Computational Composition in Architecture

Marcos J. Novak
Graduate School of Architecture and Urban Planning
University of California, Los Angeles
Los Angeles, California 90024

Abstract

The impact of computers on architectural design, manifested through the creation of architectural designs that equal or surpass in quality and sophistication architecture generated using traditional means, has yet to be demonstrated. Computer-aided design is conspicuously absent from theoretical discussions of architectural design per se, and there exist no major built or published buildings that have been designed using the computer in a true design capacity. Most prominent architects continue to ignore computer-aided design. This paper argues that the issues of architectural theory and composition must be addressed directly by the computer-aided design field, and that until this occurs computer-aided architectural design will necessarily be peripheral to architecture because it does not address the central problems of architecture as an expressive medium. To this end, it proposes a shift to the paradigm of computational composition, and discusses recent work in this direction.

The paper is divided into four parts. Part I identifies fundamental theoretical problems, contrasts the application of computation to architecture and to music, and draws upon several different areas for insight into the nature of making; Part II reviews particular architectural implications of these considerations, introduces the concept of computational composition in architecture, and presents a brief overview of important precedents; Part III proposes new goals for computer-aided architectural design and presents a framework for computational composition; finally, Part IV presents recent work directly related to the ideas presented in the previous parts and leads to the Conclusion. The appendices contain a pseudo-Prolog expression of Alvar Aalto's architectural language and notes on features of the PADL-2 solid modeller that are architecturally interesting.

Part I: Theoretical considerations.

1.1: Two Modernities; Two Postmodernities

There is widespread agreement that we have come to the end of an era, the modern era, an era that goes back to the Renaissance, and that we are within a transitional period, the postmodern period4. The prefix "post-" indicates a time that lacks positive identifying characteristics, a time whose main features are responses to a more distinctly defined past. In architecture, the current theoretical and critical debate attempts to grasp the characteristics of this period in order to articulate the aspirations and goals of architectural production. What is the place of computer-aided design in this debate?
In order to answer this question we must examine modernity more carefully. In his book "The Five Faces of Modernity," Matei Calinescu describes that there are in fact two modernities, "bitterly conflicting." We quote at length:

With regard to the first, bourgeois idea of modernity, we may say that it has by and large continued the outstanding traditions of earlier periods in the history of the modern idea. The doctrine of progress, the confidence in the beneficial possibilities of science and technology, the concern with time (a measurable thing, a time that can be bought and sold and therefore has, like any other commodity, a calculable equivalent in money), the cult of reason, and the ideal of freedom defined within the framework of an abstract humanism, but also the orientation toward pragmatism and the cult of action and success - all have been associated in various degrees with the battle for the modern and were kept alive and promoted as key values in the triumphant civilization established by the middle class.

By contrast, the other modernity, the one that was to bring into being the avant-gardes, was from its romantic beginnings inclined toward radical antibourgeois attitudes. It was disgusted with the middle-class scale of values and expressed its disgust through the most diverse means, ranging from rebellion, anarchy, and apocalypticism to aristocratic self-exile. So, more than its positive aspirations (which often have very little in common), what defines modernity is its outright rejection of bourgeois modernity, its consuming negative passion.

It is clear that computer-aided design is rooted in the first modernity, while the current theoretical discussion in architecture has its foundations in cultural modernity, the second modernity.

As a consequence of the duality established above, the postmodern period is also split along two paths, with two branches of postmodernity corresponding to the two modernities. At present, these two branches are reenacting the byzantine conflict of iconotary and iconoclastm. The first, iconoclastic, postmodernity combines the icons of historical architecture and technological development at a surface level, and produces the now familiar image of "postmodern" architecture. The second, iconoclastic, postmodernity extends the attitudes of the Dada, surrealists, and the avant-garde, as "a tradition against itself," and constantly seeks to create work that questions any developing standard. While the iconoclastic postmodernity is often at once superficial and conformist, the critical stance of iconoclastic postmodernity, by always challenging settled views, extends the realm of the possible in architecture.

Already, from this macroscopic view, we can see that there is a discrepancy between architectural thought and the present state of computer-aided architectural design. Computer-aided design has tried to solve functional, pragmatic problems, that, though important, are secondary to the cultural pursuits of architecture - it has shown a positivist trust in the ability of reason, science and technology to solve all problems (corresponding to the the first modernity); it has, more recently, attempted to encompass some of the iconography of architecture at a surface level (corresponding to the iconoclastic postmodernity); but, it has yet to become a means for examining the limits of architecture (corresponding to the iconoclastic postmodernity); it does not fully recognize that if it

1 There is, of course, much that is built that has nothing to do with either of these categories. Architecture has always been distinguished from building. But we are not talking about computer-aided building design.
were to even approach what seemed to be a comprehensive definition of architectural design, it would find architectural theory advancing to some other uncharted space.

If we consider architecture more closely, from a detailed, microscopic view, we find similar problems. Of the Vitruvian triad of firmitas, utilitas, venustas, firmitas can be delegated to engineers and consultants, utilitas can change with a simple change of tenants or needs, and only venustas remains as the *sine qua non* of architecture, the tradition to be encompassed, if only to be overcome. Architecture is a medium, and as such it has its own expressive logic, and a realm of possible expressions. A sequence of rooms, or even a single room can become architecture if it attends to issues of light, proportion, order, detail, transformation, and so on. Presently, the specifically architectural knowledge that is used in creation of form, space and order and the manipulation of material and light, is not accommodated at a level high enough to expand the capabilities of the architect, and the fine distinctions that characterize design are all but abandoned. In other words, computer-aided design has yet to prove that it is aware of and can extend the tradition of architecture as an art.

We have, then, a twofold problem: first, within the discipline of architecture, the problem of representing and extending the tradition of making spaces and buildings as art; and second, within the overall culture, the problem of responding critically to the challenges of "a tradition against itself." Clearly, the solution is not to expect to grasp architecture within a set of programs, it is to use programs to explore the uncharted territories of architecture, to join and redefine the avant-garde. It is to incorporate all the knowledge of making architecture that we have, with the express purpose of going beyond that knowledge. It is to participate in the production of "an architecture that blazes".

