URBANISM AS COMPUTATION

Nikos A. Salingaros
Department of Mathematics
University of Texas at San Antonio

Keynote speech at the "Complexity Theories of Cities" Conference, Delft, Holland, 24-27 September 2009. Published as a chapter in: J. Portugali, H. Meyer, E. Stolk & E. Tan, Editors., "Complexity Theories of Cities Have Come of Age: An Overview with Implications to Urban Planning and Design", Springer, Berlin (2012), pages 245-268.

Abstract.

Successful urban configurations are the result of a complex sequence of implicit computations that transform unorganized input into organized output. Although that is exactly what they do, few urbanists discuss their work in these terms. I propose a fundamental distinction between the principal methods of urban computation. One algorithmic process for urbanism leads to formal planning, which lacks the complex organizational structures that support essential adaptability. This closed computational method uses a set of fixed, or formal, rules to compute a configuration that does not adapt interactively during execution. Such algorithms perform each computation based upon predetermined rules, and those rules cannot be changed via any interaction. The other method of urban generation is achieved by means of interactive computing, which is the basis of human intelligence. An interactive computational method generates adaptive organic urban fabric, as seen in both traditional cities and squatter settlements. Adaptive computational systems necessarily rely upon interaction with their situational environment. In this interactive approach, the result of each step in the sequence of computations is fed back into the algorithm so as to influence the subsequent step. The algorithm itself changes by interacting with whatever it is computing. Interactive or intelligent computing, therefore, is not equivalent to computations that rely exclusively upon a fixed algorithm. These two diverse computational methods design two morphologically distinct types of urban fabric. Also included in this discussion are urban morphologies that have no computational basis, as well as those that are deliberately random.

Introduction.

I suggest re-thinking the discipline of urban design using algorithms as a model. Different categories of urban fabric are classified here according to the type of algorithm used to generate them. With this classification, urban morphology is no longer based on formal suppositions, and visual patterns can be used to judge the presence or absence of living urban fabric. This applies universally to all instances of the built environment. Urbanists will be able to utilize processes found in other algorithmic structures such as those in computer science, biology, robotic design, artificial intelligence, etc. All these techniques and mechanisms new to urbanism can be used to effect better cities.

The model presented here considers urban morphogenesis as a sequence of extended computations. A piece of urban fabric arises as the result of many steps, just like the application of an algorithm that computes the solution to a mathematical problem. The interactive computational procedure produces an adaptive urban fabric that is highly complex such that the end product necessarily cannot be reached all at once. Therefore, we should conceptualize urban design not as one fully formed vision, but as a computational process employing algorithms consisting of very many interactive steps. In an effort to make this natural process more explicit, and thus manageable, consider the systems of computation that serve to establish a healthy built environment on the scale of a city.

An algorithm is a sequence of prescriptive steps that eventually leads to a result as the solution of a problem. An urban computation uses data (e.g. rules; constraints from the site, brief, and planning legislation; a previously-built example as a model; intuition, etc.) to make a design decision. In a mathematical sense, the 'computation' has transformed the initial data and constraints into a result. A final design is the accumulation of a large number of individual design decisions. In the urban context, our 'result" is a geometrical plan for the shape and positioning of buildings in a portion of urban fabric. The specific shapes of buildings (interior plans, exterior elevations) we can leave to architecture, whereas the placement and situation of buildings is the proper domain of urbanism. Those architectural design steps are important as part of a more complex design algorithm, but some of the steps, particularly those that compute the smaller scales, lie outside the scope of the present analysis.

Urban computations are steps in practice: siting a city with respect to major transportation arteries; planning roadway connections; situating the city relative to the geography of the site; taking advantage of weather patterns, etc. Each step of the design algorithm is a prescription based on empirically derived urban planning practice. Traditional urbanism acted with rules based on precedent. Starting around the 1930s, however, urban improvements more often than not effectively destroyed the positive qualities of existing urban fabric because they did not correctly anticipate their own consequences on urban use.

A city 'computes' its structure by implementing different cumulative steps. Human needs and commercial forces establish the criteria of the computation, adjusting at each step of the way so as to satisfy legal and other constraints. Sometimes it is the government that builds the city, as in centralized state systems. Otherwise, it is private capital. If we took more care, we could compute every new building's shape and the position of its entrances, the building's detailing using an appropriate form language, etc. This proposed computational procedure should be adopted in terms of legal codes. Existing post World-War II zoning codes are simplistic and have sought only to maximize vehicular flux and the number of storeys. The presently formalized legal planning rubric has thus eliminated several layers of pedestrian ground level use and has led to the dehumanization of downtowns.

Design often appears to lack any computational basis, but this impression can be misleading. In the arts, a visual form, sound, or dance movement is frequently invented as an expression of one's interior feelings and not as a response to the environment; in this case there is a direct externalization of a state of mind and nothing is actually 'computed'. In classic artistic traditions the world over, however, a formal structure requires that innovation necessarily follows certain internal and external structural rules, so that creativity does indeed follow a computational process. It is only in the artistic world of recent decades that all computation has been abandoned to free invention without any constraints. This attitude has influenced urbanism in the same period.

Every urban design involves computations (even if those happen to be trivial — where there is hardly any computation at all). A trivial algorithm may consist of only one or two steps. I will argue that living urban fabric is the result of interactive computations that generate ordered complexity. Comparing different approaches to planning and urban design according to their number of computational steps, we can discover which urban algorithms generate the highest degree of ordered complexity, which of them generate no complexity at all, and which of them generate disordered complexity via randomness.

Urbanism's computational basis.

Our most recent understanding of urban structure recognizes a city to be a highly complex system, both in a static structural sense as well as in the dynamic sense of movement and continuous change (Alexander, 2001-2005; Batty, 2005; Hillier, 1999; Marshall, 2009; Portugali, 2000; Salingaros, 2005). Researchers around the world analyze and compare successful and failed pieces of urban fabric by applying techniques of complexity theory. At the same time, however, cities are being built that ignore all those findings. Even the most basic concepts necessary for the understanding and design of complex systems are violated in the design of new urban regions. This contradiction may be explained by the unfortunate disconnect that exists between urban practice and a theoretical understanding of urban morphology.

Let me review different types of algorithms in general before applying them to urban situations. The simplest algorithms are called deterministic, in which both the set of instructions and the input data are completely known beforehand. Like a mathematical function, the same data inputs will give the same output. There are, moreover, trivial examples of deterministic algorithms that give the *same* output for all *different* inputs, and these are constant algorithms (i.e. an algorithm that always gives the same fixed answer). The most simplistic urban algorithm is represented in International Style urbanism, which dispenses with all computational complexity since the result is every time the same (Figure 1). This widespread example corresponds to a constant deterministic algorithm where little or no computation is taking place.

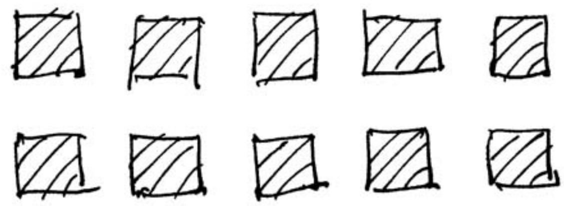

Figure 1. Simplistic modular buildings distributed along a regular but irrelevant grid.

