Harvard Graduate School of Design
Cambridge, Massachusetts

AutoPLAN: a stochastic generator of architectural plans from a building program

by Kostas Terzidis

In the world of design, computer programs have taken over many traditionally human intellectual tasks leaving fewer tasks for traditional designers. From Photoshop filters to modeling applications and from simulation programs to virtual reality animation and even more mundane tasks that used to need a certain talent to take on such as rendering, paper cutting, or 3D sculpting the list of tasks diminishes day by day only to be replaced by their computational counterparts. What used to be a basis to judge somebody as a talent or a genius is no more applicable. No longer are dexterity, adeptness, memorization, fast calculation, and aptitude sought after in a designer's skills set, nor do they elicit admiration and genius-level praise. The focus has shifted far away from what it used to be toward new territories. In the process many take advantage of the ephemeral awe that the new computational tools bring to design by using them as means to establish a new concept or form only to be revealed later that their power was based on the tool they used and not on their own intellectual ability. After all, the tool was developed by somebody else, the programmer who discovered the tool's mechanism, and should, perhaps, be considered the innovator instead.

As a result of the use and abuse of design tools, many have started to worry about the direction that design will take in the coming years. As one-by-one all design tasks are becoming computational, some regard this as a danger, misfortune, or an appropriation of what design should be and others as a liberation, freedom, and power toward what design should be: i.e. conceptualization. According to the latter, the designer does not need to worry anymore about the construction documents, schedules, databases, modeling, rendering, animation, etc. and can now concentrate on what is most important: the concept. But what if that is also replaced? What if one day a new piece of software appears that allows one to input the building program and then produces valid designs, i.e. plan, elevation, and sections that work. And, worse, what if they are better than the designer would have ever done by himself or herself? (Even though most designers would never admit publicly that something is better than what

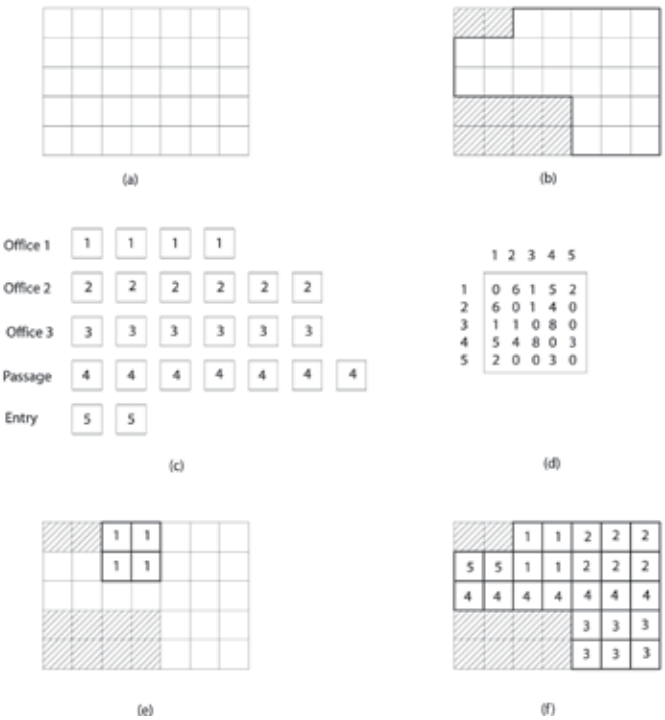


Figure 1: **(a)** A grid, **(b)** the site, **(c)** the spaces, **(d)** the adjacency matrix, **(e)** placing a space, and **(f)** one possible solution.

they would have designed, yet what if deep inside them they would admit the opposite). What then? Are we still going to continue demonizing the computer and seeking to promote geniuses when they really don't exist?

During the peak of enthusiasm for possibilities that opened up for computational design in the early 1970s, a series of innovative projects were set as potential targets. One of them was the automatic generation of plans from building programs that was proposed by Dietz (1974). It involved a unit system, a site, a program, and an adjacency matrix and then the computer system would produce multiple solutions by trying various combinations of space allocation based on the neighborhood rules (see image below). This possibility apart from clever, innovative, productive, and effective, it also introduced indirectly a radical view on the role of the designer and the process of design itself. In its simplest manifestation it calls for the production of an architectural plan without human guidance. In its so-called "automatic" nature, it negates the very premise upon which architecture, and design by extension, has established its existence, identity and authority throughout the ages. It poses a strange paradox where design is redefined not as an intentional articulation of form in pursuit of an objective, but as a random reshuffling of information under constraining rules until a possibility is met that satisfies a function. Despite its promising potential, automated design did not take off as one would perhaps expect. Instead, computers simply became tools that enhanced the productivity, efficiency, and presentation of design that led eventually to enhancing the ego of the designer instead of challenging it.