Kandinsky dreamed of "a great city built according to all the rules of architecture and then suddenly shaken by a force that defies all calculation." This dream sums up the double aspiration of architecture and architectural theory today and the double failing of computer-aided design. On the one hand, the ability to design according to all the rules of architecture, the vast storehouse of accumulated compositional knowledge, on the other hand the ability to shake the rules, reconsider them, dislocate them, invent them, play.

This double failing, we feel, is not the result of technological but of ideological and theoretical differences. Certainly, a great deal of necessary foundation work has been done that must be praised, but, even so, it is evident that computer-aided design has yet to escape the influences of its birth as an aspect of computer-aided manufacturing, a birth in the lion's den of the first modernity.

We are not yet in the debate.

---

1 Coop HimmelBlau, motto.
I.2.: New Digressions, Diversions, Directions

On the way to making a positive proposal for the role of computer-aided architectural design in this fundamental conflict we will now gather reinforcements and questions, drawn from sources related only by their insight and concern for the activity of making.

1.2.1: Music: Amériques

In 1921 Edgar Varèse composed a piece of music entitled "Amériques" to "signify the affirmation of new worlds on the earth, in the sky, or in the minds of men."\(^9\)

In 1939, anticipating possibilities in electronic music that are only now being realized, he wrote:

"And here are the advantages I anticipate from such a machine: liberation from the arbitrary, paralyzing tempered system; the possibility of obtaining any number of cycles or, if still desired, subdivisions of the octave, consequently the formation of any desired scale; unsuspected range in low and high registers, new harmonic splendours obtainable from the sub-harmonic combinations now impossible, the possibility of obtaining any differentiation of timbre, of sound-combinations, new dynamics far beyond the present human-power orchestra, a sense of sound projection in space by means of the emission of sound in any part or in many parts of may be required by the score, cross rhythms unrelated to each other, treated simultaneously.... - all these in a given unit of measure or time which is humanly impossible to attain."\(^9\)

We see a musical/technological imagination, an imagination that foresees the musical promise of technology - in clear musical terms.

What is the equivalent architectural/technological imagination? What is the architectural promise of computer-aided design - in architectural terms?

Of course, Varèse is not remembered for making "tools for music": Galileo is not remembered for improving the refracting telescope.

Have we lost our passion for architecture?

1.2.2: Poetry: is 5

When e.e. cummings was asked to write an introduction to his book of poetry cryptically entitled "is 5" he responded as follows:

"On the assumption that my technique is either complicated or original or both, the publishers have politely requested me to write an introduction to this book. At least my
theory of technique, if I have one, is very far from original; nor is it complicated. I can express it in fifteen words, by quoting The Eternal Question And Immortal Answer of burlesk, viz. "Would you hit a woman with a child? - No, I'd hit her with a brick." Like the burlesk comedian, I am abnormally fond of that precision which creates movement.

If a poet is anybody, he is somebody to whom things made matter very little - somebody who is obsessed by Making. Like all obsessions, the Making obsession has disadvantages; for instance, my only interest in making money would be to make it. Fortunately, however, I should prefer to make almost anything else, including locomotives and roses. It is with roses and locomotives (not to mention acrobats, Spring, electricity, Coney Island the 4th of July the eyes of mice and Niagara Falls) that my "poems" are competing.

They are also competing with each other, with elephants, and with El Greco.

Ineluctable preoccupation with The Verb gives a poet one priceless advantage: whereas nonmakers must content themselves with the merely undeniable fact that two times two is four, he rejoices in a purely irresistible truth (to be found, in abbreviated costume, upon the title page of the present volume)." 4

What does architecture compete with? What is "The Verb" in architecture? The "precision which creates movement"? What is architecture's "priceless advantage"?

In architecture, when do we say that 2 plus 2 is 5?

Architecture is poetry.

Poetry is not Truth:
It is the resurrection of presences, history transfigured in the truth of undated time.14

Poetry is making.

1.2.3: Deconstruction/Redestruction

In 1977, when deconstruction was already eleven, Jacques Derrida wrote the following:

"every sign, linguistic or non-linguistic, spoken or written (in the current sense of this opposition), in a smaller or larger unit, can be cited, put between quotation marks; in so doing it can break with every given context, engendering an infinity of new contexts in a manner which is absolutely illimitable."5

A year later, in his book The Truth in Painting, while discussing the role in art of the "parergon", that which is beside the work, both part and not part of it, such as the frame around the painting, there is a solitary sentence fragment, lost between two paragraphs, as if in the crack between the frames of two paintings:
a theory which would run along as if on wheels

The mechanics by which architecture is rendered in computational terms involve the "citation" of architectural signs, the meaning of which is necessarily "illimitable". The structures of programs embody theories. "No permutation is innocent," Tschumi writes. Each juxtaposition of signs carries meaning, whether we like it or not. To accommodate the play of the architectural signifier and signified, this willful play, shouldn't these theories "run along as if on wheels"?

We are making theories about computer-aided architectural design. Is it not clear that the best architects will always try to confound these theories, as a matter of course? Simply because they are good architects?

1.2.4: Science: Noise, Chaos, Entropy, Dynamic Systems

Voss: "There are no simple mathematical models that produce 1/f noise other than the tautological assumption of a specific distribution of time constants. Little is also known about the physical origins of 1/f, but it is found in many physical systems: in almost all electronic components from simple carbon resistors to vacuum tubes and all semiconducting devices; in all time standards from the most accurate atomic clocks and quartz oscillators to the ancient hourglass; in ocean flows and the changes in yearly flood levels of the river Nile as recorded by the ancient Egyptians; in the small voltages measurable across nerve membranes due to sodium and potassium flow; and even in the flow of automobiles on an expressway. 1/f noise is also found in music."27

Ruelle: "I have not spoken of the aesthetic appeal of strange attractors. These systems of curves, these clouds of points suggest sometimes fireworks or galaxies, sometimes strange and disquieting vegetal proliferations. A realm lies there of forms to explore, and harmonies to discover."17

Voss: "With the exception on very modern composers like Stockhausen, Jojet, Carter (where the melody fluctuations approach white noise at low frequencies), all types of music share this 1/f noise base. Such a view of melody fluctuations emphasizes the common element of music and suggests an answer to a question that has long troubled philosophers. In the words of Plato, " For when there are no words (accompanying music), it is very difficult to recognize the meaning of the harmony and rhythm, or to see that any worthy object is imitated by them". Greek philosophers generally agreed on the imitative nature of the arts. It seemed obvious that painting, sculpture or drama imitated nature. But what does music imitate? The measurements suggest that music is imitating the characteristic way our world changes in time. Both music and 1/f noise are intermediate between randomness and predictability. Like fractal shapes there is something interesting on all (in this case, time) scales. Even the smallest phrase reflects the whole." 27 [my emphasis]

If it is the nature of the activity of making to challenge, to extend, to revise, on what ground can we build a relevant computer-aided architectural design theory?
Does not science also challenge, extend, revise? How is the avant-garde similar to science?

