier Luigi Nervi

RENAUD DANHAIVE
Massachusetts Institute of Technology

CAITLIN MUELLER
Massachusetts Institute of Technology

INTRODUCTION

In architectural design, the past decades have been marked by the evolution of the computer from a drawing instrument to a design tool. It is clear that computation has had, and will continue to have, a significant impact on the design process. Bill Addis (2007) identifies the post-1960's period as the Computer Age in structural design, and this classification is, if not absolute, relevant for architecture as well. Computers have not only enabled the construction of some of the world's most daring structures, but also have also given birth to new styles and visions. Styles were constructed based on the new modeling possibilities offered by software and the computer played a crucial role in the development of new architectural visions in the second part of the twentieth century (Picon, 2010). However, computation and, specifically, the development of numerical solvers for structural analysis, did not mean that engineers designed more efficient structures. On te contrary, the rise of finite element analysis enabled a *'make-it-work'* approach, exemplified first by the Sydney Opera House of the Utzon-Arup duo and many projects after. In terms of design thinking, this represented a huge shift compared to the philosophy of engineers and architects of the 1950's, or even of the Architectural Engineering Age in general as described by Addis (2007). Computation was a revolution that would deeply influence the architect-engineer collaboration. Today, the revolution has matured and both practitioners and academics now have the necessary perspective to grasp the influence of the computer on design professions. With the development of new computational paradigms for design, along with the contemporary economic, social and, environmental context, there is compelling evidence that the barriers existing between the two professions can now be broken down with the use of computation.

Through an abridged history of architecture, structure, and computation, this paper will briefly examine how structure and architecture interacted in the recent past and how a renewed synthesis of the two disciplines can occur in the near future. It is structured in two parts. First, we will progress from the 1950's until today to succinctly explain how computation has changed the collaboration between architects and engineers, while focusing on the generally overlooked idea that, until recently, computation has had a negative impact on the synthesis of architecture and technological disciplines. We will then use Picon's concept of social imagination (2001) to explain how conditions are in place for a renewed synergy of architecture and engineering. The second part will succinctly present research on new computational ideas and tools to explore architectural and structural design intent symbiotically.

STRUCTURAL RATIONALITY AND ARCHITECTURE

Modern architects and engineers had varying perspectives on the meaning of rationality in structures (see Figure 1). For some, rationality was concerned with the clear and legible layout of structural elements in plan and elevation or, more generally, order. This was especially the case of avant-garde modernists advocating for, or at least making use of, the Corbusian free plan and its variations. In terms of structural mechanics, however, the Miesian structural rationale is unfounded. Rational structural design, understood as efficiency in terms of material use, does not mean and, in fact, usually avoids repetition and orthogonality. For many, rationality in structures was to be found in the strict application of physical laws, an idea that was very popular in the post-war context.

1

Figure 1: Two different perspectives on structural rationality: Nervi's 'natural' structures (left) opposed to Mies Van der Rohe's structural order (right, *credits: Flickr user 96dpi*)

A well-regarded defender of the latter interpretation, Pier Luigi Nervi claimed the supremacy of the physical laws on structural morphogenesis, even if he acknowledged the importance of constructability (he was himself a contractor). He referred to structural design as the one and only acceptable generating principle for large-scale architectures such as stadia or transit stations, although he was kind enough to leave housing and urban planning out of his dogma. In his essay aptly titled '*Is Architecture Moving Toward Unchangeable Forms?*', Nervi tells us that:

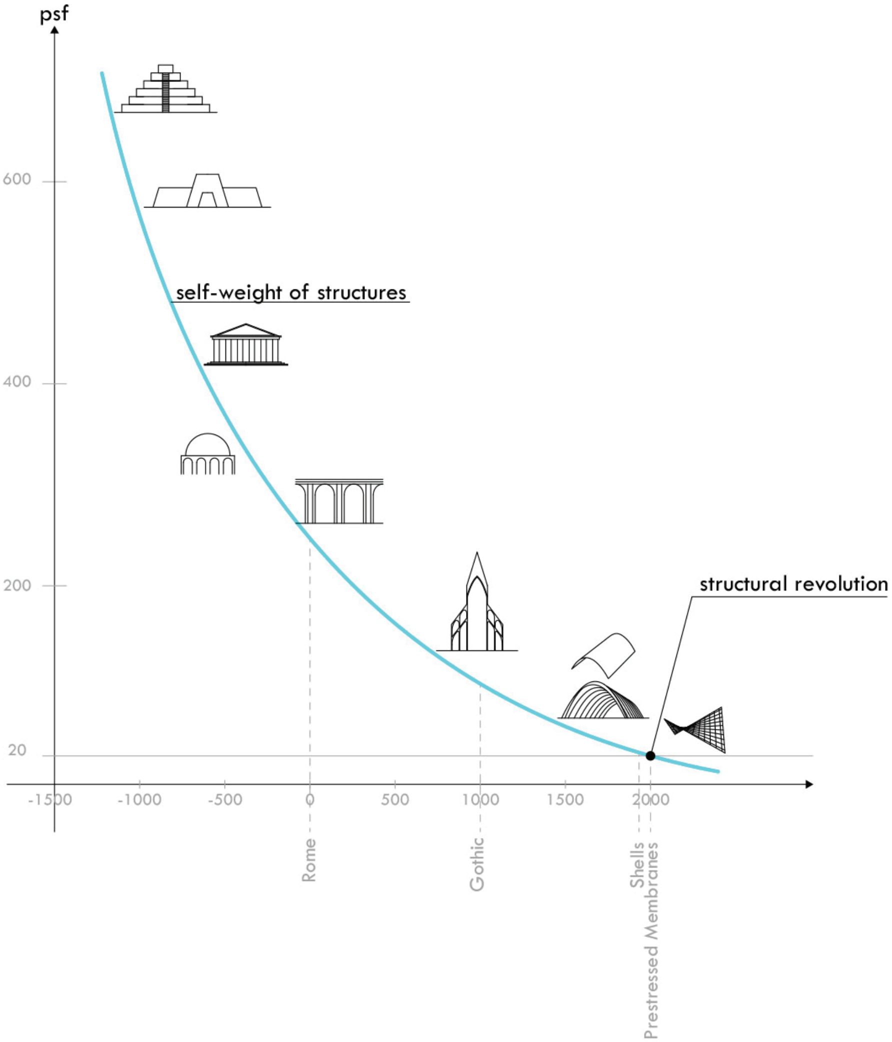
> "[...] Humanity is heading towards forms […] which, once reached, will forever remain unchanged and unchangeable in time."

These unchangeable forms, according to Nervi, were to be dictated by physical laws, a concept that would soon be vehemently rejected by post-modernists. However, Nervi also stated that :

> "[One must not] fear that the adoption of forms and volumes closely following natural laws must lead to a monotonous and unsupportable uniformity of products.",

an idea that undermines the rigidity and permanence of his previous claims. Given the diversity of creations by celebrated structural designers of the past—Eiffel, Maillart, Nervi, Torroja, Candela, Dieste, Esquillan, Otto, etc. —and the present—Schlaich, Conzett, Ney, Baker, Sasaki, etc. —who use and hijack nature laws to shape diverse, efficient and innovative structures, it is obvious that this assurance is valid.

At the dawn of post-modernism, a *'philosophy'* of structures had emerged organically from the doctrines of diverse international engineers and was hailed by many as the future of architecture. Particularly edifying is the graph drawn by Rene Sarger in 1967 in the *'Cahiers d'Etudes Architecturales'* (in English, Books of Architectural Studies). Sarger believed that a structural revolution had taken place through the discovery of shells and hanging structures. The future of human construction was going be the heritage of this revolution. A few years later, for most designers, these ideas belonged to the past. This coincided with the rise of computation in engineering practice.



2

Figure 2: The Structural Revolution as envisioned by Rene Sarger (*adopted and translated to English, based on (Sarger, 1967)*)

Indeed, structural rationality left post-modernists frustrated. In general, the structural rationale was another manifestation of the modernist objectivism, even if nuanced by Nervi's accent on the poetic qualities of structures, and thus became absent from the mainstream post-modernist discourse, only to find itself later super-evaluated in high-tech architecture. Still, post-modern architects had to cope with the unfortunate necessity of making their buildings *'stand up'*. However, the context was fundamentally different than before as computer simulations had pervaded the daily practice of engineers. With the newly available computing power, they were able to analyze any shape and, by any means, make it work. Thus, computation had exacerbated the architect-engineer dichotomy: the architect as a form-giver and the engineer as a form-verifier. The Sydney Opera House constitutes a milestone in that shift from a philosophy to a submission of structures, especially because it is the first major project that used the newly available computational power as an enabling technology.