Going one step further in computational complexity, a formula that associates a distinct urban typology with each different situation corresponds to a simple deterministic algorithm. It is more sophisticated than the constant deterministic algorithm. Simple typologies have been used throughout history to build according to fixed templates (e.g. house, temple, shed, school, tavern, etc.). These templates are still limited because, if there is no further computation, the result will not adapt to the site and conditions.

More sophisticated types of algorithms will include internal branches and selections (alternatives), where the computation chooses to proceed among different sequences of instructions at certain points. These choices may be one-way computational branches, with no return possible, or the algorithm may include loops that allow iteration and feedback. With increasing sophistication, an algorithm becomes more complex and acquires an internal logical structure consisting of branches and loops that enables a combination of selections and conditions. Exactly which choices are made to execute different portions of the algorithm will depend on each individual case.

Going beyond deterministic algorithms, dynamic or online algorithms add new dimensions of computational complexity since they adjust to handle constantly changing data inputs. Complex algorithms cannot be represented by analogy to something as simple as a mathematical function. There is no formula for finding the final state. The more interactive an algorithm is, the more its results exhibit emergence, i.e. unexpected behavior that could not have been predicted at the beginning of the computation. But even with emergence we have distinct cases, where a system may either develop its internal complexity in isolation from its surroundings, or embed itself to become a complex part of its environment through adaptation.

Different types of urban fabric found around the world represent processes with varying degrees of intrinsically complex computation. I believe that every distinct observed category of urban morphology can be usefully classified according to how much and what type of computation was involved in creating it. This classification is potentially significant due to being universal and inclusive, as it relates each of the very different examples of urbanism within one rubric. My goal is to educate the reader sufficiently so that, after understanding this model, he or she can very easily deduce the type of computational procedure that generated an observed portion of urban fabric. The knowledge obtained from studying the plan and geometry of the urban fabric can then be used as a diagnostic tool.

Four-way classification of different urban categories.

In order to make sense of a wide variety of urban morphologies, I propose a general division of planning and urban design methods, based upon their computational content (Figure 2). Urban morphologies can be roughly divided into four groups, with each category transitioning between and merging with any of the others. Three of these categories have a computational basis and one does not.

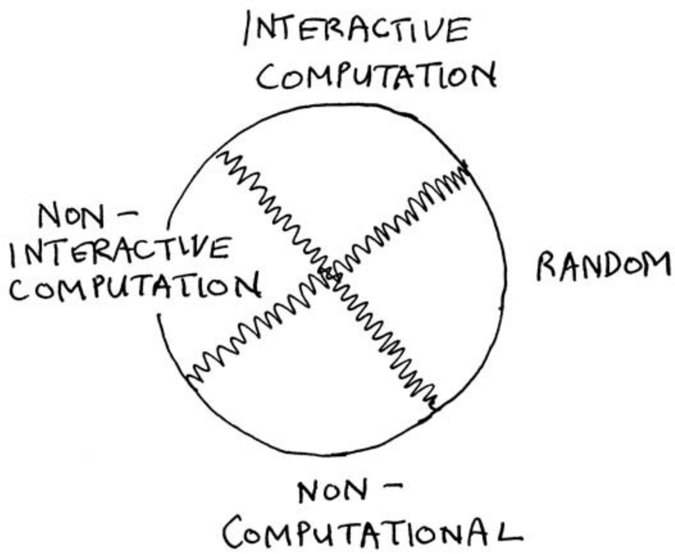

Figure 2. Four-way classification of distinct categories of urban morphology according to their computational complexity.

The four categories of urban morphologies are labeled on the basis of their underlying computational structure as follows, in order of decreasing measure of computational overhead. Another way of thinking about these four categories is to say that their ranking corresponds to decreasing computational complexity:

A) Interactive computations.

B) Non-interactive computations.

C) Random.

D) Non-computational.

I will discuss each of these design categories in detail, and explore the relationships among them. Historically, different types of algorithmic procedures have been used to generate the urban plan in examples of each of the four separate groups. Planning each particular case of urban fabric on the ground (i.e. a particular group of buildings and connecting roads) will likely need a distinct algorithm, but we are discussing the general algorithmic *types* rather than specific algorithms. The focus here will be on the morphology of the plan, hence other variables, such as the height of the buildings involved, will not be considered.

The first category of "Interactive computations" represents urban fabric that arises from dynamic computations with feedback, leading to an adapted form according to the multiplicity of existing conditions on the ground, and to constraints dictated by established human needs. It turns out that traditional methods of urban design, before the industrialization of the twentieth century, belong to this category of interactive computations. Biophilic and sustainable design methods now being developed as an answer to unsustainability are also based upon interactive computations.

The first two urban categories both involve computations, with "Interactive computations" being the most complex of all. The difference with the second category — "Non-interactive computations" — is that those use deterministic algorithms which are self-contained and do not take into account feedback from the form during its modeling computations. Such computations are therefore pre-determined, since they admit no feedback from the partial intermediate result and hence no adaptive adjustment is possible during a computation.

The third urban category — "Random" — may involve computations in order to generate the observed randomness, but this type of computation is unrelated to urbanism, being solely responsible for a random visual/graphic (as opposed to urban) design. Thus, whereas random building configurations might actually be the result of computations, all the effort in this category is directed at a graphic design and not at the essential elements of urban adaptivity. Any such computations are therefore programmatically irrelevant to the basic purpose of the city.

The fourth, or "Non-computational", urban category has little relevant computational basis, and may involve no true computations at all. Its morphology is simplistic in the extreme. This is the constant deterministic algorithm (analogous to a function that always gives the same result for whatever input) that generates International Style urbanism, military camps, warehouses, and some industrial installations (Figure 1).

The four-way classification could alternatively be introduced in terms of a mathematical tree structure. That is, a first general division between "Computational" and "Non-computational" categories, followed by further subdivision of the "Computational" branch into "Interactive" and "Non-interactive" branches. While this does represent a more elegant theoretical explanation, I wish to diagnose all types of urban structures as seen 'on the ground', where they occur mixed in with each other from different periods and by different architects, making the simultaneous classification (shown in Figure 2) more practical.

Design adaptivity as interactive computation.

It is possible to measure design adaptivity on an axis superimposed upon the four-way classification of computational categories (Figure 3). Since we are not dealing with abstract concepts but instead with the lives of human beings, the morphology of what is being experienced will affect users. The geometry of different types of living environments has a profound effect on humanity. Distinct urban categories in general range from being detrimental to daily living, to being experienced as neutral, or towards enhancing daily life. These human qualities can be identified according to the urban category's computational adaptivity to human needs and activities.

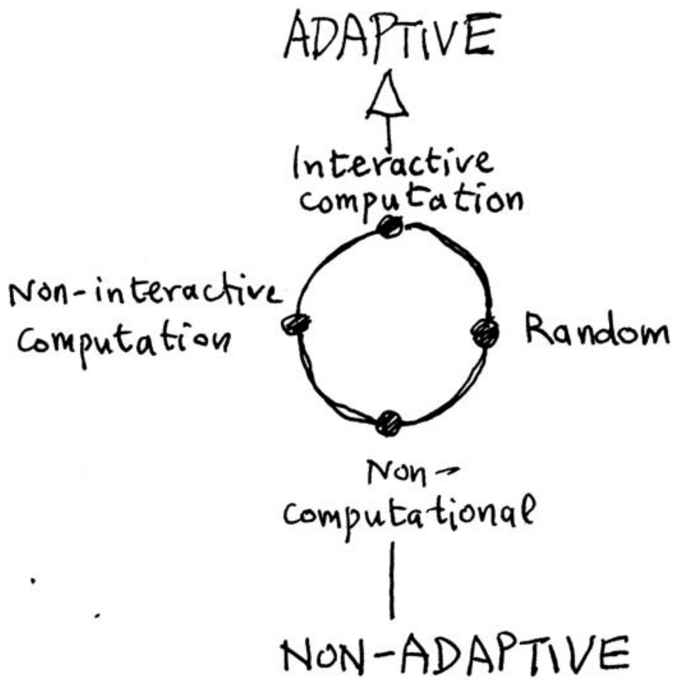

Figure 3. Adaptive design axis superimposed on the four-way classification of design methods.