What is pure research in architecture?

1.2.5: Psychology: Creativity.

"Creativity is like murder- both depend on motive, means, and opportunity."\(^{10}\)

A 1988 review of the psychological findings on creativity shows an emerging recognition of major common elements in the behavior of creative persons. Sternberg's results are perhaps the most concise:

1. **Lack of conventionality** (e.g. one makes up the rules as one goes along; has a free spirit; is unorthodox)
2. **Integration and intellectuality** (e.g., makes connections and distinctions between ideas and things; has the ability to recognize similarities and differences; is able to put old information, theories, etc., together in a new way).
3. **Aesthetic taste and imagination** (e.g., has an appreciation of art, music, etc., can write, draw, compose music; has good taste).
4. **Decisional skill and flexibility** (e.g., follows gut feelings in making decisions after weighing pros and cons; has the ability to change directions and use another procedure).
5. **Perspicacity** (e.g., questions societal norms, truisms, assumptions; is willing to take a stand).
6. **Drive for accomplishment and recognition** (e.g., is motivated by goals; likes to be complimented for work; is energetic).\(^{22}\)

Has psychology finally discovered what the avant-garde has always known? Has it discovered the necessity of destruction to creation? In any case, now we have results, and the validation of science. How shall we use them?

1.3: Small Distinctions

Architecture communicates through the interplay of order and disorder, solid and void, material and absence of material, that which has form and that which is amorphous.

"Ordinateur": **ordering device**: this is the French word for what we call "computer".

Architecture is pattern. Architecture is structuring activity. It is musical composition. It is poetry.

Architecture is always "top-down". Even when we are designing architectural details, we are considering the abstract structure of the parts.
1.4: Gathering and Synthesis: Computers in Architecture and Music: The HandMaiden and the Avant-Garde

Why have we invoked so many voices? Are we converging on a synthesis? What do these texts have in common? Let us sum up:

Varèse, e.e. cummings, Derrida, too many others to mention, bring to our attention the interplay between a known but evolving structure and the necessity of directly manipulating that structure in order to extend the range of possible expression. For every medium there is a language, for every expression there are motives, codes and contexts which differ for the maker, the user, the critic.
There is no firm ground, and precisely in that inherent absence of a firm ground we find delight. Complex works of art, be they novels, symphonies, architecture, or paintings, are better, richer, if they are hard to decipher, more enriching if they require that we return to them again and again, trying to uncover their secrets with new keys, always failing, but always being rewarded for the effort.

The known but evolving structure mentioned above is the specialized knowledge of the particular field, seen in a broad cultural and technological context. In this respect, both in retrospect and in prospect, the parallel of architecture with music is illuminating, especially in regard to how the two arts have adopted computers as part of their evolution.

Nietzsche criticized Wagner's work as decadent, and Debussy described the same work as a "beautiful sunset which was mistaken for a dawn." The realization that Western music had somehow exhausted itself with Wagner urged Schoenberg to directly manipulate the structure of music of his time, prompting him to enter the unmapped waters of atonal music even though, as he recalled, "Personally, I had the feeling as if I had fallen into an ocean of boiling water ... it burned not only my skin, it burned also internally." The traditional goals of Western music were no longer seen as valid, and following Schoenberg, a series of innovations took place that not only challenged but also freed music. The advent first of electronic instruments and then of computers directly became an important means of carrying this project of redefinition forward. Clearly, increased technological capability has meant increased musical opportunity. The use of computers in music illustrates the convergence of philosophical, technological and musical concerns.

It is clear, then, that composers have long been aware of the musical implications of using computers in the creation of music, and have concentrated on the creation of tools that allow them to manipulate their particular medium in creative, expressive, and meaningful ways. Thus, computer-assisted musical composition is directly concerned with musical structure, sound synthesis, rhythm, harmony, melody, pattern, phrasing, and the investigation of new, heretofore unattainable musical possibilities. The important point here is that the problem for composers is not simply the representation of sound or the automation of notation, but the generation of music, hopefully great music, or, in architectural terms, not the representation of buildings, but the generation of architecture, hopefully great architecture, using the most difficult and demanding criteria. Major composers are working with computers, and several internationally renowned centers for research in computer-aided composition exist (e.g. IRCAM, in Paris, the Institut de Recherche et de Coordination Acoustique/Musique, founded in the mid-seventies and directed by Pierre Boulez). Their work aspires to be directly comparable in musical content, quality, and ambition to conventionally composed music of this and any other century. By contrast, computer-
aided design in architecture has dealt primarily with the craft of building, not the art of architecture, and thus has yet to help produce work that can validate it as a field of true architectural significance. In music, the use of computers is clearly the extension of an avant-garde project. In architecture it is a handmaiden.

Part II: Architectural Considerations

Let us now concentrate on architecture itself.

2.1: The Language of Architecture: Compositional Structure

First, consider a few simple cases. The design of a single room is a valid architectural problem, as so many great public and private rooms attest. The design of a single facade, an entrance, the response to a corner condition, a detail, these are all valid and interesting architectural problems, problems that, in terms of complexity, can currently be handled even by microcomputers. When one considers such problems, one sees that the emphasis of most current computer-aided design tools is on the representation of objects. We can model the architecture. We can make computations on this model, calculate the structure, the lighting, the HVAC... but, have we done anything to design the architecture? Does computer-aided design help us develop ideas about architecture? What distinguishes a well designed room from one that is poorly designed can only be found in the relations that the well designed room embodies, the proportions, the intangible compositional structure. At present, even tools that have sophisticated ways of creating elaborate object hierarchies stop short of helping determine the precise formal relations that distinguish the sublime from the ordinary.