THE REVOLUTION OF COMPUTATION IN STRUCTURAL ENGINEERING: THE CASE OF THE SYDNEY OPERA HOUSE

The Sydney Opera House, which entered conceptual design in 1955 and was completed in 1973 is the first major architecture project that made largely use of computation (Addis, 2007). The project would not have come out of the ground were it not for the structural design calculations performed using a computer, a fact that was acknowledged by Ove Arup himself. Speaking about the design of the Sydney Opera House, he said:

> "The interplay of surfaces made an assessment of structural feasibility by normal approximations difficult and of dubious value."

Arup also stated that:

> "Utzon was quite willing to change his shapes in order to reduce the moments, but any major deviation from the architect's proposal would not have been the design which won the competition… I therefore advised him to retain his basic idea and we would somehow make it work." (Messent, 1967)

The two citations above have profound implications in terms of design thinking and are particularly representative of the new paradigm for architecture and technology in the age of computation. Each of them hints at the developing core values of a new generation of structural engineers: a technological expertise based on the use of advanced simulations and a desire to make architects' formal desires stand up. Ove Arup's firm has now become a large international practice, and the principles of the Sydney Opera House design are still evident in recent projects. Architectural visions, like Koolhaas' CCTV building in Beijing (2008), are engineered to work, in spite of their structural soundness. Such projects are exemplary of the influence that computer simulations had on the practice of engineering during the final decades of the twentieth century. From this perspective, it is particularly interesting to examine how engineers from Arup speculated on the impact that computation would have on the architecture and engineering professions, during their *'Symposium On the Use of Computers'* in 1963. In an account of the symposium conversation given by Loukissas (2012), it is clear that symposium participants had conflicting views on the influence that the computer would have on their collaboration with architects. Some were worried that the computer was a threat to the architecture profession. One attendee stated that

> "The computer couldn't really take the act of designing away from the architect… the essential thing to remember was that a computer didn't really possess an imagination."

The Arup engineers did not, however, fear the influence that computation was about to have on their own practice, relegating their work to verification rather than design.

In his discussions on the Sydney Opera House, Bill Addis (2007) writes about the architectural intentions of the project:

> "It was essential to find a way of creating the illusion of a thin shell […] and the appearance of standing on one corner."

Two words, *illusion* and *appearance*, are essential in this remark. From the outset, structural honesty was disregarded; instead, what truly mattered was the illusion of an elegant structure. Indeed, what makes the Sydney Opera House so exemplary in the history of architecture and engineering is that its poetic forms are in fact generated by the structure, thus misleading the untrained eye into thinking of the building as the sensible juxtaposition of naturally efficient shells.

The misappropriation of structural principles by architects was relatively new, but would quickly gain traction. It appears that the structures of Nervi, Isler, and others influenced architects for their seductive forms rather than for their structural principles. Still today, the expressivity of certain structural types, shells in particular, exert a fascination on many architects, the Heydar Aliyev Center of Zaha Hadid Architects in Baku (2012) being one recent example. There is no doubt that Utzon was influenced by the thin shells of Candela and Isler, although he was not the first to misuse this building technology. One of the earliest examples of this misappropriation was the Kresge Auditorium (1954) by Eero Saarinen on the MIT campus. Not unlike Utzon, Saarinen wanted to exploit thin shells for their plastic qualities without understanding their structural logic. Specifically, Saarinen wanted to use a portion of a sphere resting on three corners for the roof instead of a true membrane-only surface, which led to unreasonably thick edge beams and uncontrollable deflections (Plunkett & Mueller, 2015). The Kresge auditorium is an inefficient structure with the attire of highly performant one, just like the Sydney Opera House, and indeed this link can be seen as more than coincidence; Saarinen was a member of the jury who chose Utzon's proposal as the winning entry for Sydney. In Kresge, the geometry was simple enough to be solved analytically, thus not requiring the use of computers—although the analytical results did not succeed in predicting the actual behavior of the shell—but the Sydney example proved that, thanks to the computer, perverting structural principles to generate forms opened a new realm of virtually limitless shape exploration.