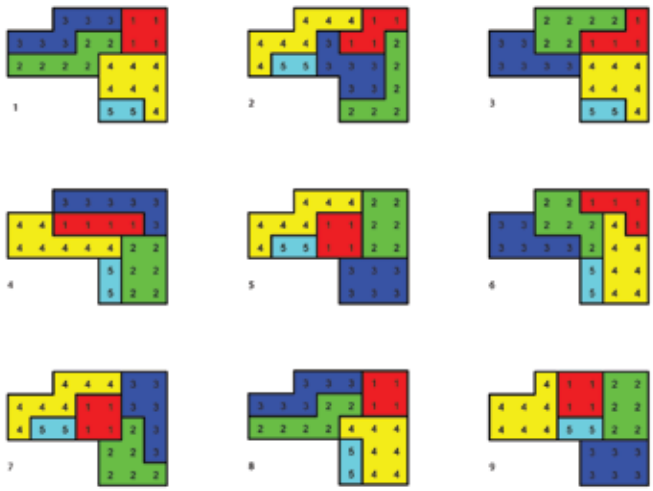
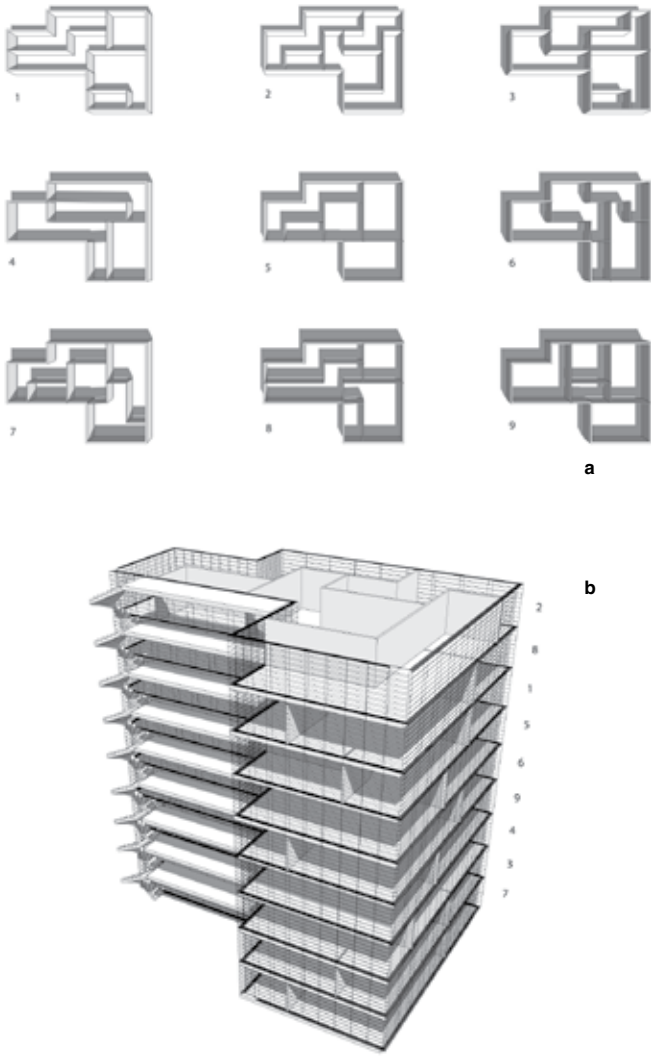


Figure 2: A series of possible solutions.

In an attempt to shed light on this missing opportunity, the author of this paper developed a computer program called autoPLAN in 2008 that generates architectural plans out of a building program and a site. The program was written in the Processing computer language and can export multiple CAD files, one for each plan that was then further enhanced using **form•Z**. A series of plans generated under autoPLAN can be seen in the figures below. AutoPLAN uses a stochastic search algorithm that searches for available space to distribute the program's rooms given the site's boundary and the adjacency matrix.

The program and its algorithm demonstrate an alternative approach to the potential of computation as a design methodology. Is it possible that a design can be accomplished through the exhaustive search of possible solutions? Consider the case of all possible combinations of black or white for nine squares in a 3x3 arrangement. They are 512. If we constrain the choice to only symmetrical configurations, those are only 32. Or perhaps all possible combinations of three continuous sets of three block of three colors in a 3x3 arrangement. Those are only 60. In that sense, the notion of randomness can



a

b

Figure 3: **(a)** A series of possible plans and **(b)** a high rise.