2.2.1: The Normative Aspect of Design

Architecture is not accommodation or problem solving: it is the willful imposition of meaningful pattern (culturally deciphered) on formal elements (architectural knowledge). Another discrepancy is clear: computer-aided architectural design does not respond to the normative aspect of design. Architecturally, the problem is never to put up four walls and a roof, it is rather to put up four walls and a roof that possess certain formal characteristics derived from an architectural idea that transcends function. The architectural intention (the what should be accomplished) precedes the technical consideration (the how it can be accomplished). The four walls and the roof must not only be locally coherent, but must also fit into a global compositional scheme. The idea dictates the compositional structure, and in that sense generates the form, and the function is only accommodated in one of many possible ways. A properly chosen or invented compositional structure ensures that there will be a coherence between the whole and the part, and provides the main key to understanding the architect's poetic intent. A change in the architectural idea or the compositional structure must have repercussions both globally and locally.

It follows therefore that computer-aided architectural design must deal with specific design knowledge related to the languages of architecture, must allow the direct manipulation of the compositional structure of a work of architecture, and must accommodate the normative aspect of design. In each case the emphasis must be on the relations between parts, not on the modeling or
rendering of particular objects. To take advantage of the computer's particular abilities, tools that allow the architect to specify and apply generative processes to systems of relations must be developed.

2.2.2: A New Architecture

In the information age a new way of conceiving architecture has become possible. There are three underlying causes:

First, the replacement of the finite object by the variable object, the object in flux, has been made possible by the ability to define architectural models parametrically, and to modify them dynamically through computation.

Second, the idea of singularity has been replaced by that of pluralism (or rather by the acceptance of a spectrum of allowable choices, from the singular to the plural); not only do we not have a dominant view of what architecture should be, we find the very idea of dominance oppressive, even within a single building.

Third, diverse aspects of knowledge have been brought into contact through the common representation of equally accessible information, and allowed to interact and modify each other, causing a blurring of the distinctions of time and place. The shift from knowledge as hierarchy and classification to knowledge as information and knowledge as equally accessible fragments collapses the traditional sense of historic, geographic, theoretic, or conceptual regions, as all sources become equally within reach. Bits of previous knowledge become information and are reassembled into new wholes.

These three ideas make it possible to create a liquid architecture, an architecture of relations, an architecture in which the final built object is wrested from an infinite continuum of possible variations by the assignment of particular values to variables that govern its internal structures. The creation of systems of relationships and the assignment of specific values becomes the foreground of the architect's activity and invention. The new infinitely variable object negates the singularity of the old finite object and asserts the freedom of an open system where change is celebrated. The creation of a building does not follow from the placement of individual elements but from the ordering of separate, independent subsystems, Varèse's "cross rhythms unrelated to each other, treated simultaneously...". The classical idea of repose is replaced by the possibility of meaningful counterpoint, consonant or dissonant, able to respond to the ad hoc nature of particular architectural problems. New emphasis is placed on interval and proportion, and therefore to harmonic and rhythmical relations. The surprise of the unexpected juxtaposition, gentle or jarring, infuses new energy to the act of designing as well as experiencing architecture.

Collage and montage, often described as the single most revolutionary formal innovation in artistic representation to occur in our century 25 pale before the prospect of computation as a formal device. Collage and montage involve the simple operation of superposition - a condition which Peter Eisenman has properly described as merely pre-compositional 26 Superposition, in essence, leaves the elements that are overlapped untouched. The formal innovation of computation allows,
for the first time, distinct elements to operate upon each other in ways that are potentially predetermined, interactive, or random.

The traditional process of creating architecture thus becomes dislocated, and a new condition of design, perhaps called meta-architecture, is in view. The postmodern is thus seen as a final flourish at the end of an era of things, and the substantive difference between our time and the past is clearly seen in the transition to a new era of manipulating knowledge using high level abstractions and nested systems of relations, rather than particular objects. The architect of the past is seen as the virtuoso performer. The future architect is the composer of symphonies in form, space, and color.

This transition places architecture in harmony with the developing era of the global village. The repose of comfortable conformity is replaced by the lively, syncopated, rhythmical harmony of pluralism.

2.3: Precedents for a computational theory of design

The development of computation as a formal device in architectural design requires a recasting of architectural knowledge in terms that can be translated into computational concepts. Precedents for a systematic approach to computational composition exist in the ideas of typology, transformation, generative systems, and architectural "languages". The following section outlines some areas that warrant further consideration.

2.3.1: Typology

Historically, typology in architecture begins with the works of Quatremère de Quincy and of Nicolas Durand. In Quincy's conception of type and model, the first vague but permanent, the second precise and particular, the logic of form is related to reason and use. Durand's combinatorial approach is more rigorous and explicitly related to convenience and economy, but as Oechslin has written even Durand shows some willingness to compromise for the sake of making a "reasonable" architecture. Moneo defines typology as the combination of formal structure with history, suggesting that types cannot be understood without reference to the circumstances under which they arose. However, Aldo Rossi, O.M. Ungers and others have recognized that types can be useful even if they are separated from their historical roots. Rossi, for example, has considered type as "functionally indifferent":

one is struck by the multiplicity of functions that a building of this type can contain over time and how these functions are entirely independent of the form. At the same time, it is precisely the form that impresses us; we live in it and experience it, and it in turn structures the city.

Other concepts of type exist. Moneo observes that for Venturi image and type are one; O.M. Ungers refers to the collective memory; Silvetti expands on Rossi by stating that not only does
typology appear to be indifferent to function but architecture seems to be stylistically indifferent to typology. This suggests that the computational manipulation of types can indeed be carried out separately from stylistic and functional considerations, the two being combined at some later stage in the generative process. Argan has insisted that "what are required are fundamental, systematic, analogic, rational and combinatorial kinds of processes in the context of the encounter with history."

1 These varying conceptions of type suggest different ways in which type can be approached computationally, and kinds of transformations that need to be provided.