Known historical situations and documentation from many distinct urban configurations the world over lead to the following unequivocal conclusion (Alexander, 2001-2005; Salingaros, 2005; 2006; 2010). Non-computational urbanism can be detrimental for society, whereas Interactive computational urbanism is beneficial for accommodating human life and activities. Adaptivity increases vertically in the four-way classification (Figure 3) because of the algorithmic content, and I will argue that this naturally implies adaptivity to human life. The other two categories, Random urbanism and Non-interactive computational urbanism, represent situations that can range from neutral to negative in their impact on residents and users, depending upon how closely they parallel the Interactive computational urban category.

If we rank the categories of urban morphologies in the reverse order: D, C, B, A, we see an upward transition in the four-way classification from simplistic, non-computed structures towards those with increasing complexity and adaptivity (Figure 3). This progression is followed in the rest of the discussion. As computational complexity can increase upwards in distinct directions (it need not go straight up), it is necessary to distinguish between complexity adaptive to human needs and complexity that is irrelevant and can therefore be potentially harmful. My thesis is that beneficial complexity arises through an interactive computational process, which evolves forms using feedback from human needs and sensibilities. Complexity for complexity's sake is often gratuitous, and its effects on human beings range from benign to causing alarm and anxiety.

Traditional urban design as interactive computation.

In traditional societies urban design tends to be highly 'intuitive'; that is, it uses human perception and intelligence for determining both the shape and positioning of buildings within an urban region. Following Alexander (Alexander, 2009), I claim that this is the result of a highly sophisticated set of complex computational processes responding to the environment, which use the human mind and perception system to carry out interactive computations.

Traditional building practices adjust the orientation and the relationship among adjacent and nearby buildings, resulting in complex urban fabric (Figures 4 & 5). In an urban setting, the density achieved after several generations of building and adjusting (computing) has generated a merged, interlaced urban fabric with a complexity that is characteristically very difficult to draw accurately (Figure 4). This is an important point, since the human mind has evolved to easily perceive a certain type of decomposable complexity, whereas evolved complexity escapes our understanding (Salingaros, 2005: Chapter 10). Even in seemingly simpler traditional situations involving detached buildings (Figure 5), their alignment and relationship to each other contains an ordered complexity that bears no relation to the superficially similar disordered complexity underlying more recent urban projects. Alignment in genuinely adaptive urban configurations is due to interactive computations, where the shape of open spaces is carefully computed and is not merely 'left over'.

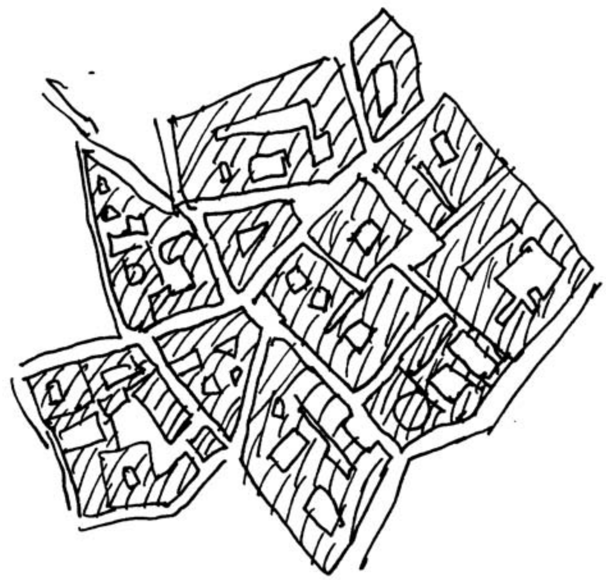

Figure 4. Traditional urban fabric is computed using interactions.

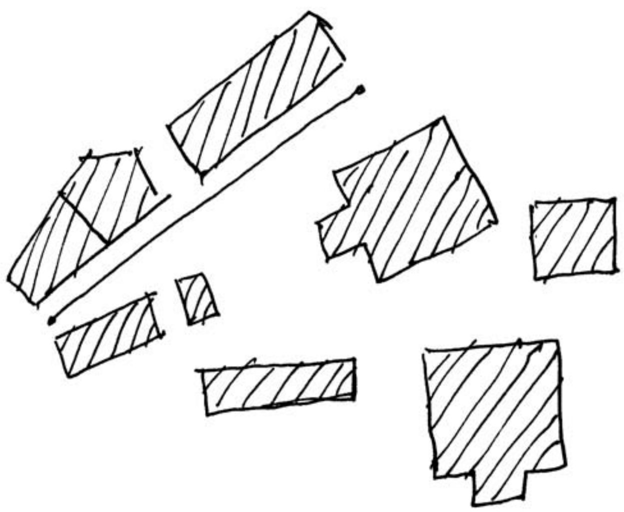

Figure 5. Traditional positioning of buildings uses interactive computation for their alignment.

Suppose that we wish to insert a new building into partially built urban fabric. The decisions regarding orientation, size, and placement of the new building should be determined by every structure that already exists, including other buildings, roads, and paths. There are legal constraints that cut down the number of choices: one has to build within certain distances set out by codes (Duany *et. al.*, 2009). An intelligent architect will take into consideration all surrounding factors so that the final building will appear to be an integral part of the urban fabric, and thus optimize its own access and use. Energy optimization according to climate and the effects of surrounding buildings influences both orientation and especially the architectural design of the new building.

Some of these decisions (interactive computations) are taken consciously, while others are subconscious. The subconscious decisions are just as important, and just as computational, as those decisions taken on the basis of measurements, and thus they are entirely deliberate (Alexander, 2001-2005). Positioning a new building according to existing streets, paths, traffic flow, and relationship to existing buildings is best done in an intuitive manner. The human mind performs the necessary complex calculations to give a very accurate result that appears 'intuitive'. Unfortunately, since intuition involves computations taking place within the subconscious mind, it is (incorrectly) not usually considered to be part of a computational process.

The example given below illustrates how to position and build a dwelling in an informal settlement (Salingaros, 2010: Chapter 12.4; Salingaros *et. al.*, 2006). Suppose the owner-builder already has the necessary scrap building materials for the structure, and has procured a piece of land (often without a legal title). He or she might follow this algorithm within the limits of what's available, and assuming other structures are already built.

1. Roughly locate the proposed building in relation to optimal access from the road or path.

2. Position the building more accurately according to determined rainwater runoff so as to avoid flooding.

3. Take advantage of topography and existing infrastructure in place (if any) for sanitation.

4. Determine intuitively the solar orientation for better heat protection and light.

5. Adjust the footprint of the proposed building to existing surrounding structures to better take advantage of, or avoid problems with, shade, noise, neighbors, etc.

6. Arrange exterior spaces, walls, and fences to optimize privacy where it is most needed for a healthy social life, in relationship to the road and surrounding structures.