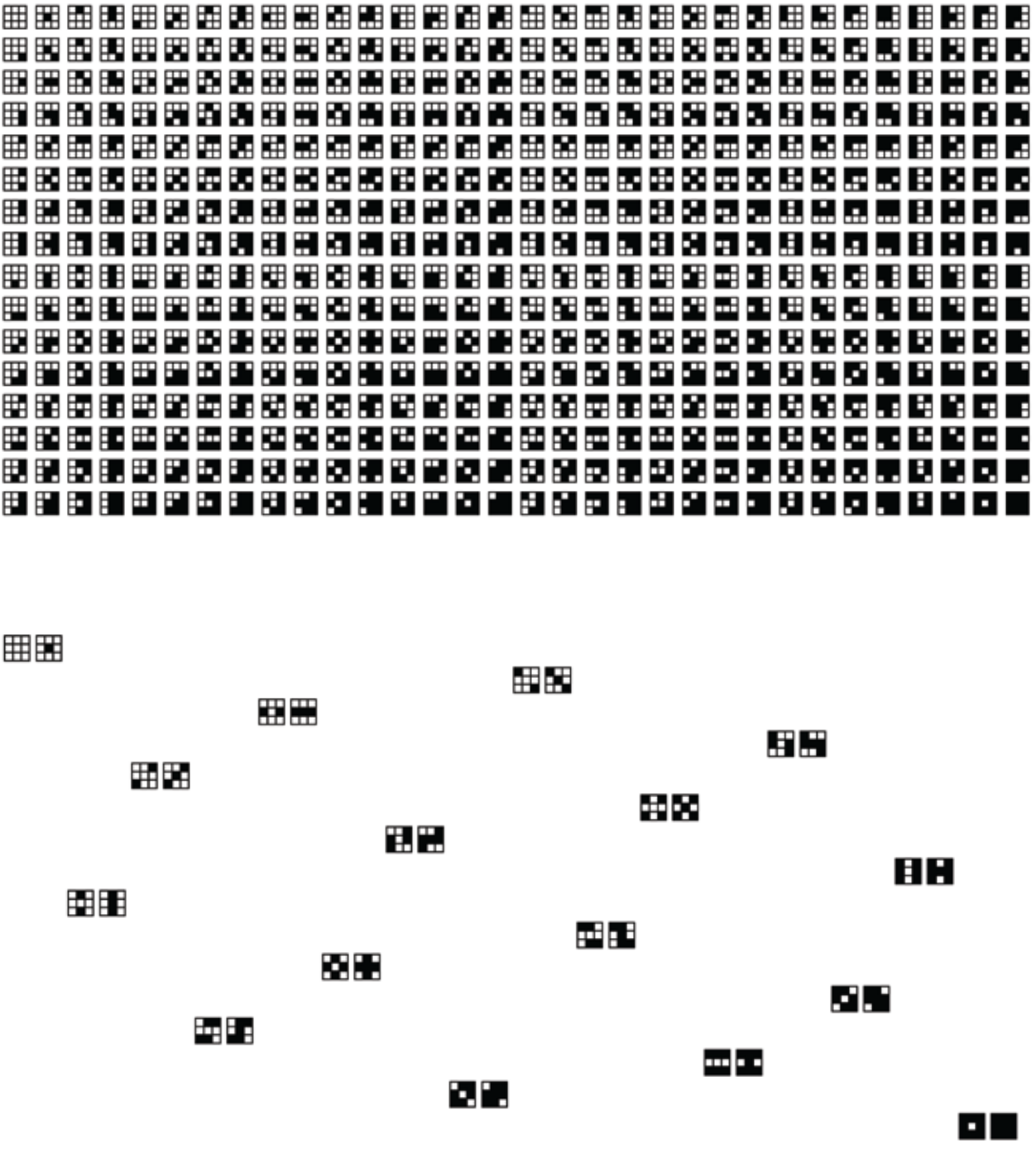


Figure 4: All possible combinations of black or white for nine squares in a 3x3 arrangement (top) and those that are symmetrical (bottom).