2.3.2: Transformation

The architecture of Terragni and Eisenman, because of the importance the idea of transformation takes in their work, is another source that needs to be examined. The architecture of Terragni, one of Peter Eisenman’s main points of intellectual departure, systematically combines classical and modern ordering systems to produce irresolvable internal tensions. Terragni’s project for the Danteum is another precursor to the systematic approach to design, especially in its use of numbers and ordering systems to convey Dante’s meaning architecturally in a process that Schumacher has described as a movement from metaphor to algebra to metaphor.

Eisenman’s work as a whole, from the early “simple” houses to the more recent experiments with deconstruction, tracings, scalings, fractal geometries, and other formal devices already contains many of the elements of a future computationally based design process. In such a process a formal hypothesis is put forth as a set and order of manipulations or transformations; the system is then allowed to cycle. The result is not known in advance but unfolds as new patterns develop and are accepted or rejected interactively. Eisenman can thus be seen as an important precursor to such an approach.

Terragni’s work, especially in the aspect of non-closure, and of contradictory systems holding each other in check, has provided the underlying principle for Eisenman’s House X, where a system is employed only to be subtly contradicted, so that at every point a dynamic tension is employed, and all simple correspondences are compromised, in order to enrich the result through paradox. In essence, another layer of intricacy is added to the generative process. Much of the power of architectural gestures is contained in such subtle operations, which therefore need to be understood and accommodated by computational compositional tools.

2.3.3: Generative systems

The idea of generative systems has its origins in the work of Aristotle. Ramon Lull was the first to use it symbolically, and Leonardo Da Vinci employed it to produce church plans from the combinations of simple shapes. Durand’s approach is one of combinations of elements as well, but, despite its influence, is rather mechanistic. In fact most generative systems are ineffective in that they are blind enumerators of possibilities, and are thus subject to “combinatorial explosions.”

A different approach is possible, one where the generative system is initially not constrained by the architect’s intent to produce “reasonable” alternatives. Such an approach was taken recently by
Bernard Tschumi. In his article entitled "Madness and the Combinative," Tschumi lists the following operations that can be used in the combinative approach to design:

- translation (quantitative transformation),
- reduction (suppression, excision, amputation, miniaturization),
- augmentation (addition, extension, rhetorical amplification, collage insertion, scale adjustments),
- substitution, distortion, contamination, permutation, disjunction, dissociation, rupture, dislocation, cut-ups.  

The architect is thus seen "first as formulator and inventor of relations"  

Bernard Tschumi's winning scheme for the Parc de la Villette competition was generated using a generative system. Three abstract systems were superimposed, each with its own logic, over a site that was ordered through the use of a regular square grid. The three systems were allowed to "contaminate" each other by being overlaid, each point on the grid becoming an architectural "folie". Tschumi writes: "no permutation is "innocent": just as the form of the text cannot be changed without altering its meaning, so no permutation of program, space or movement fails to achieve a shift in meaning."  

The areas of structural and geometric conflict that inevitably appear in this kind of approach are actually not problems, they are indeed opportunities. It is precisely in these areas that creative design takes place, and the attempt to remove them through standardized details that can be computerized amounts to "neuter"ing the architecture. The goal, then, is to set the generative systems in action, and if the result is unsatisfactory, to resist "fixing" it at the detail level; instead, to go back and alter the generative systems themselves.  

2.3.4: Languages of Architecture  

A fourth major avenue of research is suggested by the existence of large bodies of knowledge constituting "languages". First among these is, of course, the language of classicism. Alexander Tzonis and Lianne Lefèvre give a coherent presentation of this language in *Classical Architecture: The Poetics of Order*. In particular they describe the structure of this language as consisting of Taxis, Genera, and Symmetry. Particularly applicable to this discussion are the concepts of Entaxis, a more complex taxis extending in all directions, and the operations of Citation, Syncretism, and Meta-Statement, which are "ways of using the classical canon as means of questioning a dogmatic or quasi-automatic, routine application of classical order." In the variable conception of architecture discussed here, it is possible to employ the structure of this language with or without its particular forms.  

Another approach to languages of architecture is offered in Andres Duany's formalization of the compositional language of Alvar Aalto in a way that lends itself to its expression as an expert system. Duany has also developed a parametric code for the development at Seaside, Florida, which, he has claimed, is already in the process of being "computerized." Several other efforts

---

1 We have included, in the appendix, our own pseudo-Prolog version of this language.
exist in this direction, from Christopher Alexander’s "A Pattern Language", already an expert system in book form, to the descriptions of the compositional languages of Wright, Palladio, and others.

Iakov Chernikov and other Constructivists elaborated a language of the combination of forms, using the basic concepts of “constructive combinations”, such as amalgamation, combination, assemblage, conjugation, and penetration, embracing, mounting, integrating, coupling, interlacing, clamping, linking, and so on, both statically and dynamically, using hard or soft materials. These concepts facilitate the three dimensional manipulation of form. Although they are primarily part of an educational approach (at last in the case of Chernikov) they also constitute a possible architectural language, one having the advantage of describing form in a manner familiar to architects from the days of their design foundation studies.

Several architectural handbooks also exist, from Vitruvius, through Alberti, past the École des Beaux Arts, to the present. Handbooks such as Francis D.K. Ching’s "Architecture: Form, Space, and Order" and Clarke and Pause’s "Precedents in Architecture" are also useful sources for establishing a foundation of distinctly architectural concepts, a foundation, like all foundations, to build upon, to exceed.

Clearly, the overall goal in this case is to allow architects to invent their own languages, including types, transformations, proportional systems, generative systems, grammars ...

2.3.5: Exposition and Development

The creation of architecture can soon cease to be seen as the creation of isolated spatial events. Architects will have tools that allow them to manipulate the overall order and organization of their buildings, thus replacing a single design with a range of designs. Ferdinand de Saussure made the distinction between lexicological and the grammatical instruments, as referring to the two extremes of the range of language, the lexiz, or word, and the grammar. In music these correspond, on the one hand, to exposition, the stringing together of loosely developed phrases, and, on the other hand, to development, the elaboration of a theme and variations, the virtuoso and the composer. Frank Gehry and Peter Eisenman may be seen not as ir reconcilable opposites, but as end markers in a continuum, with the building oscillating in between until a final value is given as input, and we return once more to the "radical solitude" of buildings 13.