7. Arrange the interior plan of the building according to the family's needs, so the geometry will depend upon local social and cultural traditions. This should influence the building's footprint.

8. Make architectural decisions such as materials, structures, windows, floors, roof system, height of interior spaces, etc.

This seemingly simple procedure represents a sequence of interactive non-trivial computations. They are by no means all the steps needed to make a building. The design algorithm will be entirely different if one wishes to create a public space instead of an individual dwelling, for example (Alexander, 2001-2005; Salingaros, 2010: Chapter 7.1). Yet a related step-by-step procedure is followed by the majority of buildings around the world, which are erected by their owners in informal settlements. What people in wealthy societies refer to as the building industry actually represents only a very small portion of construction worldwide, in which the owners themselves carry out most building activity. An entire settlement built by individual owners or squatters follows a distributed or peer-to-peer computation without central control.

Buildings historically used an interactive algorithm to optimize both their adaptation to human needs and functions, and their relationship to existing buildings and paths in their surroundings. Adaptation to local conditions such as climate, locally available materials, local geometry of the urban fabric, and transportation network requires extensive interactive computations in order to achieve the final design. The same method can be used today (Alexander, 2001-2005; Portugali, 2006). The infrequent monumental building could afford to deliberately break this computational rule, precisely in order to stand out and thus make a 'statement'. Computational steps similar to those presented above were applied for millennia to build an organic urban fabric the world over. Of course, the larger and more complex the building, the longer the algorithm will be (containing more computational steps). It is only in recent times that this practice was deliberately avoided for the sake of stylistic (visual) novelty, and adaptation is now unfortunately lost due to lack of use (Salingaros, 2010).

Non-interactive computations.

Most mathematical algorithms are non-interactive. Their computational process does not need to refer to any outside input after the initial data set, because the result is reached by the repeated application of steps defined by a deterministic algorithm. This is the basis of the Turing Machine: a universal computer whose program is stored in memory (Wolfram, 2001). The computer executes a number of pre-determined steps to deliver a final result. Non-interactive algorithms, while perfectly fine for most mathematics and computer applications, provide a misleading precedent for architects and planners who wish to compute urban configurations. Creating a city by applying a sequence of steps (a deterministic algorithm) is more intelligent than the brute imposition of a non-computed design, yet it can lead to the same dysfunctionality because it does not adapt to the complex environmental needs of human beings. For such essential adaptation to occur, it is absolutely necessary to program feedback from human spatial experience during computation. Otherwise, what is computed does not respond to the situation on the ground, and may not even respond to the users' needs.

There are numerous examples of non-interactive urban computations. All of these arise as 'rational' attempts by some central planning authority to generate an ordered and comprehensible urban environment containing all the necessary components. What the algorithm is computing is a plan with aerial symmetries that might look interesting, but there is no other goal to this computation. The resulting configurations (Figures 6 & 7) look ordered from the air, showing varying degrees of complexity. Such urban models include what are called 'ideal cities', 'garden cities', 'campuses', etc., none of which have ever been very successful in producing a living urban fabric. Those planned cities range from a dull, sterile environment to an oppressive one, depending on the distribution of scales. It is very rare to find a product of non-interactive computations that provides a genuinely human environment.

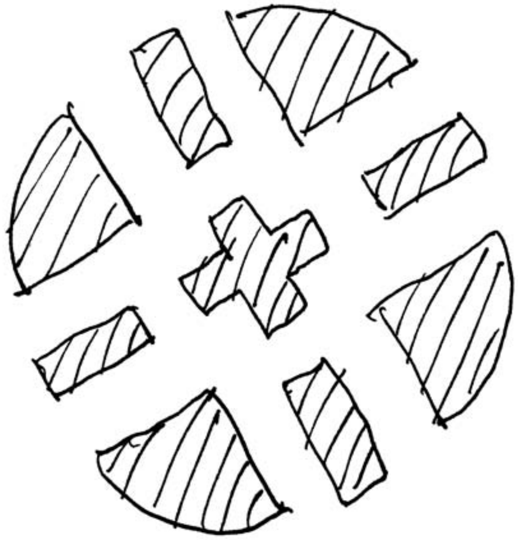

Figure 6. Formal planning creates an ordered but irrelevant geometry.

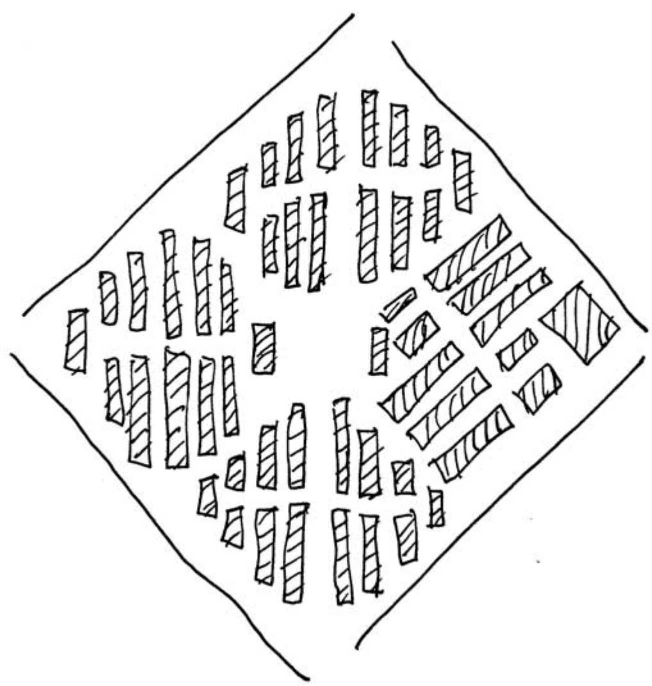

Figure 7. Example of formal planning showing non-adaptive building footprints.

Simplistic cases that have little computation are morphologically closer to the Non-computational category. Examples include giant urban forms of monolithic shape that ignore the human range of scales; these tend to be the worst from a human experience. Computations that generate a complex hierarchy of structural scales, by contrast, might provide better environments than those having only a limited range of built scales. Cases that compute fine-grained structure, at times poorly matched to human needs and activities, at least introduce scales on the human level that could be used for purposes other than those for which they have been designed. For example, the success of some plazas is helped by the presence of low walls used for sitting, originally planned as purely decorative elements; the same holds true for stairs used as seats. Nevertheless, those are instances of accidental success rather than adaptive design.

Non-interactive computations include numerous cases where an interactive computation from one context is merely copied onto another very different context. That is equivalent to using some previously obtained mathematical result, but applying it to an entirely different problem while skipping any new computation. It doesn't make any sense but it certainly saves time! In the same way, builders take a solution developed elsewhere and mindlessly repeat it: for example, a shoe factory in Germany from the 1930s is used as a model for all new hospitals in Texas in the 2000s. This is ridiculous but it is occurring today. Another example would be solutions computed for an auto-dependent landscape, such as a restaurant adapted for a highway truck stop, being inserted into dense urban settings, destroying pedestrian urban fabric in the process.

Here is the basic problem: ***What appears to work and connect on paper in an abstract, formalistic manner does not necessarily work and connect on the ground. This is the first law of human-scale urbanism.*** Moreover, there is no way to predict whether some plan drawn on paper will be successful or not without testing it at least in part at full scale. Informal settlements actually work because they are computed at full scale on the ground. On the other hand, non-interactive algorithms used to build urban fabric turn out to be irrelevant to human actions and needs. Whether some elements of this design strategy are going to be successful or not cannot be predicted in advance.