CONTEMPORARY SOCIAL IMAGINATION: DIVERSITY AND EFFICIENCY

The philosophy of structures that was theorized until the early 1960's became largely absent from mainstream architectural discourse by the late twentieth century. As we have seen, it can largely be explained by the two following observations: (1) Post-modern architects rejected structural rationality as a founding principle of architecture. (2) The computer enabled and enforced a 'make-it-work' approach to structural engineering.

The reversal of the expensive material/cheap labor paradigm that occurred at the end of the twentieth century also supports this argument. Indeed, reducing material weight was not valuable economically compared to the imperative of easy constructability. Today, however, these three observations do not hold true anymore. Instead, the contemporary context is marked by their counterpoints:

First, the current generation of architects has assimilated the principles of both modernism and post-modernism. They understand that objective—performance, technology, function—and subjective—context, meaning, form—qualities can be sensibly integrated. Structural rationality is not incompatible with subjectivity in architectural design and not synonymous with uniformity.

Second, computation has matured into a media of increased collaboration between architects and engineers. New developments are still needed to empower designers to synthesize architectural and technological choices but the latest developments in modeling and analysis software are heading towards more integration.

Finally, while it is accepted that economy in construction is mostly related to ease of construction rather than material efficiency, growing environmental concerns are refocusing performance goals on reducing embodied energy through a minimization of material usage, bringing classical ideas of structural efficiency back to relevance.

These three points alone cannot fully characterize the contemporary social imagination, but give clues that allow one to speculate on the near future of architecture and technology. In

'Architecture, Technology and Imagination' (2001), Antoine Picon uses Bronislaw Backzko's definition of social imagination to describe this concept as:

> "[...] the system of general representations of social order that prevail in a given society. [It] is about the ethical values recognized by a society, and the way that society should evolve in order to conform to these values."

In the same essay, he argues that social imagination is conveyed in architecture through technological choices. In regard of the arguments presented above, the contemporary social imagination is characterized by several priorities: (1) plurality and diversity as the expression of a democratic and individualist society, (2) economic efficiency, and (3) environmental efficiency. If we accept Picon's argument that architecture and technology are linked by social imagination, these values, in combination with new enabling technologies of fabrication, constitute the conditions that are now in place for a renewed synthesis between architecture and structural design through computation, for the formulation of a richer, more explorative and experimental philosophy of structures.

NEW COMPUTATIONAL PHILOSOPHY OF STRUCTURES

New computational means can help explore design options in a way that integrates structural performance in the conceptual design phase without inhibiting the designer's creative freedom. Computation can help in creating platforms that relate early design intentions in a *'shared space of alternatives'*, in the words of Herbert Simon (1947), or, as we will call it more broadly, the design space. A participant of the 1963 Arup symposium already saw the potential of the computer for design exploration:

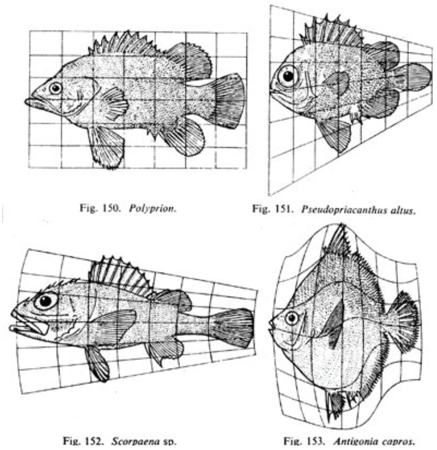
> "A computer ought to give the architect more choice, rather than simply produce an optimum solution for him."

New computational directions for the construction and exploration of design spaces are presented in the following sections.

INTERACTIVE EVOLUTIONARY OPTIMIZATION AND PARAMETRIC DESIGN

Two computational design strategies, parametric design and interactive evolutionary optimization, are particularly interesting and relevant to the goals of synthesizing architecture and engineering in creative ways. Indeed, these approaches are naturally oriented towards exploration, are theoretically applicable to any design problem, and are well-suited to make ill-defined criteria meet quantifiable objectives in architectural design.