Part III: Designata: A Proposal for Computational Composition

Everything that has been said up to this point boils down to this: in language there are only differences. Even more important: a difference generally implies positive terms between which the difference is set up; but in language there are only relations, without positive terms. 18

Ferdinand De Saussure
We can now formulate an outline for the long term goals of computer-aided design. We propose:

1) That computer-aided architectural design should become a means for investigating pure and not just applied architecture;

2) that it make provisions for the full range of possible designed artifacts while imposing the least possible "functional fixedness" and allowing the most interpretations;

3) that it should seek to encompass the accumulated knowledge of architectural tradition;

4) that it should encourage, even promote, a questioning attitude with respect to that architectural tradition;

5) that it should carry on this questioning attitude by looking at architecture in broad, cultural and historical terms;

6) that it should seek influences external to architecture and to computing, especially from music, science, literature, criticism, philosophy, the arts, poetry, history;

7) that it should seek to lead the ongoing architectural discussion rather than accommodate any particular dominant school;

8) that it adopt and invent computational paradigms on the basis of their eventual potential in enlarging the range of architectural form, since all architectural ideas must eventually be expressed in built form;

9) that it emphasize the new paradigm of architect as computational composer, not rejecting the previous paradigm of the architect as performer, but simply recognizing that traditional media, tools and processes already quite well suited for its demands;

10) that it should seek to provide new ways of structuring relations between terms, differences rather than just positive terms.

11) that it should recognize that the issues of computational composition are, to a great extent, independent not only of drafting and rendering, but also geometric modeling, though they rely on these technologies for the display of results. (This observation has great implications for the architecture of the next generation of computer aided design systems, as it introduces a new layer between the interface and the modeler, and requires that systems be able to import compositional data computed using standalone programs. It also allows research in computational composition to proceed independently.)

How can these goals be met? Throughout this paper we have presented various dualities, and we will employ one more. Parallel to the tradition and to the overcoming of the tradition, we suggest, on the one hand, architectural knowledge based tools, and on the other, the direct questioning of these tools through programming (as composition), by the architects themselves. We propose the working model of the designata. Let us explain.

The range of designed artifacts, which we call designata can be specified as follows:
In this list, the first group specified is made up of design artifacts commonly identifiable as objects (in one, two or three dimensions, solid or void); the second, entities identifiable as events (sequences of actions pertinent to a task, configurational designs, such as temporal reasoning, plans of action); the third, systems of relations (games, simulations, organizational structures, etc); and the fourth, systems of reference (artificial, notational, textual, etc.). The four broad categories that constitute the designata help ensure completeness. For each designatum, we distinguish the matrix shown in figure 1:

![Designata Matrix](image)

**Figure 1: Designata**

This is to say that there are at least four levels of primitives that are of interest to the architect, as shown in the the rows of the matrix. For instance, an architectural design may be described as consisting of elements such as columns and walls organized on a grid, ordered symmetrically, and exploring the formal idea of having analogous plan and section relationships.

The columns of the matrix show how the variation of these elements is structured. First, for each row-item we can recognize existing archetypes, such as different column types (doric, ionic, corinthian, piloth) in the element row, grid types (radial, central, orthogonal, triangular) in the organization row, ordering principles (symmetry, asymmetry, clustering) in the ordering row; and formal ideas (plan to section relations, overlap, transformation) in the formal idea row.

Second, each pre-existing type, and any that are added to this framework through its use, can be parameterized, as implied by column two. Third, as column three suggests, those parameters can be systematically varied by employing appropriate, normative generative systems to determine, for
instance, proportions, spacings, transformations, and other operations. Finally, column four suggests that the system has provisions for keeping track of combinations of simpler types at each level.

One more level of distinctions is necessary. For each item in the matrix, we recognize that variations can deal with relative or absolute values or, stated differently, with the locus or the measure of a primitive. Depending on the kind of artifact being designed, this distinction may apply to spatial relations (hence to topology vs. geometry or topometry), to temporal relations (thus to chronology vs. chronometry), and finally to relations of energy (and thus to energy loci, such as equilibrium vs. energy measures, such as particular input values to a circuit).

This framework provides an encompassing view of the kind of primitives and operations we think are necessary to support a true computer-aided architectural design. Architectural investigations can take place within this framework in at least three levels:

1) selection among pre-existing types, using direct parametric variation;
2) selection among pre-existing types, using computed values for parametric variation;
3) direct definition of new types and subsequent direct or computed parametric variation.

Using a system that employs such a framework the architect can select the elements, grid, and proportional system for a design; for example, columns on a rectangular grid proportioned according to the golden mean, may be selected, and experimentation can proceed in the following ways:

1) vary the bay size of the grid
2) substitute planar elements in place of the columns
3) switch from a rectangular grid to a radial grid
4) compute a new proportional system in place of the golden mean
5) apply progressive transformations to the elements on the grid, or to the grid itself
6) alter the symmetry of the entire composition
7) alter the relationships of solids to voids
8) alter the plan-to-section relationships
9) apply topological transformations to elements
10) invent new computed types of elements, grids, symmetries, transformations.

These, of course, are but a few of the possibilities inherent in this structure.

Part IV: Recent Work

This section of the paper outlines investigations on the variation of proportion and order, the algebra of solids and voids, and experiments with algorithmically composed music.
4.1: Variations of Proportion and Order

Consider chess and checkers: for the same game layout and number of pieces, it is the abstract parametric structure, the set of game rules, that determines whether you are playing one or the other. In either case, however, it is the particular values entered for that game that distinguish a winning game from a losing game. The creation of architecturally interesting objects requires both an appropriate parametric structure and a very specific set of values. Figures 2 and 3 show the same object responding to computed variations in the values placed into a hierarchic parametric structure of proportions, order and placement. These images emphasize the computation of the parameters itself, both to dimensions, orientations, placements, and order of the parts, and suggest a field on inquiry examining architectural rhythm, harmony, proportion.¹

![Diagrams showing variations in structure and proportions](image)

Figure 2: Original quadrant, rotational symmetry, and compositions derived using changes of proportions.