The second law of human-scale urbanism is that adapted computed solutions are not transferable. General, common constraints do apply in helping to compute each result, but the computation has to be done in every case under very specific local conditions, otherwise the result can never be adaptive. Even quite similar situations, if independently computed, will evolve to show substantial individual differences and modifications. The results as built on the ground are going to be different every time. These elementary lessons have been ignored by generations of post World-War II urbanists.

Random urban design.

It is common nowadays to generate visual complexity via some simple algorithm, and then apply the result — a visual plan in two dimensions — to build urban fabric. This method produces the Random category of urban morphology in the four-way classification (Figures 2 & 3). Any algorithm for producing randomness relies on a random-number generator that is then fed into a visual form. This superficial procedure in no way ‘computes’ the design of urban fabric; it is an artistic game that only computes a random graphic, which is then imposed upon the ground. ***The third law of human-scale urbanism is that genuinely adaptive computation is based on complex urban algorithms, not algorithms for generating visual graphic effects.*** Urban morphology is meant to contain and promote human activities and should not be confused with visual sculptural art.

Randomness results in disordered complexity. For example, monofunctional housing subdivisions consist of repeating module houses. Their positioning is in most cases arrived at through a random computation: a whimsical arrangement that is part of a marketing strategy. The developer draws up some curved roads in the office when the land is first purchased, then gives that plan to the government so that local roads and the sewerage infrastructure can be laid down. These roads are randomly drawn on a plan, not as a computation according to local topography, as they would be when following a river or land contour (Figure 8). It is believed that an artistically curved road grid will be more attractive than the boring rectangular grid of post World-War II suburban housing regions (Figure 1), which may be true. Since no interactive algorithm is responsible for generating the curved roads, and there is no input from either the site conditions, or human use patterns, the positioning is entirely random.

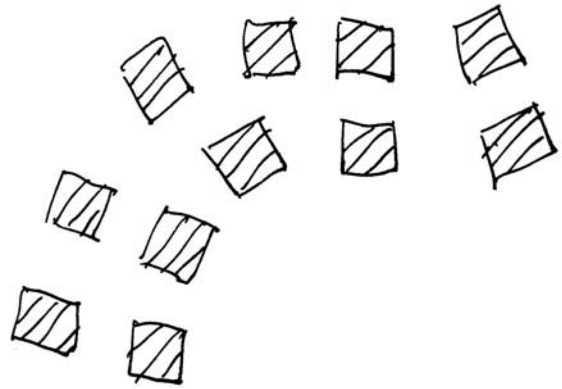

Figure 8. Simplistic modular buildings distributed along a randomly-applied curve.

This arbitrary design method is far more prevalent than might at first appear. Typically, an architect (rather than an urbanist) draws a graphic for the footprints and placement of a cluster of buildings, which are then constructed as drawn (Figures 9 & 10). Famous ‘star’ architects are commissioned to design urban fabric containing their own showcase buildings. They conceive a graphic design with injected randomness and prepare a visually striking computer-generated virtual presentation. Their project then wins a competition solely on its futuristic look. Graphics are substituted here for architecture and urbanism. It would be incorrect to term this design basis as artistic, because it represents a narrow and peculiar aesthetic that is certainly not shared.

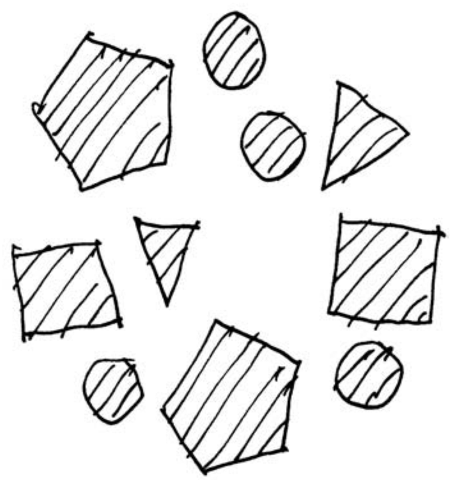

Figure 9. Buildings with Platonic geometries (but programmatically irrelevant plans) are positioned randomly.

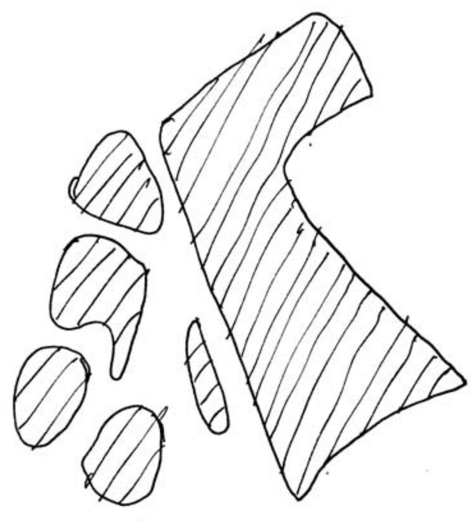

Figure 10. Buildings with amorphous (and programmatically irrelevant) plans are positioned randomly.

Random designs disguised as 'contemporary forms' are in fact arbitrary, because they are not adapted to any priorities of actual people on the ground. The way in which the final buildings, roads, paths, and open spaces are actually experienced is usually a surprise to users, after everything is built and it is too late to make any adjustments. The surprise could in fact be most unpleasant, to the point of condemning the award-winning project as dysfunctional. Here, for example, is my own interpretation of one such algorithm (Figure 9):

1. Take a few buildings having plans of Platonic figures such as a triangle, square, pentagon, and circle.

2. Arrange them randomly within the area of a disk.

These buildings' footprints are programmatically irrelevant. Any apparent visual ordering from an aerial perspective gives the misleading impression that this design is the result of intelligent decisions towards resolving problems of urban morphology. Evidence of the architect's intelligence in creating the interesting graphic plan is mistaken for an intelligent approach to solving an urban problem, but in fact the intelligence has been misapplied. No effort has been directed to computing urban morphology as it relates to human users. Another random algorithm might run as follows (Figure 10):

1. Using computer software, randomly draw 3-dimensional blobs on the ground, connecting some of them into larger, more continuous blobs.

2. Build low buildings on the large blob footprints, and high buildings on the small blob footprints.

The result of this random design game is then constructed as an urban region. Again, the buildings' footprints are programmatically irrelevant. No thought is given to computing the connections between the buildings, nor to shaping the spaces created in-between the buildings, nor to adjusting the relationship of both buildings and urban spaces to the surroundings, nor to any other key aspect of adaptive urban design. This method of graphic design is not adaptive, since the result could be dropped into any location in the world. With its self-referentiality, Random urbanism is akin to both Non-computational and Non-interactive computational categories of urbanism in generating deficient urban fabric.

Architects introduce further confusion by creating and building random designs, while at the same time labeling them as 'formal'. The term 'formal' is not unambiguously defined. What architects mean is that those configurations (their random designs) are the result of a sequence of transformations based upon arbitrarily defined criteria and rules. Since the steps leading to such designs are arbitrary, the end result is random, although it tends to look 'ordered'. Unless the steps of a computation where each step computes a configuration adapt the configuration to the problem at hand, a sequence of steps cannot be accepted as a relevant approach to a solution.

Non-computational International Style urbanism.