Parametric design is a scheme of design in which an overall concept is parametrized according to a set of properties, or parameters, which can either vary or change. One of the earliest examples of parametric design can be found in D'Arcy Thompson's On Growth and Form (1917), where he shows how geometrical, and thus parametric, transformations can be used to generate different species of related animals. This early example shows the power of parametric approaches for design exploration.



3

Figure 3: Transformation of the Polyprion into other fish species (*D'Arcy Thompson, 1917*)

Parametric design systems are not new to architecture. Since the introduction of CATIA (Dassault Systèmes, 2015), they have gained more and more popularity, culminating in the present context in which nearly every architecture student has used Grasshopper (Robert McNeel & Associates, 2015). In general, parametric design has emerged as one of the most widely used computational methodologies for early-stage design. Platforms usually grow from the development of inter-related tools. Specifically, 3D modeling software, such as Rhinoceros (Robert McNeel & Associates, 2015), can be combined with visual programming interfaces, supplementing the modeling workspace to constitute parametric modeling environments. These allow the user to script complex generative algorithms without prior programming knowledge and can help in steering design space exploration. Exploring different solutions can be done in a timely manner as the parametric design process is by essence non-destructive, meaning that one model contains all the previously explored solutions as well as the ones yet to evaluate. Furthermore, these parametric modeling environments can be used in combination with analysis and optimization plug-ins to link geometry and performance. Such integrated environments constitute a compelling common ground for architects and engineers. However, parametric design space exploration usually remains limited to manual manipulation of parameters through sliders and initiation of computational search. At the other extreme is optimization, which automates the design process completely and yields a single best result. However, this is also dissatisfying as a design approach, and stokes fears about computation as a replacement for human designers. Such automated procedures that fail to take advantage of the designer's expertise (Scott, Lesh, & Klau, 2002) can never fully capture the complexity and relate to the culture of architectural design.

Another strategy, interactive evolutionary optimization, has recently gained popularity for optimization in structural design. While standard evolutionary algorithms seek to find the optimal solution using heuristic methods in a closed-loop workflow, interactive optimization incorporates the designer's input in the optimization process for the selection of parent solutions. This approach accounts for ill-defined objectives, such as aesthetics, which makes it very suitable for applications in architectural and structural design where design complexity goes beyond quantifiable metrics. This strategy may lead to sub-optimal solutions which are more valuable to the designer in terms of non-measurable criteria important to the architectural con.