The top left image in Figure 2 shows the generic quadrant that forms the basis for all the steps that follow. It is composed of several primitives, some visible and some invisible. The proportions of

¹A brief overview of the PADL-2 solid modeler, and a description of how these images were generated using it, can be found in the Appendix.
the primitive shapes are interrelated in various ways. First, there are global parameters that govern the primary ratio of the composition (f.e. the golden mean), and which are transmitted from the root of the CSG tree to each of the branches. Second, there are intermediate parameters that govern the assemblies of parts, so that, for example, it is possible to dictate that only ratios of prime numbers (f.e. harmonic ratios) will be used. Third, there are local parameters, such as the radius of a column, the height of a block, etc. Significantly, the parameters are used not only to specify the dimensions of parts but also their relative locations in space. It is thus possible to alter the composition at three levels: by altering a fundamental ratio, by modifying the relations within an assembly of parts, or by modifying any single part.

Figure 3: Additional compositions derived using changes of order and placement.

The examples shown above show primarily geometric variations, though the topology of the final result is indeed altered through the solid modeling operations employed. More intentional variations of topology are also possible through the conditional inclusion or omission of parts, the conditional decision to use union, intersection or difference between volumes, and so forth, as is demonstrated in the solid/void algebra described below.
4.2: Solid/Void Algebra

The following series of images illustrates another compositional investigation. We have found that it is possible to generalize the relationship between solids and voids into a solid/void algebra, echoing an architect's intuitive reading of the spatial implication of the overlap of figures in plan and section.

Five clear relations between solids and voids were discovered. For the sake of simplicity they are presented below as the relationship between two objects, A and B, each consisting of a solid and an associated void. It should be noted that these relations can be generalized in at least the following ways: a) each object can consist of a set of solids associated with a set of voids; b) the relationships can be easily extended to express the relations between any number of sets of objects, not just two.

![Five basic relations of a solid/void algebra](image)

Figure 4: Five basic relations of a solid/void algebra
The five relations are as follows, as in figure 4, from left to right:

1) \(((\text{solids of } A \text{ dif voids of } A) \text{ un } (\text{solids of } B \text{ dif voids of } B)) \text{ un } (\text{voids of } A \text{ int voids of } B)\)

2) \((\text{solids of } A \text{ un solids of } B) \text{ dif } (\text{voids of } A \text{ un voids of } B)\);

3) \(((\text{solids of } A \text{ dif voids of } A) \text{ un solids of } B) \text{ dif voids of } B\);

4) \((\text{solids of } A \text{ dif voids of } A) \text{ un } (\text{solids of } B \text{ dif voids of } B)\)

5) \(((\text{solids of } A \text{ dif voids of } A) \text{ un } (\text{solids of } B \text{ dif voids of } B)) \text{ un } (\text{solids of } A \text{ int solids of } B)\)

where \text{ un, dif, and int} stand for union, difference, and intersection, respectively.

Figure 5: Application of the solid/void algebra to separately parameterized solids and voids.

Figure 5 shows applications of some of these relationships to two more complex combinations of shapes. Shape A consists of a cylindrical solid with three voids, and shape B consists of a rectangular shape with four voids. Notice that since these are defined parameterically it is possible to vary independently the size, position and orientation of both solids and voids, as well as to investigate different spatial interpretations of the same diagram. Later figures show an additional level of complexity, in that the voids of the rectangular shape have been divided into two layers so that the voids of B now total eight, in two groups of four. Each of these groups can now be manipulated independently of each other, as well as independently of the shell.
4.3: Computational Music: Testing architectural ideas

The work undertaken using PADL-2 made clear the importance of learning to manipulate relationships computationally. In light of our previous theoretical analysis, as presented in the first parts of this paper, and the computational parallels with music, we attempted to manipulate relations directly, using sounds instead of images.

The choice of sound may appear arbitrary at first. Upon examination it becomes evident that the mechanisms needed to generate and select musical rhythms, harmonies, melodies, counterpoint, parallel voices, and so on, can be quite similar. The particulars may differ as, for example, the harmonic relation between two sounds may not yield a harmonic relations between two shapes, but the process of determining the compositional structure, what criteria to employ, and at what steps they are to be employed, is identical.

An important finding has been that the old problems of generate-and-test, combinatorial explosions, and so on, may not be the formidable obstacles they have been made to appear. This is because the issue is simply avoided by dictating what relationships are desirable at the onset. Design is not search-it is pattern imposition. This is not as tautological or trivial as it may appear. With a well chosen set of rules the complexity of the finished piece can very easily become quite rich and the result almost impossible to anticipate, while still being entirely determined. Of course, all sorts of mechanisms can be employed to provide conditional or random changes of direction to take place in the program.

Another important observation has to do with the opening up of new areas of investigation. In the brief time we spent writing these programs, we have experimented with fractal music, several variations with transcendental numbers (pi is quite musical), Fourier transforms, and the musical equivalents of architectural grids and proportions. We are now elaborating on the fractal approach to increase our control over multiple voices. We are also investigating genetically and chaotically inspired mechanisms.

Abstractly, a most interesting and intellectually stimulating problem has been raised, one that is a challenge to both computer-aided design and architecture. At its highest level, the problem is that of understanding how to generate complex, interesting, intricate, interwoven patterns of numbers. Such patterns seem to have intrinsic interest for all people, and may justify the independent study of the aesthetics of pattern. Once these have been generated, they can be mapped into any sensual system, be it visual, tactile, auditory, perhaps even olfactory or olfactory. There is a purity to the problem that is very stimulating, in architectural, mathematical, cognitive, and computational ways, and at the same time the pragmatic implications are enormous. To comprehend how immense these implications are, it suffices to consider the cityscape as pattern.

\[\text{\footnotesize \begin{itemize}
\item We have speculated elsewhere, that intelligence itself is the ability to observe, manipulate, and generate patterns, at many levels.}
\end{itemize}\]
Conclusion

If we have sounded polemical throughout this paper it is not because we do not recognize the contributions that have been made so far, or because we do not share in the enthusiasm that the promise of computer aided design can generate. Quite the contrary, it is on the basis of those contributions, and that enthusiasm that we can now articulate this position. We are indebted to all the pioneering work that we have consciously not mentioned in the effort to keep our own position clear; our criticisms apply to our own work so far most of all.

In summation, we have tried to present the view that computer aided design must enter a new generation. In this journey in a new direction, we seek kindred spirits.