The International Style is represented by glass-and-steel cubes, concrete slabs, and boxes. For reasons to do with the politics of architecture and ideology, and the operational forces of commercial globalization, this style has spread around the world, so that a new office building in Buenos Aires is bound to look identical to an apartment building in Shanghai. This architectural style, though extremely widespread, is arrived at by a very limited and formally fixed algorithm. Since the design is basically a box made from the same materials, employing an engineering construction that covers a supportive steel frame with glass curtain walls, there is little to compute. A simplistic module is repeated the world over.

The reductivist non-computational aspect of International Style buildings extends beyond their form, to include their positioning in the urban fabric. Position is invariably determined by formalistic arguments; that is, following the visual arrangement of a grid on the plan (Figure 1). While that might appear to be a 'rational' decision, it actually involves only trivial computations. There is no adjustment of the building's placement according to existing path structure, seeking to encourage the spontaneous development of new paths, paying attention to climatic conditions and solar orientation, enhancing the possibilities for communication with adjacent buildings, or, most importantly, collaborating with existing structures in order to define a usable urban space. More often than not, existing complex structure is destroyed so that the new building does *not* have to adapt to it.

It requires little or no computation to draw a square or a rectangular grid. Clearly, a generic design that is repeated all over the world with no variations cannot possibly adapt to any of the above-mentioned human factors; there exist multiple independent constraints that a new building has to adapt to in order to truly fit into the urban fabric.

One might wonder why non-adaptive structures are so popular, and continue to be built in great numbers. The reason is that the architectural and urban professions, as well as the public at large, have come to accept non-adaptivity as an unassailable feature of a certain style of contemporary design, and therefore deceived clients naively sponsor such non-adaptive buildings. An obsession for this symbolic visual style obscures its serious urban failings. Clients and the building industry just love to repeat the same industrial model all over the world, saving themselves the trouble (and money) of having to design genuinely adaptive human environments. Celebrating the extreme contradiction with older traditional models, which are themselves highly adaptive, the world goes on replacing its previously adapted urban fabric with very expensive new but non-adapted buildings.

Moving from non-computational towards random design.

The paradigmatic example of non-computational urban design is the 'cookie-cutter' method, which repeats the same simplistic module placed regularly over a rectangular grid (Figure 1). So much of post World-War II urban growth looks like this from the air, with the same typology repeated endlessly in a large monofunctional residential zone. It makes no difference whether these are middle-class suburbs or government-built social housing for the poor. Allowing a functional mix with a little design freedom would alter the monotony of this widespread example, but that is not usually possible. Government zoning regulations legally forbid what are obviously necessary and beneficial design variations. Those laws, widely implemented after World War II, have more to do with architectural ideology than with urbanism.

The sprawl typology of monotonously repeating square or rectangular modules (Figure 1) exists on every scale: one-family houses in suburban sprawl; modular apartment blocks of four residential units each; or buildings on a much larger scale such as modular apartment blocks containing twenty, forty, or more residential units. The same simplistic modular idea lies behind a city of skyscrapers set in hard concrete plazas (Figure 11), as was proposed in the well-known dystopian schemes implemented after the Second World War. That obsession with vertical gigantism occurred at the same time as the proliferation of sprawl consisting of single-storey modular houses. Identical skyscrapers repeating on a rectangular (or any other regular) grid require little or no urban computation (Figures 1 & 11), from which follows the computational equivalence between horizontal sprawl and skyscrapers.

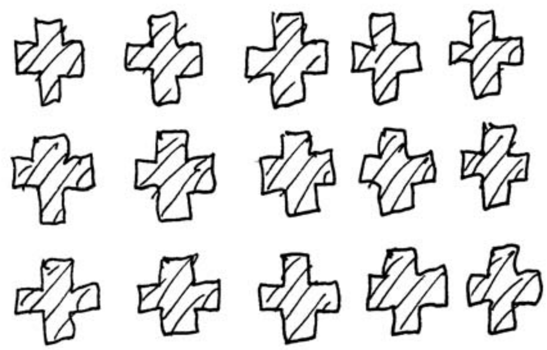

Figure 11. Modular buildings with irrelevant complexity distributed along a regular irrelevant grid.

Let me outline an experiment in urban morphology. Injecting randomness into a repetitive urban design could be done in either of two distinct ways, both of them programmatically irrelevant. Begin with a non-computational urban region consisting of simplistic repeating modules (Figure 1). One could randomize the buildings' positioning on the ground slightly (Figure 8), or more deliberately (Figure 12). Another method of injecting randomness is to keep the grid but make the buildings' footprint more random (Figure 13). The non-interactively computed footprint (Figure 11) looks more regular but is no more adaptive than the random case (Figure 13). Those projects are again victims of the flawed conception that a programmatically irrelevant graphic will somehow improve the functionality of a repeating module.

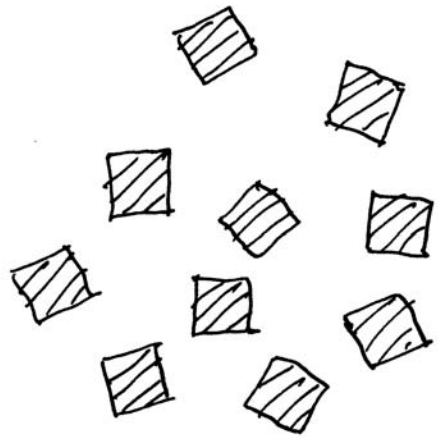

Figure 12. Randomness injected into the positioning and alignment of modular buildings.

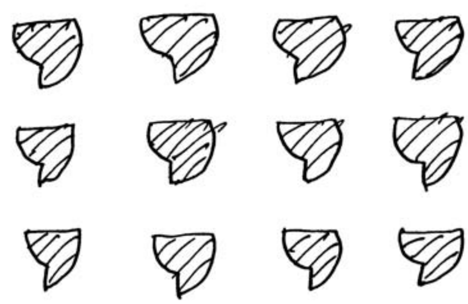

Figure 13. Randomness injected into the shape (plan) of modular buildings, but their regular positioning is maintained.

A random region looks complex, but then so does living urban fabric that results from Interactive computational urbanism. ***The fourth law of human-scale urbanism is that adaptive computations generate complex urban fabric with sufficient geometric diversity to not need any imposition of randomness.*** Moving towards Random urbanism is not the solution to the monotony of the simplistic repetition of Non-computational urbanism, an unfortunate mistake made by many clients. Only introducing complexity using computational rules for adaptivity works to enliven monotonous urban fabric.

Conversely, a complex but disorganized and problematic urban region cannot be improved just by making it more superficially ordered. Geometrical organization is part of the answer, but it has to be carried out using adaptive computations, not graphic design that is irrelevant to human use. Designers of urban regions too often generate a graphic intervention on a plan, and then implement it despite any possible non-adaptability to the functions of that particular complex of buildings, to the people and to the human movement they are supposed to accommodate.

Traditional urbanism is adaptive and not random.

Any discussion of urbanism as computation has to fight against several decades of basic misunderstandings about the nature of traditional urbanism. The modernist movement was based upon non-computational urbanism, and its champions declared, without any proper understanding, that traditional urban form was 'random' and proceeded to recommend its replacement with simplistic non-computed forms. This was a propaganda strategy aimed at gaining commissions, in a bid to free up land for new modernist buildings. A negative psychological association was used to condemn existing urban fabric: the false accusation of 'random' versus the proposed but equally false 'rational' alternative being offered. Unfortunately, generations of urbanists have mistaken these polemics for authoritative statements about urban form (Salingaros, 2005; 2006; 2010).