4

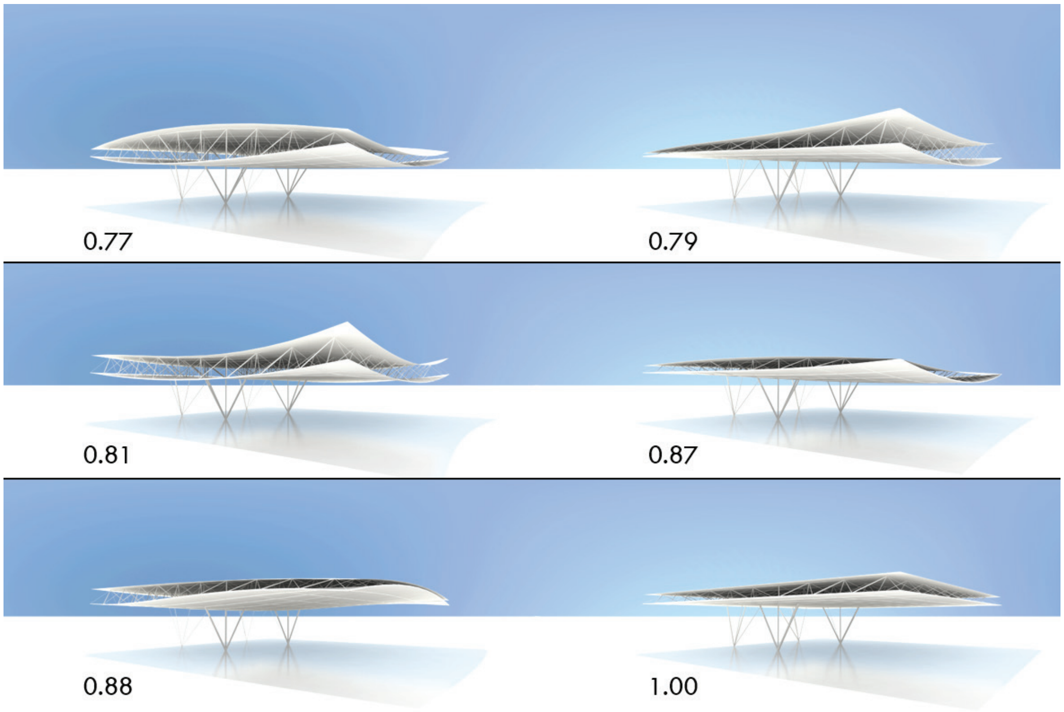


Figure 4: Variations of an initial structural concept with normalized required weight

Given the advantages of the strategies discussed above and their complementarity, there is an obvious potential for connecting interactive evolutionary optimization and parametric design. Previous works, such as ParaGen, a tool developed by von Buelow (2012), have illustrated the benefits of combining parametric modeling, evolutionary algorithms, and designer's input for design space exploration. More recently, these concepts were linked in a new tool called *Stormcloud*, developed by the authors. *Stormcloud* is a plug-in for the Rhino/Grasshopper environment that supplements the Grasshopper visual scripting interface with a window dedicated to design space exploration.

Figure 4 shows the diversity of designs generated using *Stormcloud*. Each design presents different aesthetic features (flatness, convexity, concavity, etc…) and is more efficient, in terms of structural weight, than the starting design. With such an environment, designers can make informed choices during the conceptual design phase while maintaining flexibility and control. The implemented design tool capitalizes on Grasshopper's versatility for geometry generation but supplements the visual programming interface with a flexible portal, increasing the designer's creative freedom through enhanced interactivity. The tool can accommodate a wide range of structural typologies and geometrical forms in an integrated environment.

The plug-in takes three different inputs—geometry, an evaluation method that produces a numerical score, and design variables—and has no output parameter. The score is normalized according the initial solution score. Populations of candidate solutions are generated by re-computing the Grasshopper script solution after setting the design variables to new values obtained after cross-over and mutation operations. Since it is blind to the nature of the problem, *Stormcloud* is not bound to any predefined parametric formulation or typology and can be used on a variety of design problems

The framework developed in this research significantly lowers the barriers for designers to adopt interactive evolutionary optimization as a performance-oriented methodology for design. Parametric design and interactive optimization are powerful approaches for the reestablishment of a symbiotic relationship between architecture and engineering through the performance-driven exploration of the shared space of alternatives during conceptual design.

GRAMMATICAL DESIGN SPACES FOR TRANS-TYPOLOGICAL EXPLORATION

A second major computational development involves moving beyond parametric frameworks to formulate systems of design alternatives, motivated for the need for even greater design diversity to improve decision-making in the earliest-stages of design. The first steps in the contemporary conceptual structural design process involve choosing a typology or system. For instance, in a long-span roof design, should the structural action be carried out with an arch, a cable, a fan-like scheme, a bending option, or with a truss? The world's best structural designers are able to brainstorm a range of creative ideas and can intuitively estimate relative performance of competing concepts. Currently, in the most successful examples, the generation of these typological ideas and the selection between them are carried out by expert practitioners with many years of experience and keen intuitions. In less successful approaches, fewer typological ideas are considered, or an ill-fitting typology is chosen without adequate consideration. There is room for bias and human error to influence this step in the process, which is arguably the most important step because it determines many characteristics of the overall form. There is therefore a strong and unaddressed need to develop computational methodologies for exploring possibilities across typological boundaries. While some masters in the structural design field excel at doing this by hand, the computer can help in several ways. First, given a broad enough design space definition, computational techniques can automatically generate a range of solutions to consider, behaving like a creative brainstorming partner. Second, computation can be used to quantitatively evaluate design options according to structural behavior. This is standard practice as a way to compare designs within a set typology, such as trusses of various configurations, but is rarely used to compare designs across typologies.