be seen not as a chaotic disorganizing principle that is often portrayed, but rather as an ordering mechanism. Perhaps, the term randomness should be redefined here to clarify its connection with order. Random presupposes an exhaustive search of all possible combinations and therefore can be seen as unexpected sampling. In the cases shown earlier, such combinations are computable within a reasonable amount of time. Yet, in other cases, the combinations are so many that it is not possible to be computed in a desirable amount of time. In such a case, randomness functions as a sampling mechanism that provides possible choices for the designer, occasionally surprising. Nevertheless, such an ordering device is based not on a careful premeditated intuitive process but rather on simple, almost naïve, attempts under extreme repetition. The process, albeit antithetical to that of traditional design, sets out a new paradigm where design is laid out, not in the mind of the user, but rather in the computer program that addresses the issue. The focus of design is not even in the process itself since that can be replaced, but rather in the replacement operation itself. In that realm the new designer constructs the tool that will enable one to design in an indirect meta-design fashion.

References

1. Dietz, A., Dwelling House Construction, Cambridge: MIT Press, 1974.

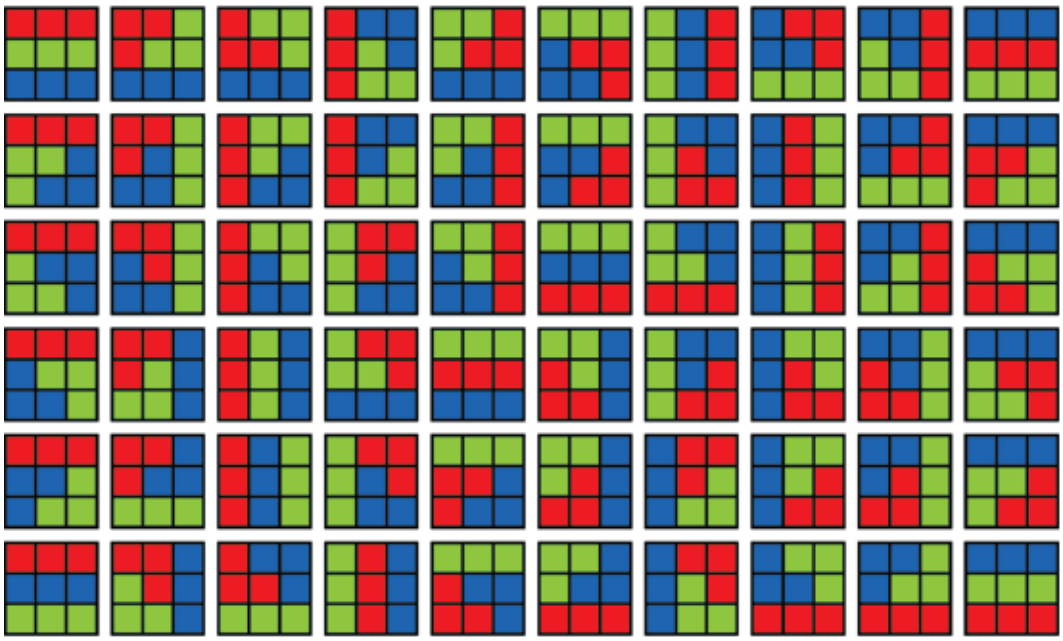


Figure 5: All possible combinations of three continuous sets of three block of three colors in a 3x3 arangement.

Kostas Terzidis is an Associate Professor at the Harvard Graduate School of Design. His current GSD courses are Kinetic Architecture, Algorithmic Architecture, Digital Media I & II, Cinematic Architecture, and Design Research Methods. He holds a PhD in Architecture from the University of Michigan (1994), a Master's of Architecture from The Ohio State University (1989), and a Diploma of Engineering from the Aristotelion University in Thessaloniki, Greece (1986). He is a registered architect in Europe where he has designed and built several commercial and residential buildings. His research work focuses on creative experimentation within the threshold between arts, architecture, and computer science. As a professional computer programmer he is the author of many computer applications on form-making, morphing, virtual reality, and self-organization. His most recent work is in the development of theories and techniques for algorithmic architecture. His book Expressive Form: A Conceptual Approach to Computational Design published by London-based Spon Press (2003) offers a unique perspective on the use of computation as it relates to aesthetics, specifically in architecture and design. His book Algorithmic Architecture, (Architectural Press/Elsevier, 2006), provides an ontological investigation into the terms, concepts, and processes of algorithmic architecture and provides a theoretical framework for design implementations. His latest book Algorithms for Visual Design by Wiley (2009) provides students, programmers, and researchers the technical, theoretical, and design means to develop computer code that will allow them to experiment with design problems.