Decades of prejudice against traditional urbanism have polarized public opinion against complex adaptive urban fabric condemned as 'messy' and 'old-fashioned', obscuring the interactive computational basis that produced it. Traditional urban fabric was replaced with non-computed simplistic structures, such as social housing in isolated high-rise apartment blocks. Those subsequently proved to be disastrous failures. In spite of these and other well-publicized fiascos, the mythology of non-computed, falsely 'rational' forms continues to be propagated in the media and in all the schools. Any attempt to re-introduce computational methods for urbanism is resisted on ideological grounds (but those arguments are always disguised in technical terms).

Traditional urbanism is in fact intensely computational and adaptive. Because of urban pressures that if left unchecked would take over every single piece of territory, the built environment is continually being negotiated and re-adjusted by following a complex computational process. Every piece of traditional urban fabric, from the self-built favelas and informal settlements around the world, to the older historic cities that act as a magnet for tourists, represents the result of interactive computations.

For example, an urban space is created by the surrounding building fronts and pattern of streets (Figure 14). This space must be primarily pedestrian, and not made into a parking lot. It must also have maximal pedestrian but restricted vehicular access. During the computational steps required to create a living urban space, buildings negotiate their position and footprints so as to partially enclose the urban space, allowing gaps for paths and roads. These adaptive computations can occur all at once in a new city, or over a millennium in a historic city center.

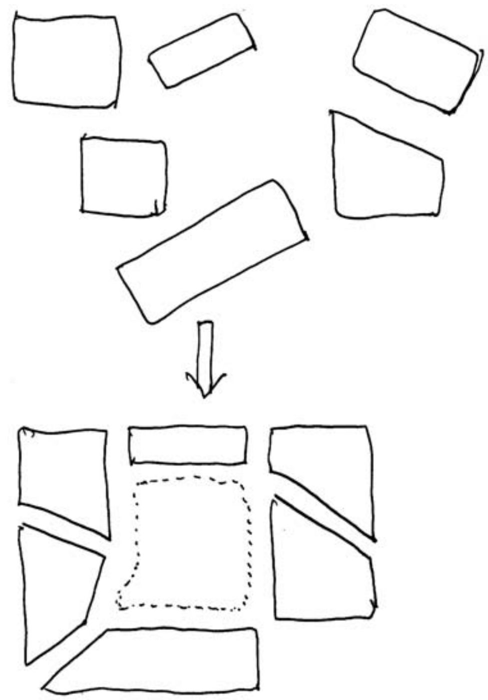

Figure 14. Plan adapts to create semi-enclosed pedestrian urban space. Vehicular traffic connects to, but should not invade this space.

Today's scientific knowledge makes it possible to identify different types of urban fabric according to the four algorithmic categories introduced above, and to dispel the confusion surrounding what merely 'looks' complex. Obviously non-computational urban fabric is, and looks, simplistic in plan. Complex-looking urban fabric, however, has to be carefully analyzed so that it can be classified into the three remaining categories: random, non-interactive computational, or interactive computational. Urban success or failure can then be understood in terms of adaptivity that follows the underlying computational basis of the morphology.

Computing before and after: how contemporary interventions destroy historic plazas.

There are two distinct periods when urban computations are implemented: before and after a building, or cluster of buildings, have been erected. Clearly, it is far easier to compute before and during construction. In traditional urbanism, the normal process of historical adjustment implemented some computations after a building's completion, as changes were made over time. The most-loved of our cities are the result of continuous re-adjustments that have reinforced each other. In today's throwaway society, however, a problematic building is seldom adjusted, but is more likely to be demolished and an equally non-adaptive building put up in its place. The point is that before and after construction any of the four different methods of urban computations listed above may be applied.

Adaptive design and adjustments during a building's history is a neglected topic (Brand, 1995). Contemporary design ideology seems focused upon erecting structures on every scale that are never meant to change, and therefore the theory of adaptive computations is not a part of present-day design discourse. The fault for this unrealistic attitude lies with the extensive media hype surrounding fashionable architecture and urbanism. The possibility that one of today's famous 'star' architects could design a showcase project that is not 'perfect' is too shocking to be contemplated. For this reason, expensive mistakes are hardly ever admitted, let alone repaired, after they are built. In this ideological mind-set, the methodology of adaptive repair that privileges function over form, along with the computational methods it contains, has been relegated to obscurity.

The opposite instance of this ideology, where 'repair' is carried out on urban fabric that doesn't need it, or by using the wrong type of algorithm, can be far more damaging. Contemporary architects believe that their sculptural forms are far superior to existing structures, and will destroy older urban fabric in order to make them possible. Sadly, in many recent examples, an urban space that has worked as a collective node for society for a millennium is deliberately destroyed by a contemporary intervention. A lot of money is spent on the renovation, the mayor is proud of the result since architectural experts praise it, yet the public space becomes dead because it is hated and avoided from that moment on. What happened is that the wrong type of algorithm was used for the recent additions or changes. The urban space becomes an example of algorithmic non-adaptivity.

An urban plaza built and modified over several centuries used Interactive computational algorithms to evolve all of its structural features. That was the secret of its success. Nowadays the current fashion is to use Random and Non-computational algorithms everywhere for urban design, and of course these are applied unthinkingly to 'renovate' a historic plaza. A random algorithm may be used to create an abstract structure such as a warped pavilion, a giant formless sculpture or fountain, etc. That alien object henceforth dominates the public space, influencing all potential users negatively. Otherwise, a Non-computational algorithm may be used to create massively rectangular concrete benches and tree planters, a hard minimalist pavement, or a metal canopy that resembles a crane used for lifting shipping containers.

Users perceive the difference in computational method intuitively; an evolved form gives the positive impression of 'growing from' the existing situation, whereas a non-computed form is seen negatively as 'inserted' into it. We know that beneficial complexity evolves through a computational process, using feedback from human needs and sensibilities. All changes made by using a non-adaptive algorithm are by definition alien, and they degrade the perceived human qualities of the geometry inherent in the urban space. ***The fifth law of human-scale urbanism is that by destroying its computational coherence, even minor built elements, if they are products of non-interactive algorithms, can switch urban morphology from being adaptive to non-adaptive.*** Contrary to any media praise for the supposed artistic value of such non-adaptive interventions (conforming to a narrow and peculiar aesthetic), in reality they are usually lethal to the life of the urban fabric.

Intelligence and interactive computations.

In this last section, I wish to focus on the qualities of the computer that actually performs urban computations. For millennia, this task has been done by the human brain, which evolved to interact dynamically with its environment. Humans created historical artifacts, buildings, and cities in a stepwise and unselfconscious manner (Alexander, 2001-2005). Artificial computers, on the other hand, are only now beginning to acquire capabilities of interactive computation. Therefore, even though computers are routinely used for non-adaptive design (as tools merely for generating a visual graphic), in most cases they are not employed for computational purposes as outlined in the present model.

Interactive computations are responsible for both biological and machine intelligence. Perception mechanisms based on communication with the outside world occur during neural computations (Wegner, 1997). This is the basis for animal biology: the ability to both perceive and navigate the environment. As human technology has recently advanced in parallel to biology, machines embodying pieces of intelligence now shape our daily lives. The virtual informational universe we inhabit today is quintessentially interactive, consisting of human-computer interfaces, the World-Wide Web, and intelligent agents, all of which function using interactive computations to create a complex global information system that includes us.