It is possible, through clever parametric formulation, to define somewhat broad design spaces that exhibit diversity in possible solutions and that are useful in exploring design decisions once the overall formal strategies and structural systems have been decided upon. However, it is practically impossible to define a parametric design space that covers the range of possibilities that one would like to consider during conceptual structural design. In the case of D'Arcy Thompson's parametric variations of fishes, parametric formulations allow us to explore a great variety of fishes but will never allow us to evolve from fishes to batrachians. This also holds true for parametric structural design, usually limiting designers to explore the variations of predefined typologies.

5

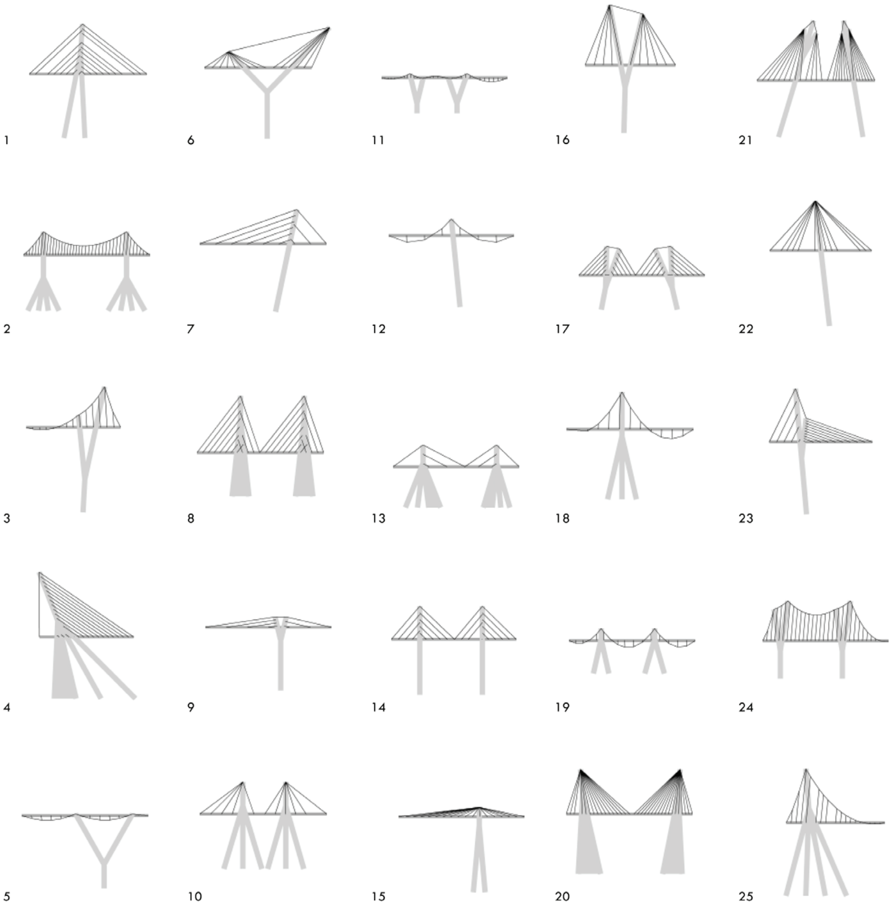


Figure 5: Example of pedestrian bridges generated randomly using a rule-based grammatical approach

A compelling way to move beyond the limitations of parametric variation is by using rule-based systems, or grammars, instead of parameter settings to generate designs. Based on Noam Chomsky's theories of generative grammars in language, George Stiny and James Gips (1972) proposed generative grammars for geometric shapes, or shape grammars. As Stiny (2006) later explained:

> "[Chomsky's] idea was that a grammar had a limited number of rules that could generate an unlimited number of different things, and that the resulting language was the set of things the rules produced".

Just as there are an unlimited number of new and creative sentences that can be uttered in

ENDNOTES

Addis, B. (2007). *Building: 3000 Years of Design Engineering and Construction*. New York: Phaidon Press Limited.

Danhaive, R., & Mueller, C. (2015). Combining parametric modeling and interactive optimization for high-performance and creative structural design. Proceedings of the International Association for Shell and Spatial Structures (IASS) Symposium 2015. Amsterdam.

Dassault Systèmes. (2015). CATIA. Retrieved from http://www.3ds.com/products-services/catia

Geyer, P. (2008). Multdisciplinary grammars supporting design optimization of building. *Research in Engineering Design*, 18(4), 197-216.