Appendix A : A Pseudo-Prolog Expression of Alvar Aalto's Architectural Language

The compositional rules of Alvar Aalto, as described by Andres Duany in his Harvard Architecture Review 5 article, "Principles in the Architecture of Alvar Aalto," can be expressed as follows in a Prolog-like language (the numbering system is just provided for clarity):

(0.0) Aalto's building -> (1.0) organize program
(2.0 ) articulate form
(3.0 ) define space

(1.0 ) organize program -> (1.1) use head/tail principle

(1.1) use head/tail principle -> (1.1.1) define head
(1.1.2) define tail

(1.1.1) define head -> (1.1.1.1) make mass positive
(1.1.1.2) make formally elaborate
(1.1.1.3) make finite

(1.1.2) define tail -> (1.1.2.1) make space positive
(1.1.2.2) allow potential for expansion
(1.1.2.3) mediate unresolved grids

(2.0) articulate form -> (2.1) create strong silhouette
(2.2) make monumental
(2.3) bind disparate elements

(2.1) create strong silhouette -> (2.1.1) make perceived silhouette diagonal OR
(2.1.2) make perceived silhouette stepped OR
(2.1.3) give high strong silhouette OR
(2.1.4) give low strong silhouette

(2.1.1) make perceived silhouette diagonal -> (2.1.1.1) use horizontal plan lines
(2.1.1.2) use plan lines parallel to visual axis OR
(2.1.1.3) make actual silhouette diagonal

(2.1.2) perceived silhouette stepped -> (2.1.2.1) use horizontal plan lines
(2.1.2.2) use plan lines perpendicular to visual axis OR
(2.1.2.3) make actual silhouette stepped
(2.1.3) high strong silhouette -> (2.1.3.1) tall building
(2.1.3.2) distorted plan

(2.1.4) low strong silhouette -> (2.1.4.1) low building
(2.1.4.2) diagonal section

(2.2) make monumental -> (2.2.1) form unique elements
(2.2.1.1) use head/tail principle
(2.2.3) inflect toward one point
(2.2.4) unify plan elements
(2.1) create strong silhouette

(2.3) bind disparate elements -> (2.2.1) uniquely formed elements
(2.3.1) precisely located elements
(2.2.3) inflect toward one point
(2.1) create strong silhouette

(3.0) define space -> (3.1) emphasize perceptual qualities of space

(3.1) emphasize perceptual qualities of space -> (3.1.1) deviate from orthogonality
(3.1.2) manipulate section
(3.1.3) offer divergent paths
(3.1.4) determine sequence of vantage points

These rules can be refined to any desired level of detail. At some point they will yield simpler spatial entities such as:

space -> space plan
  space section
  space openings
space plan -> plan outline
  wall thickness
  floor thickness
  ceiling thickness
space section -> section outline
  wall thickness
  floor thickness
  ceiling thickness

At the lowest level graphic primitives and actions can be defined. While this example is far from being a complete system, it is sufficient to show that procedural considerations are replaced by declarative considerations. The resulting structure is roughly hierarchic, but it remains open-ended at both the conceptual and the detail level. Redundancy is also employed, and, in both OPS5 and Prolog, the built-in inference engines and conflict resolution schemes take care of the precise workings of programs, once robust structures have been developed. It is possible, of course, to break out of these languages and resume conventional programming for particular calculation intensive purposes, as needed, and then return to the expert system environment.
Appendix B : A Partial Overview of PADL-2

PADL-2 (Part and Assembly Description Language), is a solid modeler developed primarily for mechanical engineering applications. It is based on mathematical definitions of half-spaces, and thus gives precise intersections between shapes, avoiding the awkward intersections found in solid modelsers based on boundary representations.

The PADL-2 solid modeler has several features that are very useful for design:

- objects can be defined parametrically
- objects can be defined as "genetics"
- positional definitions can be defined parametrically
- values parameters can be computed within the object definition or externally
- conditional structures are accommodated
- the set of primitive objects can be extended using built-in "meta-primitives" consisting of planar, spherical, cylindrical and toroidal half-spaces.

As with other solid modelers, objects are defined with a constructive solid geometry (CSG) tree definition, employing the primitive objects block, cylinder, sphere, wedge, and torus. This set can be extended through the use of "genetics" and "meta-primitives," which, once defined, can be treated in exactly the same manner as the original primitives. The usual operations of union, operation and difference are available.

Objects are defined in a CSG tree of files. Each file contains the part description, default values for all the parameters necessary to define the part, and references to other generic files. Dimensions can be given either as constants, variables or expressions, allowing the dimensions of one part to be dependent on the dimensions of other parts. Conditional statements are allowed within files, so that, for example, the same file can produce a three legged and a four legged table (i.e. allowing variations in topology as well as geometry). Looping, however, is not supported. Parameters can be passed from parent files to children files, overriding the defaults given.

Parts are defined as follows:

```
generic column (column);              [file name]
...                                     
  base_width = 2;                     [default values for parameters]
  base_height = 1;                   
  base_depth = 2;                    
...                                     
  ratio = 2;                         
  height = 6*radius;                 
  radius = ratio*base_width;         
...                                     
  column = base union shaft un capitol; [main definition]
  base = block(z=base_width, y=base_height, z=base_depth) at base_location;
  shaft = cylinder(r=radius, h=height);
```

Once an object has been defined it can be used by being invoked multiple times from within another generic definition, stored in a separate file:

```
...                                    
  four_new_columns = column(radius=6) at column_location_1 union
  column(ratio=3, height=12) at column_location_2 union  column() at column_location_3 union
  column() at column_location_4;         
...                                    
```

29
Notice that the previously defined object can be invoked several times, each time with none, some, or all of the parameters that define it being instantiated. If no parameter is included, the defaults given in the original definition are given; if new parameters are given, they override the defaults. Similar freedom is available in the definition of locations: it is possible to define a generic grid, associate a bay dimension and a skew angle with it, and generate a multiple variations of objects associated with the grid simply by manipulating the grid's defining parameters.

It is thus possible to build parametric libraries of archetypes of both organizational devices and of objects, as well as to define operations of particular interest, such as the solid/void algebra we will present below. Finally, external programs can be used to compute values for particular parameters or to define entirely new generic objects. The architect can thus write programs that investigate architectural ideas directly without having to be concerned with the operation of the solid modeler itself.

References