This principle holds true for both biological and machine intelligence. As was discovered in the development of mobile robots (Brooks, 1999, 2002), this type of robot functions by interactively computing its own situation in its environment. Those computations have to occur in real time, in an ongoing interactive process. Such computations are far more complex, and fundamentally distinct from, the self-contained Turing computations characterizing a non-interactive algorithm. Turing computations have an initial informational input, then perform their computations in a closed environment, and finally output their result. In a Turing computation, no input or output occurs during computation (Wolfram, 2001).

Robots are commonly misunderstood to be (isolated) machines that privilege their internal program over their surroundings. One branch of Robotics does use extensive internal storage of data and possibilities, to which the robot refers for decisions about its next action (static industrial robots are programmed internally to perform a specific repetitive task), but this has proved to be tremendously costly in terms of devoting computing resources to memory. The most successful mobile robots contain no internal representation of the outside world at all, and make their decisions by interacting with their real surroundings (Brooks, 1999, 2002). The latest generation of mobile robots work in this way, using the environment as the robot's memory. It turns out that the model of no internal representation created a breakthrough because of the saved computing power that can be devoted to decision-making.

The urban computations that I described as being responsible for traditional and vernacular urban fabrics follow mechanisms of precisely this interactive type. They are not Turing algorithms, but rather complex hierarchical algorithmic sequences that interact with the environment at every computational step. Computing power is devoted to the decision-making necessary in the exploration of the solution space. In contrast with

Turing algorithms (which are closed, and do not exchange information with the environment), interactive algorithms depend upon continuous in-out informational flow (Wegner, 1997; Wegner & Goldin, 2003). Each step in their sequence of computations depends on both informational inputs and outputs from preceding steps.

An interactive computation harnesses the information content of the environment, which is itself being used for the computation. Since there is no way to control the external environment, we might as well utilize it instead of ignoring it (and often thereby destroying it). Unfortunately, an interactive approach to urbanism runs counter to much of contemporary planning philosophy, where central control is the overriding concern. What urbanists know as design follows a self-referential model: a top-down procedure that produces buildings and urban fabric based upon some legally defined prototypes such as streets and sidewalks decided upon by the planning department, building lots again defined by legislation, buildings themselves correlating with formal codes, etc.

The goal of interactive computations in building biological, architectural, and urban systems is to encourage the emergence of large-scale system coherence (Alexander, 2009). Large-scale system coherence is created by the step-wise interactive computations, and this coherence is the goal of the computational sequence. Therefore, contrary to naïve expectations, adaptive design does not compute a specific form, but rather a state of coherence and connectivity. Adaptive urbanism is characterized by a mathematical quality showing 'intelligence' intrinsic to the system itself, not a particular geometry. There exist constraints and instructions that contain elements of the result within a defined language, yet the individual result is free to evolve but entirely coherent (Salingaros, 2010).

Conclusion.

I introduced a four-way classification of urban categories according to the type of algorithm responsible for their design. Using this model, we can judge the effectiveness of urban fabric in large part by its visual appearance on a plan. I argued for adaptive complexity that is generated via an interactive computational process, where human needs and the state of the environment are fed back into the computations at each step. This process is similar to that which allows a mobile robot to navigate its environment. Interactive computation is, in fact, the basis for biological and machine intelligence. Traditional urban fabric results from interactive computations, not random acts and occurrences. The four-way classification made possible a discussion of all different types of urban fabric, from formal to traditional, and to self-built housing found in the world's informal settlements. Each urban type fits somewhere in this classification. Most usefully, we can now track the evolution of one urban type towards another which has distinct characteristics: for example, transforming homogeneous modular houses on a rectangular grid into more randomly-positioned settings, or increasing the complexity of an individual building module while repeating it on the same grid. Understanding this evolution in terms of the underlying computational characteristics goes much further than the previous practice of interpreting urbanism in formal terms.

APPENDIX: Five laws of human-scale urbanism.

1. What appears to work and connect on paper in an abstract, formalistic manner does not necessarily work and connect on the ground.

2. Adapted computed solutions are not transferable.

3. Genuinely adaptive computation is based on complex urban algorithms, not algorithms for generating visual graphic effects.

4. Adaptive computations generate complex urban fabric with sufficient geometric diversity to not need any imposition of randomness.

5. By destroying its computational coherence, even minor built elements, if they are products of non-interactive algorithms, can switch urban morphology from being adaptive to non-adaptive.

Acknowledgments.

I thank Rajendra V. Boppana, Turgay Korkmaz, Stephen Marshall, Kenneth G. Masden, and George Papanikolaou for their very useful comments.

REFERENCES.

Alexander, C. (2001-2005) *The Nature of Order, Books 1-4*, Center for Environmental Structure, Berkeley, California, USA. Book 1: *The Phenomenon of Life*, 2001; Book 2: *The Process of Creating Life*, 2002; Book 3: *A Vision of a Living World*, 2005; Book 4: *The Luminous Ground*, 2004.

Alexander, C. (2009) "Harmony-Seeking Computations", *International Journal of Unconventional Computing*, Volume 6, in press. Earlier draft available from: <http://www.livingneighborhoods.org/library/harmony-seeking-computations.pdf>

Alexander, C., Ishikawa, S., Silverstein, M., Jacobson, M., Fiksdahl-King, I. & Angel, S. (1977) *A Pattern Language*, Oxford University Press, New York.

Batty, M. (2005) *Cities and Complexity*, MIT Press, Cambridge, Massachusetts.

Brand, S. (1995) *How Buildings Learn*, Penguin Books, New York.

Brooks, R. A. (1999) *Cambrian Intelligence*, MIT Press, Cambridge, Massachusetts.

Brooks, R. A. (2002) *Flesh and Machines*, Pantheon Books, New York.

Duany, A., Wright, W. & Sorlien, S. (2009) *Smart Code*, Version 9.2, available from

Duany-Plater-Zyberk: <http://smartcodecentral.com/smartfilesv9_2.html>

Hillier, B. (1999) *Space is the Machine*, Cambridge University Press, Cambridge, UK.

Marshall, S. (2009) *Cities, Design and Evolution*, Routledge, London, UK.

Portugali, J. (2000) *Self-Organization and the City*, Springer-Verlag, Berlin.

Portugali, N. (2006) *The Act of Creation and the Spirit of a Place: A Holistic-Phenomenological Approach to Architecture*, Edition Axel Menges, Berlin.

Salingaros, N. A. (2005) *Principles of Urban Structure*, Techne Press, Amsterdam, Holland.

Salingaros, N. A. (2006) *A Theory of Architecture*, Umbau-Verlag, Solingen, Germany.

Salingaros, N. A. (2010) *Twelve Lectures on Architecture: Algorithmic Sustainable Design*, Umbau-Verlag, Solingen, Germany.

Salingaros, N. A., Brain, D., Duany, A. M., Mehaffy M. W. & Philibert-Petit, E. (2006) *Socially organized housing: a new approach to urban structure* (book in preparation).

Wegner, P. (1997) "Why Interaction is More Powerful than Algorithms", *Communications of the ACM*, Vol. 40, No. 5: pages 80-91.

Wegner, P. & Goldin, D. (2003) "Computation Beyond Turing Machines", *Communications of the ACM*, Vol. 46, No. 4: pages 100-102.

Wolfram, S. (2001) *A New Kind of Science*, Wolfram Media, Champaign, Illinois.