Kepes, G. (Ed.). (1965). *Structure in Art and Science*. New York: George Braziller.

Loukissas, Y. A. (2012). Co-Designers. Routledge.

Messent, D. (1967). Opera House Act One. David Messent Photography.

Mitchell, W. J. (1991). Functional Grammars: An Introduction. *ACADIA Reality and Virtual Reality*, (pp. 167-176).

Mueller, C., & Ochsendorf, J. (2013). An Integrated Computational Approach for Creative Conceptual Structural Design. Proceedings of the International Association for Shell and Spatial Structures (IASS) Symposium. Wroclaw.

Picon, A. (2001). Architecture, technology and imagination. In H. Louw (Ed.), *The place of technology in architectural history*. London: Society of Architectural Historians and Construction History Society.

Picon, A. (2010). Digital Culture in Architecture. Basel: Birkhauser.

Plunkett, W., & Mueller, C. (2015). Thin concrete shells at MIT: Kresge Auditorium and the 1954 conference. Proceedings of the 5th International Congress on Construction History. Chicago.

Robert McNeel & Associates. (2015). Rhinoceros. Retrieved from https://www.rhino3d.com/

Robert McNeel and Associates. (2015). Grasshopper 3D. Retrieved from http://www.grasshopper3d.com/

Sarger, R. (1967). Structures Nouvelles en Architecture: Volume 1 des Cahiers du CEA. Brussels: Centre d'Etudes Architecturales (CEA).

a language, a grammar for shapes can yield an infinite number of new and creative designs.

Because of the breadth and richness of design spaces defined by grammars and rules, they are a better candidate for enabling trans-typological explorations than parametric design spaces. Grammars in architectural and engineering domains that move beyond shapes were first suggested by Mitchell (1991), who proposed functional grammars with rules that incorporate engineering and fabrication knowledge. The trans-typology grammar approach presented here involves three types of computational classes: shapes, grammars, and analysis engines. A particular type of shape is operated upon by a particular grammar, and analyzed for structural performance by a particular analysis engine. With this rule-based grammatical approach, a designer can start generating diverse and unexpected designs.

To demonstrate the power of this approach to generate diverse and interesting designs, the approach was applied in a realistic and complex trans-typology structural grammar developed to generate designs for short- and medium-span pedestrian bridges. Using the grammar, a variety of designs can be generated, as shown through the examples in Figure 5, which were all generated randomly from a single design space formulation. These designs demonstrate the breadth of the grammatical design space, including both the cable-stayed bridge typology, the suspension bridge typology, and space in between the two. Such techniques offer the possibility of improving architecture-engineering synthesis by supporting design diversity and selection, without limiting choice or constraining creativity. Indeed, as designers shift from designing objects to designing collaborative computational systems of possibilities, the role of creativity grows, nwwwwow newly empowered to also achieve performance.

CONCLUSION

In multiple ways, the computer has influenced architectural and structural engineering since its first use by practitioners. Instead of further polarizing our professions, computation will now play a role in a renewed synthesis of technology and architecture in the near future. Parametric and grammatical design strategies demonstrate how computation can open new perspectives to explore appealing and diverse structural concepts early in the design process, shifting the domain of structural engineering from analytical verification to collaborative design.

Scott, S. D., Lesh, N., & Klau, G. W. (2002). Investigating Human-Computer Optimization. Proceedings of the SIGCHI Conference on Human Factors in Computing Systems , (pp. 155-162). New York, USA.

Simon, H. (1947). *Administrative Behavior: a Study of Decision-Making Processes in Administrative Organization*. Macmillan.

Stiny, G. (2006). Shape: *Talking about Seeing and Doing*. Cambridge: MIT Press.

Stiny, G., & Gips, J. (1972). Shape Grammars and The Generative Specification of Painting and Sculpture. Information Processing ‘71, (pp. 1460-1465).

D’Arcy Thompson, W. (1917). *On Growth and Form*. Cambridge: Cambridge University Press.

von Buelow, P. (2012, December 4). Paragen: Performative Exploration of Generative Systems. Journal of the International Association for Shell and Spatial Structures.